

|              |  |
|--------------|--|
| DATE         | 18 NOVEMBER 2022                                     |
| TEAM ID      | PNT2022TMID00517                                     |
| PROJECT NAME | Developing a Flight delay prediction model           |
| TEAM MEMBERS | Priyadharshini,Manjari,Lavanya,Mrinalini,Madhu mathi |

## APPLICATION BUILDING USING FLASK:

```
from flask import Flask,render_template,request
```

```
import requests
```

# NOTE: you must manually set API\_KEY below using information retrieved from your IBM Cloud account.

# NOTE: you must manually set API\_KEY below using information retrieved from your IBM Cloud account.

```
API_KEY = "2SNGxCC84_SnT4w-  
CK18BSgHa22dH7hgM673se9fq57B"
```

```
token_response =  
requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":  
    API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})  
mltoken = token_response.json()["access_token"]
```

```
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' +  
mltoken}
```

```
app=Flask(__name__)
```

```
@app.route('/')  
def index():
```

```
    return render_template('index.html')
```

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

        dayofmonth=request.form["dayofmonth"]
        if(int(dayofmonth)>31):
            ans="Please Enter the correct Day of Month"
            return render_template("index.html" ,y=ans)

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

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

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

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

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

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

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

if(destination=="dtw"):

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

if(destination=="jfk"):

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

if(destination=="sea"):

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

if(destination=="alt"):

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

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

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

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

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

else:

dept15=1

print(dept15)

total=[[month,dayofmonth,dayofweek,origin1,origin2,origin3,origin4,origin5,destination1,destination2,destination3,destination4,destination5,dept,actdept,dept15,arrrtime]]

# NOTE: manually define and pass the array(s) of values to be scored in the next line

payload\_scoring = {"input\_data": [{"fields": ["f0", "f1", "f2", "f3", "f4", "f5", "f6", "f7", "f8", "f9", "f10", "f11", "f12", "f13", "f14", "f15", "f16"], "values": total}]}

response\_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/74fb0eec-a7f5-4bb6-ab8b-d423e91a872c/predictions?version=2022-11-16', json=payload\_scoring,

headers={'Authorization': 'Bearer ' + mltoken})

print("Scoring response")

print(response\_scoring.json())

pred = response\_scoring.json()

value = pred['predictions'][0]['values'][0][0]

print(value)

if(value==[0.]):

```
    ans="THE FLIGHT WILL BE ON TIME"  
else:  
    ans="THE FLIGHT WILL BE DELAYED"
```

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

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