PROJECT REPORT

PROJECT TITLE **: DEVELOPING A FLIGHT DELAY PREDICTION MODEL USING MACHINE LEARNING.**

TEAM ID : PNT2022TMID14481.

TEAM MEMBERS : VIDHYAPATHI N (TEAMLEAD)

SAKTHIVEL S

SAIRANJAN S

SREE SIVA GURU A R

1. INTRODUCTION

1.1 Project overview

In the past two decades, air travel has become increasingly popular and has become increasingly accessible to people all over the world. Aviation has evolved to become one of the most important forms of transportation, with its efficiency and reliability making it the preferred choice for long-distance travel. However, flight delays are a major problem in the aviation industry, and they are becoming more and more common. In the United States, the average delay has increased by 30% since 2000, and the cost of delays has risen to $32 billion per year. There are many factors that can contribute to flight delays, such as weather, air traffic control, and maintenance. However, the most common cause of delays is simply that the plane is not ready to take off on time. This is usually due to the fact that the plane is not fully loaded with passengers, baggage, and fuel. It can also be due to technical problems with the plane itself. The goal of this project is to develop a machine learning model that can predict flight delays. The model will be trained on a dataset of historical flight data, and it will be used to predict the delay of a flight before it even takes off.

1.2 Purpose

The purpose of this project is to develop a machine learning model that can predict flight delays. The model will be trained on a dataset of flight information, and will be used to predict the arrival delay of flights. The project is divided into two parts:

1. Data pre-processing and feature engineering

2. Model training and testing

In the first part, the data will be pre-processed and features will be engineered. This part will be focused on cleaning the data and making sure that the

features are suitable for training the machine learning model. In the second part, the machine learning model will be trained and tested. This part will focus on tuning the model to get the best performance possible.

2. LITERATURE SURVEY

2.1 Existing problem

Airlines, airports, and passengers would all benefit from a more accurate flight delay prediction model. Currently, models used by airlines to predict flight delays are based on historical data and do not take into account real-time data such as weather conditions. This can lead to delays and cancellations, as well as increased costs for airlines.

2.2 References

[1] Khaksar, H., & Sheikholeslami, A. (2017). Airline delay prediction by machine learning algorithms. *Scientia*

[2] Esmaeilzadeh, E., & Mokhtarimousavi, S. (2020). Machine learning approach for flight departure delay prediction and analysis. *Transportation Research Record: Journal of the Transportation Research Board*, *2674*(8), 145–159.

[3] M. Al-Tabbakh, S., M. Mohamed, H., & H. El, Z. (2018). Machine learning techniques for analysis of Egyptian flight delay. *International Journal of Data Mining & Knowledge Management Process*, *8*(3), 01–14’

[4] Ye, B., Liu, B., Tian, Y., & Wan, L. (2020). A methodology for predicting aggregate flight departure delays in airports based on supervised learning. *Sustainability*, *12*(7), 2749.

[5] ATLIOĞLU, M. C., BOLAT, M., ŞAHİN, M., TUNALI, V., & KILINÇ, D. (2020). Supervised learning approaches to flight delay prediction. *Sakarya University Journal of Science*.

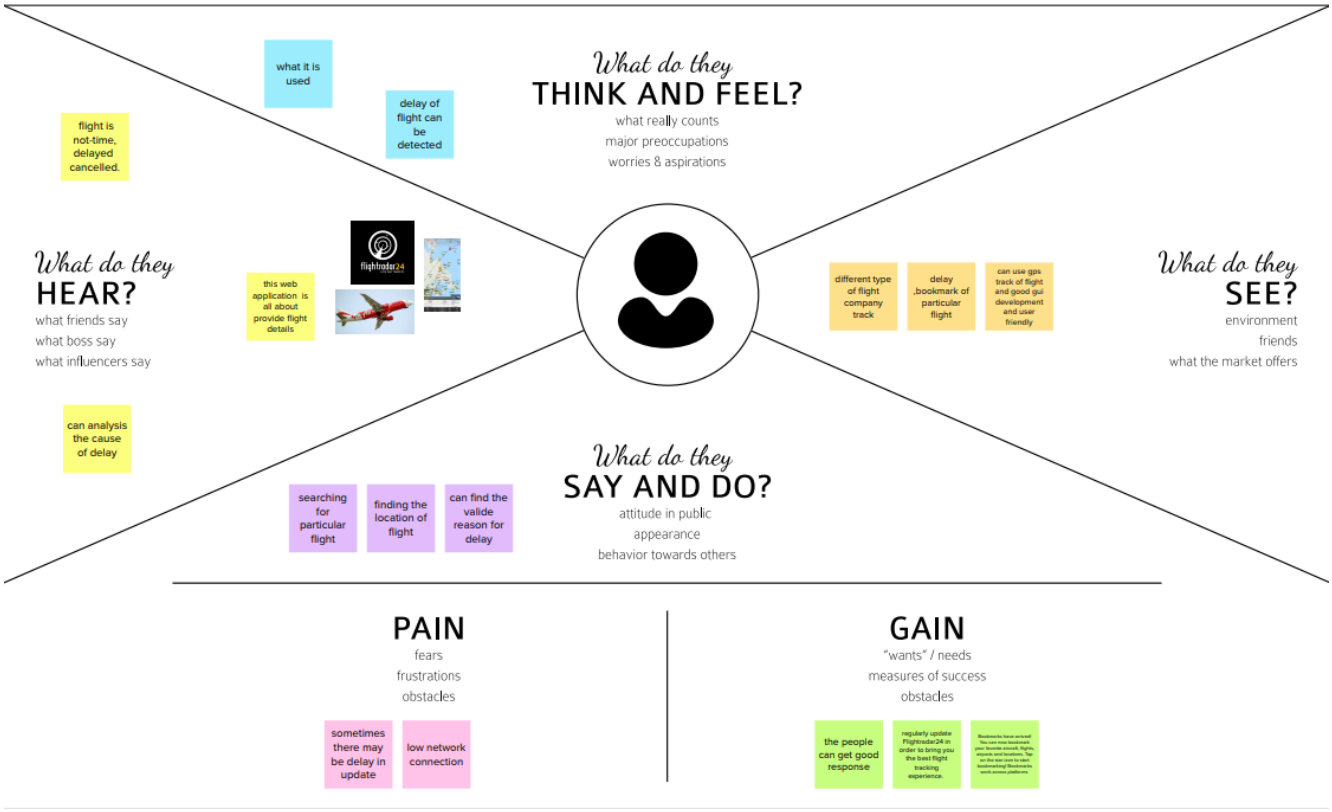
[6] Yu, B., Guo, Z., Asian, S., Wang, H., & Chen, G. (2019). Flight delay prediction for commercial air transport: A deep learning approach. *Transportation Research Part E: Logistics and Transportation Review*, *125*, 203–221.

2.3 Problem Statement Definition

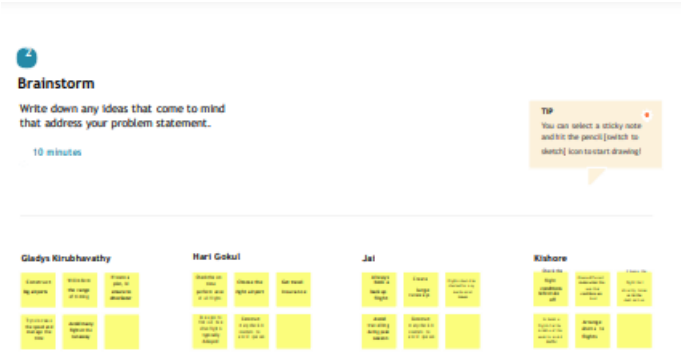
The main objective of the model is to predict flight delays accurately in order to optimize flight operations and minimize delays. These delays are responsible for large economic and environmental losses.

3. IDEATIOIN AND PROPOSED SOLUTION

3.1 Empathy Map Canvas



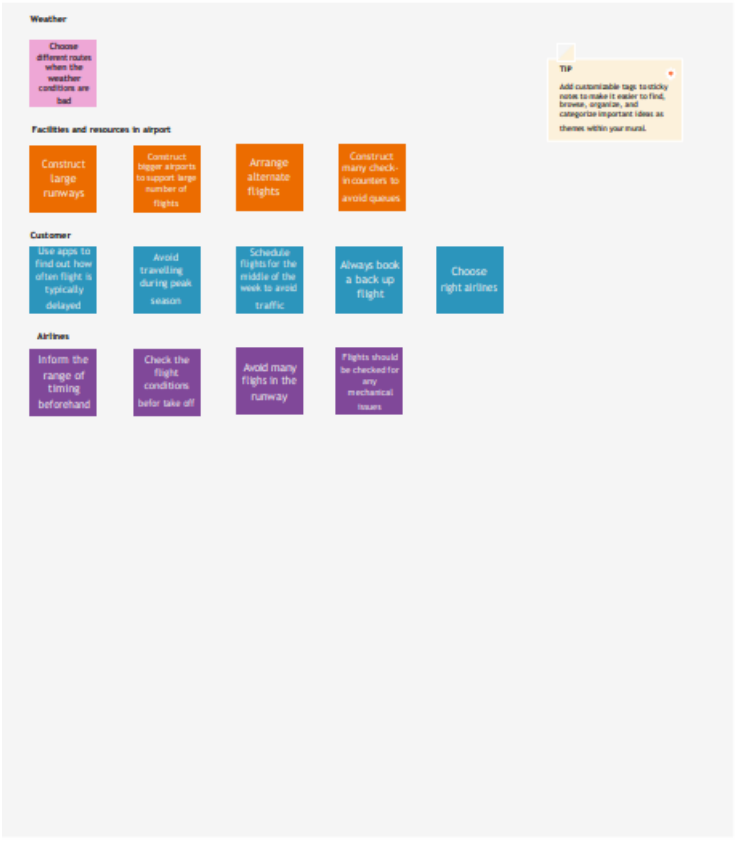
3.2 Ideation and Brainstroming

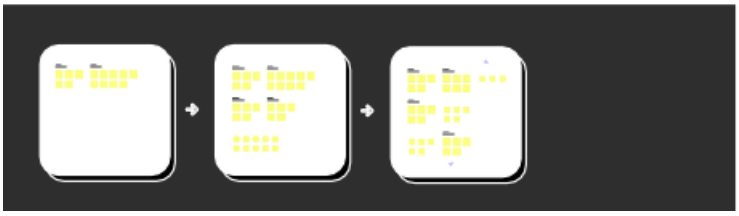


Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

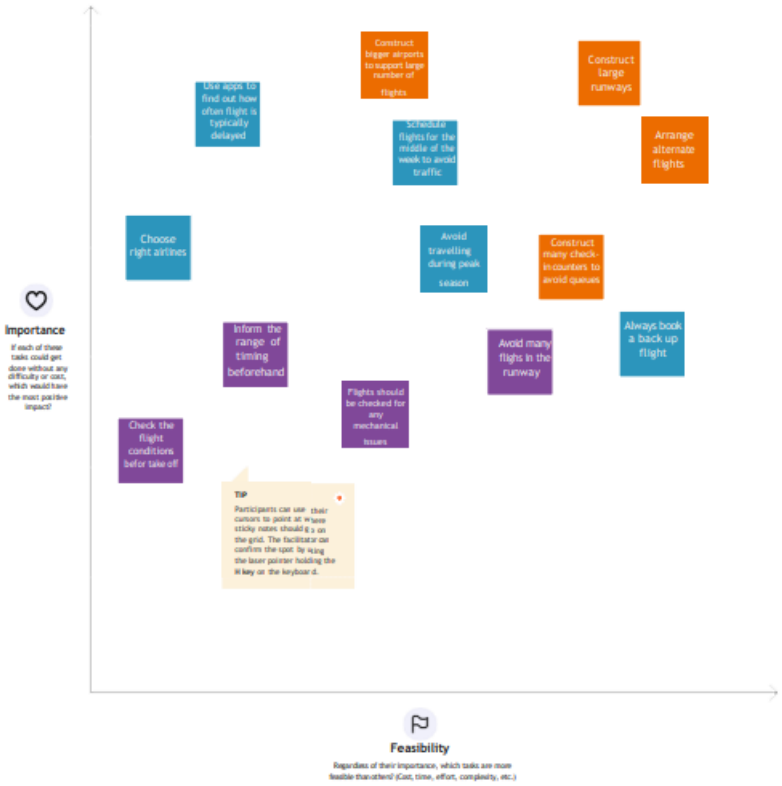




Prioritize

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



After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template

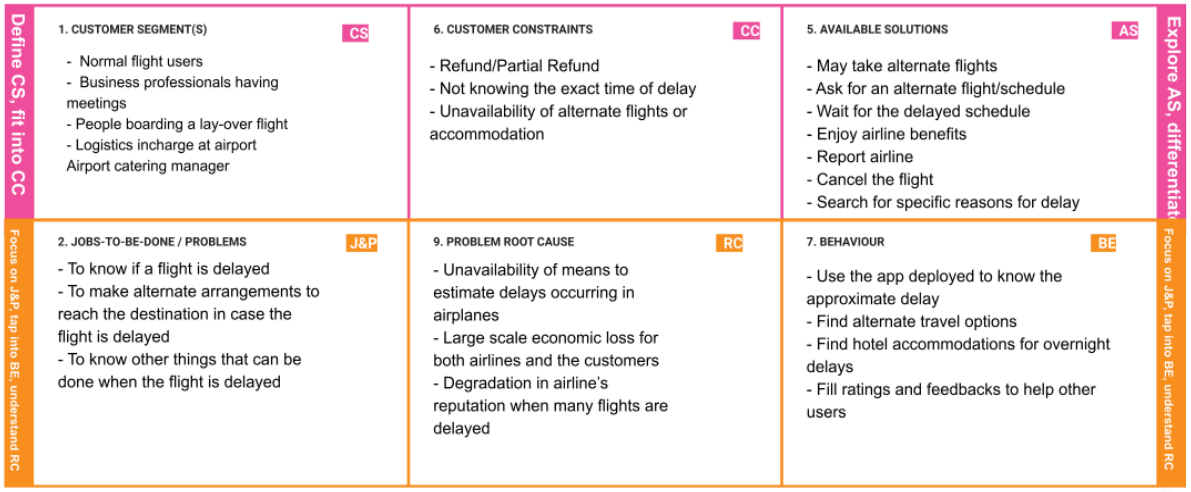
Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template

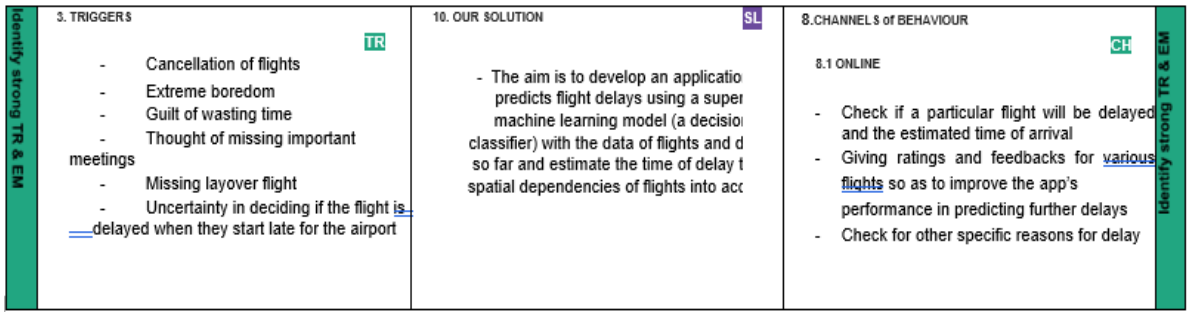
Share template feedback

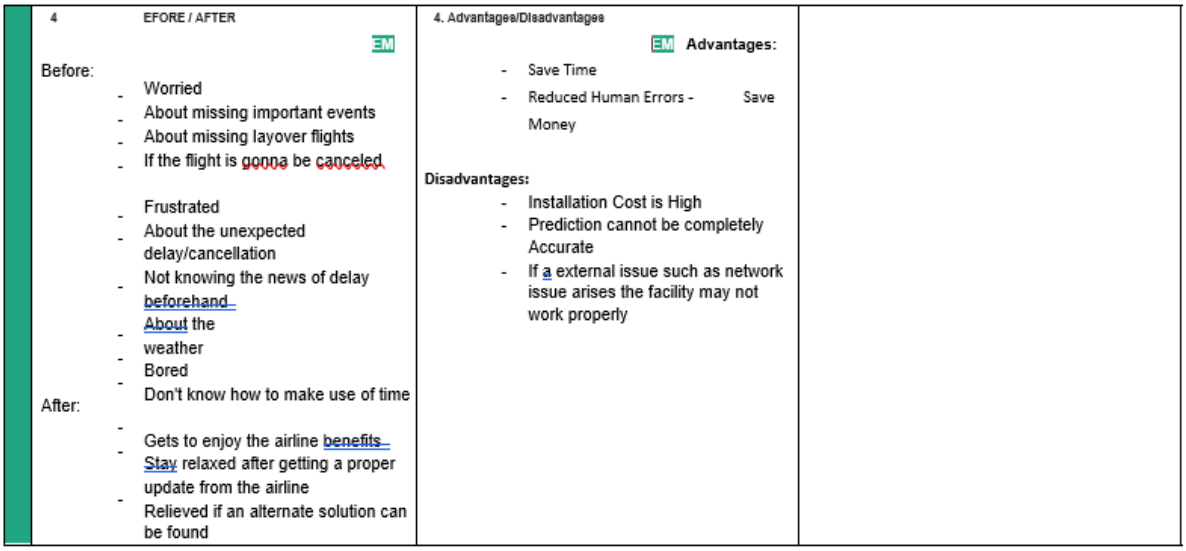
3.3 Proposed Solution

S.No	Parameter	Description
1.	Problem Statement	The main objective of the model is to predict flight delays accurately in order to optimize flight operations and minimize delays. These delays are responsible for large economic and environmental losses.
2.	Idea/Solution description	Using a machine learning model, we can predict flight arrival delays. A flight is considered to be delayed when the difference between scheduled and actual arrival times is greater than 15 minutes.
3.	Novelty/Uniqueness	A user friendly app that provides accurate predictions of the delay time which can be easel accessible.
4.	Social Impact/Customer Satisfaction	Predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy.
5.	Business Model (Revenue Model)	Make revenue from commercial advertisements and sell the model to airline companies.
6.	Scalability of the solution	This model can handle any number of inputs and provide the respective outputs.

3.4 Problem Solution Fit







4.REQUIREMENT ANALYSIS

4.1 Functional requirement

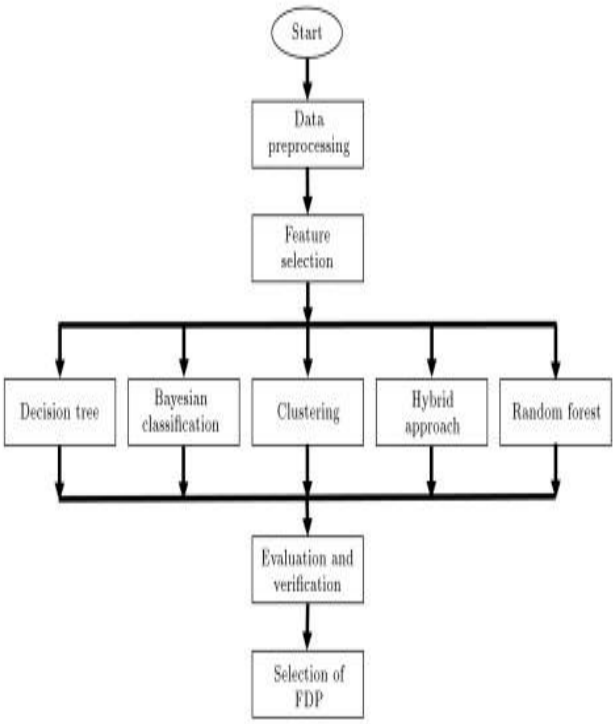
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User registration & login	Registration &login of passengers via Google with email id and password
FR-2	Detailed arrival and departure time of flights	With the flight no and name, the passenger cam see thedetails (time, boarding station, etc)of his/her in the dashboard.
FR-3	Intimate the accurate flight timings to passengers	With the help of various machine learning algorithms, when given the right input features (actual arrival time & departure time, scheduled time, etc) we can predictthe delay in time of the flight which will also be shown in the dashboard and updated time-to-time.
FR-4	Airline helpdesk provide alternatives	The contact details of different airlines will be provided,The passenger will also be able to look for any alternative flight in case the flights get cancelled.
FR-5	Passenger feedback	The feedback will be got from the users or how theapplication was to use, with their feedback and suggestions, we can improve the application further.

4.2 Non-Functional requirement

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application will have an easy-to-use GUI. Users will find it simple to comprehend and utilize all the capabilities of the application.
NFR-2	**Security**	The technique known as database replication will be utilised for the application security to ensure the safety of all crucial data
NFR-3	**Reliability**	The application will be consistent in all scenarios and work without fail in any environment
NFR-4	**Performance**	The applications response time is direct &faster which is determined by the efficiency of the implemented machine algorithm.
NFR-5	**Availability**	The application will be accessible to users 24 hours a day,7 days a week without interruption. They can access it from any part of the world with proper internet.
NFR-6	**Scalability**	The application will be able to handle a rise in the no.of users & generate higher versions.

5.PROJECT DESIGN

5.1 Data Flow Diagram



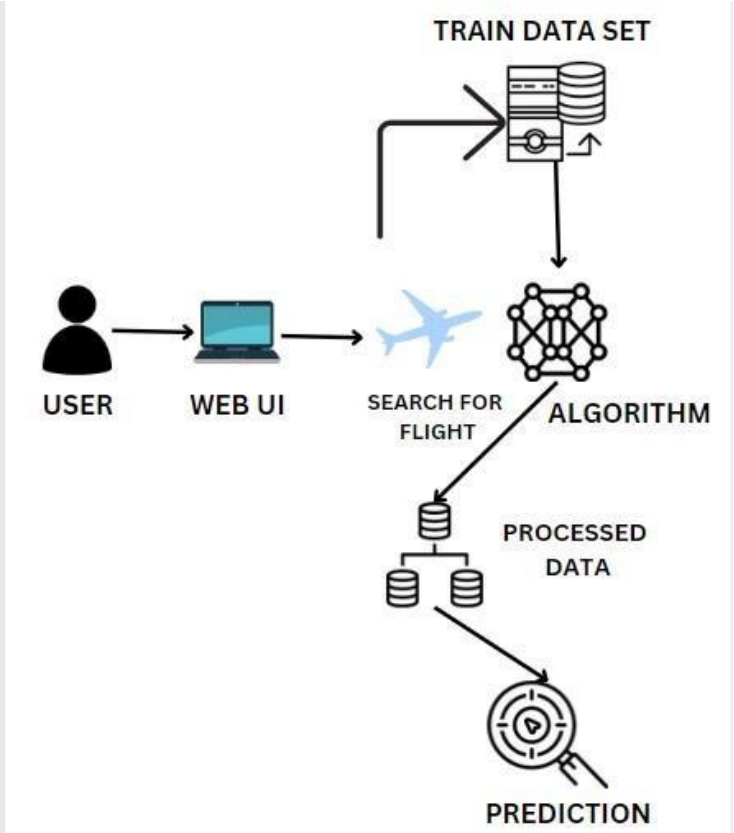
5.2 Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.

- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

Solution Architecture Diagram:



5.3 User Stories

User Type	Functional requirement	User story number	User story/task	Acceptance criteria	Priority	Release
Customer (Mobile user, Web user, Care executive, Administrator)	Registration	USN-1	As a user, I can register for the application by entering my mail, 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
	Dashboard	USN-3	As a user, I can register for the application through internet	I can register & access the dashboard with Internet login	Low	Sprint-2
		USN-4	As a user, I can register for the	I can confirm the registration in	Medium	Sprint-1

			application through Gmail	Gmail		
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login with my id and password	High	Sprint-1

6.PROJECT PLANNING &SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story
Sprint-1	Data Collection and Preprocessing	USN-1	As a user, I am unable to engage with anything.	
Sprint-1	Build HTML Pages	USN-2	As a user, I can view the web pages to enter flight details.	
Sprint-2	Build Python Pages	USN-3	As a user, I am unable to engage with anything.	
Sprint-2	Execute And Test Your Model	USN-4	As a user, I can predict flight delays using the best created ML models.	

Sprint-3	Train The ML Model	USN-6	As a user, I can predict flight delays using the best created ML models.	
Sprint-3	Integrate Flask with Model	USN-5	As a user, I can predict flight delays using the user interface.	
Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story**
Sprint-4	Model Deployment on IBM Cloud using IBM Watson	USN-8	As a user, I can use the model by requesting the deployed model on Cloud.	

7.CODING & SOLUTIONING

7.1 Feature 1

- IBM Watson Platform
- Node red
- Web UI
- Python code-Flask
- HTML

- CSS

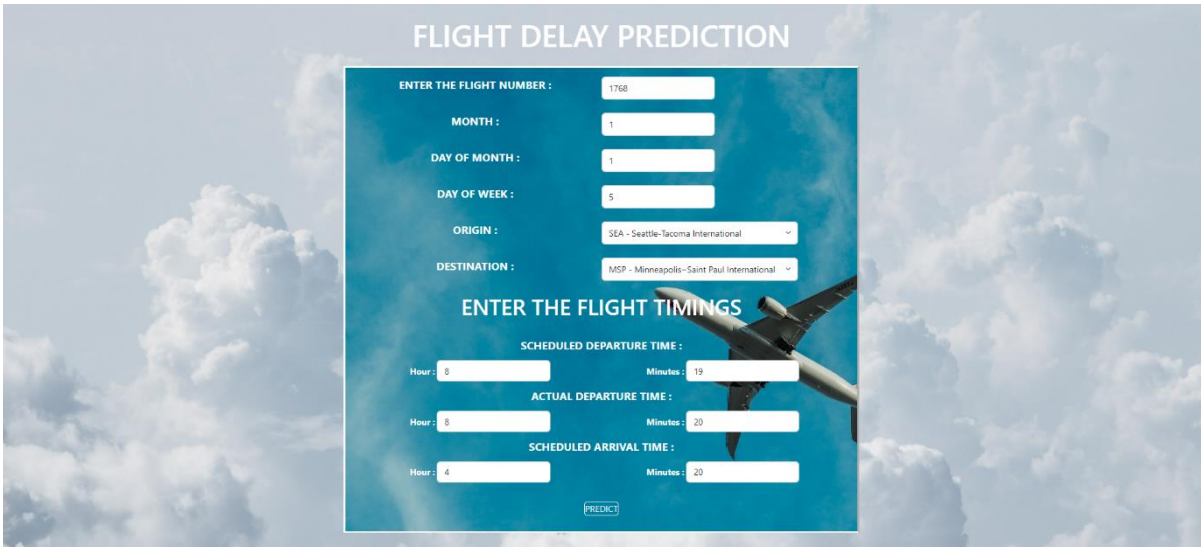
7.2 Feature 2

- Data entry page
- Prediction result page

8.TESTING AND RESULTS

8.1 Test Cases

Test case 1:



Test case 2:

9.ADVANTAGES

- Machine learning can predict flight delays with a high degree of accuracy.
- Machine learning can help identify causes of flight delays.
- Machine learning can help reduce the number of flight delays.
- Machine learning can help improve the efficiency of airport operations.

10.DISADVANTAGES

- Machine learning models can be complex and difficult to understand.
- Machine learning models require a large amount of data to train and can be time-consuming to develop.
- Machine learning models can be prone to overfitting, meaning they may not generalize well to new data.
- Machine learning models can be expensive to develop and maintain.

11.CONCLUSION

In this project, we use flight data, weather, and demand data to predict flight departure delay. Our result shows that the Random Forest method yields the best performance compared to the SVM model. Somehow the SVM model is very time consuming and does not necessarily produce better

results. In the end, our model correctly predicts 91% of the non-delayed flights. However, the delayed flights are only correctly predicted 41% of time. As a result, there can be additional features related to the causes of flight delay that are not yet discovered using our existing data sources.

In the second part of the project, we can see that it is possible to predict flight delay patterns from just the volume of concurrently published tweets, and their sentiment and objectivity. This is not unreasonable; people tend to post about airport delays on Twitter; it stands to reason that these posts would become more frequent, and more profoundly emotional, as the delays get worse. Without more data, we cannot make a robust model and find out the role of related factors and chance on these results. However, as a proof of concept, there is potential for these results. It may be possible to routinely use tweets to ascertain an understanding of concurrent airline delays and traffic patterns, which could be useful in a variety of circumstances.

12.FUTURE SCOPE

This project is based on data analysis from year 2008. A large dataset is available from 1987-2008 but handling a bigger dataset requires a great amount of preprocessing and cleaning of the data. Therefore, the future work of this project includes incorporating a larger dataset. There are many different ways to preprocess a larger dataset like running a Spark cluster over a server or using a cloud-based services like AWS and Azure to process the data. With the new advancement in the field of deep learning, we can use Neural Networks algorithm on the flight and weather data. Neural Network works on the pattern matching methodology. It is divided into three basic parts for data modelling that includes feed forward networks, feedback networks, and selforganization network. Feed-forward and feedback networks are generally used in the areas of prediction, pattern recognition, associative memory, and optimization calculation, whereas self-organization networks are generally used in cluster analysis. Neural Network offers distributed computer architecture with important learning abilities to represent nonlinear relationships.

Also, the scope of this project is very much confined to flight and weather data of United States, but we can include more countries like China, India, and Russia. Expanding the scope of this project, we can also add the flight data from international flights and not just restrict our self to the domestic flights.

13.APPENDIX

13.1 Source Code

```
from flask import Flask,render_template,request

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
```

```
# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "2SNGxCC84_SnT4w-CK18BSgHa22dH7hgM673se9fq57B"
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.route('/')
def index():
    return render_template('index.html')

@app.route('/prediction',methods=["POST"])
def predict():
    if request.method=="POST":
        name=request.form["name"]
        month=request.form["month"]
        if(int(month)>12):
            ans="Please Enter the correct Month"
            return render_template("index.html" ,y=ans)

        dayofmonth=request.form["dayofmonth"]
        if(int(dayofmonth)>31):
```

```
    ans="Please Enter the correct Day of Month"
    return render_template("index.html" ,y=ans)

dayofweek=request.form["dayofweek"]
if(int(dayofweek)>7):
    ans="Please Enter the correct Day of Week"
    return render_template("index.html" ,y=ans)



origin=request.form["origin"]
destination=request.form['destination']

if(origin==destination):
    ans="Origin airport and destination airport can't be same"
    return render_template("index.html" ,y=ans)

if(origin=="msp"):
    origin1,origin2,origin3,origin4,origin5=0,0,0,1,0
if(origin=="dtw"):
    origin1,origin2,origin3,origin4,origin5=0,1,0,0,0
if(origin=="jfk"):
    origin1,origin2,origin3,origin4,origin5=0,0,1,0,0
if(origin=="sea"):
    origin1,origin2,origin3,origin4,origin5=0,0,0,0,1
if(origin=="alt"):
    origin1,origin2,origin3,origin4,origin5=1,0,0,0,0
```

```
        if(destination=="msp"):

destination1,destination2,destination3,destination4,destination5=0,0,0,1,0
        if(destination=="dtw"):

destination1,destination2,destination3,destination4,destination5=0,1,0,0,0
        if(destination=="jfk"):

destination1,destination2,destination3,destination4,destination5=0,0,1,0,0
        if(destination=="sea"):

destination1,destination2,destination3,destination4,destination5=0,0,0,0,1
        if(destination=="alt"):

destination1,destination2,destination3,destination4,destination5=1,0,0,0,0


        depthr=request.form['depthr']
        deptmin=request.form['deptmin']
        if(int(depthr)>23 or int(deptmin)>59):
            ans="Please enter the correct Departure time"
            return render_template("index.html" ,y=ans)
        else:
            dept=depthr+deptmin


        actdepthr=request.form['actdepthr']
```

```
    actdeptmin=request.form['actdeptmin']
    if(int(actdepthr)>23 or int(actdeptmin)>59):
        ans="Please enter the correct Actual Departure time"
        return render_template("index.html" ,y=ans)
    else:
        actdept=actdepthr+actdeptmin



    arrtimehr=request.form['arrtimehr']
    arrtimemin=request.form['arrtimemin']
    if(int(arrtimehr)>23 or int(arrtimemin)>59):
        ans="Please enter the correct Arrival time"
        return render_template("index.html" ,y=ans)
    else:
        arrtime=arrtimehr+arrtimemin



    if((int(actdept)-int(dept))<15):
        dept15=0
    else:
        dept15=1

    print(dept15)

total=[[month,dayofmonth,dayofweek,origin1,origin2,origin3,origin4,origin5,d
```

```
estination1,destination2,destination3,destination4,destination5,dept,actdept,dep
t15,arrtime]]


    # NOTE: manually define and pass the array(s) of values to be scored in the
next line
    payload_scoring = {"input_data": [{"fields":
["f0","f1","f2","f3","f4","f5","f6","f7","f8","f9","f10","f11","f12","f13","f14","
f15","f16"], "values": total}]}


    response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/74fb0eec-a7f5-4bb6-ab8b-
d423e91a872c/predictions?version=2022-11-16', json=payload_scoring,
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    print(response_scoring.json())




    pred = response_scoring.json()
    value = pred['predictions'][0]['values'][0][0]


    print(value)
    if(value==[0.]):
       ans="THE FLIGHT WILL BE ON TIME"
    else:
       ans="THE FLIGHT WILL BE DELAYED"
```

```
    return render_template("results.html" ,y=ans)


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

13.2 Github Link

Github Link : https://github.com/IBM-EPBL/IBM-Project-10948-1659246792

Project demo Link : https://github.com/IBM-EPBL/IBM-Project-10948-1659246792/blob/main/FINAL%20DELIVARABLES/DemoVideo.mp4