PROJECT DEVELOPMENT PHASE

Sprint – IV Code and Test cases

Date	16-Nov-2022
Team ID	PNT2022TMID18451
Project Name	Developing a Flight Delay Model Using Machine Learning
Maximum Marks	8 Marks

Integration the Deployed Model with Flask

Web Application using IBM Cloud

App_ibm.py :

```
from flask import Flask,request,
render_template  import numpy as np import
pandas as pd import pickle import os import
requests

# NOTE: you must manually set API_KEY below using information retrieved from
your IBM Cloud account.
API_KEY = "ET9hVXAyLGZV0zduulIuo5ZEx_fzg6Q_4w721luSJ_9r"
token_response =
requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
API_KEY, "grant_type": 'urn:ibm:params:oauth:granttype:apikey'}) mltoken
= token_response.json()["access_token"]
 header = {'Content-Type': 'application/json', 'Authorization': 'Bearer '
+ mltoken}


app=Flask(__name__)
@app.route('/') def
home():
```

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

@app.route('/predicts', methods=['POST','GET'])   def
predict():
    name=request.form['name']
month=request.form['month']
dayofmonth=request.form['dayofmonth']
dayofweek=request.form['dayofweek']
origin=request.form['origin']     if(origin=="msp"):
        origin1,origin2,origin3,origin4,origin5=0,0,0,0,1
if(origin=="dtw"):
        origin1,origin2,origin3,origin4,origin5=1,0,0,0,0
if(origin=="jfk"):
        origin1,origin2,origin3,origin4,origin5=0,0,1,0,0
if(origin=="sea"):
        origin1,origin2,origin3,origin4,origin5=0,1,0,0,0
if(origin=="alt"):
        origin1,origin2,origin3,origin4,origin5=0,0,0,1,0


    destination=request.form['destination']
if(destination=="msp"):
        destination1,destination2,destination3,destination4,destination5=0,0,0
,0,1
if(destination=="dtw"):
        destination1,destination2,destination3,destination4,destination5=1,0,0
,0,0
if(destination=="jfk"):
        destination1,destination2,destination3,destination4,destination5=0,0,1
,0,0
if(destination=="sea"):
        destination1,destination2,destination3,destination4,destination5=0,1,0
,0,0
if(destination=="atl"):
        destination1,destination2,destination3,destination4,destination5=0,0,0
,1,0
    dept=request.form['dept']
arrtime=request.form['arrtime']
actdept=request.form['actdept']        dept15
= int(dept) - int(actdept)
    total=[[name,month,dayofmonth,dayofweek,origin1,origin2,origin3,origin4,or
igin5,destination1,destination2,destination3,destination4,destination5,dept,ar
rtime]]
    # y_pred=model.predict(total)
    # print(y_pred)
```

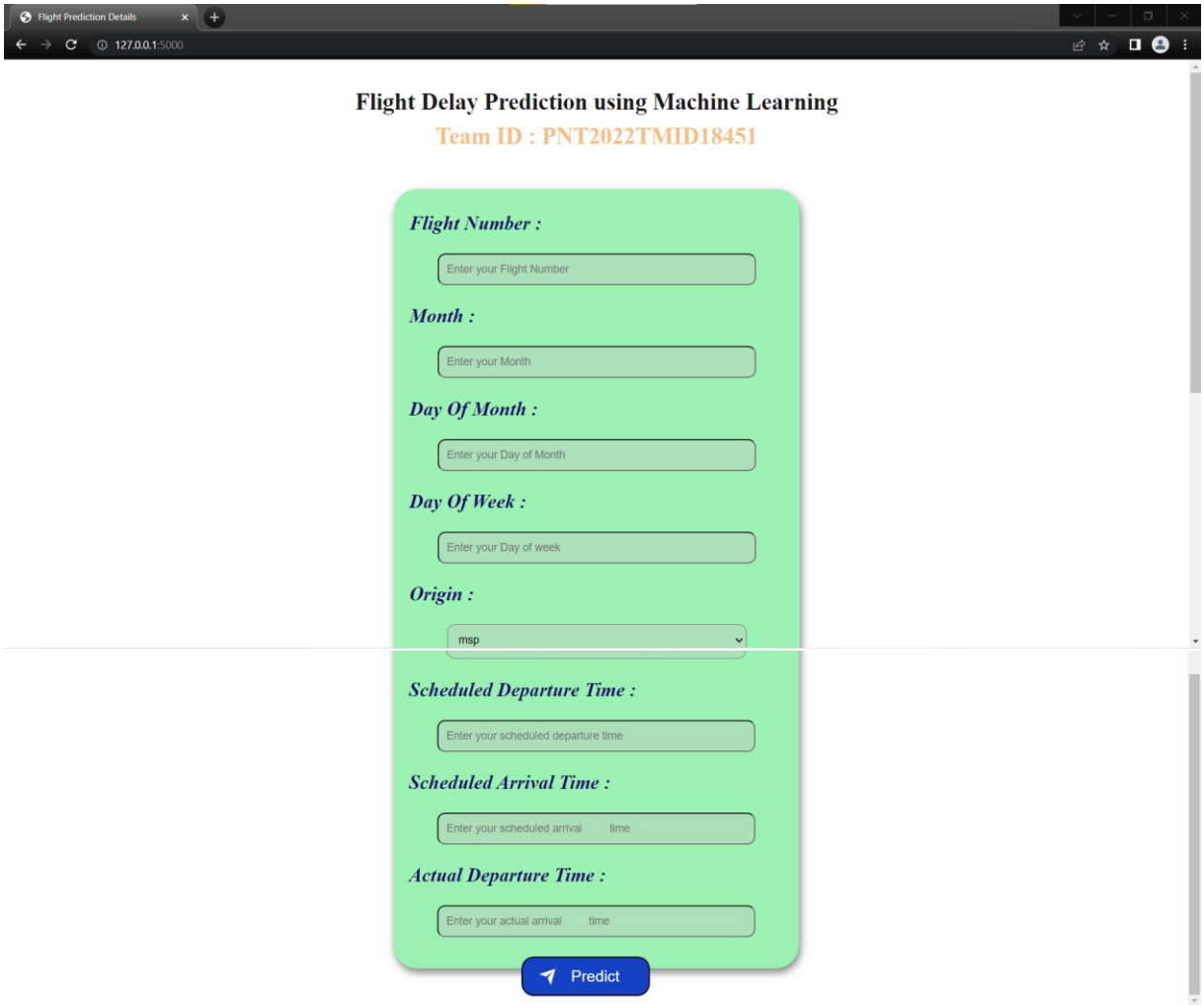
```
    payload_scoring = {"input_data": [{"field":
[['name','month','dayofmonth','dayofweek','origin1','origin2','origin3','origi
n4','origin5','destination1','destination2','destination3','destination4','des
tination5','dept','arrtime']], "values": total}]}     response_scoring =
requests.post('https://ussouth.ml.cloud.ibm.com/ml/v4/deployments/a7a269f3-
d3c1-4e2d-85b247e1bf6bbfee/predictions?version=2022-10-13',
json=payload_scoring, headers={'Authorization': 'Bearer ' + mltoken})
print(response_scoring)     predictions = response_scoring.json()     output =
predictions['predictions'][0]['values'][0][0]     print(output)

         if(output
==[0.]):
        ans="The Flight will be on time"
else:
        ans="The Flight will be Delayed"

          return
render_template("predict.html",showcase=ans)

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

Search web page :



Result web page :

The Flight will be Delayed

Go Back