Sprint-3

Date	18 November 2022
Team ID	PNT2022TMID37323
Project Name	Predicting the energy output of wind turbine based on weather condition

app.py

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import joblib
import requests
from dotenv import load_dotenv
from os import getenv


load_dotenv()
apikey=getenv('API_KEY')

app = Flask(__name__)
model = joblib.load('Power_Prediction.sav')


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


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


@app.route('/windapi',methods=['POST'])
def windapi():
    city=request.form.get('city')
```

```
    url="http://api.openweathermap.org/data/2.5/weather?q="+city+"&appid="+apikey
    resp = requests.get(url)
    resp=resp.json()
    temp = str((resp["main"]["temp"])-273.15) +" °C"
    humid = str(resp["main"]["humidity"])+" %"
    pressure = str(resp["main"]["pressure"])+" mmHG"
    speed = str((resp["wind"]["speed"])*3.6)+" Km/s"
    return render_template('predict.html', temp=temp, humid=humid, pressure=pressure,speed=speed)
@app.route('/y_predict',methods=['POST'])
def y_predict():
    '''
    For rendering results on HTML GUI
    '''
    x_test = [[float(x) for x in request.form.values()]]
    prediction = model.predict(x_test)
    print(prediction)
    output = prediction[0]
    return render_template('predict.html', prediction_text='The energy predicted is {:.2f} KWh'.format(output))


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