DEVELOPING A FLIGHT DELAY PREDICTION MODEL USING MACHINE LEARNING

Team ID	PNT2022TMID41699
Project Name	Flight Delay Prediction
Team Members	Sivaranjani L(621819104012) Boomika K(621819104001) Gowtyhami S(621819104003) Harini M(621819104004) Sneka B(621819104013)

Table Of Contents

1 INTRODUCTION

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 travelers and the airline industry in 2007 was $32.9 billion in the US, resulting in a $4 billion reduction in GDP [1]. Therefore, predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy.

1.1 PROJECT OVERVIEW:

In this project I looked at different ML techniques/algorithms to try to predict if a flight will be delayed or not before it is even announced on the departure boards. So I will not be aiming to get the highest accuracy possible, because if I wanted to do that, it would be quite easy by adding a series of features/categories that will biased the model in terms of predictive power. Examples of these are “departure delays” and “arrival delays”. Think about it. If you go into a plane knowing already that there is a departure delay, chances are that your flight will be late at arrival. The same happens if you already know that the plane has an arrival delay. So this information was looked at as part of the Exploratory Data Analysis (EDA), but was taken out of the main models. Just to show you my point, I ran the models twice, the second time adding only one of these features that increases the predictive power of the algorithms and as a result I got an average increase of 15%. Now imagine how much this would have increased if I had added all the features. However, how useful is it to know that you will arrive late at your destination when it has been already announced on the Departing Boards or when you are already sitting in the plane and about to take off?

1.2 PURPOSE:

A flight delay is when an airline flight takes off and/or lands later than its scheduled time. The Federal Aviation Administration (FAA) considers a flight to be delayed when it is 15 minutes later than its scheduled time. A cancellation occurs when the airline does not operate the flight at all for a certain reason. This has led to phenomenal growth in air traffic and on the ground. An increase in air traffic growth has also resulted in massive levels of aircraft delays on the ground and in the air. These delays are responsible for large economic and environmental losses. The main objective of the model is to predict flight delays accurately in order to optimize flight operations and minimize delays. The problem of flight delay prediction is approached most often by predicting a delay class or value. However, the aviation industry can benefit greatly from probabilistic delay predictions on an individual flight basis, as these give insight into the uncertainty of the delay predictions.

2 LITERATURE SURVEY

2.1 EXISTING PROBLEM:

Commercial aviation is a complex distributed transportation system. It deals with valuable resources, demand fluctuations, and a sophisticated origin-destination matrix that needs orchestration to provide smooth and safety operations. Furthermore, individual passengers follow her itineraries while airlines plan various schedules for aircrafts, pilots and flight attendants**.** Stages can take place at terminal boundaries, airports, runways, and airspace, being susceptible to different kinds of delays. Some examples include mechanical problems, weather conditions, ground delays, air traffic control, runway queues and capacity constraints.

2.2 REFERENCE:

http://scientiairanica.sharif.edu/article_20020_0.html

https://ieeexplore.ieee.org/document/8903554

https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020 -00380

https://ieeexplore.ieee.org/document/9633571

https://ieeexplore.ieee.org/document/9512525

https://ieeexplore.ieee.org/document/8373742

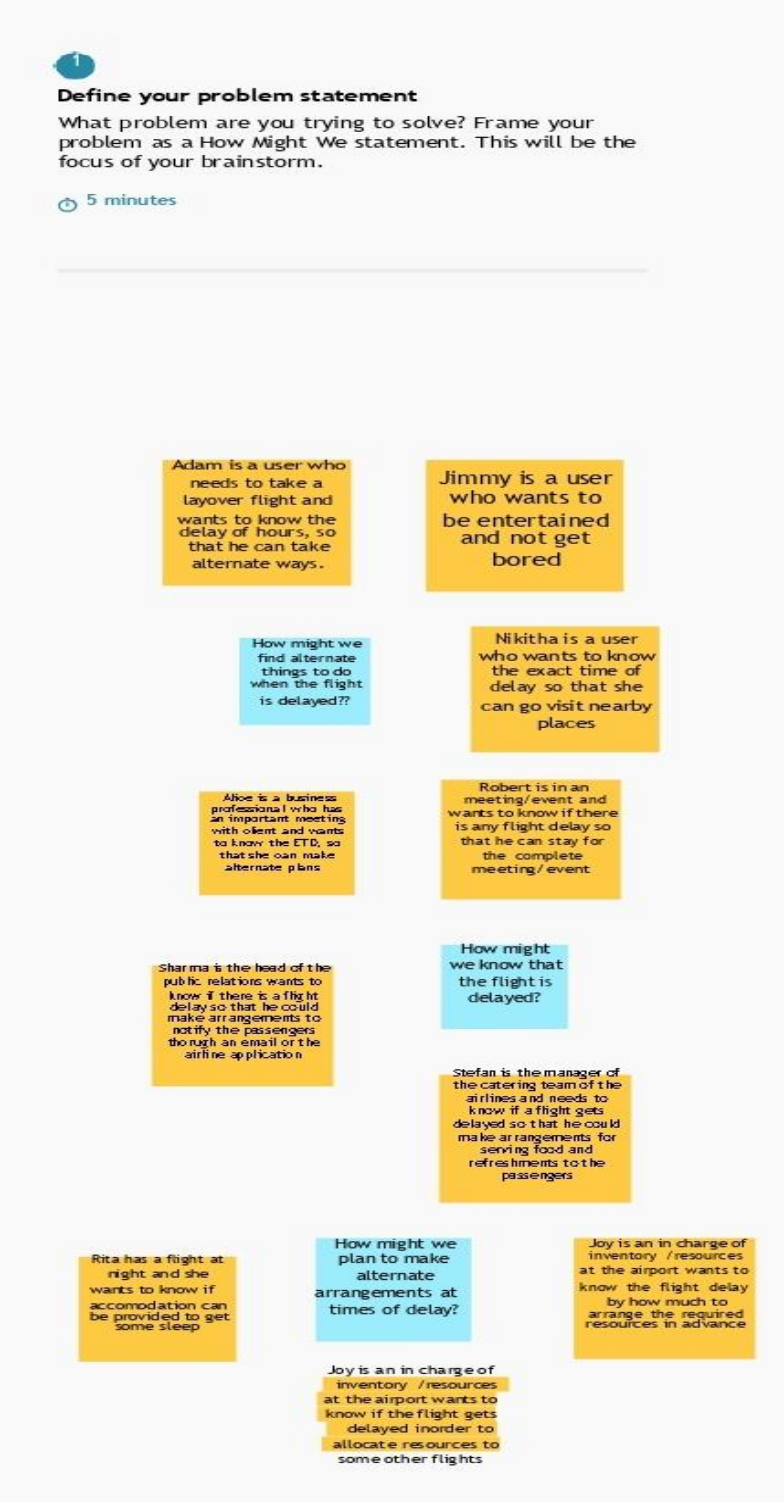
3 IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:

Build empathy and keep your focus on the user by putting yourself in their shoes.

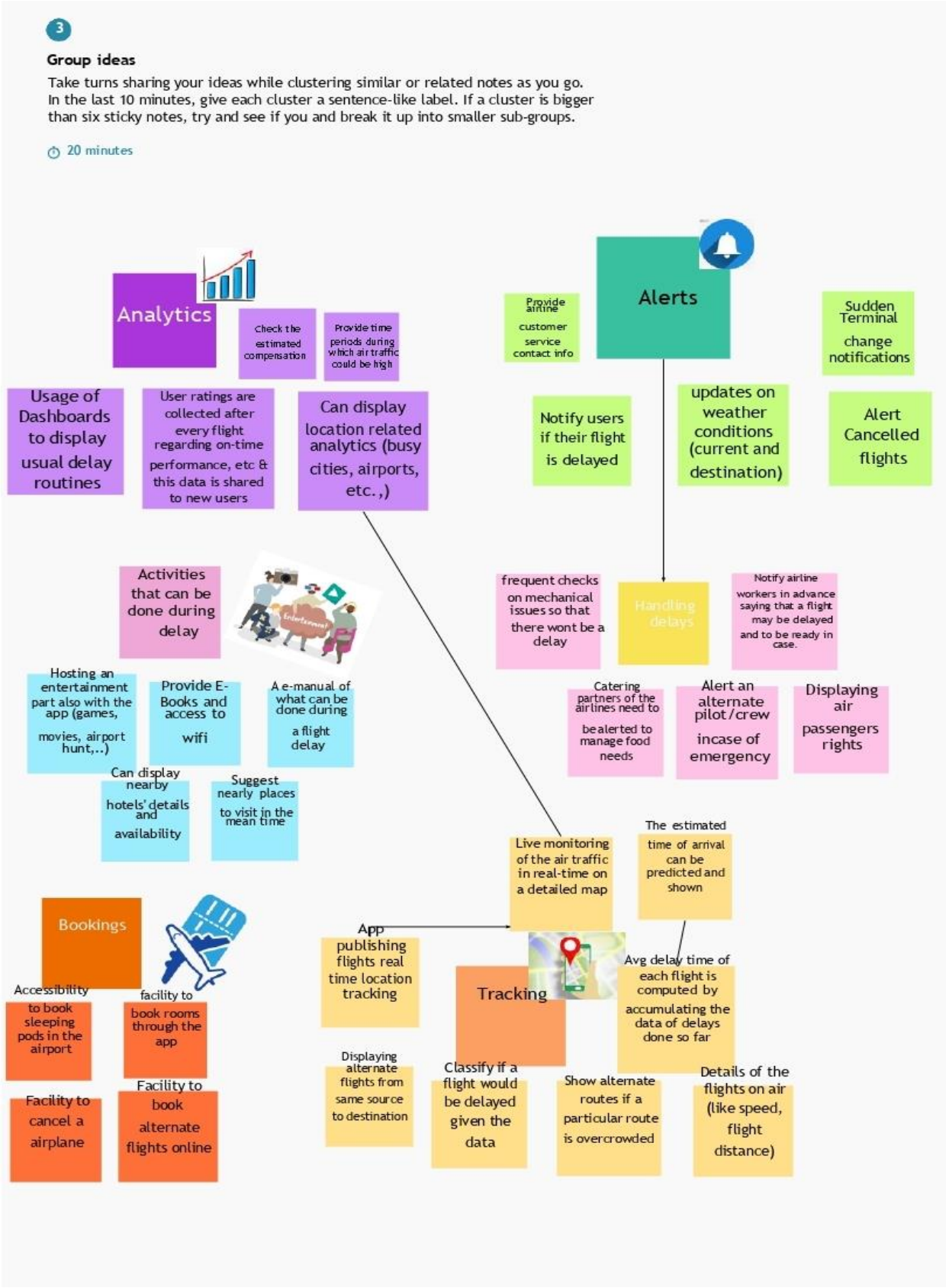
3.2 Brainstorm & Idea Prioritization Template:

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping

Step-3: Idea Prioritization



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

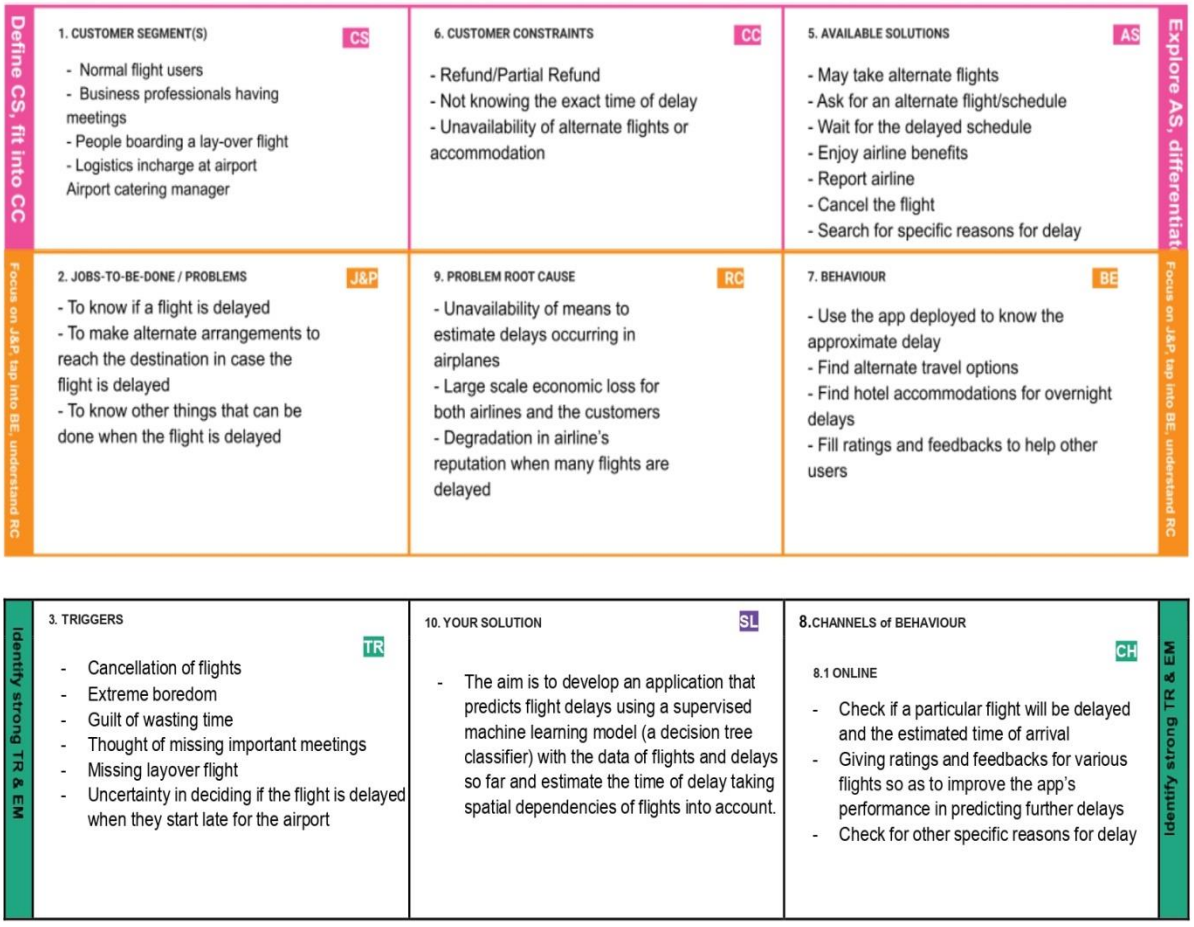
3.3 Proposed Solution Template:

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	● Flight delays have been the most challenging area for airlines to improve. ● They have been affecting the air industry directly and indirectly causing unforeseen expenses thereby reducing the reputation of the industry and the airlines. ● Thus, knowing if a flight would be delayed beforehand can let passengers and airlines be prepared for the circumstances. ● This solution aims at making it possible by predicting arrival and departure delays using Machine learning.
2.	Idea/Solution description	● Building an application interface for customers(passengers and airlines) to know if a flight is delayed by implementing a machine learning based model to predict departure and arrival delays of an aircraft considering spatial, temporal and other dependencies causing the delay..
3.	Novelty/ Uniqueness	● The solution takes into account all possible reasons for delay(crew delays, weather, air traffic, aircraft type) to provide an accurate prediction. ● Apart from predicting arrival delays, departure delays are also predicted in order for the passengers to prepare accordingly and for the airline to make arrangements suitably.

4.	Social Impact / Customer Satisfaction	● A lot of time and money can be saved for the customers and the loyalty and trust of customers towards the company increases. ● Improves airline operations by letting the company prepare in prior to adversaries (like crew illness, timeouts, rescheduling) leading to passenger satisfaction which will result positively on the economy and brand value.
5.	Business Model(Revenue Model)	● Business to Consumer model ● The solution is a low-cost airline model planned to be created as an application with which the consumers can interact directly to know the details of their flight. ● It follows a non-monetary revenue model where the consumers aren't charged for what they get but are asked to provide their flight details and ratings which can be used to improve the model and shared with the airline in return for airline's flight data.
6.	Scalability of the Solution	● The present solution is drafted with the aim of experimenting with airlines based out of the United States of America. ● If there is a possibility to acquire data of a broader region (say North America, other continents), then the solution can be developed to benefit a wider range of people. ● International flight depend on both temporal and spatial focus can be derived from that data to provide more accurate predictions. ● Presence of ADS-B data can further increase the efficiency of system making it reach global audience and live time tracking of flights..

3.4 PROBLEM SOLUTION FIT:

Project Title: Developing a flight delay model using Machine Learning **Project Design Phase-I - Solution Fit Template** **Team ID:** *PNT2022TMID41699*



4 REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Sign up	sign up through Form sign up through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	user login	login through form
FR-4	forgot password	OTP via email.
FR-5	Data Collection	Flight delay data uploads to the database

4.2 NON-FUNCTIONALREQUIREMENTS:

Following are the non-functional requirements of the proposed solution.

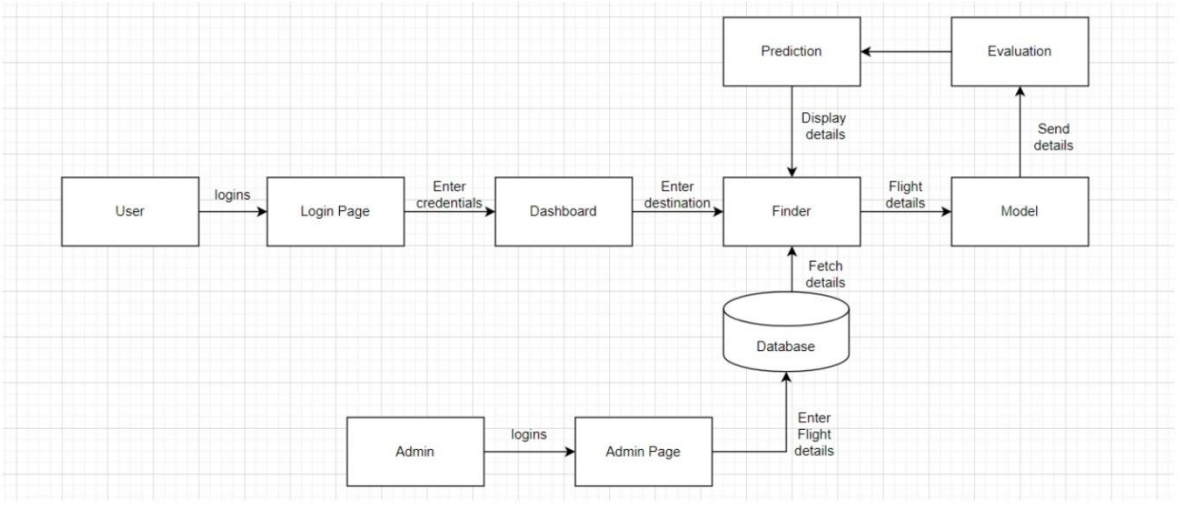
NFR No.	Non-Functional Requirement	Description
NFR-1	Security	If a flight is about to leave and a passenger is still at security the airline decides whether to wait for that passenger or not.
NFR-2	Reliability	The reason you want to arrive to the airport before yourflights original time because flight delays are usually not reliable estimates
NFR-3	Performance	The system should provide accurate delays of the Flight.

NFR-4	Maintainability	The system watching and upkeep should be fundamental and focus in its approach.

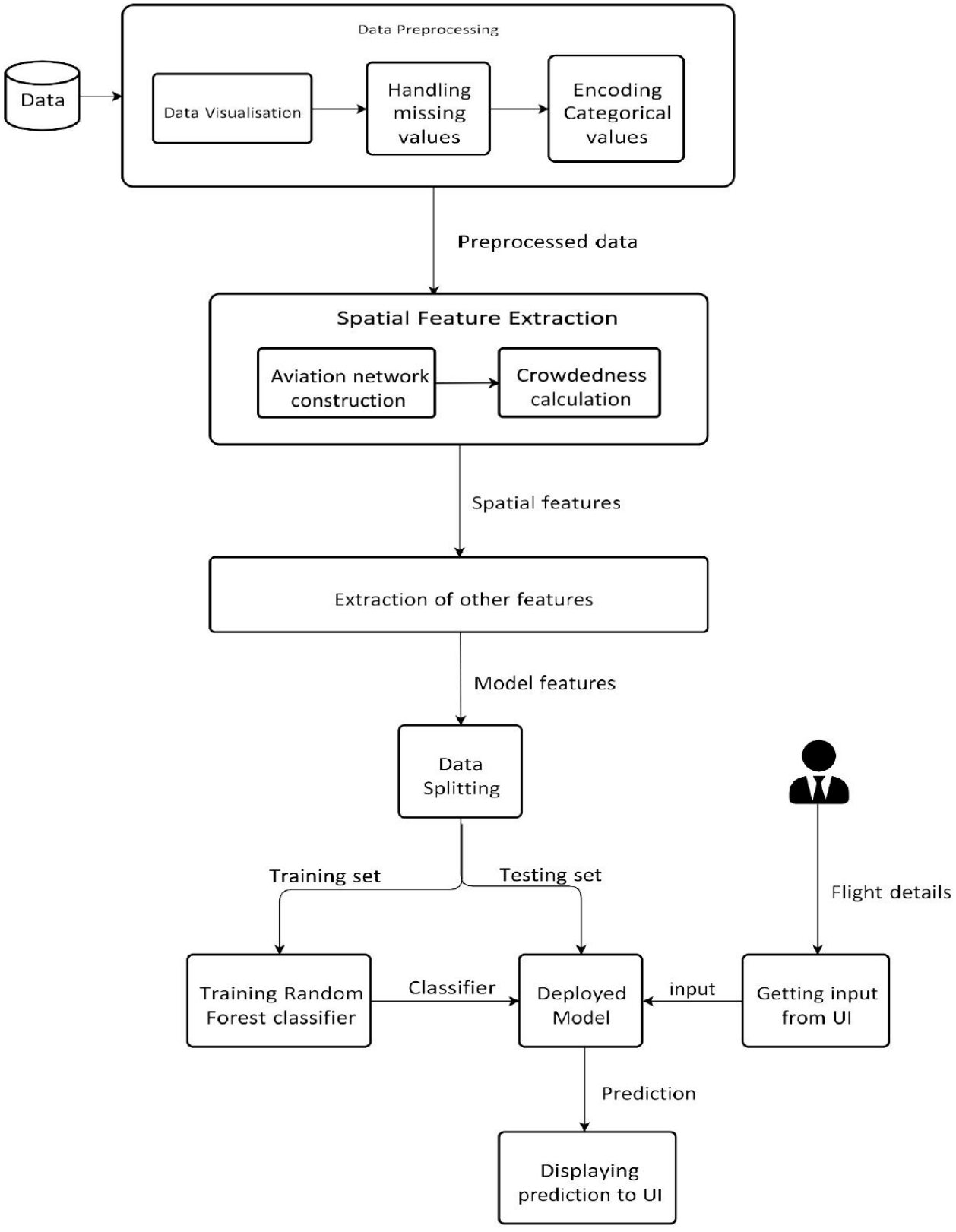
5 PROJECT DESIGN

5.1 Data Flow Diagram:

Data Flow Diagrams:



5.2 Solution & Technical Architecture:



5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile 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 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 Gmail	I can receive confirmation notifications through Gmail	Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password	I can access into my User profile and view details indashboard	High	Sprint-1
	Dashboard	USN-5	As a user, I can send the proper requests todonate and obtain plasma.	I can receive appropriate notifications through email	High	Sprint-1
Customer (Web user)	Login	USN-6	As a user, I can register and log into the application by entering email & password to view the profile	I can access into my User profile and view details indashboard	High	Sprint-1

	Dashboard	USN-7	As a user, I can send the proper requests todonate and obtain plasma.	I can receive appropriate notifications through email	High	Sprint-1
Customer Care Executive	Applicatio n	USN-8	As a customer care executive, I can try to address user's concerns and questions	I can view and address their concerns and questions	Mediu m	Sprint-2
Administr ator	Applicatio n	USN-9	As an administrator I can help with user-facing aspects of a website, like its appearance, navigation and use of media.	I can change appearance friendly manner	Mediu m	Sprint-3
		USN-10	As an administrator, I can involve working withthe technical side of websites.	I can help with such as troubleshooti ng issues, setting up web hosts, ensuring users have access and programming servers	Mediu m	Sprint-1

6 PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

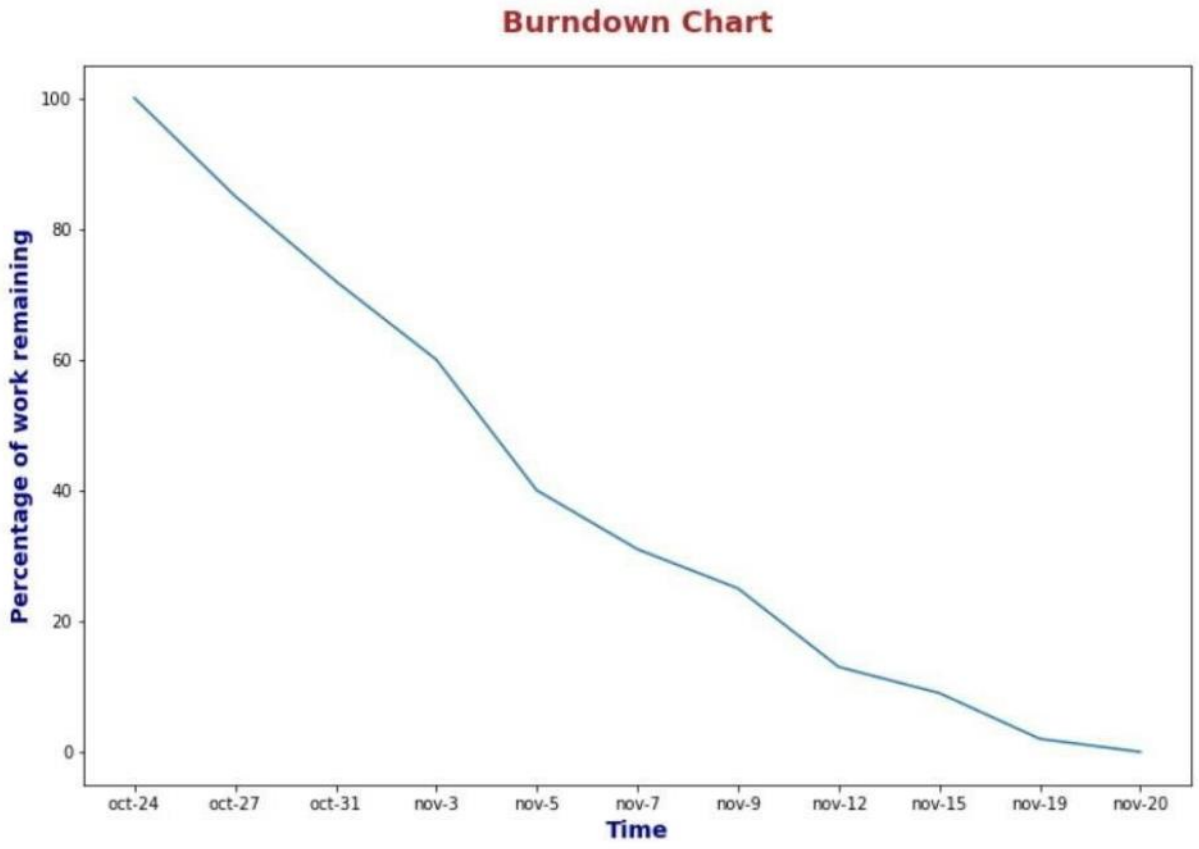
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team members
Sprint-1	Data Collection and Pre-processing	USN-1	As a user, I can't interact anything. Waiting is user's task. User can listen the relationship exist between the various attributes of data by presentation of developer	2	High	Sivaranjani Boomika
Sprint-1	Model Building	USN-2	As a user, I can predict flight delay by various developed ML models by console	1	High	Sivaranjani Boomika
Sprint-2	Model Evaluation	USN-3	As a user, I can predict flight delay by best Model in various developed ML model by console	2	High	Sivaranjani Boomika
Sprint-2	Model Deployment on IBM Cloud using IBM Watson	USN-4	As a user, I can use the model by requesting the deployed model on Cloud	1	Medium	Sivaranjani Boomika
Sprint-2	Basic user interaction Dashboard	USN-5	As a user, I can use the model or prediction from model by interacting with dashboard	2	High	Harini, Gowthami
Sprint-3	Improved Dashboard and GUI	USN-6	As a user, I can use the model or prediction from model by interacting with improved dashboard	1	Medium	Harini, Sneka
Sprint-3	Registration	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Gowthami, Sneka

Sprint-3	Registration	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Harini, Gowthami
Sprint-3	Login	USN-8	As a user, I can log into the application by entering email & password and I can register .login to the application through Gmail	2	Medium	Sneka, Harini
Sprint-4	Raise query/complaint and give feedback	USN-9	As a user, I can raise complaint or query and give feedback	1	Medium	Gowthami, Sneka
Sprint-4	Improve overall web app	USN-10	As a user, I can user revised and improved version of web application	1	High	Sivaranjani, Boomika, Gowthami, Harini, Sneka

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	20	6 Days	24 Oct 2022	29 Nov 2022	20	31 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	07 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA



7 CODING & SOLUTIONING

7.1 FEATURE 1:

Python

It is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.[33]

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming.

It is often described as a "batteries included" language due to its comprehensive standard library.[34][35]

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.[36]

Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020.[37]

Python consistently ranks as one of the most popular programming languages

7.2 FEATURE 2:

Flask

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries.[2]

It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

7.3 Database Schema

IBM Db2 -

a hybrid ANSI-compliant data virtualization tool for accessing, querying and summarizing data across the enterprise which:

• Provides a massively parallel processing (MPP) architecture Exploits Hive, HBase and Apache Spark concurrently for best-in-class analytic capabilities

• Requires only a single database connection or query to connect disparate sources such as HDFS, RDMS, NoSQL databases, object stores and Web HDFS

• Provides low latency support for ad-hoc and complex queries, high performance, and federation capabilities

• Understands dialects from other vendors and various products from Oracle, IBM® Db2® and IBM Netezza®

• Enables advanced row and column security

KUBERNATES-

Kubernetes — also known as “k8s” or “kube” — is a container orchestration platform for scheduling and automating the deployment, management, and scaling of containerized applications.

Kubernetes was first developed by engineers at Google before being open sourced in 2014. It is a descendant of Borg, a container orchestration platform used internally at Google. Kubernetes is Greek for *helmsman* or *pilot*, hence the helm in the Kubernetes logo (link resides outside IBM).

Today, Kubernetes and the broader container ecosystem are maturing into a general-purpose computing platform and ecosystem that rivals — if not surpasses — virtual machines (VMs) as the basic building blocks of modern cloud infrastructure and applications.

This ecosystem enables organizations to deliver a high-productivity Platform-as-a-Service (PaaS) that addresses multiple infrastructure-related and operations-related tasks and issues surrounding cloud-native development so that development teams can focus solely on coding and innovation.

8 TESTING

8.1 TESTING CASE:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product.

It provides a way to check the functional it your components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectation and does not fail in an unacceptable manner.

There are various types of test. Each test type addresses a specific testing requirement.

8.2 ACCEPTANCE TESTING

Acceptance Testing
UAT Execution & Report Submission

1.Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [developing a flight delay prediction using machine learning] project at the time of the release to User Acceptance Testing (UAT).

2 .Defect Analysis

This report execute our user scheduling and their approaches.

Task	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
Login	5	1	2	4	12
Home page	1	1	7	5	17
Model building	1	0	3	0	4
Execute the model	1	0	0	1	2
Flask(app.py)	1	2	2	2	7
Flask(app.py)	0	0	1	0	1
Deploying the model	0	0	1	1	2
Totals	12	4	16	13	45

3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Login	1	0	0	pass
Homepage	1	0	0	pass

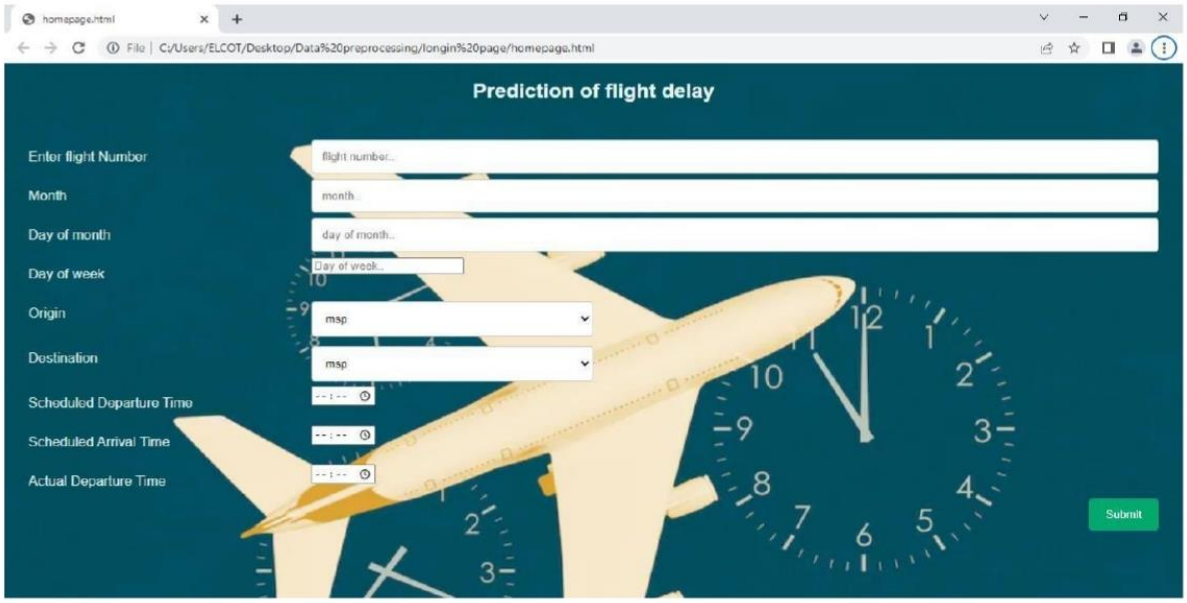
9 RESULTS

9.1 PERFORMANCE METRICS:

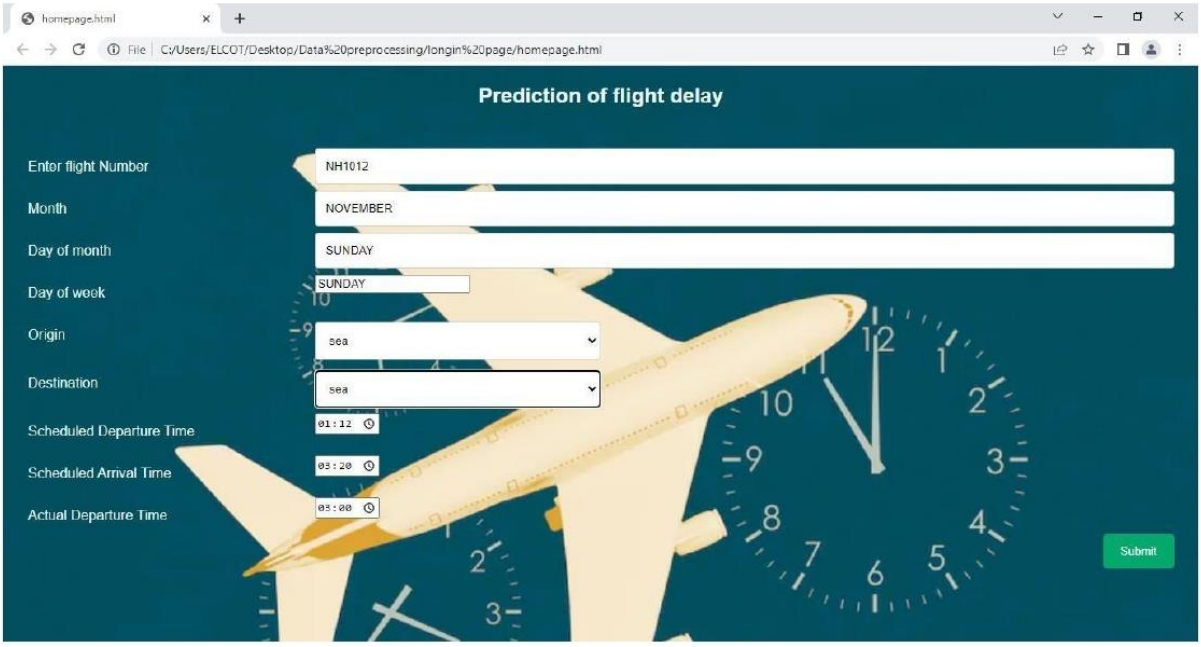
- Project metrics are used to track the progress and performance of a project.
- Monitoring parts of a project like **productivity, scheduling, and scope** make it easier for team leaders to see what's on track.
- As a project evolves, managers need access to changing deadlines or budgets to meet their client's expectations

OUTPUT SCREENS:

Register Page:



While entering details:



10 ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- With this model, we can easily simplify the extensive traffic at the airport and can prevent the major confusions over flight delays.
- This can enable customer satisfaction and incomes of major airlines.
- Accuracy is measured with the previous models and we have analyzed that this model is much more effective in every way.
- The delay prediction model can make the concerned authorities be well prepared for any possible problem.
- The model can easily be understood by a layman: the model is simple and effective.

DISADVANTAGES:

- This model needs to be more compact and flexible. The interoperability feature should be more enhanced.
- The model can be automated instead of manually entering data from the user. Manually entering data is hectic work for the user.

.

11 CONCLUSIONS

In the present world, the major components of any transportation system include passenger airline, cargo airline and air traffic control system. They all face difficulties due to some sort of miscommunication. Our model has been made with the motive of simplifying complex situations due to flight delays and increasing customer satisfaction. With delays being predicted before, the passengers can easily schedule their plans well before. Our model works with an accuracy of 84% and is considered as an efficient model.

\

12 FUTURE ENHANCEMENTS

The project can be extended to a wider range of airports. Current model only supports the data from 5 airports. If the dataset is extended by a vast quantity that has data from airports worldwide then the model can predict any flight delay across the globe. But to do so the complexity of power required will be much greater and the model needs to be trained better to have a higher speed and accuracy of computing results.

13 APPENDIXES

13.1 SOURCE CODE:

Home page:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
h2 {text-align: center;}
* {
box-sizing: border-box;
}
body{
font-family: Arial, Helvetica, sans-serif;

background-image: url(https://www.washingtonpost.com/wp-
apps/imrs.php?src=https://arc-anglerfish-washpost-prod-
washpost.s3.amazonaws.com/public/PBKJ5C6KJJC75BO46RZEWUGL6A.jpg&w

=860);
background-size: cover;
```

```
background-attachment: fixed;
}
input[type=text], select, textarea {
width: 100%;
padding: 12px;

border: 1px solid #ccc;
border-radius: 4px;
resize: vertical;
}

label {
padding: 12px 12px 12px 0;
display: inline-block;
}

input[type=submit] {
background-color: #04AA6D;
color:white;
padding: 12px 20px;
border: none;
border-radius: 4px;
cursor: pointer;
```

```
float: right;
}

input[type=submit]:hover {
background-color: #45a049;
}

.container {
border-radius: 5px;

background-color:transparent;
padding: 20px;
}

.col-25 {
float: left;
width: 25%;
margin-top: 6px;
}

.col-75 {
float: left;
width: 75%;
```

```
margin-top: 6px;
}

/* Clear floats after the columns */
.row:after {
content: "";
display: table;
clear: both;
}

</style>
</head>

<body>

<h2 style="color: #ebf7f3">Prediction of flight delay</h2>

<div class="container">
<form action="/action_page.php">
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Enter flight Number</label>
</div>
```

```
<div class="col-75">
<input type="text" id="fname" numbers="flight number" placeholder="flight
number..">
</div>
</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Month</label>
</div>
<div class="col-75">
<input type="text" id="" name="month" placeholder="month..">
</div>

</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Day of month</label>
</div>
<div class="col-75">
<input type="text" id="" name="month" placeholder="day of month..">
</div>
</div>
<div class="row">
```

```
<div class="col-25">
<label style="color: #ebf7f3">Day of week</label>
</div>
<div class="col-75">
<input type="calender" id="fname" numbers="Day of week"
placeholder="Day of week..">
</div>
</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Origin</label>
</div>
<div class="col-75">

</div>
<div class="col-25">

<select id="country" name="origin">
<option value="region">msp</option>
<option value="region">sea</option>
<option value="region">dtw</option>
<option value="region">jfk</option>
<option value="region">alt</option>
```

```
</select>
</div>
</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Destination</label>
</div>
<div class="col-75">
</div>
<div class="col-25">
<select id="region" name="origin">
<option value="region">msp</option>
<option value="region">sea</option>
<option value="region">dtw</option>
<option value="region">jfk</option>
<option value="region">alt</option>
</select>
</div>
</div>
<div class="row">

<div class="col-25">
<label style="color: #ebf7f3">Scheduled Departure Time</label>
```

```
</div>
<div class="col-75">
<input type="time" id="fname" numbers="predict" placeholder="scheduled
Depatureb Time..">
</div>
</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3"> Scheduled Arrival Time</label>
</div>
<div class="col-75">
<input type="time" id="fname" numbers="predict" placeholder="Arrival
Depatureb Time..">
</div>
</div>
<div class="row">
<div class="col-25">
<label style="color: #ebf7f3">Actual Departure Time</label>
</div>
<div class="col-75">
<input type="time" id="fname" numbers="predict" placeholder="..">
</div>
</div>
```

```
<div class="row">

<input type="submit" value="Submit">
</div>

</form>
</div>

</body>
</html>
```

FLASK:

```
from flask import Flask ,request,render_template
import numpy as np
import pandas as pd
import pickle
import os
model=pickle.load(open('flight.pkl','rb'))
app=Flask( name )
@app.route('homepage.html')
def home():
return render_template("index.html")
@app.route('/prediction',methods=['POST'])
```

```
def predict():
name=request.form['name']
month=request.form['month']
dayofmonth = request .form['dayofmonth']
dayofweek = request.form['origin']
if(origin1=="msp"):
origin1,origin2,origin3,origin4,origin5 = 0,0,0,0,1
if(origin2 == "dtw"):
origin1,origin2,origin3,origin4,origin5= 1,0,0,0,0

if(origin3== "jfk"):
origin1,origin2,origin3,origin4,origin5 = 0,0,1,0,0
if(origin4 == "sea"):
origin1,origin2,origin3,origin4,orgin5 = 0,1,0,0,0
if(origin5 == "alt"):
origin1,origin2,origin3,origin4,origin5 = 0,0,0,1,0

destination = request.form['destination']
if(destination == "msp"):
destination1,destination2, destination3,destination4,destination5 =0,0,0,0,1
if(destination == "dtw"):
destination1,destination2, destination3,destination4,destination5 = 1,0,0,0,0
if(destination == "jfk"):
```

```
destination1,destination2, destination3,destination4,destination5 = 0,0,1,0,0
if(destination == "sea"):
destination1,destination2, destination3,destination4,destination5 =0,1,0,0,0
if(destination == "alt"):
destination1,destination2, destination3,destination4,destination5 =0,0,0,1,0
dept= request.form['dept']
arrtime = request.form['arrtime']
actdept = request.form['actdept']
dept15=int(dept)-int(actdept)
total=('Name,month,day of month,dayofweek,
origin1,origin2,origin3,origin4,origin5,destination1,destination2,
destination3,destination4,destination5')

y_pred = model.predict(total)
print(y_pred)

if (y_pred == [0.1]):
ans="The Flight will be on time"
else:
ans="The Flight will be delayed"
def index():
return render_template('homepage.html'
)
```

13.2 GitHub & Project Demo Link

GitHub link:

https://github.com/IBM-EPBL/IBM-Project-53687-1661488994