**Project Report**

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose
2. **LITERATURE SURVEY**
   1. Existing problem
   2. References
   3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   1. Empathy Map Canvas
   2. Ideation & Brainstorming
   3. Proposed Solution
   4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   1. Functional requirement
   2. Non-Functional requirements
5. **PROJECT DESIGN**
   1. Data Flow Diagrams
   2. Solution & Technical Architecture
   3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
   1. Sprint Planning & Estimation
   2. Sprint Delivery Schedule
   3. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   1. Feature 1
   2. Feature 2
   3. Database Schema (if Applicable)
8. **TESTING**
   1. Test Cases
   2. User Acceptance Testing
9. **RESULTS**
   1. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

# PROJECT REPORT

**Airlines Data Analytics for Avaition Industry**

## 1. INTRODUCTIO

### 1.1PROJECT OVERVIEW

In simple words, Airlines Data Analytics for Avaition Industry entails all the delays and activities that makes your life easier by helping you by giving analysis of flight delays, departure time , arrival time.

The project mainly focuses on the air traffic and problems based on the events that causes discomfort to passengers and to reduce the high prolific economic losses. This applications will provide better Airline and AirPort services and to avoid delays in Air Travel across different locations at Municipality level. The aim is to provide airports, airlines, and the travelling public with a neutral, third-party view of which airlines are delivering on their promise to get passengers from Point A to Point B on-time.

### 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

**1.AIRLINE MEMBER CUSTOMER VALUE ANALYSIS:**

In recent years, the vigorous development of the transportation industry has attracted a large number of customers, especially those in the aviation industry. However, for airlines, the pressure of competition has increased year by year; on the other hand, there are also competing relationships among different airlines. Therefore, for airlines, how to retain customers has become the key to the problem. In fact, using the various customer factors provided by the existing churn customer information data set can use the data visualization means of data analysis to analyse the behaviour of churn customers. In addition, relevant marketing strategies can be proposed to improve the business level as much as possible.

**2.PREDICTIVE ANALYTICS PLATFORM FOR AIRLINE INDUSTRY:**

The research is to develop accurate demand forecasting model to control the availability in Airline industry. The primary outcome of the model is that the airline organization can maximize the revenue by controlling the availability. The product in airline industry is the seat, which is an expensive, unstock able product. The demand for the seats is almost uncertain, the capacity is constraint and difficult to increase and the variable costs are very high. Hence the priority of the expected demand forecast is very high for airline industry. An accurate mechanism is to predict the revenue for future months of ODs is done using fare and passenger data. The revenue is derived by the number of passengers and fares they pay which vary for each flight. Hence most of the information is available, however changing market conditions is an unknown variable which can have a significant impact on passenger travel patterns. Through this research they are going to design and develop the best fit model to forecast flight OD level passenger demand based on the historical data.

1. **EXPLORATORY DATA ANALYSIS ON AVIATION DATASET**

The usage of big data analytics is booming today, with its ability to be used to draw useful insights from past data research. Its uses in the aviation industry have a wide array of applic ations ranging from predicting flight delays to detecting faults in airplane parts. In this paper, we conducted exploratory data analysis on flight dataset to draw inferences on arrival and departure delays and to identify relationships between flight timings and delays. Using the flight delay data, we identified which flight is mostly prone to delays. The arrived upon conclusions are useful for selecting flights in the future.

1. **DATA SCIENCE AND ANALYTICS IN AVIATION**

Data science and analytics are attracting more and more attention from researchers and practitioners in recent years. Due to the rapid development of advanced technologies nowadays, a massive amount of real time data regarding flight information, flight performance, airport conditions, air traffic conditions, weather, ticket prices, passenger’s comments, crew comments, etc., are all available in different flight performance monitoring systems, operational systems of airlines and airports, and social media platforms. Development of data analytics in aviation and related applications are also growing rapidly. This paper concisely examines data science and analytics in aviation studies in several critical areas, namely big data analysis, air transport network management, forecasting, and machine learning. The papers featured in this special issue are also introduced and reviewed, and future directions for data science and analytics in aviation are discussed.

**2.2 REFERENCES.**

[**https://ieeexplore.ieee.org/document/9410686**](https://ieeexplore.ieee.org/document/9410686) [**https://ieeexplore.ieee.org/document/9357244**](https://ieeexplore.ieee.org/document/9357244) [**https://ieeexplore.ieee.org/document/9738868**](https://ieeexplore.ieee.org/document/9738868)

**AIRLINE MEMBER CUSTOMER VALUE ANALYSIS:**

**Published in:**[ISCTT 2021; 6th International Conference on Information Science, Computer](https://ieeexplore.ieee.org/xpl/conhome/9738856/proceeding)

[Technology and Transportation](https://ieeexplore.ieee.org/xpl/conhome/9738856/proceeding)

**Date of Conference:** 26-28 November 2021

**Date Added to IEEE *Xplore*:** 22 March 2022

**Print ISBN:**978-3-8007-5727-5

**Publisher:** VDE

**Conference Location:** Xishuangbanna, China

**Authors**

**PREDICTIVE ANALYTICS PLATFORM FOR AIRLINE INDUSTRY**

**Published in:**[2020 2nd International Conference on Advancements in Computing (ICAC)](https://ieeexplore.ieee.org/xpl/conhome/9357062/proceeding)

**Date of Conference:** 10-11 December 2020 **Date Added to IEEE *Xplore*:** 26 February 2021 **ISBN Information:**

**INSPEC Accession Number:** 20491004

**DOI:**[10.1109/ICAC51239.2020.9357244](https://doi.org/10.1109/ICAC51239.2020.9357244) **Publisher:** IEEE

**Conference Location:** Malabe, Sri Lanka

[P. H. K Tissera-](https://ieeexplore.ieee.org/author/37088851617)Faculty of computing, SLIIT, Malabe, Sri Lanka

[A.N.M.R.S.P. llwana-](https://ieeexplore.ieee.org/author/37088851268)Faculty of computing, SLIIT, Malabe, Sri Lanka

[K.T. Waduge-](https://ieeexplore.ieee.org/author/37088851107)Faculty of computing, SLIIT, Malabe, Sri Lanka

[M.A.l. Perera-](https://ieeexplore.ieee.org/author/37088851228)Faculty of computing, SLIIT, Malabe, Sri Lanka

[D.P. Nawinna-](https://ieeexplore.ieee.org/author/37891650100)Faculty of computing, SLIIT, Malabe, Sri Lanka

**EXPLORATORY DATA ANALYSIS ON AVIATION DATASET**

**Published in:**[2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)](https://ieeexplore.ieee.org/xpl/conhome/9410659/proceeding)

**Date of Conference:** 17-18 March 2021 **Date Added to IEEE *Xplore*:** 28 April 2021 **ISBN Information:**

**INSPEC Accession Number:** 20654292

**DOI:**[10.1109/ICCIKE51210.2021.9410686](https://doi.org/10.1109/ICCIKE51210.2021.9410686)

**Publisher:** IEEE

**Conference Location:** Dubai, United Arab Emirates

### 2.3 PROBLEM STATEMENT DEFINITION

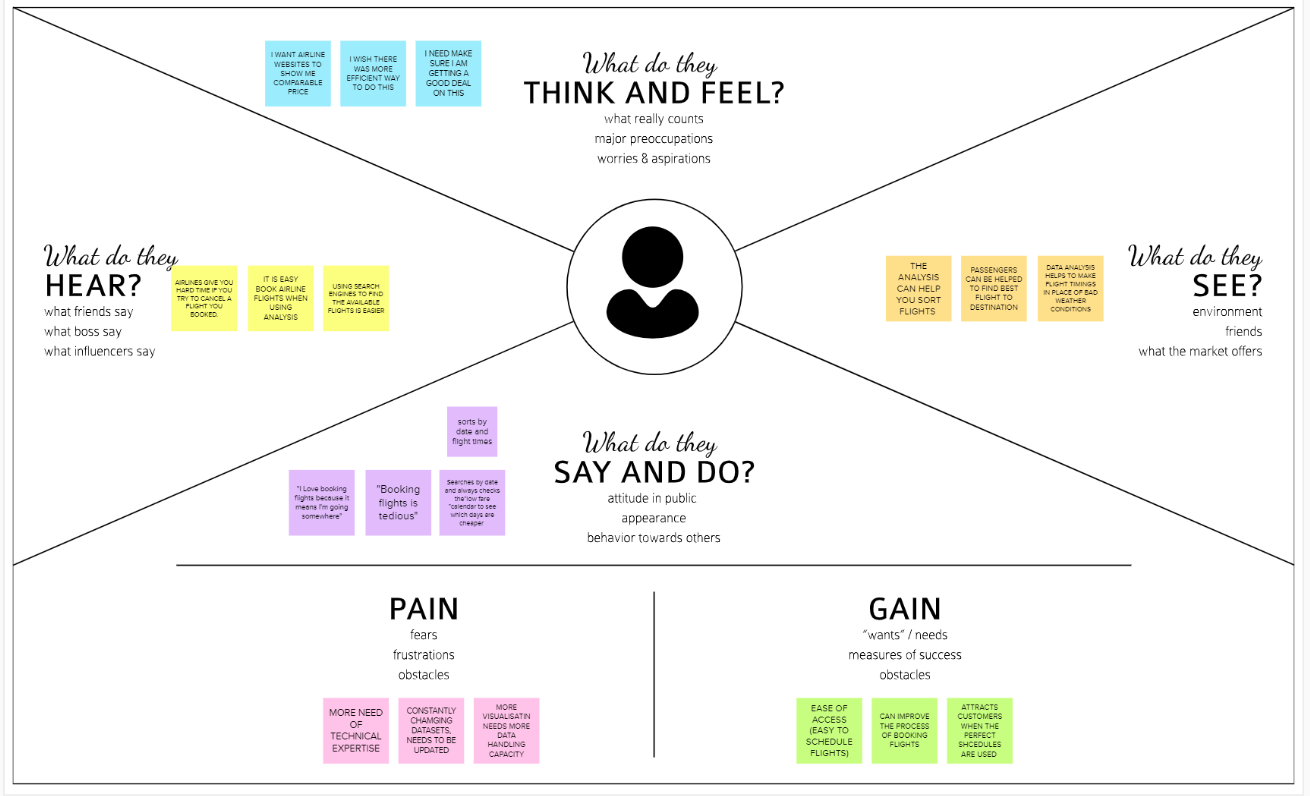
Analysis of flight delay and causal factors is crucial in maintaining airspace efficiency and safety. However, delay samples are not independent since they always show a certain aggregation pattern. Therefore, this study develops a novel spatial analysis approach to explore the delay and causal factors which is able to take dependence and the possible problem involved

including error correlation and variable lag effect of causal factors on delay into account using data Analytics. 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.

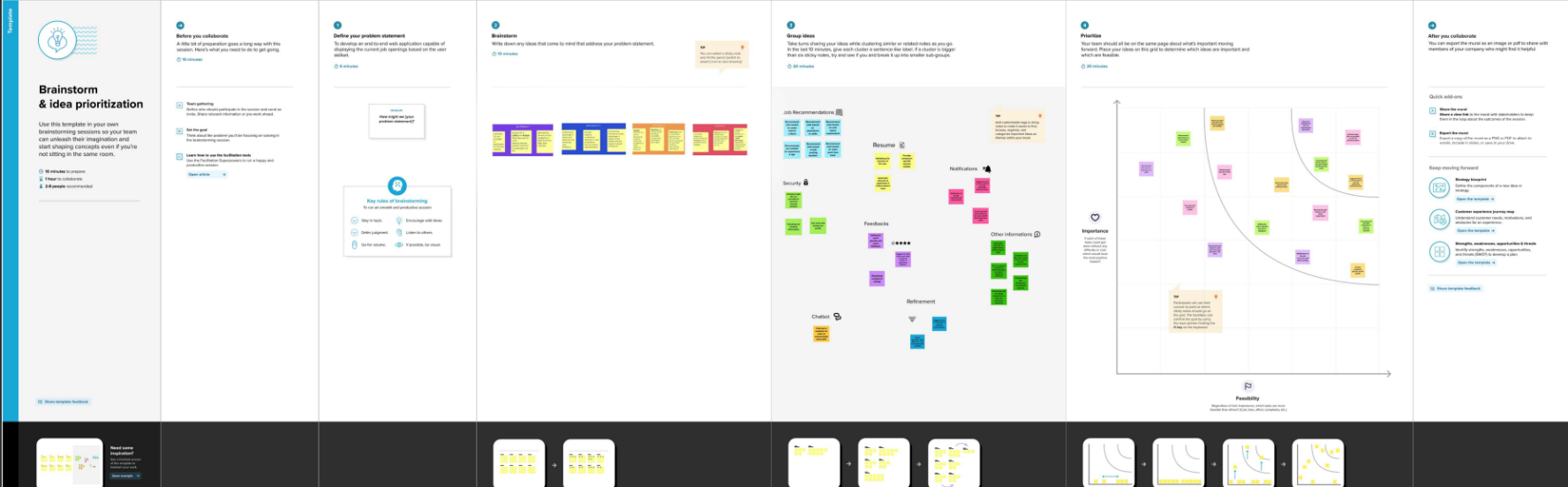
• These delays are responsible for large economic losses. It's important to provide better Airline and AirPort services and avoid delays in Air Travel across different locations and promise to get passengers from Location A to Location B on time.

**IDEATION & PROPOSED SOLUTION**

### 3.1 EMPATHY MAP CANVAS



### 3.2 IDEATION AND BRAINSTORMING



**3.3 PROBLEM SOLUTION FIT**

1. CUSTOMER SEGMENT Airlines literally bear high costs due to delays and cancellations that includes expenses on maintenance and compensations to travelers stuck in airports. With nearly 30 percent of the total delay time caused by unplanned maintenance, predictive analytics applied to fleet technical support is a reasonable solution. Customers are airline and airport services who are struggling to keep track of their forecasting data and planes arrival and departure.

2. JOBS-TO-BE-DONE / PROBLEMS Collecting data related to flight operations and inventory. You will use proprietary software like Airmax, for instance, or simple tools like Microsoft Excel to gather statistics related to important metrics called Key Performance Indicators (KPI). Optimising flight operations based on quantitative analysis. You will have to advise your management on trends and bottlenecks that you observe from data analysis so they can take the necessary action.

3. TRIGGERS In Aviation Industry,due to incidents like flight delays passenger may face delays in departure and arrival of flight. It is very hard to maintain the overall data.But if they use Data Analytics Report,Performance and Quality are reliable and profitable.

4. EMOTIONS: BEFORE / AFTER After:They feel like success after making increased profits,reducing the mistakes that happen in manual process. Before:They feel lost due to losses which occur due to improper management of Airline Analytics for Aviation Industry.

5. AVAILABLE Planning and Schedule Analytics: Provides in-depth analysis of ticket sales, operational expense and profitability of airline routes. It helps in fleet rebalancing, fuel needs and crew planning for a flight. Flight Turnaround Analytics: Provides insights on process inefficiencies in a flight turnover. The video annotation service helps to capture the time taken by each specific activity within flight turnover using video monitoring used for ground activities

6. CUSTOMER CONSTRAINTS Customer experience in the airline industry is often defined as what the customer perceives and experiences while traveling through the different departure stages and arrival in an airport. Mid-air: It is the best time to engage with passengers and understand their in-flight expectations. Start with the basics like seating comfort and crew etiquette. Post landing: Inspect through passengers' eyes and listen to their opinion. That's a great way to enhance your online reputation, post flight.

7.BEHAVIOUR Using airport analytics, data analysts can collect information on people who pass through various checks, like their gender, arrival times, baggage-check in times and the type of flight they take to better understand passenger behaviour. A better understanding of how passengers operate can be used to improve services.

8. ONLINE CHANNELS Online Airline Analytics for Aviation Industry which come for free may steal personal information of users and it may also contains a lot of ads. Security is not authenticated.

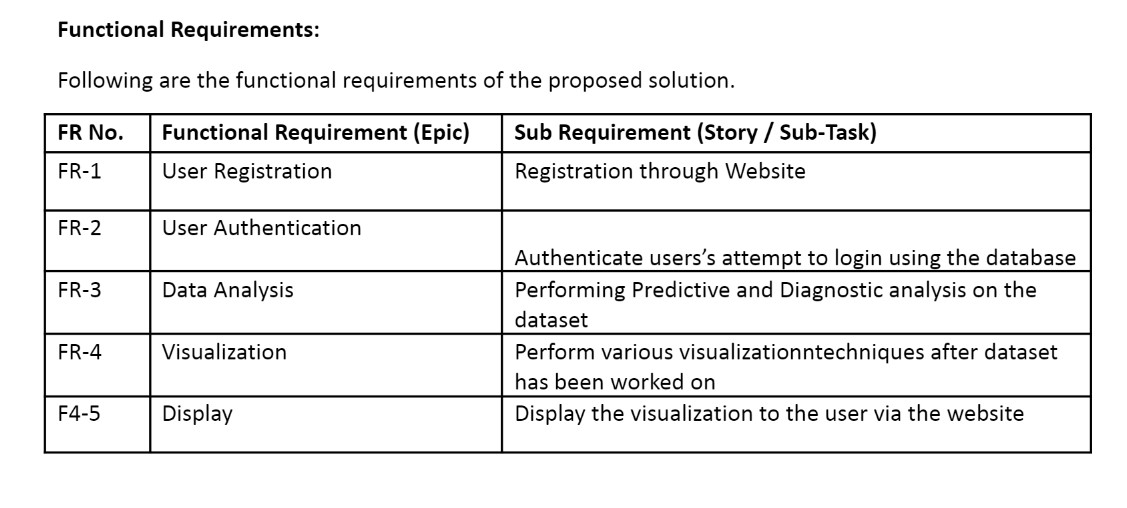
9. OFFLINE CHANNELS Manual logs can be maintained.Employees can be hired to maintain the airline analytics for aviation industry system logs when the business grows.

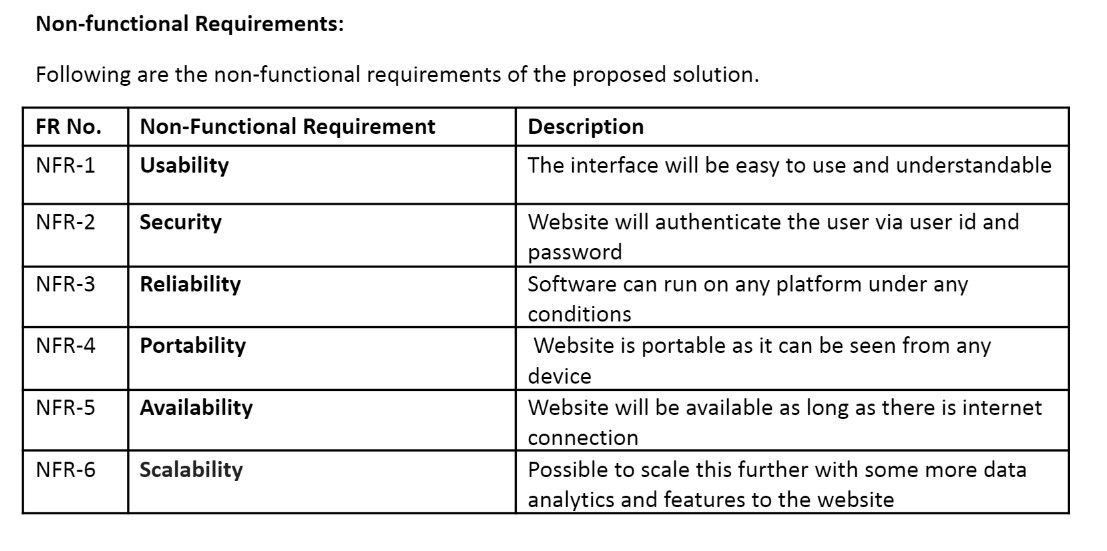
10. PROBLEM ROOT CAUSE A root cause analysis is performed as a reaction to risk management processes as defined in your aviation SMS manual. The purpose of the analysis is to understand the causal factors that trigger substandard safety performance within a particular event, whether the event is an, accident, minor incident, or close call.

11. YOUR SOLUTION To design an Airline Data Analytics Report for Aviation Industry using Cognos Analytics. Enable Email based alerts for arrival and departure of flight and it also sends messages related to the changes in configuration of flight path parameters. Provide a option for graphical view of aviation industry

# 4. REQUIREMENT ANALYSIS

### 4.1Functional Requirements And Non Functional Requirements



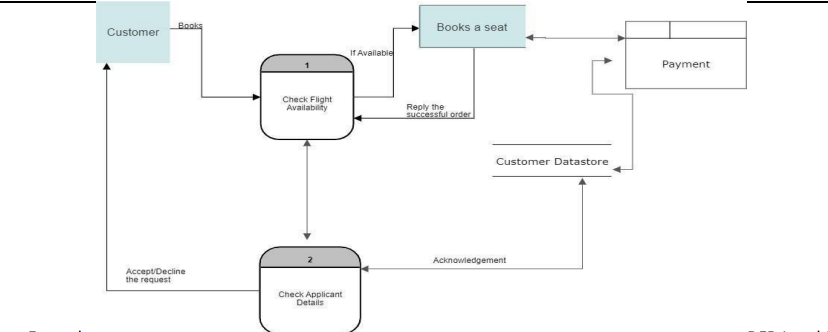


### 5.PROJECT DESIGN

#### 5.1 DATA FLOW DIAGRAMS

**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one.

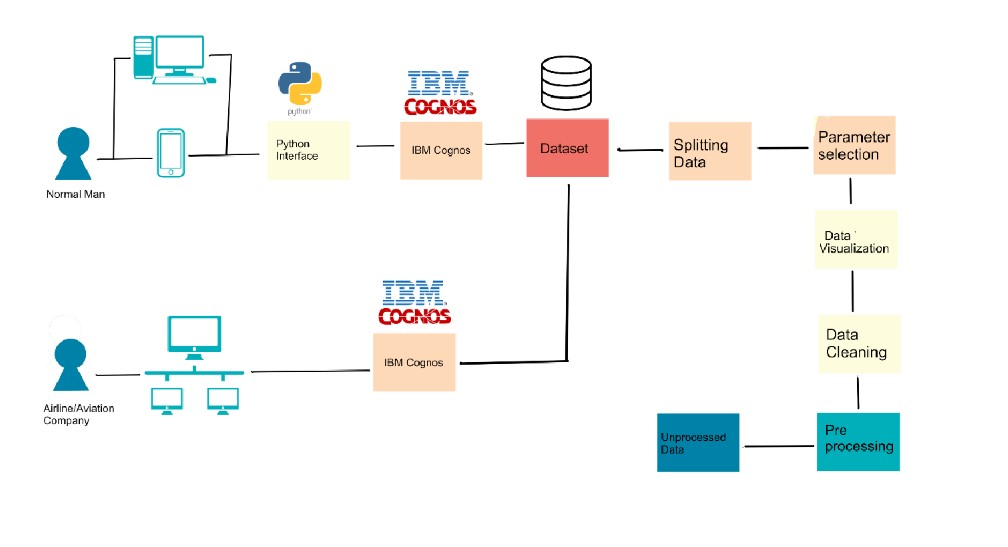


**5.2 SOLUTION AND TECHNICAL ARCHITECTURE Solution Architecture:**

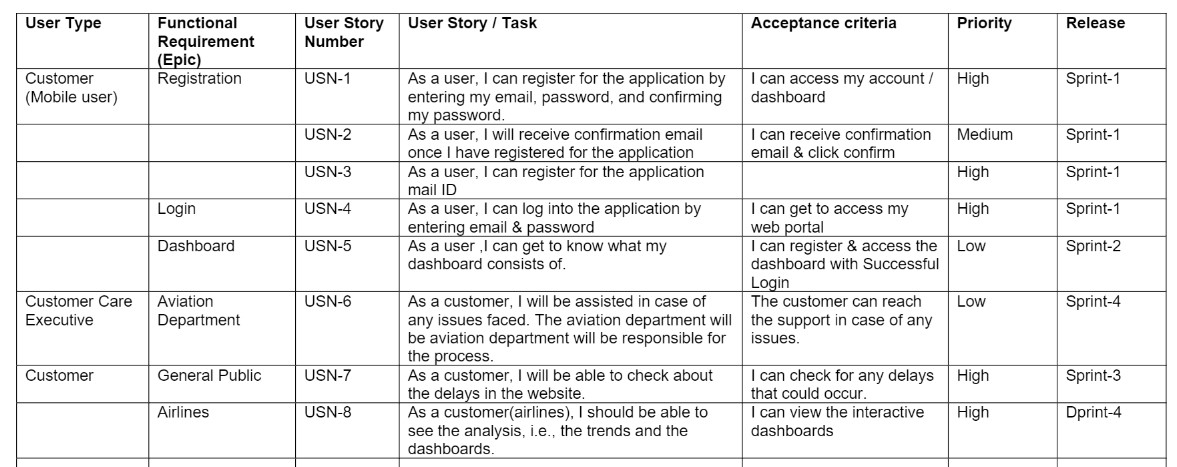
|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Component | Description | Technology |
| 1 | User Interface | User interacts using web application UI | HTML,CSS,JavaScript |
| 2 | Application logic1 | Logic for preprocessing | Python,numpy |
| 3 | Application logic2 | Logic for analyzing | Data visualization analysis using bigdata python |

|  |  |  |  |
| --- | --- | --- | --- |
| 4 | Application Logic3 | Delay prediction | Python using pandas and  matplotlib,seaborn |
| 5 | Database | Data base contains flight details and user credentials | Mysql |
| 6 | storage | File storage requirements | IBM block storage ,local file system |

**TECHNICAL ARCHITECTURE**

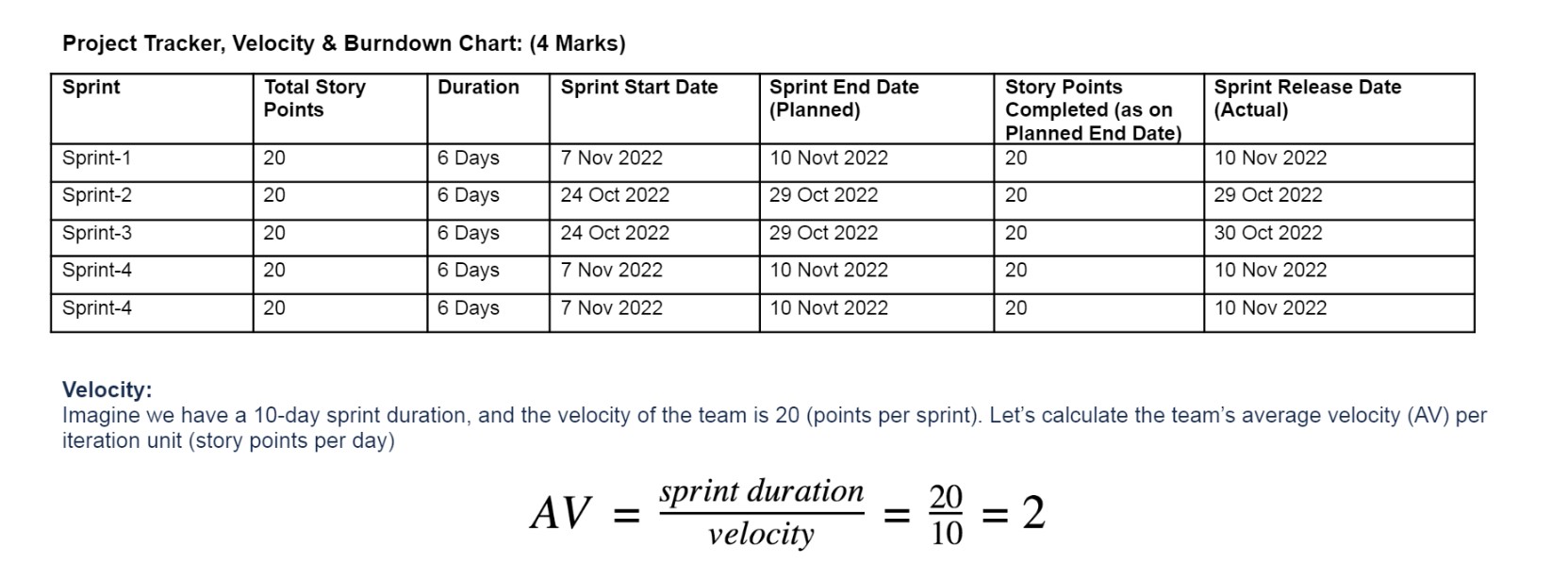
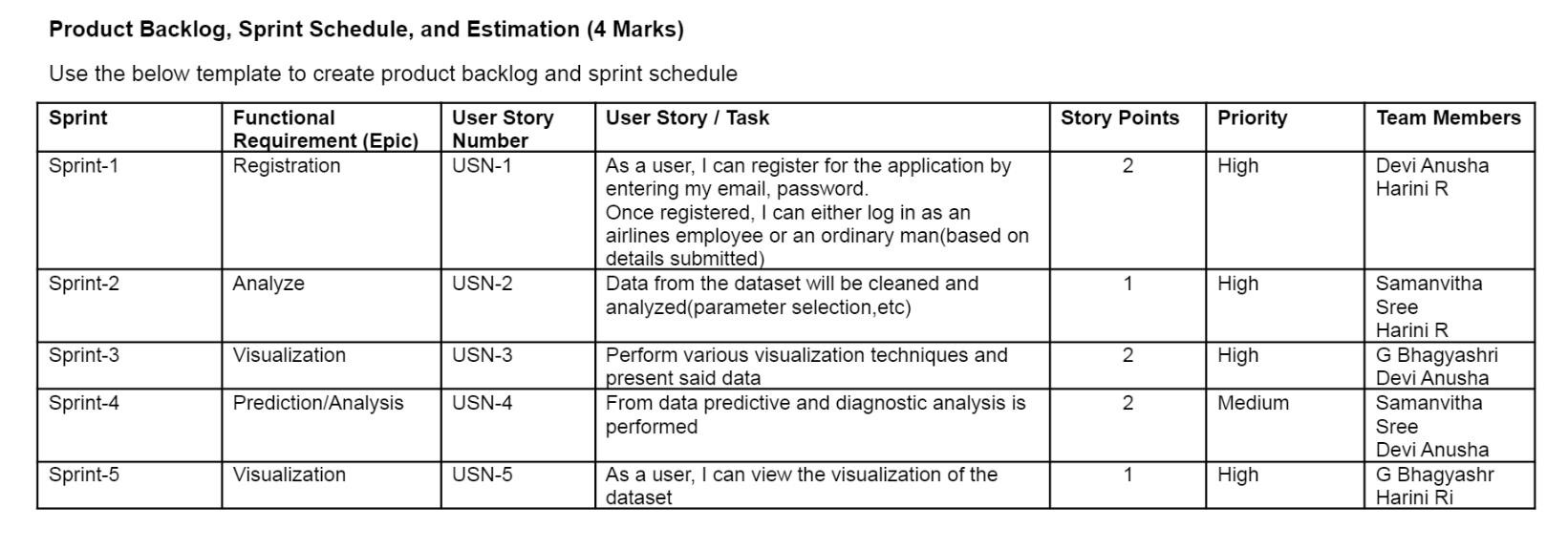


**5.3 User Stories**



## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation



### 6.2 Sprint Delivery Schedule

Phase 1

* Register or sign in the application by entering username , new password and confirming the same password ● Receive confirmation alert message once registered for the application. ● Log into the application by entering email and password

Phase 2

* Analytics to show delay of flights
* Comparing with other airlines which flights are frequently delayed
* Analytics to show weekly and everyday flight arrival,departure,airtime

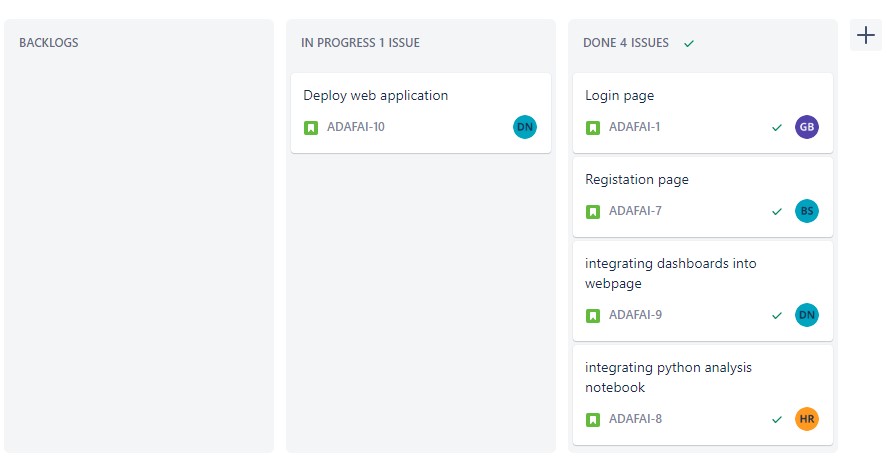
Phase 3

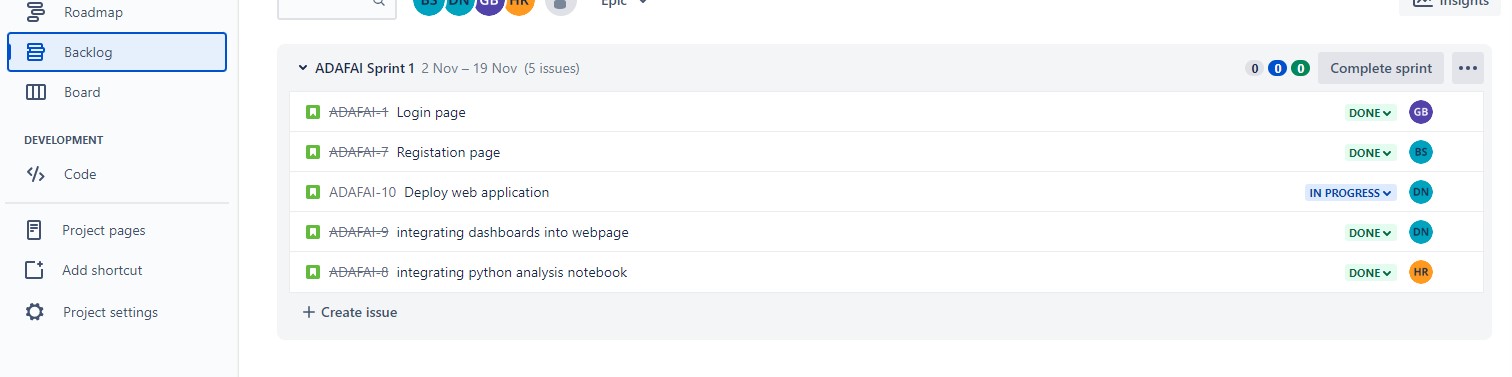
* Linking database with dashboard
* Secure the passwords
* Keep track of the delays

Phase 4

* Intergrate the IBM DB2, Sendgrid and other services
* Containerize the app and use IBM cloud to host the web app

# 6.3Reports from JIRA





### 7. CODIND AND SOLUTIONING

**Analysis:**

#### Login.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initialscale=1.0">

<!--<title> Responsive Login and Signup Form </title>-->

<!-- CSS -->

<link rel="stylesheet" href="style.css">

<!-- Boxicons CSS -->

<link href='https://unpkg.com/boxicons@2.1.2/css/boxicons.min.css' rel='stylesheet'>

</head>

<body style=" background: url(plane.png);background-size: auto;">

<section class="container forms">

<div class="form login">

<div class="form-content">

<header>Login</header>

<div id="login-error-msg-holder"> <p id="login-error-msg"></p>

</div>

<form id="login\_form">

<div class="field input-field">

<input type="text" id="username" placeholder="User name" class="input">

</div>

<div class="field input-field">

<input type="password" id="password" placeholder="Password" class="password">

<i class='bx bx-hide eye-icon'></i>

</div>

<div class="form-link">

<a href="#" class="forgot-pass">Forgot password?</a>

</div>

<div id="login-form-submit" class="field buttonfield">

<button onclick="login()">Login</button>

</div>

</form>

<div class="form-link">

<span>Don't have an account? <a href="main2.html" class="link signup-link">Signup</a></span>

</div>

</div>

<!-- Signup Form -->

</section>

<!-- JavaScript -->

<script src="script.js"></script>

</body>

</html>

#### SCRIPT.js

const loginForm = document.getElementById("login-form"); const loginButton = document.getElementById("login-form-submit"); const loginButton1 = document.getElementById("login-form-submit1");

var valid={"ad":"ad123","admin":"admin123","ibm":"ibm123"} function login(){ event.preventDefault();

var u = document.getElementById("username").value; var p = document.getElementById("password").value; console.log(u); console.log(p); if(u=="" && p==""){

alert("Enter username and password")

}

else if(u==""){

alert("Enter username")

}

else if(p==""){

alert("Enter password")

} f=0

for (let x in valid){ if(x==u){ if(valid[x]==p){

location.replace("index.html") f=1; break; } else{

document.getElementById("login-error-msg").innerHTML='Invalid password';

document.getElementsByName("username").value=""; document.getElementsByName("username").value="";

}

} } if(f==0){

document.getElementById("login-error-msg").innerHTML='Invalid username';

document.getElementsByName("username").value=""; document.getElementsByName("username").value="";

}

}

function signup(){ event.preventDefault();

var un = document.getElementById("uname").value; var ps = document.getElementById("pass").value; var rps = document.getElementById("rpass").value; var i=1;

for (let x in valid) { if(x==un) { i=0;

document.getElementById("login-error-msg1").innerHTML='User already exists';

document.getElementsByName("username").value=""; document.getElementsByName("username").value="";

} } if(i==1) { if(ps!=rps) {

document.getElementById("login-error-msg1").innerHTML='Password Mismatch';

document.getElementsByName("uname").value=""; document.getElementsByName("username").value="";

} else { valid[un]=ps; console.log(valid)

window.alert("Signup successful") event.preventDefault(); location.replace("main.html")

}

}

}

**SIGNUP.html**

<!DOCTYPE html>

<!-- Coding by CodingLab | www.codinglabweb.com-->

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<!--<title> Responsive Login and Signup Form </title>-->

<!-- CSS -->

<link rel="stylesheet" href="style.css">

<!-- Boxicons CSS -->

<link href='https://unpkg.com/boxicons@2.1.2/css/boxicons.min.css' rel='stylesheet'>

</head>

<body>

<section class="container forms"> <div class="form login">

<div class="form-content">

<header>Sign up</header>

<div id="login-error-msg-holder1">

<p id="login-error-msg1"></p>

</div>

<form id="signinform">

<div class="field input-field">

<input type="text" placeholder="Username" id="uname" class="input">

</div>

<div class="field input-field">

<input type="password" placeholder="Password" id="pass" class="password"> <i class='bx bx-hide eye-icon'></i>

</div>

<div class="field input-field">

<input type="password" placeholder="Retype Password" id="rpass" class="password">

<i class='bx bx-hide eye-icon'></i> </div>

<div class="form-link">

<a href="#" class="forgot-pass">Forgot password?</a>

</div>

<div id="login-form-submit1" class="field button-field">

<button onclick="signup()">Sign up</button>

</div>

</form>

<div class="form-link">

<span>Don't have an account? <a href="main.html" class="link signuplink">SignUp</a></span>

</div>

</div>

<!-- Signup Form -->

</section>

<!-- JavaScript -->

<script src="script.js"></script>

</body>

</html>

<!—

**<!DOCTYPE html>**

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="style.css">

<link href='https://unpkg.com/boxicons@2.1.2/css/boxicons.min.css' rel='stylesheet'>

</head>

<body>

<section class="container forms">

<div class="form signup">

<div class="form-content">

<header>Signup</header>

<form action="#">

<div class="field input-field">

<input type="email" placeholder="Email" class="input">

</div>

<div class="field input-field">

<input type="password" placeholder="Create password" class="password">

</div>

<div class="field input-field">

<input type="password" placeholder="Confirm password" class="password"> <i class='bx bx-hide eye-icon'></i> </div>

<div class="field button-field">

<button>Signup</button>

</div>

</form>

<div class="form-link">

<span>Already have an account? <a href="main.html" class="link loginlink">Login</a></span>

</div>

</div>

</div>

</section>

<!-- JavaScript -->

</body>

</html>-->

**Style.css**

@import

url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600&display=swap');

body{ background: url(plane.png);

background-size: auto;

} \*{ margin: 0; padding: 0; box-sizing: border-box;

font-family: 'Poppins', sans-serif;

}

.container{ height: 100vh; width: 100%; display: flex; align-items: center; justify-content: center; background-image: url(plane.png); background-size: auto;

column-gap: 30px;

}

.form{

position: absolute; max-width: 430px; width: 100%; padding: 30px; border-radius: 6px;

background-color:rgba(255, 255, 255, 0.75);

}

.form.signup{ opacity: 0;

pointer-events: none;

}

.forms.show-signup .form.signup{ opacity: 1;

pointer-events: auto;

}

.forms.show-signup .form.login{

opacity: 0;

pointer-events: none;

}

header{ font-size: 28px; font-weight: 600; color: #232836; text-align: center;

}

form{ margin-top: 30px;

}

.form .field{ position: relative; height: 50px; width: 100%; margin-top: 20px;

border-radius: 6px;

}

.field input, .field button{ height: 100%; width: 100%; border: none; font-size: 16px; font-weight: 400;

border-radius: 6px;

}

.field input{ outline: none; padding: 0 15px; border: 1px solid#CACACA;

}

.field input:focus{ border-bottom-width: 2px;

}

.eye-icon{ position: absolute; top: 50%; right: 10px; transform: translateY(-50%); font-size: 18px; color: #8b8b8b; cursor: pointer;

padding: 5px;

}

.field button{ color: #fff; background-color: #0171d3; transition: all 0.3s ease; cursor: pointer;

}

.field button:hover{

background-color: #016dcb;

}

.form-link{ text-align: center;

margin-top: 10px;

}

.form-link span, .form-link a{ font-size: 14px; font-weight: 400;

color: #232836;

} .form a{ color: #0171d3;

text-decoration: none;

}

.form-content a:hover{

text-decoration: underline;

} .line{ position: relative; height: 1px; width: 100%; margin: 36px 0;

background-color: #d4d4d4;

}

.line::before{ content: 'Or'; position: absolute; top: 50%; left: 50%;

transform: translate(-50%, -50%); background-color: #FFF; color: #8b8b8b;

padding: 0 15px;

}

.media-options a{ display: flex; align-items: center; justify-content: center;

}

a.facebook{

color: #fff;

background-color: #4267b2;

}

a.facebook .facebook-icon{ height: 28px; width: 28px; color: #0171d3; font-size: 20px; border-radius: 50%; display: flex; align-items: center; justify-content: center;

background-color: #fff;

}

.facebook-icon, img.google-img{ position: absolute; top: 50%;

left: 15px;

transform: translateY(-50%);

}

img.google-img{ height: 20px; width: 20px; object-fit: cover;

}

a.google{

border: 1px solid #CACACA;

}

a.google span{ font-weight: 500; opacity: 0.6; color: #232836;

}

@media screen and (max-width: 400px) {

.form{

padding: 20px 10px;

}

}

**Analysis.ipynb**

from mpl\_toolkits.mplot3d import Axes3D from sklearn.preprocessing import StandardScaler import matplotlib.pyplot as plt *# plotting* import numpy as np *# linear algebra*

import os *# accessing directory structure*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)* import matplotlib.pyplot as plt import seaborn as sns

There are 3 csv files in the current version of the dataset:

In [40]: for dirname, \_, filenames in os.walk('/kaggle/input'): for filename in filenames: print(os.path.join(dirname, filename))

/kaggle/input/airports.csv

/kaggle/input/airlines.csv

/kaggle/input/flights.csv

In [41]:*# Distribution graphs (histogram/bar graph) of column data*

def plotPerColumnDistribution(df, nGraphShown, nGraphPerRow):

nunique = df.nunique()

df = df[[col for col in df if nunique[col] > 1 and nunique[col] < 50]] *# For displaying purposes, pick columns that have between 1 and 50 unique values* nRow, nCol = df.shape columnNames = list(df)

nGraphRow = (nCol + nGraphPerRow - 1) / nGraphPerRow

plt.figure(num = None, figsize = (6 \* nGraphPerRow, 8 \* nGraphRow), dpi = 80, facecolor

= 'w', edgecolor = 'k') for i in range(min(nCol, nGraphShown)): plt.subplot(nGraphRow, nGraphPerRow, i + 1) columnDf = df.iloc[:, i] if (not np.issubdtype(type(columnDf.iloc[0]), np.number)): valueCounts = columnDf.value\_counts() valueCounts.plot.bar() else: columnDf.hist() plt.ylabel('counts') plt.xticks(rotation = 90)

plt.title(f'{columnNames[i]} (column {i})') plt.tight\_layout(pad = 1.0, w\_pad = 1.0, h\_pad = 1.0) plt.show()

In [42]:

*# Correlation matrix* def plotCorrelationMatrix(df, graphWidth): filename = df.dataframeName

df = df.dropna('columns') *# drop columns with NaN*

df = df[[col for col in df if df[col].nunique() > 1]] *# keep columns where there are more than 1 unique values* if df.shape[1] < 2: print(f'No correlation plots shown: The number of non-NaN or constant columns

({df.shape[1]}) is less than 2') return corr = df.corr()

plt.figure(num=None, figsize=(graphWidth, graphWidth), dpi=80, facecolor='w', edgecolor='k')

corrMat = plt.matshow(corr, fignum = 1)

plt.xticks(range(len(corr.columns)), corr.columns, rotation=90) plt.yticks(range(len(corr.columns)), corr.columns) plt.gca().xaxis.tick\_bottom() plt.colorbar(corrMat)

plt.title(f'Correlation Matrix for {filename}', fontsize=15) plt.show()

In [43]:

*# Scatter and density plots* def plotScatterMatrix(df, plotSize, textSize):

df = df.select\_dtypes(include =[np.number]) *# keep only numerical columns*

*# Remove rows and columns that would lead to df being singular* df = df.dropna('columns')

df = df[[col for col in df if df[col].nunique() > 1]] *# keep columns where there are more than 1 unique values* columnNames = list(df)

if len(columnNames) > 10: *# reduce the number of columns for matrix inversion of kernel density plots*

columnNames = columnNames[:10] df = df[columnNames]

ax = pd.plotting.scatter\_matrix(df, alpha=0.75, figsize=[plotSize, plotSize], diagonal='kde')

corrs = df.corr().values for i, j in zip(\*plt.np.triu\_indices\_from(ax, k = 1)): ax[i, j].annotate('Corr. coef = %.3f' % corrs[i, j], (0.8, 0.2), xycoords='axes fraction', ha='center', va='center', size=textSize) plt.suptitle('Scatter and Density Plot') plt.show()

Now you're ready to read in the data and use the plotting functions to visualize the data.

In [44]: nRowsRead = 1000 *# specify 'None' if want to read whole file*

*# airlines.csv may have more rows in reality, but we are only loading/previewing the first*

*1000 rows*

df1 = pd.read\_csv('/kaggle/input/airlines.csv', delimiter=',', nrows = nRowsRead) df1.dataframeName = 'airlines.csv' nRow, nCol = df1.shape

print(f'There are {nRow} rows and {nCol} columns')

There are 14 rows and 2 columns

In [45]:

Out[45]: In [46]: plotPerColumnDistribution(df1, 10, 5)

In [47]: nRowsRead = 1000 *# specify 'None' if want to read whole file*

*# airports.csv may have more rows in reality, but we are only loading/previewing the first*

*1000 rows*

df2 = pd.read\_csv('/kaggle/input/airports.csv', delimiter=',', nrows = nRowsRead) df2.dataframeName = 'airports.csv' nRow, nCol = df2.shape

print(f'There are {nRow} rows and {nCol} columns')

nRowsRead = 1000 *# specify 'None' if want to read whole file* *# flights.csv may have more rows in reality, but we are only loading/previewing the first 1000 rows*

df3 = pd.read\_csv('/kaggle/input/flights.csv', delimiter=',', nrows = nRowsRead)

df3.dataframeName = 'flights.csv' nRow, nCol = df3.shape

print(f'There are {nRow} rows and {nCol} columns')

plotPerColumnDistribution(df3, 10, 5) plotCorrelationMatrix(df3, 8) plotScatterMatrix(df3, 20, 10) def get\_shape(df): print('Now there are', df3.shape[0], 'rows and',df3.shape[1],'columns in this dataset')

from mpl\_toolkits.mplot3d import Axes3D from sklearn.preprocessing import StandardScaler import matplotlib.pyplot as plt *# plotting* import numpy as np *# linear algebra* import os *# accessing directory structure*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)* import matplotlib.pyplot as plt import seaborn as sns nRowsRead =100

df3 = pd.read\_csv('/kaggle/input/flights.csv', delimiter=',', nrows = nRowsRead)

df3.dataframeName = 'flights.csv'

fig, axss = plt.subplots(5,4, figsize=[15,10]) sns.boxplot(x='WEATHER\_DELAY', y ='AIRLINE', data=df3, ax=axss[0][0],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='FLIGHT\_NUMBER', data=df3, ax=axss[0][1],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='TAIL\_NUMBER', data=df3, ax=axss[0][2],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='ORIGIN\_AIRPORT', data=df3, ax=axss[0][3],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='DESTINATION\_AIRPORT', data=df3, ax=axss[1][0],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='SCHEDULED\_DEPARTURE', data=df3, ax=axss[1][1],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='DEPARTURE\_TIME', data=df3, ax=axss[1][2],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='DEPARTURE\_DELAY', data=df3, ax=axss[1][3],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='DISTANCE', data=df3, ax=axss[2][0],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='SCHEDULED\_ARRIVAL', data=df3, ax=axss[2][1],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='ARRIVAL\_TIME', data=df3, ax=axss[2][2],palette="Blues") sns.boxplot(x='WEATHER\_DELAY', y ='ARRIVAL\_DELAY', data=df3, ax=axss[2][3],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='CANCELLATION\_REASON', data=df3, ax=axss[3][0],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='AIR\_SYSTEM\_DELAY', data=df3, ax=axss[3][1],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='SECURITY\_DELAY', data=df3, ax=axss[3][2],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='AIRLINE\_DELAY', data=df3, ax=axss[3][3],palette="Blues")

sns.boxplot(x='WEATHER\_DELAY', y ='LATE\_AIRCRAFT\_DELAY', data=df3, ax=axss[4][0],palette="Blues")

*#sns.boxplot(x='Attrition', y ='', data=df, ax=axss[4][1],palette="Blues")*

plt.tight\_layout()

plt.savefig('numerical\_dist.png'); from mpl\_toolkits.mplot3d import Axes3D from sklearn.preprocessing import StandardScaler import matplotlib.pyplot as plt *# plotting* import numpy as np *# linear algebra* import os *# accessing directory structure*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)* import matplotlib.pyplot as plt import seaborn as sns

fig,axss = plt.subplots(2,4, figsize=[15,10]) nRowsRead = 10

df2 = pd.read\_csv('/kaggle/input/airports.csv', delimiter=',', nrows = nRowsRead)

df2.dataframeName = 'airports.csv'

sns.countplot(x='COUNTRY', hue='CITY', data=df2, ax=axss[0][0]) sns.countplot(x='COUNTRY', hue='STATE', data=df2, ax=axss[0][1]) sns.countplot(x='COUNTRY', hue='AIRPORT', data=df2, ax=axss[0][2]) sns.countplot(x='COUNTRY', hue='IATA\_CODE', data=df2, ax=axss[0][3]) *#sns.countplot(x='COUNTRY', hue='LALITUDE', data=df2, ax=axss[1][0])*

*#sns.countplot(x='COUNTRY', hue='LONGITUDE', data=df2, ax=axss[1][1])* plt.tight\_layout() plt.savefig('cate\_dist.png');

df2 = pd.read\_csv('/kaggle/input/airports.csv') df2.dataframeName = 'airports.csv' nRow, nCol = df2.shape

print(f'There are {nRow} rows and {nCol} columns') sns.heatmap(df2.corr(),annot=True)

**splitting\_data.ipynb**

import numpy as np *# linear algebra*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)*

*# Input data files are available in the read-only "../input/" directory* *# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directory*

import os for dirname, \_, filenames in os.walk('/kaggle/input'): for filename in filenames: print(os.path.join(dirname, filename))

*# You can write up to 20GB to the current directory (/kaggle/working/) that gets preserved as output when you create a version using "Save & Run All"*  *# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of the current session*

In [3]: import datetime, warnings, scipy import pandas as pd import numpy as np import seaborn as sns

import matplotlib.pyplot as plt

In [4]: import pandas as pd

data = pd.read\_csv("../input/flight-delays/flights.csv")

keep\_col=['MONTH','WEATHER\_DELAY','ORIGIN\_AIRPORT','DESTINATION\_AIRPORT'] new\_f=data[keep\_col]

new\_f.to\_csv("weather.csv",index=False) new\_f.head()

keep\_col=['ORIGIN\_AIRPORT','DEPARTURE\_DELAY','ARRIVAL\_DELAY'] new\_f=data[keep\_col]

new\_f.to\_csv("airportdelay.csv",index=False) new\_f.info()

keep\_col=['AIRLINE','DIVERTED','CANCELLED','CANCELLATION\_REASON','AIR\_SYSTEM\_

DELAY','SECURITY\_DELAY','AIRLINE\_DELAY','LATE\_AIRCRAFT\_DELAY'] new\_f=data[keep\_col]

new\_f.to\_csv("airline.csv",index=False) new\_f.head() index.html

<body

border="10px">

<title >AIRLINES ANALYSIS FOR AVIATION INDUSTRY</title>

<center><h3>AIRLINES ANALYSIS FOR AVIATION INDUSTRY</h3> <div style="width: 500px;height:20px; border:1px solid black;padding:20px;postion:absolute;left:35%;top:100px;background-

color:#016dcb"><a href="main1.html"><button>Merged

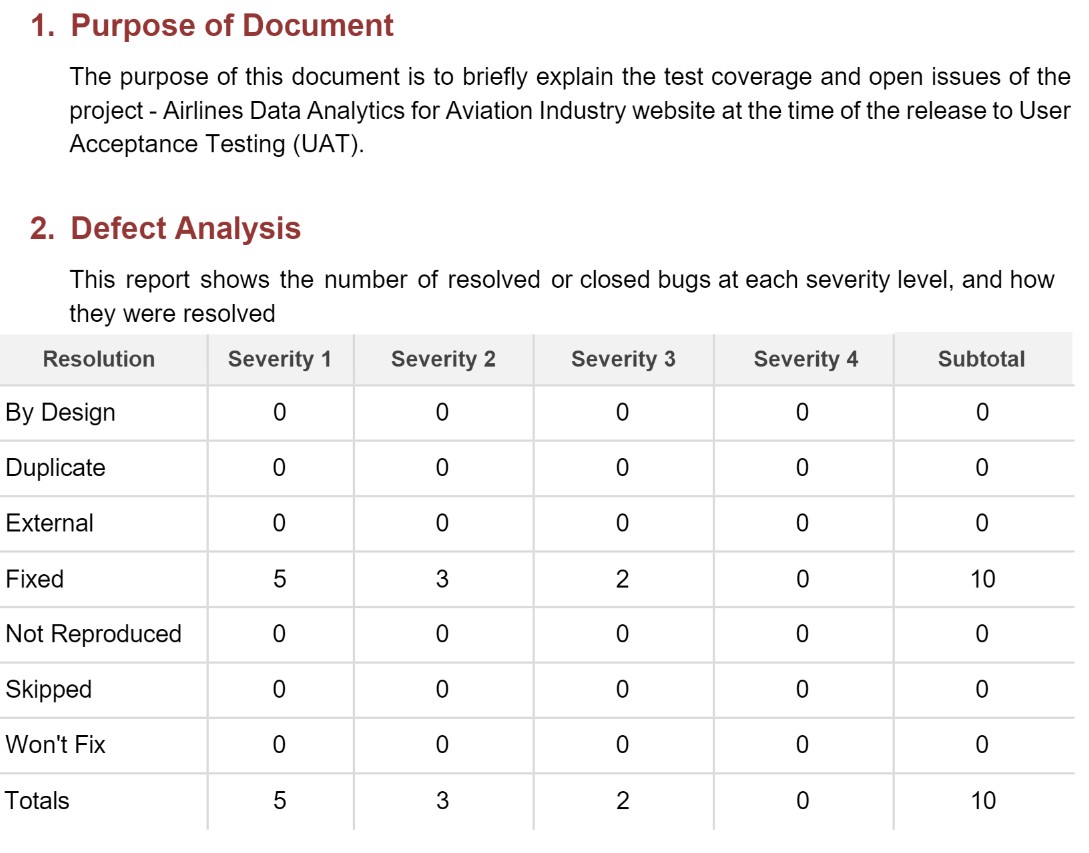
Analysis</button></a>

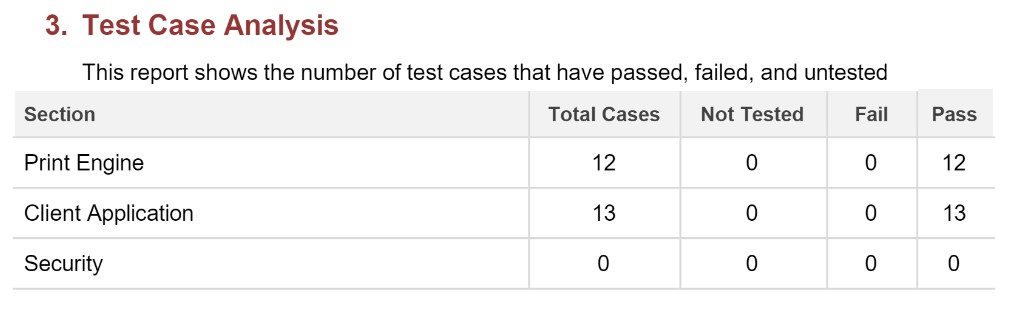
<a href="analysisAirlines.html"><button>Simple Analysis</button></a> <a href="dash.html"><button>Dashboard2</button></a></div></center> <br><br>

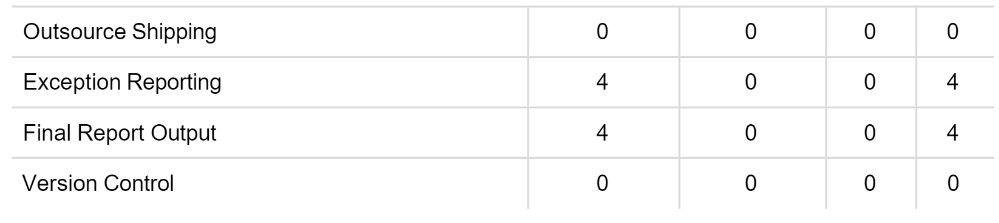
<embed src="Sprint\_dashboard.pdf" width="100%" height="2100px" />

</body>

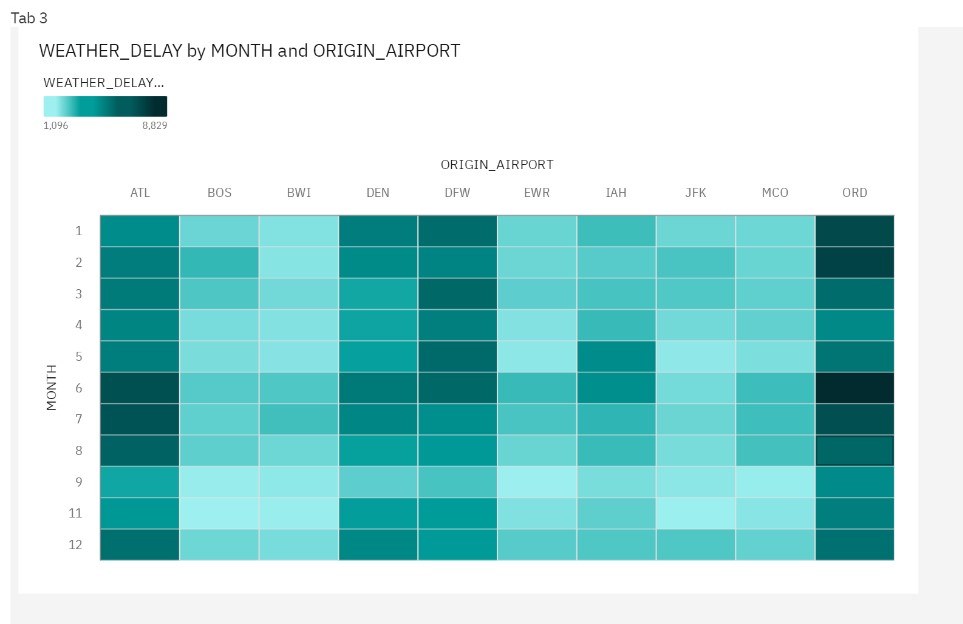
### 8. TESTING

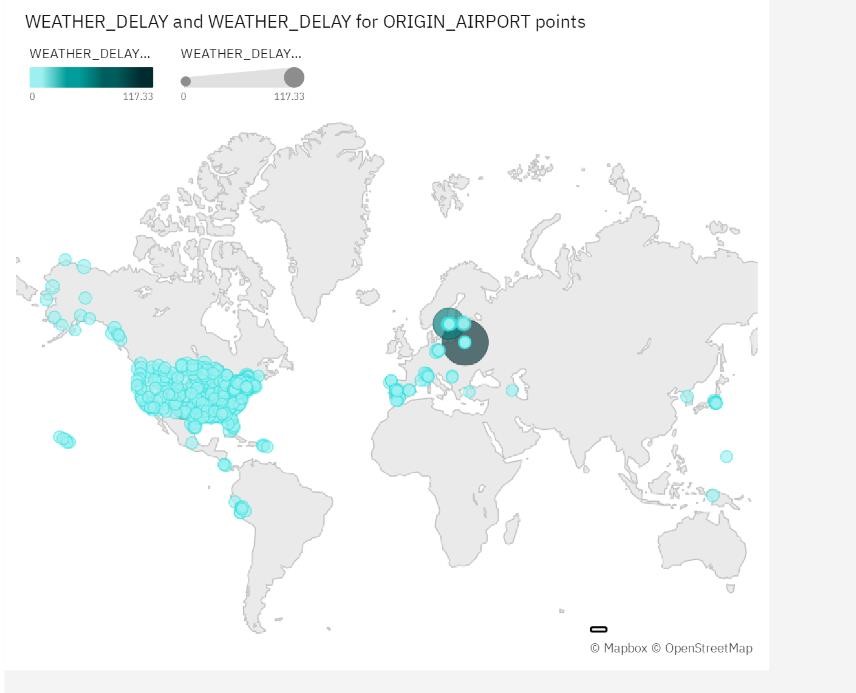
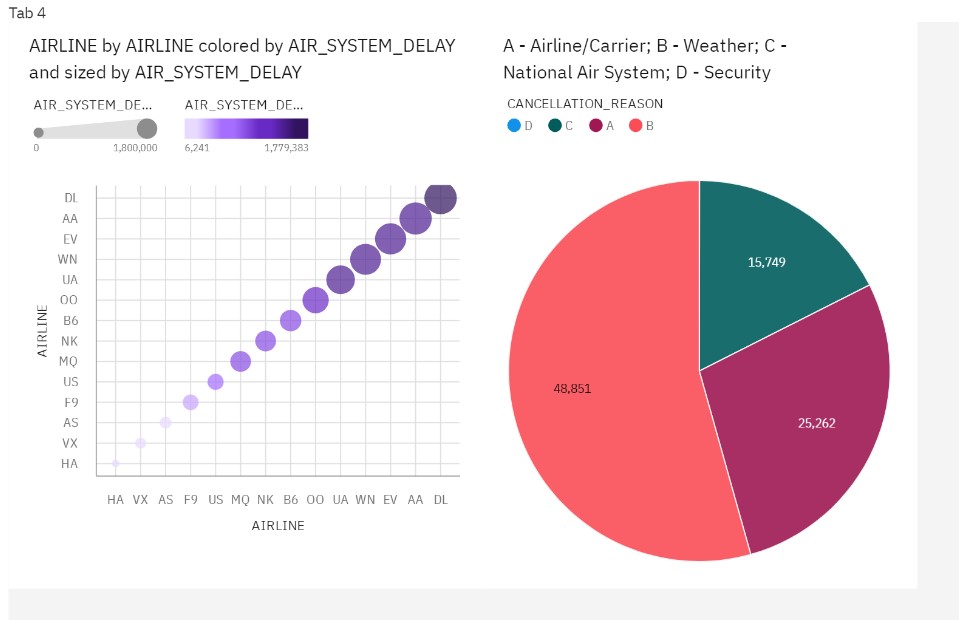
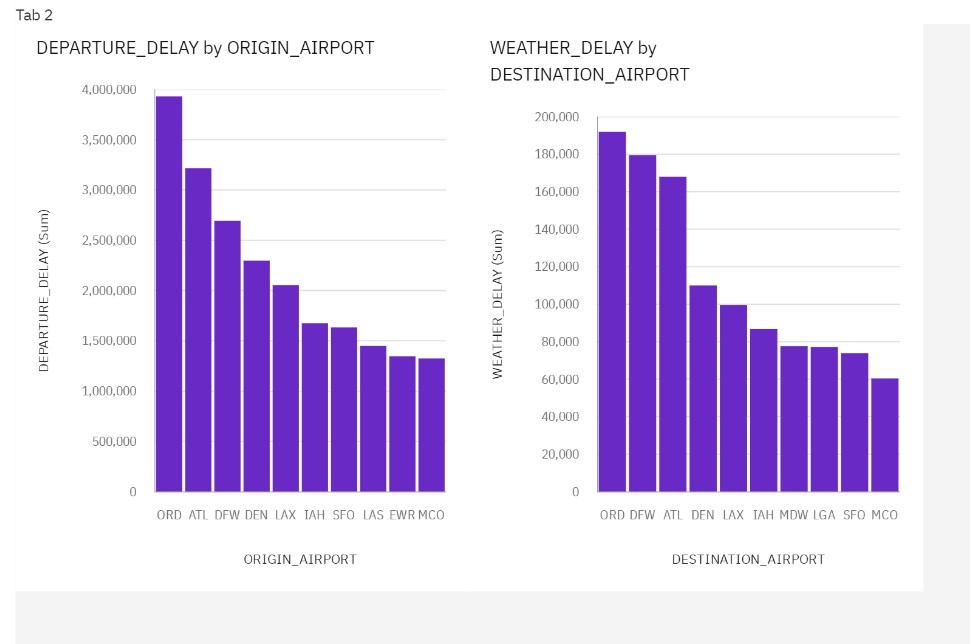
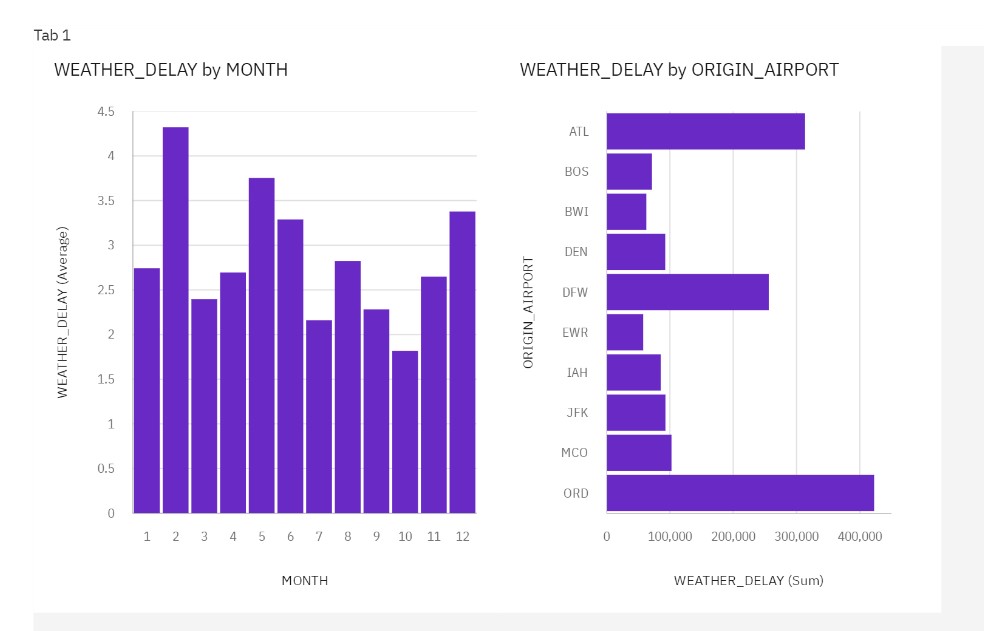






#### 9.RESULTS





#### 10.ADVANTAGES AND DISADVANTAGES

**Advantage:**

1) With this application, we can easily analyze flight delays and simplify the extensive traffic at the airport and can prevent the major confusions over flight delays. 2) This can enable customer satisfaction and incomes of major airlines.

1. Accuracy is measured with the previous models and we have analyzed that this model is much more effective in every way.
2. The delay prediction can make the concerned authorities be well prepared for any possible problem.
3. can easily be understood by a layman: the model is simple and effective.

#### Disadvantages

1.This application needs to be more compact and flexible. The interoperability feature should be more enhanced.

2. The application can be automated instead of static data from the user(airport authorities).

### 11. CONCLUSION

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.

### 12. FUTURE SCOPE

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.APPENDIX

**GITHUB AND PROJECT DEMO LINK**

<https://github.com/IBM-EPBL/IBM-Project-27782-1660065879>