

PROJECT DEVELOPMENT PHASE

SPRINT 4 SOURCE CODE

Date	15-Nov-2022
Team ID	PNT2022TMID27775
Project Name	Developing a Flight Delay Model Using Machine Learning
Maximum Marks	8 Marks

Integrating Deployed Model with Flask Web Application

App.py

```
from flask import Flask, render_template, request

import requests

import requests

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

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM
Cloud account.
API_KEY = "gyOvc0l0Hde4zdTmNc47N4Vh1zmMTFh7FIK8BEcKPADB"
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} import

mysql.connector

app = Flask(__name__)

conn=mysql.connector.connect(host="localhost", user="root", password="", database="login")
cursor=conn.cursor()

@app.route('/') def
index():
    return render_template('index.html')

@app.route('/login')
def login(): # put application's code here return
    render_template('login.html')

@app.route('/register')
def register():
    return render_template('register.html')

@app.route('/home')
def home():
    return render_template('home.html')

@app.route('/service')
def service():
    return render_template('service.html')

@app.route('/about') def
about():
    return render_template('about.html')

@app.route('/login_validation', methods=['POST'])
def login_validation():
    email=request.form.get('email')
    password=request.form.get('password')

    cursor.execute("""SELECT * FROM `users` WHERE `email` LIKE'{' AND `password`
LIKE '{ }'""".format(email,password)) users
    = cursor.fetchall()

```

```

if len(users)>0: return
    render_template('home.html')
else:
    return render_template('login.html', prediction_text = "1" )
@app.route('/add_user', methods=['POST'])
def add_user():
    name=request.form.get('name')
    email=request.form.get('email')
    password=request.form.get('password')

    cursor.execute("""INSERT INTO `users`(`id`,`name`,`email`,`password`) VALUES
(NULL,'{}','{}','{}')""".format(name,email,password)) conn.commit()
    return render_template('login.html', prediction_text = "0")

@app.route('/predict',methods=['POST'])
def predict():

    year = request.form['year']
    month = request.form['month']
    day = request.form['day']
    carrier = request.form['carrier']
    origin = request.form['origin']
    dest = request.form['dest']

    if (carrier=="UA"):
        carrier=11
    if (carrier=="AA"):
        carrier=1
    if (carrier=="B6"):
        carrier=3
    if (carrier=="DL"):
        carrier=4
    if (carrier=="EV"):
        carrier=8
    if (carrier=="MQ"):
        carrier=9
    if (carrier=="US"):
        carrier=12
    if (carrier=="WN"):
        carrier=14

    if

    if

    if

```

```
if (carrier=="VX"):
    carrier=13
(carrier=="FL"):
    carrier=7
(carrier=="AS"):
    carrier=2
(carrier=="9E"):
    carrier=0
```

```
(carrier=="F9"):
carrier=9
(carrier=="HA"):
carrier=4
(carrier=="OO"):
carrier=5
if (carrier=="YV"):
carrier=15
```

```
if (origin=="EWR"):
origin=0
if (origin=="LGA"):
origin=2
if (origin=="JFK"):
origin=1
```

```
if (dest=="ATL"):
dest=4
if (dest=="IAH"):
dest=43
if (dest=="MIA"):
dest=57
if (dest=="BQN"):
dest=12
if (dest=="ORD"):
dest=68
if (dest=="FLL"):
dest=35
if (dest=="IAD"):
dest=42
if (dest=="MCO"):
dest=53
if (dest=="PBI"):
dest=70
if (dest=="TPA"):
dest=99
if (dest=="LAX"):
dest=49
if (dest=="SFO"):
dest=89
```

```
if
```

```
if
```

```
if
```

if

if

if

```
if (dest=="DFW"):
    dest=30
if (dest=="BOS"):
    dest=11
    if
    (dest=="LAS"):
        dest=48
if (dest=="MSP"):
    dest=60
if (dest=="DTW"):
    dest=32
if (dest=="RSW"):
    dest=82
if (dest=="SJU"):
    dest=91
if (dest=="PHX"):
    dest=73
if (dest=="BWI"):
    dest=16
if (dest=="CLT"):
    dest=23
if (dest=="BOS"):
    dest=11
if (dest=="BUF"):
    dest=14
if (dest=="DEN"):
    dest=29
if (dest=="SNA"):
    dest=94
if (dest=="LAS"):
    dest=48
if (dest=="MSY"):
    dest=61
if (dest=="SLC"):
    dest=92
if (dest=="SEA"):
    dest=88
if (dest=="ROC"):
    dest=99
if (dest=="ATL"):
    dest=4
```

```
if (dest=="DCA"):
    dest=33
if (dest=="RDU"):
    dest=4
    (dest=="BNA"):
    dest=4 (dest=="CLE"):
    dest=88
    (dest=="STL"):
    dest=82
    (dest=="MDW"):
    dest=99
    (dest=="CVG"):
    dest=68
    (dest=="CMH"):
    dest=68
if (dest=="CHS"):
    dest=99
if (dest=="PIT"):
    dest=1
if (dest=="SAN"):
    dest=82
if (dest=="MKE"):
    dest=11
if (dest=="JAX"):
    dest=88
if (dest=="BTV"):
    dest=4
if (dest=="AUS"):
    dest=23
if (dest=="RIC"):
    dest=64
if (dest=="PWM"):
    dest=83
if (dest=="HOU"):
    dest=89
if (dest=="IND"):
    dest=47
if (dest=="MCI"):
    dest=80
if (dest=="SYR"):
    dest=78
```

if

if

if

if

if

if

```
if (dest=="BWI"):
    dest=4
if (dest=="MEM"):
    dest=23
if (dest=="PHL"):
    dest=14
if (dest=="GSO"):
    dest=96
if (dest=="ORF"):
    dest=23
if (dest=="DAY"):
    dest=57
    if
    (dest=="PDX"):
    dest=83
if (dest=="SRQ"):
    dest=91
if (dest=="SDF"):
    dest=29
if (dest=="XNA"):
    dest=88
if (dest=="MHT"):
    dest=43
if (dest=="BDL"):
    dest=23
if (dest=="OMA"):
    dest=4
if (dest=="GSP"):
    dest=57
if (dest=="SAV"):
    dest=28
if (dest=="GRR"):
    dest=16
if (dest=="HNL"):
    dest=24
if (dest=="SAT"):
    dest=30
if (dest=="TYS"):
    dest=99
if (dest=="MSN"):
    dest=55
```



```

if (dest=="DSM"):
    dest=23
if (dest=="STT"):
    dest=23
if (dest=="ALB"):
    dest=99
if (dest=="BUR"):
    dest=41
if (dest=="PVD"):
    dest=32
if (dest=="PSE"):
    dest=96
    (dest=="OKC"):
    dest=61
    (dest=="TUL"):
    dest=60
    (dest=="SMF"):
    dest=88
    (dest=="ACK"):
    dest=11
    (dest=="AVL"):
    dest=10
    (dest=="ABQ"):
    dest=30
if (dest=="MVY"):
    dest=68
if (dest=="EGE"):
    dest=32
if (dest=="CRW"):
    dest=4
if (dest=="ILM"):
    dest=79
if (dest=="CAE"):
    dest=69

```

```

t=[[int(year),int(month),int(day),int(carrier),int(origin),int(dest)]]

```

```

payload_scoring = {"input_data": [{"fields": ["f0","f1","f2","f3","f4","f5"], "values":t }]}
#payload_scoring = {"input_data": [{"fields": [array_of_input_fields], "values":
[array_of_values_to_be_scored, another_array_of_values_to_be_scored] ]}]

```

```

if

```

```

if

```

```

if

```

if

if

if

```
response_scoring = requests.post('https://us-  
south.ml.cloud.ibm.com/ml/v4/deployments/f4014f53-d84e-4c2a-9dd2-  
e36cd70e6b22/predictions?version=2022-11-04', json=payload_scoring,  
headers={'Authorization': 'Bearer ' + mltoken})
```

```
print("Scoring response")
```

```
payload_scoring = {"input_data": [{"fields": ["f0", "f1", "f2", "f3", "f4", "f5"], "values": t }]}  
pred= response_scoring.json() output=pred['predictions'][0]['values'][0][0] print(output)  
return render_template('home.html', prediction_text = output)
```

```
if __name__ == '__main__':
```

```
    app.run(debug=True)
```

```
# For mac, make 'app.run(debug=True)'
```