PROJECT REPORT

DEVELOP ING A FLIGHT DELAY PREDICTION MODEL USING MACHINE LEARNING

Submitted by

PNT2022TMID07489

Santhosh .T - 721719104072

Pradeep surya .R - 721719104064

Palanisamy .V - 721719104061

Nishanth . M.L - 721719104058

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 PROJECT OVERVIEW

Machine learning and deep learning play an important role in computer technology and artificial intelligence. With the use of deep learning and machine learning, human effort can be reduced in recognizing, learning, predictions and in many more areas.

Ensemble learning is embedded in the developed model to perform a supervised fine-tuning within the presented predictive architecture. The proposed method has proven to be highly capable of handling the challenges of large datasets and capturing the key factors influencing delays. This ultimately enables connected airports to collectively alleviate delay propagation within their network through collaborative efforts (e.g., delay prediction synchronization).

1.2PURPOSE

Flight delay prediction systems are capable of predicting the delay from different sources like airports, heliports etc… to alleviate delay propagation within their network through collaborative efforts ,to let people know the information about change in the flight schedule and to plan their time accordingly.

CHAPTER 2
LITERATURE SURVEY

2.1 EXISTING PROBLEM

The traditional approaches can model the delay propagation of air traffic operation, but they lack the ability to analytically analyze the huge volume of traffic and weather dataset. On the other side, the machine learning approaches has the ability to discover the hidden patterns in the data, but they lack the inner relationships between different airports, such as delay propagation.

2.2 REFERENCES

Title : Handwritten English Character and Digit Recognition(2021)

Author : Al-Mahmud; Asnuva Tanvin; Sazia Rahman

one of the most sought-after technologies is a handwritten character recognition system. It has the potential to solve a wide range of issues and bring about radical change in our lives. We used Convolutional Neural Networks (CNNs) to recognize handwritten English capital letters and digits in this research. We improved a previously developed CNN architecture by adjusting hyperparameters and minimizing the model's overfitting. The MNIST digit dataset is used to evaluate the experiments, which are then compared to different methods. On the MNIST dataset, 99.47 percent test accuracy was attained , which is superior to other approaches. The research was then expanded upon by the addition of a new dataset for recognizing English capital letters. 98.94 percent accuracy was achieved on this extended dataset.

Title**: Image Classification using Deep Learning: An Experimental Study on Handwritten Digit Recognition**

Author : Mukesh Kumar Rohil; Raju Singh

This paper presents an experimental study of the use of Deep Learning using Convolution Neural Networks (CNNs) for Image Classification. Specially, the problem being addressed here is of recognition of handwritten digits. The objective is to report variations in testing errors and accuracies with varying kernel size and varying number of feature maps. We performed handwritten digit classification using neural network and deep learning for a subset from the MNIST dataset, which contains 60,000 training images and 10,000 test images in all. It is observed that the accuracy and loss are stabilizing with minor change in the kernel size and the number of feature maps.

Title : **Real Time Handwritten Digits Recognition Using Convolutional Neural Network.**

Author:Kaveti Upender; Venkata Siva Kumar Pasupuleti.

Reading handwritten information like examination answer sheets is still a difficult task for many of us, because each one of us is having a different interpretation style. As the world is moving towards digitization, converting the handwritten information to a readable digital format reduces the difficulty. This approach will be beneficial for the readers as it gives a better understanding of the information. With the help of machine learning and deep learning algorithms, the handwritten patterns can be recognized and classify them accordingly to a digital format with human level accuracy. This research paper deals with predicting the real time handwritten digits only. To classify the handwritten digits MNIST data set is used for training the model. OpenCV python library is used for detecting the patterns in the real time handwritten digits. These detected patterns are predicted to human level accuracy with the help of a Convolutional Neural Network model.

Title: Prediction of weather-induced airline delays based on machine learning algorithms(2019)

Author: Sun Choi; Young Jin Kim; Simon Briceno; Dimitri Mavris and others

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

Title: Flight delay prediction from spatial and temporal perspectiveAlgorithm (2020)

Author: Wang, Yuxiang and Wang, Ruijin and Li, Dongfen and Adu-Gyamfi, Daniel and Tian, Kaibin and Zhu, Yixin

In this paper, we propose a novel prediction framework (ST-Random Forest) for flight delay prediction from temporal and spatial perspective. We first apply complex network theory to extract the spatial feature of the aviation network at edge-, node-, and network-level. Furthermore, considering the temporal correlation of weather condition and airport crowdedness on flight delays, we create a prediction framework based on LSTM units to extract the temporal property of crowdedness and weather condition. Finally, we use the factors (e.g., spatial, temporal and extrinsic) that affect flight delays as inputs and apply Random Forest as classifier to predict flight delays. We apply and test our approach in a case study at China domestic flights between Jun and Aug 2016; after evaluation, we find that the accuracy of our proposed model reaches 92.39%. For the on-time samples, approximately 86% are correct identified; for the delayed samples, the classification accuracy reaches 95%. The ST-Random Forest model contributes to aviation authorities and airport regulators by creating real-time monitoring and high accuracy prediction system to alleviate flight delays and providing insightful suggestions to develop effective air traffic control strategies.

Title: Predicting flight delay based on multiple linear regression (2021)

Author: Yi Ding

Delay of flight has been regarded as one of the toughest difficulties in aviation control. How to establish an effective model to handle the delay prediction problem is a significant work. To solve the problem that the flight delay is difficult to predict, this study 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. Experiments based on a realistic dataset of domestic airports show that the accuracy of the proposed model approximates 80%, which is further improved than the Naive-Bayes and C4.5 approach approaches. The result testing shows that this method is convenient for calculation, and also can predict the flight delays effectively. It can provide decision basis for airport authorities.

2.3 PROBLEM STATEMENT DEFINITION

Over the last twenty years, air travel has been increasingly preferred among travellers, mainly because of its speed and in some cases comfort. 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. According to the Bureau of Statistics HOS), hoor 20% of all flights are delayed by 15 minutes or more. Flight delays causes a negative impact, mainly economical for airport arities, commuters and airline industries as well. Therefore, this study develops a novel spatial analysis approach to explore the delay and canal factors which is able to take dependence and the possible problem involved including error correlation and variable lag effect of canal factors on delay into account. The main objective of the model is to predict flight delays accurately in order to optimize flight operations and minimize delays.

Using a machine learning model, we can predict flight arrival delays. The input to our algorithm is rows of feature vector like departure date, departure delay, distance between the two airports, scheduled arrival time etc. We then use decision tree classifier to predict if the flight arrival will be delayed or not. A flight is considered to be delayed when difference between scheduled and actual arrival times is greater than 15 minutes. Furthermore, we compare decision tree classifier with logistic regression and a simple neural network for various figures of merit.

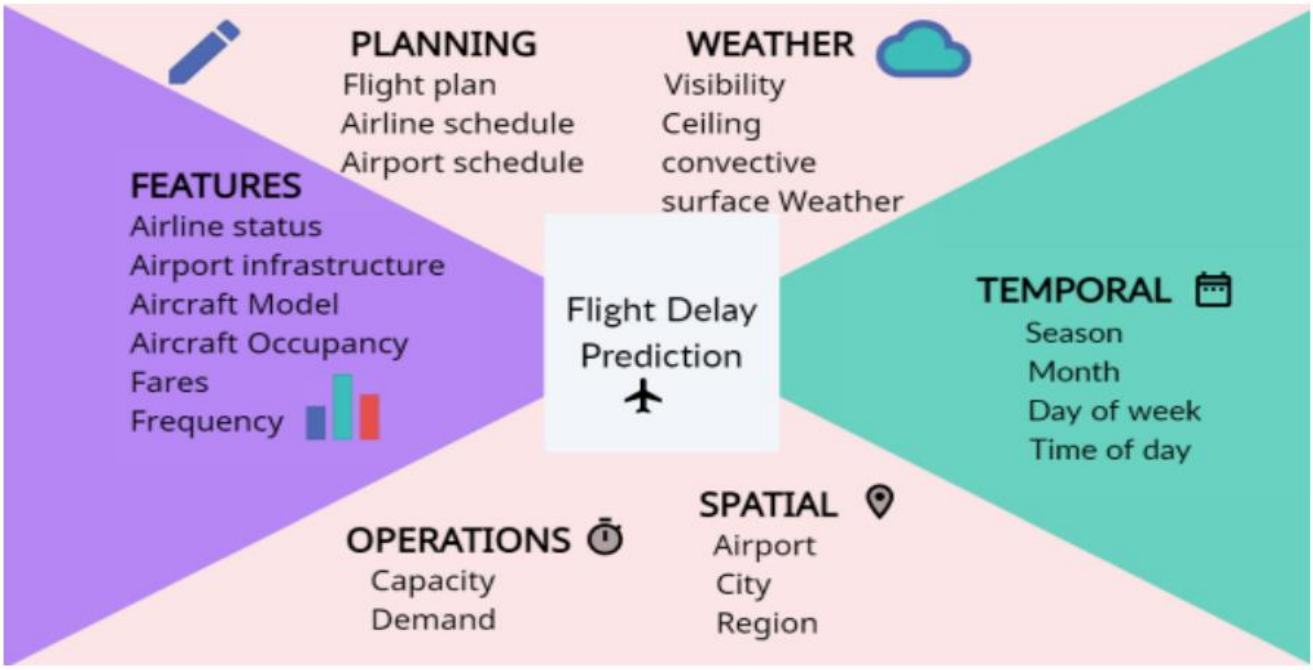
I am (Customer)	I'm trying to	But	Because	Which makes me feel
A Traveler	travel to my native	the flight is getting late	of the unusual weather condition	disappointed.

CHAPTER 3

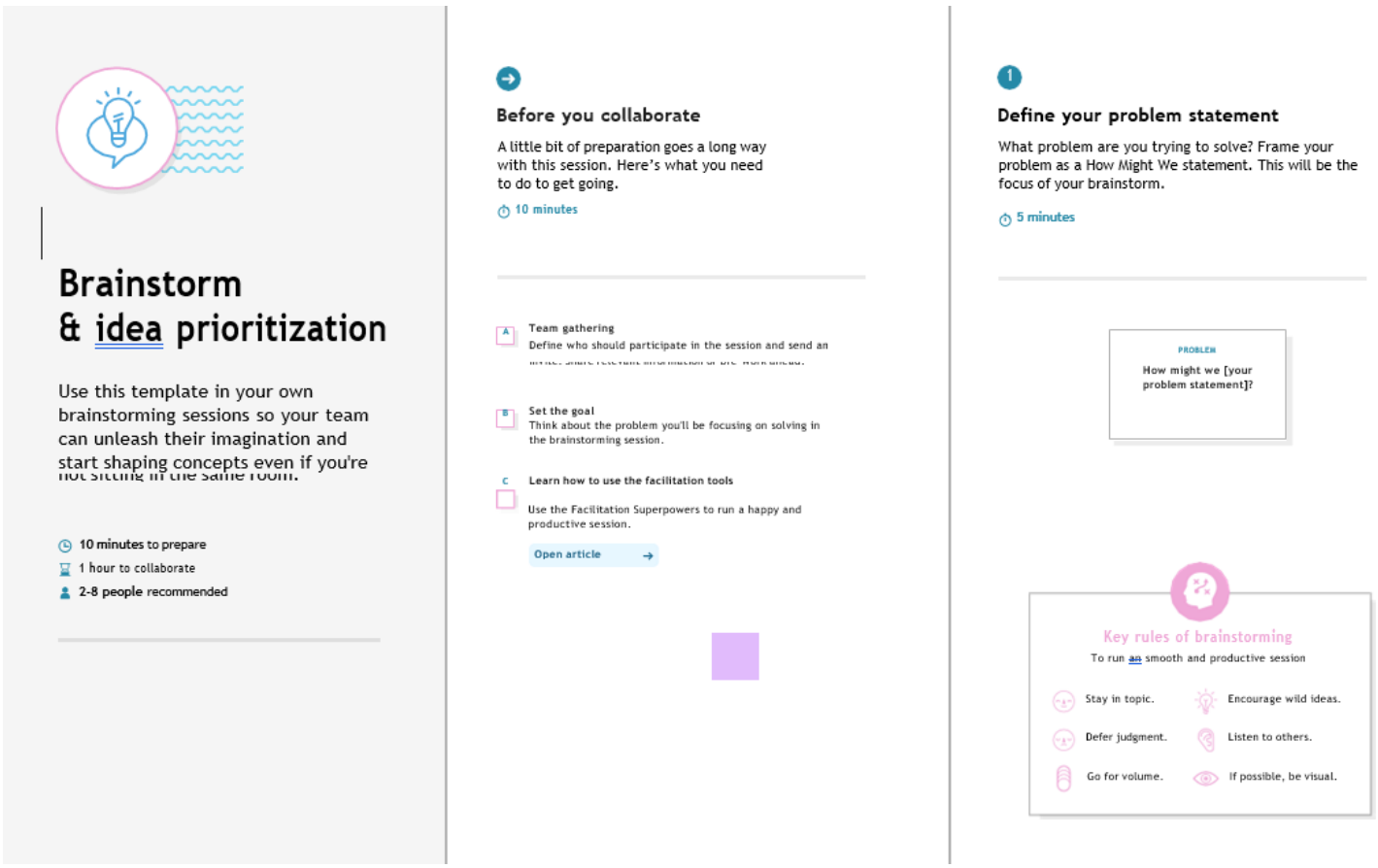
IDEATION AND PROPOSED SOLUTION

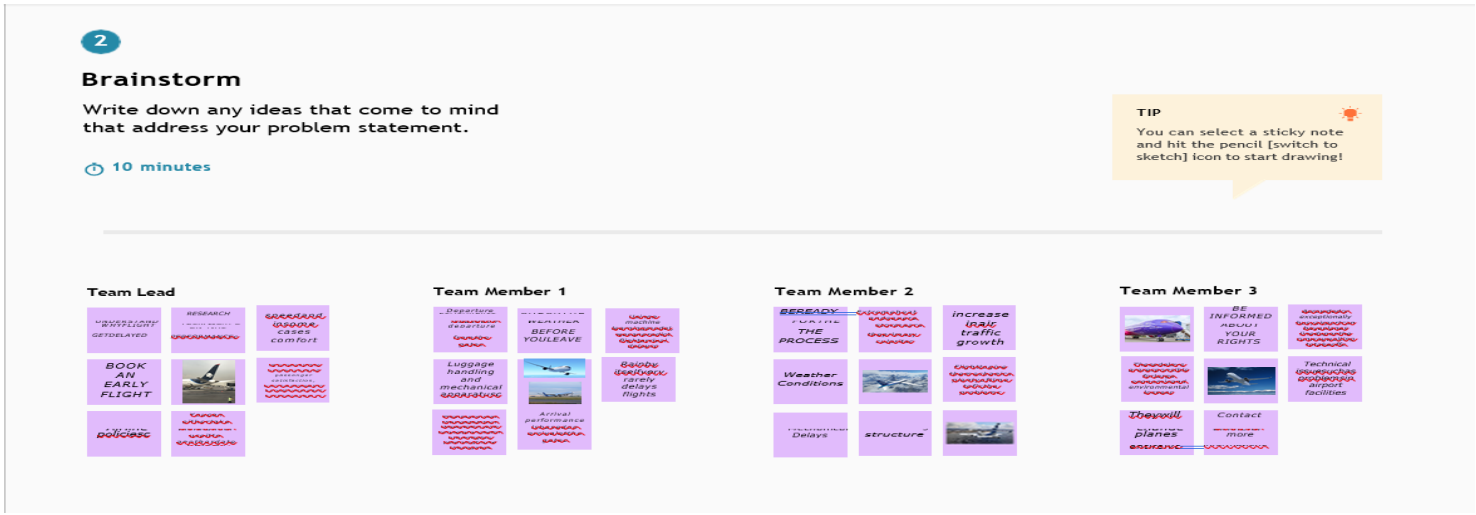
3.1 EMPATHY MAP CANVAS

EMPATHY MAP:



3.2 IDEATION & BRAINSTORMING





3

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

TIP

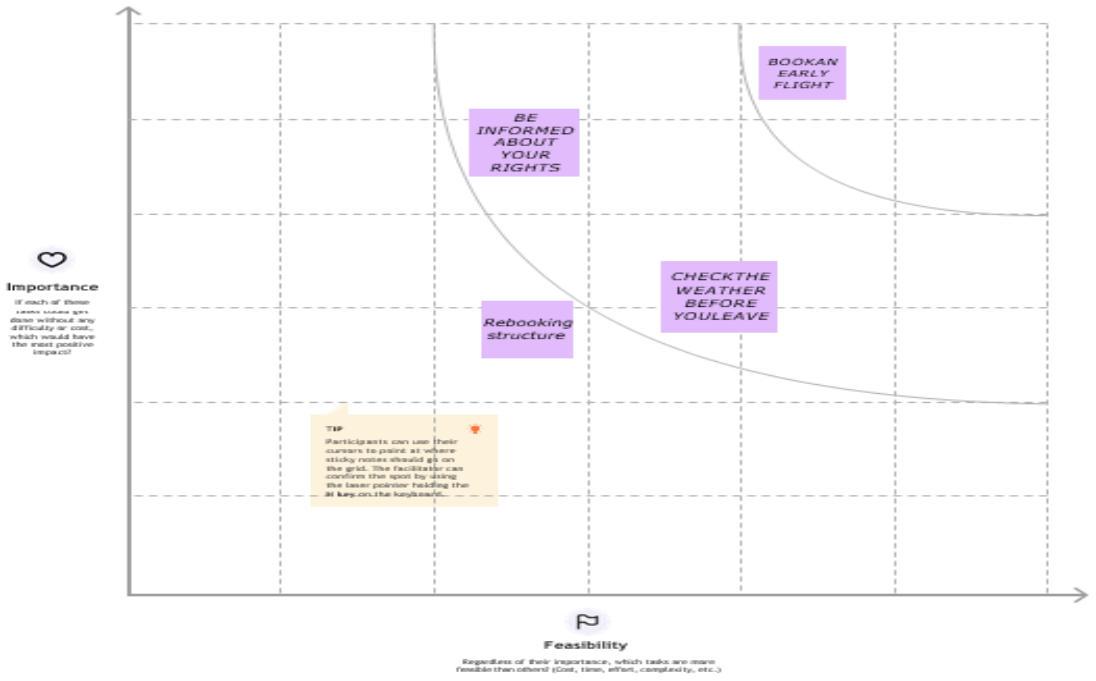
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

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



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

A **Share the mural**
them in the loop about the outcomes of the session.

B **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

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Developing a flight delay prediction model
2.	Idea / Solution description	The main objective of the model is to predict flight delays accurately in order to optimize flight operations and minimize delays. Using a machine learning model, we can predict flight arrival delays. The input to our algorithm is rows of feature vector like departure date, departure delay, distance between the two airports, scheduled arrival time etc. We then use decision tree classifier to predict if the flight arrival will be delayed or not. A flight is considered to be delayed when difference between scheduled and actual arrival times is greater than 15 minutes. Furthermore, we compare decision tree classifier with logistic regression and a simple neural network for various figures of merit.
3.	Novelty / Uniqueness	Object detection using Deep Learning.
4.	Social Impact / Customer Satisfaction	By predicting the flight delay with more accuracy, the optimised results will help the passengers by alerting them, which will not lead them to miss the flight. In the case of the medical field, if a doctor misses a flight, it can cause issues in the life or health of a patient. Our project helps them to stay aware of their flights. 1. Ease of customer to find out available flights. 2. Happiness of customer. 3. Protect the reputation of the airlines. 4. Reduce the extra expenses for the customer and airlines.
5.	Business Model (Revenue Model)	The model used is Paywall(Subscription) where customers will be able to use some features of the
		app for free and other premium features as a subscriber.
6.	Scalability of the Solution	This makes the passengers to take preventive action when the status of the flight is notified and this improves the business value of the passengers, time management, and more.

3.3 PROBLEM SOLUTION FIT

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

- Normal flight users
- Business professionals having meetings
- People boarding a lay-over flight
- Logistics incharge at airport

Airport catering manager

6. CUSTOMER CONSTRAINTS CC

- Refund/Partial Refund
- Not knowing the exact time of delay
- Unavailability of alternate flights or accommodation

5. AVAILABLE SOLUTIONS AS

- May take alternate flights
- Ask for an alternate flight/schedule
- Wait for the delayed schedule
- Enjoy airline benefits
- Report airline
- Cancel the flight
- Search for specific reasons for delay

Explore AS, differentiate

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

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

- To know if a flight is delayed
- To make alternate arrangements to reach the destination in case the flight is delayed
- To know other things that can be done when the flight is delayed

9. PROBLEM ROOT CAUSE RC

- Unavailability of means to estimate delays occurring in airplanes
- Large scale economic loss for both airlines and the customers
- Degradation in airline's reputation when many flights are delayed

7. BEHAVIOUR BE

- Use the app deployed to know the approximate delay
- Find alternate travel options
- Find hotel accommodations for overnight delays
- Fill ratings and feedbacks to help other users

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

Identify strong TR & EM

3. TRIGGERS TR

- Cancellation of flights
- Extreme boredom
- Guilt of wasting time
- Thought of missing important meetings
- Missing layover flight
- Uncertainty in deciding if the flight is delayed when they start late for the airport

10. YOUR SOLUTION SL

- The aim is to develop an application that predicts flight delays using a supervised machine learning model (a decision tree classifier) with the data of flights and delays so far and estimate the time of delay taking spatial dependencies of flights into account.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

- Check if a particular flight will be delayed and the estimated time of arrival
- Giving ratings and feedbacks for various flights so as to improve the app's performance in predicting further delays
- Check for other specific reasons for delay

Identify strong TR & EM

4. EMOTIONS: BEFORE / AFTER EM

Before:

- Worried
  - About missing important events
  - About missing layover flights
  - If the flight is gonna be canceled
- Frustrated
  - About the unexpected delay/cancellation
  - Not knowing the news of delay beforehand
  - About the weather
- Bored
  - Don't know how to make use of time

After:

- Gets to enjoy the airline benefits
- Stay relaxed after getting a proper update from the airline
- Relieved if an alternate solution can be found

8.2 OFFLINE

- Finding alternate travel routes in the airport
- Hotels near the airport can be visit for overnight stays during delays

CHAPTER 4

REQUIREMENT ANALYSIS

4.1FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
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	Book Flights	The flight ticket booking is done and receipt of booking is sent to email of user
FR-6	Request Cancellation	The user wants to cancel the ticket reserved due to delay

4.2 NON FUNCTIONAL REQUIREMENTS

Non-functional Requirements:

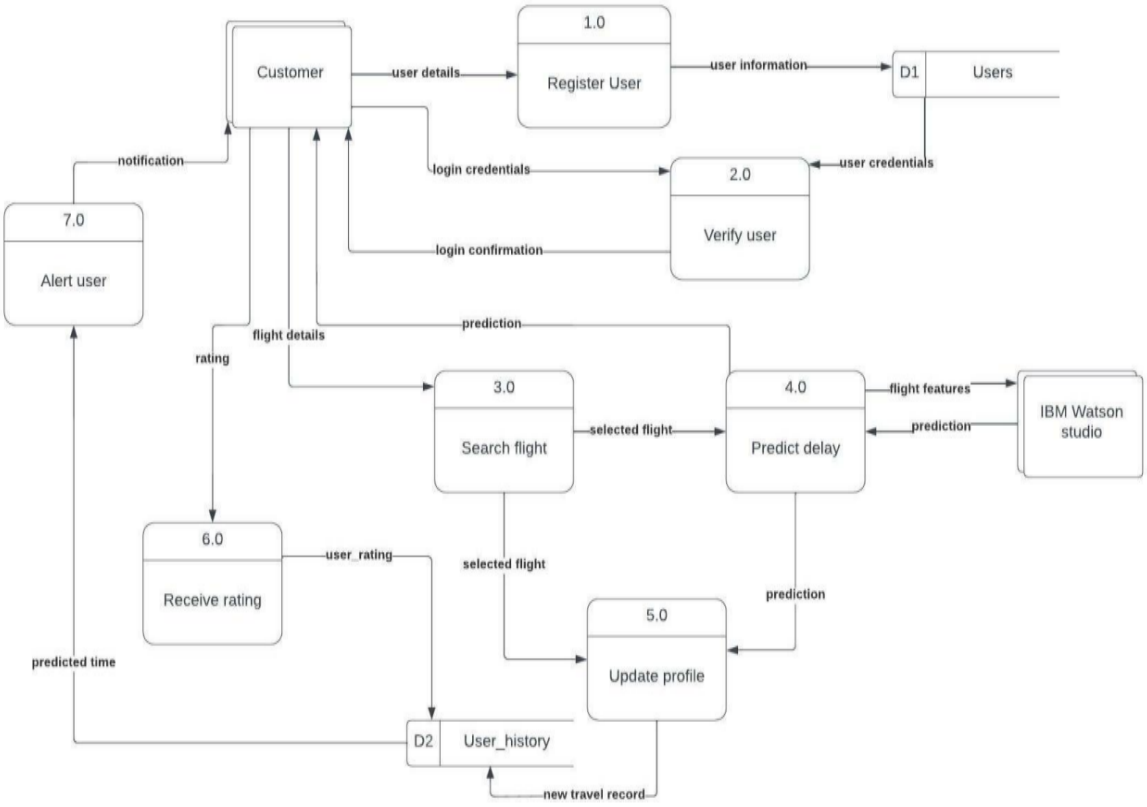
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	In this project, we use flight data, weather and demand data to predict flight departure delay
NFR-2	**Security**	If a flight is about to leave and a passenger is still at security the airline decides whether to wait for the passenger or not
NFR-3	**Reliability**	The reason you want to arrive to the airport before your flights original time because flight delays are usually not reliable estimates
NFR-4	**Performance**	Performance defines how fast a software system or a particular piece of it responds to certain users' actions under a certain workload. The system should provide accurate delays of the Flight
NFR-5	**Availability**	Availability describes how likely the system is accessible to a user at a given point in time. 24/7 available
NFR-6	**Scalability**	Scalability assesses the highest workloads under which the system will still meet the performance requirements. Can handle multiple users at the same time.

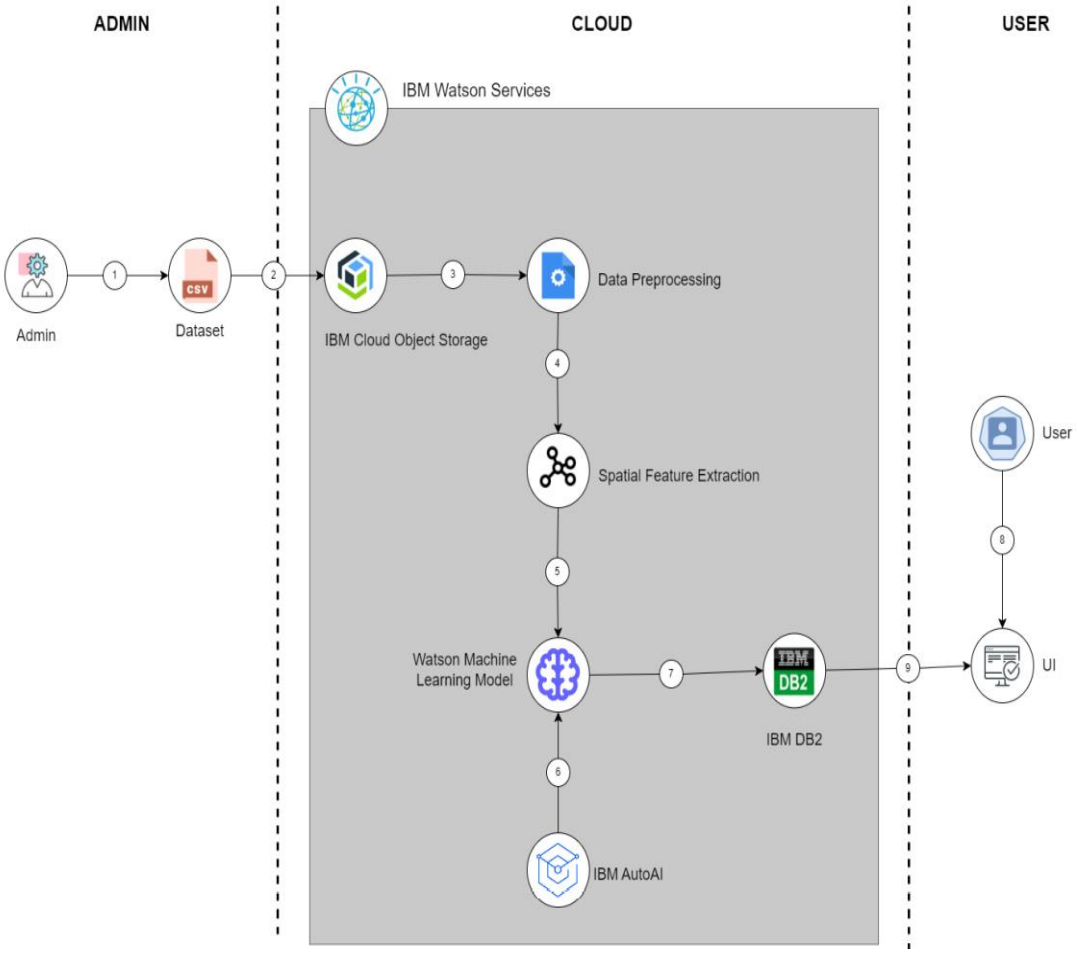
CHAPTER 5

PROJECT DESIGN

5.1DATA FLOW DIAGRAM



5.2 SOLUTION & TECHNICAL ARCHITECTURE



Guidelines:

- The HTML and CSS are used for the user interface for the user to use the application
- The user can see the information on how the image is being recognized.
- Once the button is launched, the user can see the screen to upload the image.
- After uploading the image, the predicted result will be displayed.

5.3 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 confirmation email once I have registered for the Application	I can receive confirmation email & click confirm	High	Sprint-1

			Application.			
		USN-3	As a user, I can register for the application	I can register & access the dashboard with	Low	Sprint-2

		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the app by entering email and password	I can access the dashboard	High	Sprint-1
	Dashboard	USN-6	As a user, I can navigate through different pages using the dashboard	I can various pages	High	Sprint-2

	Search	USN-7	As a user, I can search for flights with destination location.	I can receive information on various flights.	High	Sprint-2
	View	USN-8	As a user, I can view the details of the flights.	I get the information such as flight no, departure and arrival time, etc.	High	Sprint-3
	Receive notifications	USN-9	As a user, I will receive notifications about the flight.	I get frequent updates of the flight's location.	Low	Sprint-3
Admin	GPS	USN-10	As an admin, I need the location of flights	I can track the flights	High	Sprint-4
	Analyse	USN-11	As an admin, I will analyse the given dataset.	I can analyse the dataset.	High	Sprint-2

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Data Collection , Cleaning and Pre-processing	USN-1	As a user, I can't interact with anything. Waiting is users task.	2	High
Sprint-1	Model Building	USN-2	As a user, I predict flight delay by using the ML models developed.	1	High
Sprint-2	Model Evaluation	USN-3	As a user, I can evaluate my models accuracy.	2	Low
Sprint-2	HTML pages	USN-4	As a user, I can access the html pages	2	Medium
Sprint-3	Sign-up	USN-5	As a user, I can register into the application by entering email & password and confirming my password.	1	High
Sprint-3	login	USN-6	As a user, I can log into the application by entering email & password	1	high
Sprint-4	Model integration and deployment	USN-8	As a user ,I can access the models onintegration with flask.	2	High

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 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 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

CHAPTER 7

CODING & SOLUTIONING

```
# import statements
import pandas as pd
import pickle
import joblib
import seaborn as sb
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.ensemble import VotingClassifier
```

```
if(diverted == 'Yes'):
    diverted = 1
else:
    diverted = 0

X = [[month, day_of_month, crs_departure_time, departure_time, departure_delay,
    dep_del15, crs_arrival_time, cancelled, diverted]]
print(X)
payload_scoring = {"input_data": [{"field": [['month', 'day_of_month', 'crs_departure_time', 'dep
    'dep_del15', 'crs_arrival_time', 'cancelled', 'diverted']], "values": [[month, day_of_month,
    dep_del15, crs_arrival_time, cancelled, diverted]]}]}

response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b1ed017a-b3
headers={'Authorization': 'Bearer ' + mltoken})
predicted=response_scoring.json()['predictions'][0]['values'][0][0]
redirect(url_for('details'))
return render_template("details.html", active_page='details', title="Details", predict=predicted,
```

CHAPTER 8

TESTING

8.1TEST CASES

RegisterPage _TC_OO2	Functional	Registration Page	Verify the users entering the unique email	1.Click on User Account dropdown button 2.Verify Register/Signup page accepts only unique email	-	allow only unique email address	Working as expected	Pass
RegisterPage _TC_OO2	Functional	Registration Page	Verify that the user can able to register with valid credentials	1.Click on User Account dropdown button 2.Click Register/Signuppopup a.Enter name b.Enter email c.Enter phone number d.Enter password e.Enter occupation f.Click Register button	-	User should navigate to sign in page	Working as expected	Pass
LoginPage_ TC_OO4	Functional	Login page	Verify user is not able to log into application with Invalid credentials	1.Click on User Account dropdown button and click on sign in/login pop up 2.Enter Valid username/email in Email text box 3.EnterInvalid password in password text box 4.Click on login button	Email:123@gmai l.com password: 12345	User will be at the same page without navigating	Working as expected	Pass
Pediction_Pa ge_TC_OO1	UI	Prediction page	Verify user is able to see the prediction form, prediction and go back button	1.Click on User Account dropdown button 2.Enter Valid email in Email text box 3.Enter valid password in password text box 4.Click on login button		Application should navigate to Prediction page and user can able to view the prediction form, predict and go back button	Working as expected	Pass

Pediction_Page_TC_OO2	UI	Prediction page	Verify the UI elements prediction form	Account dropdown button 2.Enter Valid email in Email text box 3.Enter valid password in password text box 4.Click on login button 5. Verify Prediction form popup with below UI elements: a.Date		the details in prediction form	Working as expected	Pass
HomePage_TC_OO1	Functional	Home Page	see the Login/Signup popup when user clicked on User accountbutton	Account icon 2.Verify login/Signuppopup displayed or not	-	display	Working as expected	Pass
RegisterPage_TC_OO1	UI	Registration Page	Verify the UI elements in Register/Signup page	1.Enter URL and clickgo 2.Click on User Account dropdown button 3.Verify Register/Signup popup with below UI elements: a.name text box b. email text box c. phone number text box d. password text box e.occupation text box f. Already a member? login	-	Application should show below UI elements: a.name text box b.email text box c.phone number textbox d.password text box e.occupation text box f.Already have an account? Click login	Working as expected	Pass

8.2 USER ACCEPTANCE TESTING

8.2.1DEFECT ANALYSIS

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	5	3	1	16
Duplicate	0	0	0	0	0
External	5	0	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	23	7	7	23	60

8.2.2 TEST CASE ANALYSIS

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

Section	Total Cases	Not Tested	Fail	Pass
Model Evaluation	10	0	0	10
Client Application	20	0	0	20
Exception Reporting	5	0	0	5
Final Report Output	2	0	0	2

CHAPTER 9

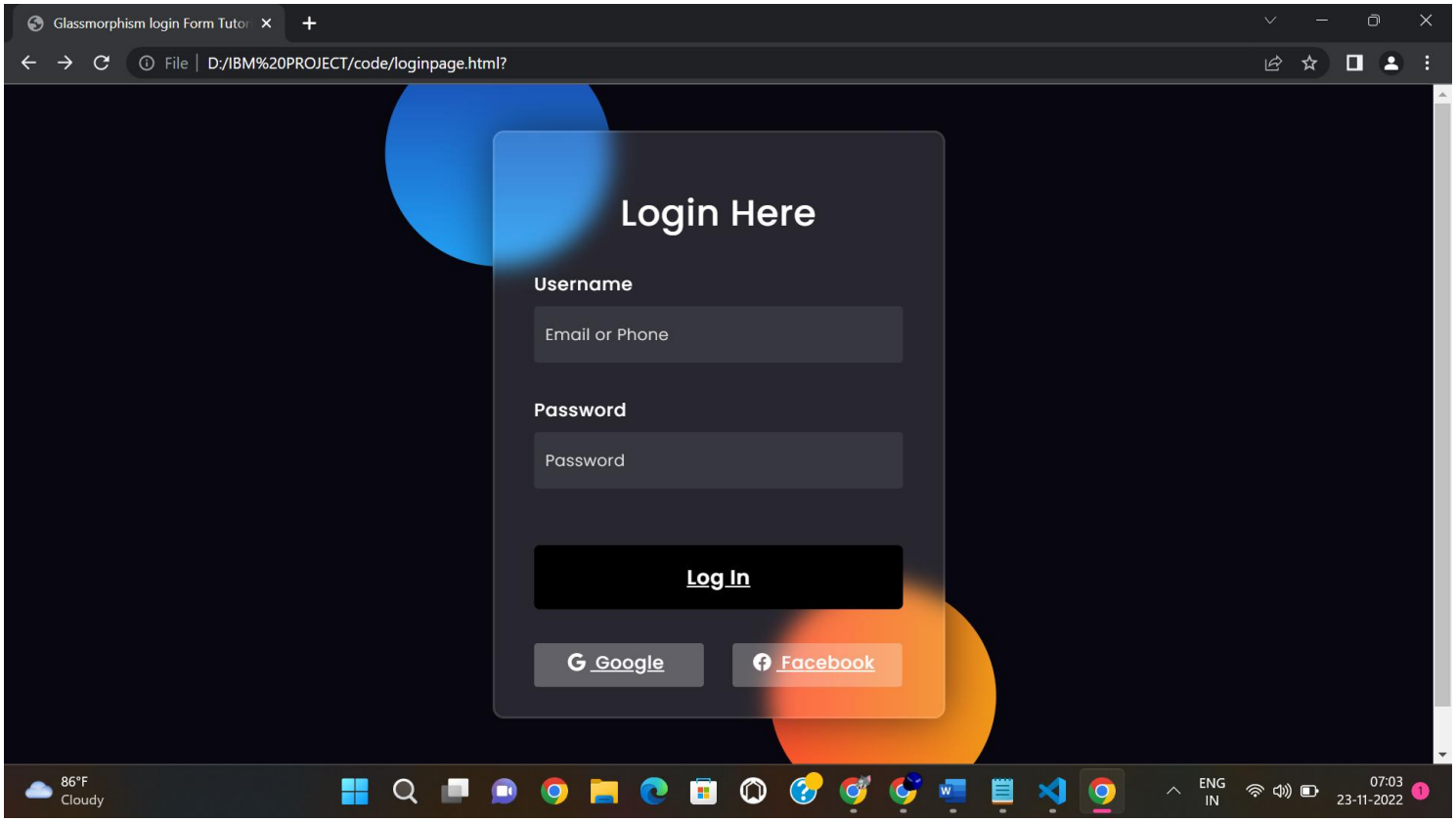
RESULTS

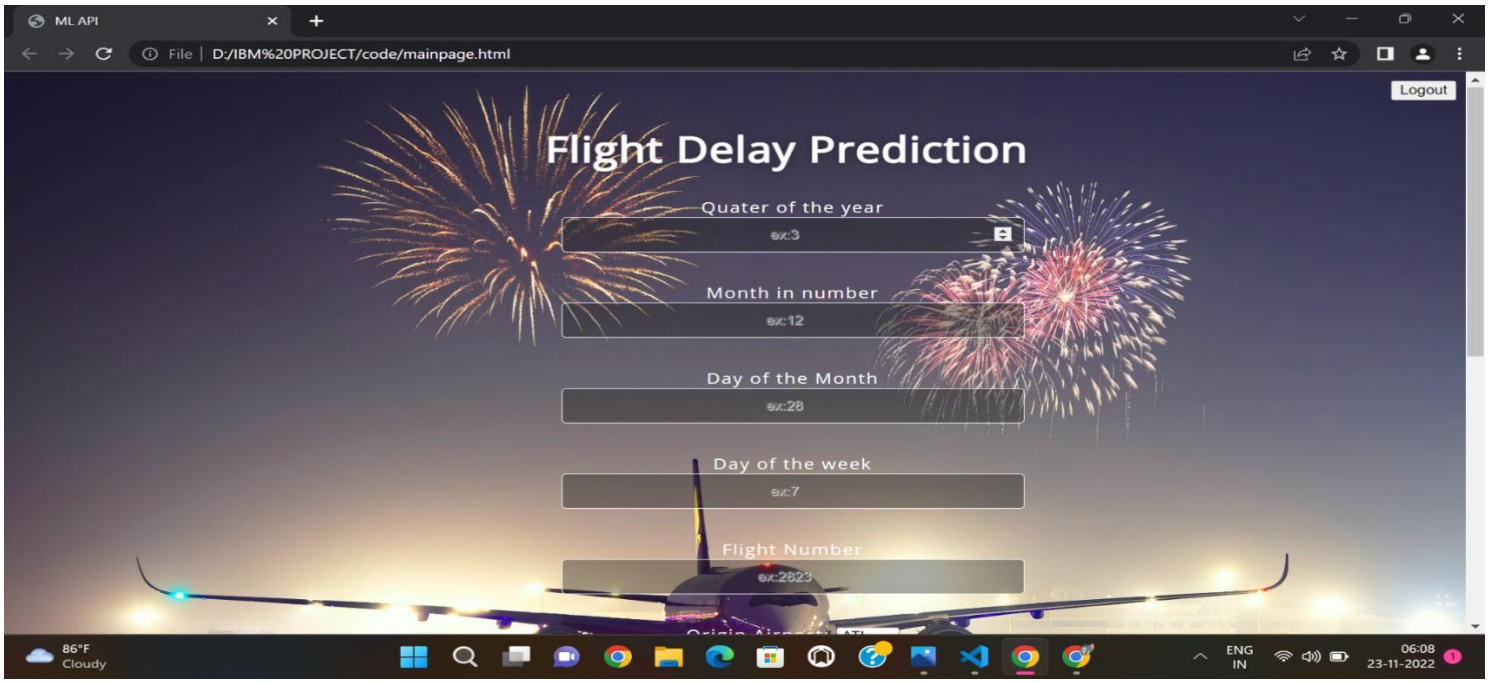
9.1PERFORMANCE METRICS

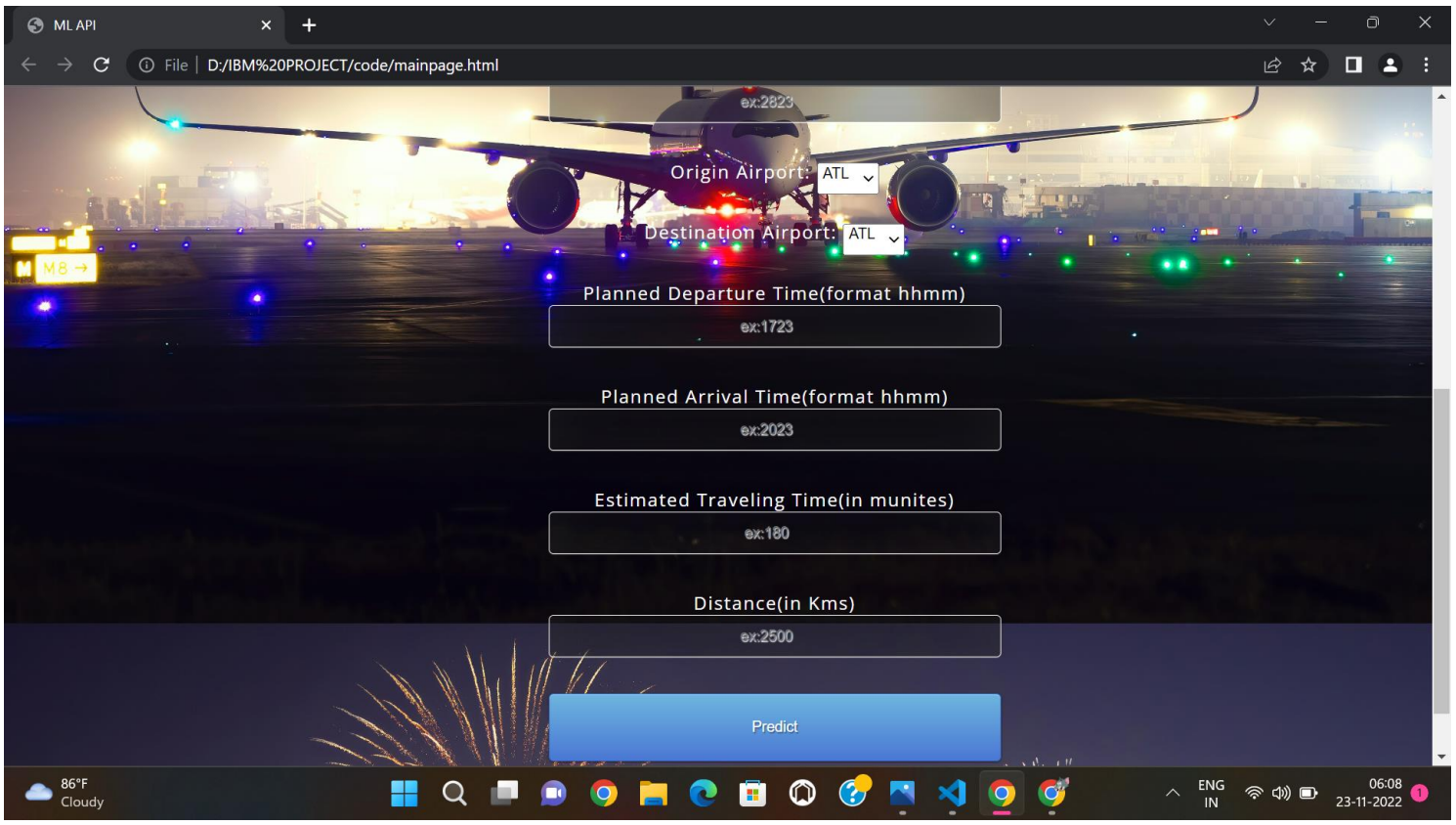
S.No.	Parameter	Values	Screenshot

1.	Metrics	**Classification Model:** Confusion Matrix , Accuracy Score & Classification Report	In [24]:
			`#Model Evaluation`
			`from sklearn.metrics import accuracy_score,confusion_matrix, classification_report`
			`print(accuracy_score(y_test, pred))`
			`0.9163899788711138`
			In [26]:
			`print(confusion_matrix(y_test, pred))`
			`[[2732 164]`
			` [ 113 304]]`
			In [27]:
			`print(classification_report(y_test, pred))`
			precision recall f1-score support
			0.0 0.96 0.94 0.95 2896
			1.0 0.65 0.73 0.69 417
			accuracy 0.92 3313
			macro avg 0.80 0.84 0.82 3313
			weighted avg 0.92 0.92 0.92 3313

2.	Tune the Model	Hyperparameter Tuning , Validation Method	In [31]:
			`from sklearn.model_selection import cross_val_score, KFold, GridSearchCV`
			`kf = KFold(n_splits = 6, shuffle = True, random_state = 25)`
			`params = {'max_depth': [4,5,6],`
			`'min_samples_split': [2,3,4],`
			`'criterion': ['gini', 'entropy', 'log_loss']}`
			In [32]:
			`grid_cv = GridSearchCV(clf, params, cv = kf)`
			`grid_cv.fit(X_train,y_train)`
			Out[32]:
			GridSearchCV
			estimator: DecisionTreeClassifier
			DecisionTreeClassifier
			In [35]:
			`grid_cv.best_params_`
			Out[35]:
			`{'criterion': 'entropy', 'max_depth': 6, 'min_samples_split': 2}`
			In [34]:
			`cv_results = cross_val_score(clf, X_train,y_train, cv = kf)`
			`print(cv_results)`
			`[0.92552366 0.9193173 0.91925466 0.92934783 0.92313665 0.9060559 ]`


Glassmorphism login Form Tutor
File | D:/IBM%20PROJECT/code/loginpage.html?
Login Here
Username
Email or Phone
Password
Password
Log In
Google
Facebook
86°F
Cloudy
ENG
IN
07:03
23-11-2022


ML API
File | D:/IBM%20PROJECT/code/mainpage.html
Logout
Flight Delay Prediction
Quater of the year
ex:3
Month in number
ex:12
Day of the Month
ex:28
Day of the week
ex:7
Flight Number
ex:2823
86°F
Cloudy
ENG
IN
06:08
23-11-2022


ML API
File | D:/IBM%20PROJECT/code/mainpage.html
ex:2823
Origin Airport: ATL
Destination Airport: ATL
Planned Departure Time(format hhmm)
ex:1723
Planned Arrival Time(format hhmm)
ex:2023
Estimated Traveling Time(in munites)
ex:180
Distance(in Kms)
ex:2500
Predict
86°F
Cloudy
ENG
IN
06:08
23-11-2022

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- Reduces manual work
- More accurate than average human
- Capable of handling a lot of data
- Can be used anywhere from any device

DISADVANTAGES:

- Cannot handle complex data
- All the data must be given by user
- Requires a high performance server for faster predictions
- Prone to occasional errors

CHAPTER 11
CONCLUSION

Flight delays are an important subject in the literature due to their economic and environmental impacts. They may increase costs to customers and operational costs to airlines. Apart from outcomes directly related to passengers, delay prediction is crucial during the decision-making process for every player in the air transportation system. In this context, researchers created flight delay models for delay prediction over the last years, and this work contributes with an analysis of these models from a Data Science perspective. We developed a ensemble learning model with an accuracy of 93% and predicted the delay.

CHAPTER 12
FUTURE SCOPE

This This project is far from complete and there is a lot of room for improvement. Some of the improvements that can be made to this project are as follows:

- Add APIs to take input automatically with flight number
- Improve model to detect delay at complex situation like chance of flight diversion
- Add support to different languages to help users from all over the world

This project has endless potential and can always be enhanced to become better. Implementing this concept in the real world will benefit several industries and reduce the workload on many workers, enhancing overall work efficiency.

APPENDIX

SOURCE CODE:

MODEL CREATION:

```
import pandas as pd
import pickle
import joblib
import seaborn as sb
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.ensemble import VotingClassifier
X = refined_dataset[['MONTH', 'DAY_OF_MONTH', 'CRS_DEP_TIME', 'DEP_TIME', 'DEP_DELAY',
        'DEP_DEL15', 'CRS_ARR_TIME', 'CANCELLED', 'DIVERTED']]
Y = refined_dataset[['ARR_DEL15']]
# splitting into train and test data
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.33, random_state=42)
y_train.value_counts()

# decision tree classifier
decision_tree_classifier = DecisionTreeClassifier()
decision_tree_classifier = decision_tree_classifier.fit(X_train,y_train)
decision_tree_prediction = decision_tree_classifier.predict(X_test)
```

```
# performance metrics
print("Confusion matrix\n", confusion_matrix(decision_tree_prediction, y_test))
print("Classification report\n", classification_report(decision_tree_prediction, y_test))
print("Accuracy score\n", accuracy_score(decision_tree_prediction, y_test))

# svc model
SVC_model = SVC()
SVC_model.fit(X_train, y_train)
SVC_prediction = SVC_model.predict(X_test)
# performance metrics
print("Confusion matrix\n", confusion_matrix(SVC_prediction, y_test))
print("Classification report\n", classification_report(SVC_prediction, y_test))
print("Accuracy score\n", accuracy_score(SVC_prediction, y_test))

# knn model
KNN_model = KNeighborsClassifier(n_neighbors=5)
KNN_model.fit(X_train, y_train)
KNN_prediction = KNN_model.predict(X_test)
# performance metrics
print("Confusion matrix\n", confusion_matrix(KNN_prediction, y_test))
print("Classification report\n", classification_report(KNN_prediction, y_test))
print("Accuracy score\n", accuracy_score(KNN_prediction, y_test))

# gaussian naive bayes model
GNB_model = GaussianNB()
GNB_model.fit(X_train, y_train)
GNB_prediction = GNB_model.predict(X_test)
# performance metrics
print("Confusion matrix\n", confusion_matrix(GNB_prediction, y_test))
print("Classification report\n", classification_report(GNB_prediction, y_test))
print("Accuracy score\n", accuracy_score(GNB_prediction, y_test))

        # ensemble model of best 3 peforming model - gnb, knn, svc
ensemble = VotingClassifier(estimators=[('gnb', GNB_model), ('knn', KNN_model), ('svc', SVC_model)], voting='hard')
```

```
ensemble.fit(X_train, y_train)
ensemble_prediction = ensemble.predict(X_test)
# performance metrics
print("Confusion matrix\n", confusion_matrix(ensemble_prediction, y_test))
print("Classification report\n", classification_report(ensemble_prediction, y_test))
print("Accuracy score\n", accuracy_score(ensemble_prediction, y_test))

joblib.dump(ensemble, 'flight.pkl')
```

FLASK APP:

```
import flask
import joblib
import pickle
import MySQLdb.cursors
import re
import pandas as pd
from flask import request, render_template, Flask, redirect, url_for, session, flash
from flask_mysqldb import MySQL
from flask_cors import CORS
import datetime
from xgboost import *
import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "8eyURsE0C1ebAbHj2NIlRGNaG0EJkD2w2t_MWiHZr4e2"
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}

# setting a new instance of Flask
app = flask.Flask(__name__, static_url_path="")
CORS(app)
```

```
# creating secret key for the app
app.secret_key = 'helloworld'

with open('secret.txt', 'r') as f:
    line = f.readline().split(',')
    user = line[0]
    password = line[1]

# setting config values
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = user
app.config['MYSQL_PASSWORD'] = password
app.config['MYSQL_DB'] = 'flights'

# creating instance of MySql
mysql = MySQL(app)

# home page route
@app.route('/', methods=["GET"])
def sendHome():
    return render_template('index.html', active_page="home", title="Welcome to FDP System!")

# Login page
@app.route('/login', methods=['GET', 'POST'])
def login():
    # to display on the website
    msg = ''
    if request.method == 'POST' and 'name' in request.form and 'password' in request.form:
        name = request.form['name']
        password = request.form['password']

        # connect to MYSQL
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM accounts WHERE name = % s AND password= % s', (name, password))
        account = cursor.fetchone()
```

```
        # if account exists
        if account:
            session['loggedin'] = True
            session['id'] = account['id']
            session['name'] = account['name']
            msg = 'Logged in successfully!'
            # if predict was selected before
            if 'pending' in session.keys():
                redirect(url_for('predict'))
                return render_template('details.html', msg=session['name'], active_page='details', title="Flight Details")
            redirect(url_for('sendHome'))
            return render_template('index.html', msg=msg, active_page='home', title="Welcome to FDP System!")
        else:
            msg = 'Incorrect username/password. Please try again'
    redirect(url_for('login'))
    return render_template('login.html', msg=msg, active_page='login', title="Login")

# logout
@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('name', None)
    session.pop('pending', None)
    msg = 'Successfully Logged out!'
    redirect(url_for('sendHome'))
    return render_template('index.html', msg=msg, active_page='home', title="Welcome to FDP System!")

# register
@app.route('/register', methods = ['GET', 'POST'])
def register():
    msg = ''
    if request.method == 'POST' and 'name' in request.form and 'email' in request.form:
```

```
        # get details from form
        name = request.form['name']
        password = request.form['password']
        email = request.form['email']

        # mysql connection and retreival
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM accounts WHERE name = % s', (name, ))
        account = cursor.fetchone()

        # check statements
        if account:
            msg = 'Account already exists'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', name):
            msg = 'Username must contain only characters and numbers !'
        elif not name or not password or not email:
            msg = 'Please fill out the form !'
        else:
            # insert into table
            cursor.execute('INSERT INTO accounts VALUES (NULL, % s, % s, % s)', (name, password, email, ))
            mysql.connection.commit()
            msg = 'You have successfully registered !'
            redirect(url_for('login'))
            return render_template('login.html', msg = msg, active_page='login', title="Log In")

    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    redirect(url_for('register'))
    return render_template('register.html', msg = msg, active_page='register', title="Signup")

# predicting the labels
@app.route('/predict', methods=["POST"])
def predict():
```

```
format_date = "%Y-%m-%d"
dep_date = datetime.datetime.strptime(request.form['dep_date'], format_date)
month = int(dep_date.date().month)
day_of_month = int(dep_date.date().day)

format_time = "%H:%M"
crs_dep_time = datetime.datetime.strptime(request.form['crs_dep_time'], format_time)
hour = crs_dep_time.time().hour
minute = crs_dep_time.time().minute
crs_departure_time = hour*100+minute

format_time = "%H:%M"
crs_arr_time = datetime.datetime.strptime(request.form['crs_arr_time'], format_time)
hour = crs_arr_time.time().hour
minute = crs_arr_time.time().minute
crs_arrival_time = hour*100+minute

format_time = "%H:%M"
dep_time = datetime.datetime.strptime(request.form['dep_time'], format_time)
hour = dep_time.time().hour
minute = dep_time.time().minute
departure_time = hour*100+minute

departure_delay = crs_departure_time - departure_time
if(departure_delay>15):
    dep_del15 = 1
else:
    dep_del15 = 0

cancelled = request.form['cancelled']
if(cancelled == 'Yes'):
    cancelled = 1
else:
    cancelled = 0
```

```
    diverted = request.form['diverted']
    if(diverted == 'Yes'):
        diverted = 1
    else:
        diverted = 0

    X = [[month, day_of_month, crs_departure_time, departure_time, departure_delay,
         dep_del15, crs_arrival_time, cancelled, diverted]]
    print(X)
    payload_scoring = {"input_data": [{"field": [['month', 'day_of_month', 'crs_departure_time', 'departure_time', 'departure_delay',
         'dep_del15', 'crs_arrival_time', 'cancelled', 'diverted']], "values": [[month, day_of_month, crs_departure_time, departure_time, departure_delay,
         dep_del15, crs_arrival_time, cancelled, diverted]]}]}

    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b1ed017a-b34d-409c-a08d-e26b0632dc30/predictions?version=2022-11-16', json=payload_scoring,
    headers={'Authorization': 'Bearer ' + mltoken})
    predicted=response_scoring.json()['predictions'][0]['values'][0][0]
    redirect(url_for('details'))
    return render_template("details.html", active_page='details', title="Details", predict=predicted, msg=session['name'])



@app.route('/details', methods=["GET"])
def details():
    msg = 'Please log in or signup to continue!'
    if 'loggedin' in session.keys():
        redirect(url_for('details'))
        return render_template("details.html", active_page='details', title="Details", msg=session['name'])
    session['pending'] = True
    redirect(url_for('login'))
    return render_template('login.html', msg=msg, active_page='login', title="Login")
```

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

BASE PAGE(HTML):

```
<!-- top navigation -->
<html>
    <head>
        <meta charset="UTF-8">
        <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
        <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">
        <title> {{ title }} </title>
        <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>
        <script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
        <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl" crossorigin="anonymous"></script>
    </head>
    <body>
    <div class="nav-header">
    <nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
            <ul class="navbar-nav navbar-center">
                <span class="nav-left-seperator"></span>
                <li class="{{ 'nav-item-active' if active_page == 'home' else 'nav-item' }}">
                <a class="nav-link" href="/">Home</a>
                </li>
                <li class="{{ 'nav-item-active' if active_page == 'login' else 'nav-item' }}">
                    <a class="nav-link" href="/login">Login</a>
```

```
                </li>
                <li class="{{ 'nav-item-active' if active_page == 'details' else 'nav-item' }}">
                   <a class="nav-link" href="/details">Predict</a>
                </li>
                <span class="nav-right-seperator"></span>
                <li class="{{ 'nav-item-active' if active_page == 'register' else 'nav-item' }}">
                   <a class="nav-link" href="/register">Signup</a>
                </li>
                <li class="{{ 'nav-item-active' if active_page == 'logout' else 'nav-item' }}">
                <a class="nav-link" href="/logout">Logout</a>
                </li>
              </ul>
         </nav>
   </div>
{% block content %}{% endblock %}
</body>
</html>
```

LOGIN PAGE(HTML):

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Design by foolishdeveloper.com -->
   <title>Glassmorphism login Form Tutorial in html css</title>

   <link rel="preconnect" href="https://fonts.gstatic.com">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
   <link                href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;500;600&display=swap"
rel="stylesheet">
   <!--Stylesheet-->
   <style media="screen">
     *,
*:before,
*:after{
```

```
    padding: 0;
    margin: 0;
    box-sizing: border-box;
}
body{
    background-color: #080710;
}
.background{
    width: 430px;
    height: 520px;
    position: absolute;
    transform: translate(-50%,-50%);
    left: 50%;
    top: 50%;
}
.background .shape{
    height: 200px;
    width: 200px;
    position: absolute;
    border-radius: 50%;
}
.shape:first-child{
    background: linear-gradient(
        #1845ad,
        #23a2f6
    );
    left: -80px;
    top: -80px;
}
.shape:last-child{
    background: linear-gradient(
        to right,
        #ff512f,
        #f09819
    );
```

```
    right: -30px;
    bottom: -80px;
}
form{
    height: 520px;
    width: 400px;
    background-color: rgba(255,255,255,0.13);
    position: absolute;
    transform: translate(-50%,-50%);
    top: 50%;
    left: 50%;
    border-radius: 10px;
    backdrop-filter: blur(10px);
    border: 2px solid rgba(255,255,255,0.1);
    box-shadow: 0 0 40px rgba(8,7,16,0.6);
    padding: 50px 35px;
}
form *{
    font-family: 'Poppins',sans-serif;
    color: #ffffff;
    letter-spacing: 0.5px;
    outline: none;
    border: none;
}
form h3{
    font-size: 32px;
    font-weight: 500;
    line-height: 42px;
    text-align: center;
}

label{
    display: block;
    margin-top: 30px;
    font-size: 16px;
```

```
    font-weight: 500;
}
input{
    display: block;
    height: 50px;
    width: 100%;
    background-color: rgba(255,255,255,0.07);
    border-radius: 3px;
    padding: 0 10px;
    margin-top: 8px;
    font-size: 14px;
    font-weight: 300;
}
::placeholder{
    color: #e5e5e5;
}
button{
    margin-top: 50px;
    width: 100%;
    background-color: black;
    color: #080710;
    padding: 15px 0;
    font-size: 18px;
    font-weight: 600;
    border-radius: 5px;
    cursor: pointer;
}
.social{
  margin-top: 30px;
  display: flex;
}
.social div{
  background: red;
  width: 150px;
  border-radius: 3px;
```

```
  padding: 5px 10px 10px 5px;
  background-color: rgba(255,255,255,0.27);
  color: #eaf0fb;
  text-align: center;
}
.social div:hover{
  background-color: rgba(255,255,255,0.47);
}
.social .fb{
  margin-left: 25px;
}
.social i{
  margin-right: 4px;
}

    </style>
</head>
<body>
    <div class="background">
        <div class="shape"></div>
        <div class="shape"></div>
    </div>
    <form>
        <h3>  Login Here </h3>

        <label for="username">Username</label>
        <input type="text" placeholder="Email or Phone" id="username">

        <label for="password">Password</label>
        <input type="password" placeholder="Password" id="password">

        <button> <a href="mainpage.html">Log In </a></button>
        <div class="social">
          <div                class="go"><a                href="https://accounts.google.com/v3/signin/identifier?dsh=S-
1493151590%3A1669210384477720&elo=1&flowName=GlifWebSignIn&flowEntry=ServiceLogin&ifkv=ARgd
```

vAsZZ0t1kW-19e_F3RprEpFY16riUif3wT-sa-W_oGrJyLWM7lcQCVp6gmxqvh31Hyj8wFLC-g"> Google

  Facebook

</form>

</body>

</html>

Login Page URL: file:///D:/IBM%20PROJECT/code/loginpage.html

REGISTER PAGE (HTML):

```
<!DOCTYPE html>
<html >

<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>
<link rel="stylesheet" href="cssstyle.css">


</head>

<body>


<div class="logout" style="display : flex;
flex-direction: row-reverse";>

<a href="{{ url_for('main')}}"><button>Logout</button></a>
</div>
 <h1 class="h1">Flight Delay Prediction</h1><br><br>
```

```
<div class="login"><br><br>


    <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}" method="post">
        Quater of the year <input type="number" name="quater" placeholder="ex:3" required="required" min='1' max='4' /><br><br>
        Month in number<input id="r1" type="number" name="month" placeholder="ex:12" required="required" min='1' max='12'/><br><br>
            Day of the Month<input  id="r2" type="number" name="daym" placeholder="ex:28" required="required" min='1' max='31'/><br><br>
            Day of the week<input id="r3" type="number" name="dayw" placeholder="ex:7" required="required" min='1' max='7'/><br><br>
        Flight Number<input id="r4" type="number" name="fnum" placeholder="ex:2823" required="required" max="9999"/><br><br>

            Origin Airport: <select name="airport" class="origin">
        <option value='1'>ATL</option>
        <option value='2'>DWT</option>
        <option value='3'>JFK</option>
        <option value='4'>MSP</option>
        <option value='5'>SEA</option>
            </select><br><br>
            Destination Airport: <select name="airportd" class="dest">
        <option value='1'>ATL</option>
        <option value='2'>DWT</option>
        <option value='3'>JFK</option>
        <option value='4'>MSP</option>
        <option value='5'>SEA</option>
            </select><br><br>
            Planned Departure Time(format hhmm)<input type="number" name="dtime" placeholder="ex:1723" required="required" max="9999"/><br><br>
            Planned Arrival Time(format hhmm)<input type="number" name="atime" placeholder="ex:2023" required="required" max="9999"/><br><br>
```

```
        Estimated Traveling Time(in munites)<input type="number" name="ttime" placeholder="ex:180" required="required" max="9999"/><br><br>
        Distance(in Kms)<input type="number" name="distance" placeholder="ex:2500" required="required"min='140' max="99999"/><br><br>
      <button type="submit" class="btn btn-primary btn-block">Predict</button>
   </form>

  <br>
  <br>
  <br>

 </div>



</body>
</html>
```

Register Page URl: file:///D:/IBM%20PROJECT/code/mainpage.html

GITHUB

https://github.com/IBM-EPBL/IBM-Project-53168-1661316602