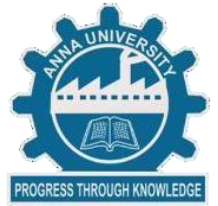# PREDICTING THE ENERGY OUTPUT OF WIND TURBINE BASED ON WEATHER CONDITION

IBM NALAIYATHIRAN REPORT

Submitted by

K. SINDHUJA -510619205038

G. KAAVYA- 510619205010

J. RABIYA TABASSUM-510619205026

I.S. SHAMEEM FAATHIMA-510619205035

B. KEERTHANA-510619205014

Date	19 November 2022
Team ID	**PNT2022TMID39678**
Project Name	**Predicting the energy output of wind turbine based on weather condition**
Maximum Marks	8 Marks

C.ABDUL HAKEEM COLLEGE OF ENGINEERING AND TECHNOLOGY

RANIPET-632509.

LIST OF CONTENTS:

1.INTRODUCTION :

Wind speed/power has received increasing attention around the earth due to its renewable nature as well as environmental friendliness.

With the global installed wind power capacity rapidly increasing, the wind industry is growing into a large-scale business. Reliable short-term wind speed forecasts play a practical and crucial role in wind energy conversion systems, such as the dynamic control of wind turbines and power system scheduling.

A precise forecast needs to overcome problems of variable energy production caused by fluctuating weather conditions.

Power generated by wind is highly dependent on the wind speed. Though it is highly nonlinear, wind speed follows a certain pattern over a certain period of time. We exploit this time series pattern to gain useful information and use it for power prediction.

1.1 PROJECT OVERVIEW :

Category:

Machine Learning

Skills Required:
Python, Python Web Frame Works, Python For Data Visualization, Data Preprocessing Techniques, Machine Learning, IBM Cloud, IBM Watson Studio, Python-Flask.

Project Description:

Wind power generation differs from conventional thermal generation due to the stochastic nature of wind. Thus wind power forecasting plays a key role in dealing with the challenges of balancing supply and demand in any electricity system, given the uncertainty associated with the wind farm power output. Accurate wind power forecasting reduces the need for additional balancing energy and reserve power to integrate wind power. For a wind farm that converts wind energy into electricity power, a real-time prediction system of the output power is significant. In this guided project , a prediction system is developed with a method of combining statistical models and physical models. In this system, the inlet condition of the wind farm is forecasted by the auto regressive model.

Technical Architecture:
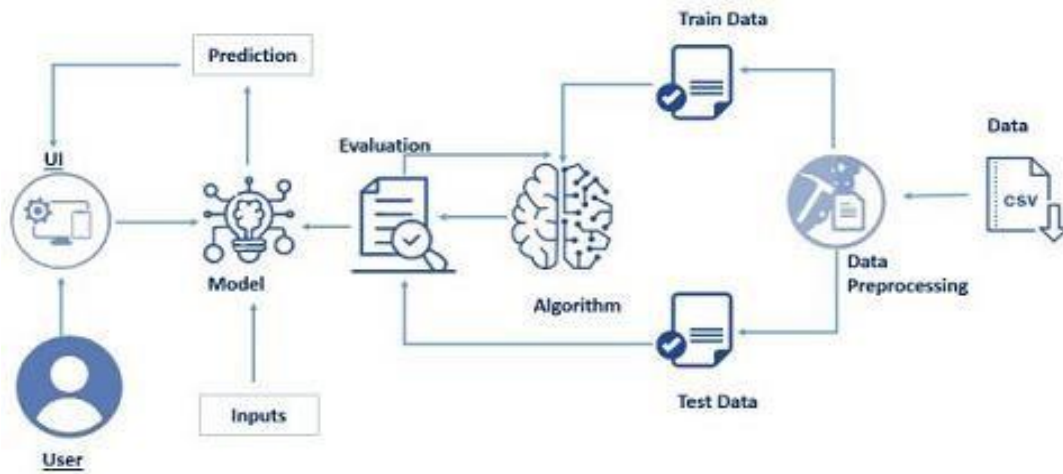
1.2 PURPOSE:

- Accurate wind power forecasting **reduces the need for additional balancing energy and reserve power to integrate wind power**.

- For a wind farm that converts wind energy into electricity power, a real-time prediction system of the output power is significant.

- Wind energy **plays an increasing role in the supply of energy worldwide**.

2. LITERATURE SURVEY :

We are going to deploy an application that will utilize multiple Watson AI Services including Cloud function, Watson Machine Learning and Node-RED/other web frame-work services. During the course of this project, we'll learn about combining Watson services, and how they can build interactive customer satisfaction interactive portals for prediction on current conditions

1.2 EXISTING PROBLEM:

- **Turbines produce noise and alter visual aesthetics :**
  Wind farms have different impacts on the environment compared to conventional power plants, but similar concerns exist over both the noise produced by the turbine blades and the visual impacts on the landscape .

- **Sound and visual impact** are the two main public health and community concerns associated with operating wind turbines. Most of the sound generated by wind turbines is aerodynamic, caused by the movement of turbine blades through the air.
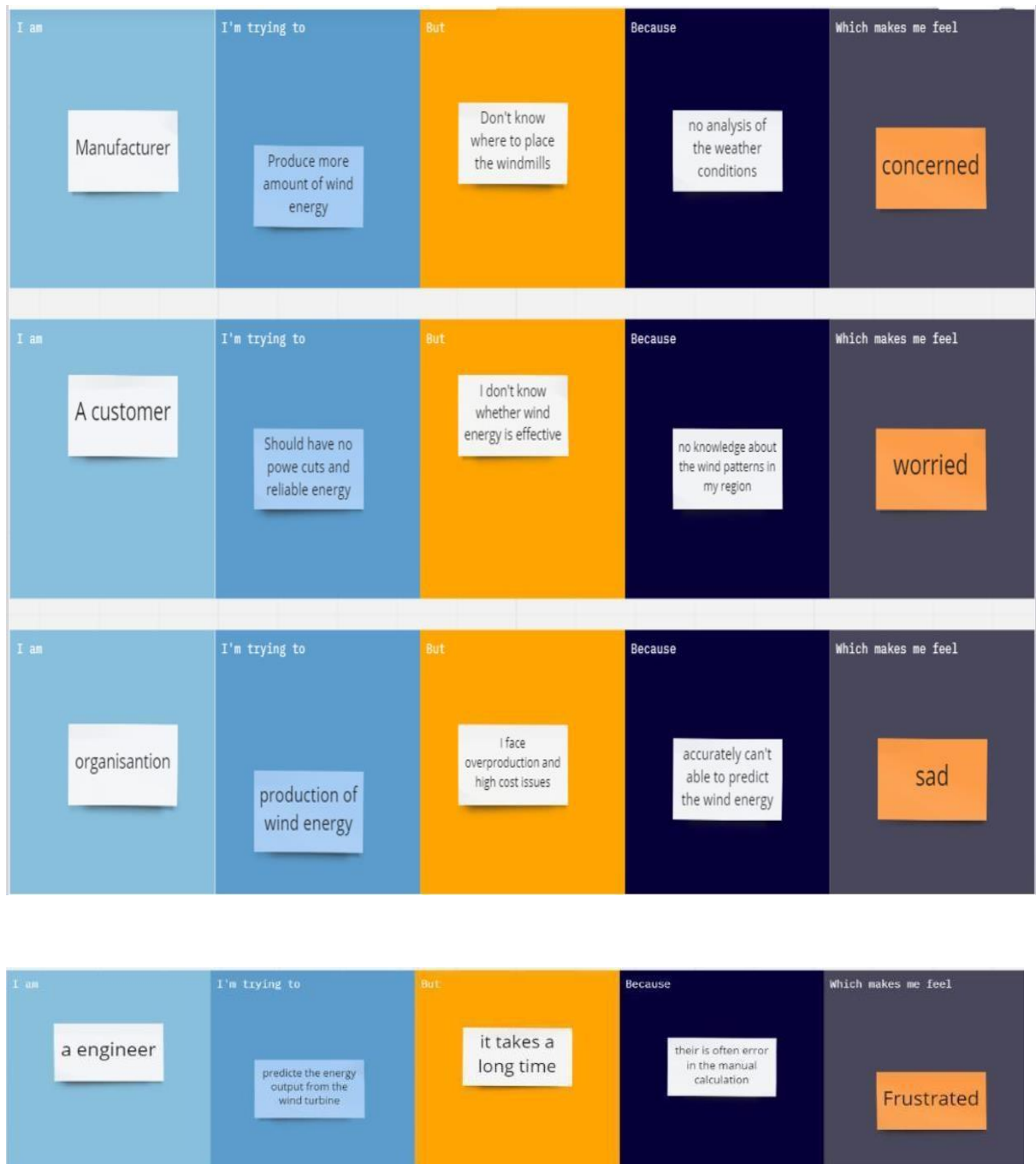
1.3 REFERENCE:

S.NO	AUTHOR/YEAR	TITTLE	TECHNIQUE USED	MERITS	DEMERITS

1.	A.Clifton/2012	Using machine learning to predict wind turbine power output.	Machine learning.	Strong function wind speed.	Affected by turbulence and shear.
2.	Aman Bahugun/2013	Predicting the energy output of wind turbine based on weather conditions watson auto AI	IBM WATSON AUTO AI machine learning.	Predicted more accurately.	However,in another study it was found that the prediction errors do not satisfy the KolmogoroveSmirnov test for normal distribution.
3.	Haroon Rashid/2020	Forecasting of wind turbine output power using machine Learning.	Machine learning	Accurate predicting of output power.	Absolute errors for the proposed model.
4.	Katya Vladislavleva/ 2019	Predicting the Energy Output of Wind Farms Based on Weather Data: Important Variables and their Correlation	wind energy, prediction, genetic programming, DataModeler	A good prediction of the energy output.	However, levels of production of wind energy are hard to predict as they rely on potentially unstable weather conditions present at the wind farm.
5.	J K Lundquist and P.Fleming1/2012	Using machine learning to predict wind turbine power output	Machine learning, classification and regression trees, wind energy, wind turbine	Reduce bias in power predictions that arise because of the different turbulence and shear at the new site, compared to the test site.	Changes of wind direction with height, non-uniform shear, and the state of the turbine were not considered here but may impact turbine deployment sites.

6.	Aoife M. Foley/2020	Review Current methods and advances in . forecasting of wind power generation	Meteorology Numerical weather prediction Probabilistic forecasting Wind integration wind power forecasting	Thus wind power forecasting plays a key role in dealing with the challenges of balancing supply and demand in any electricity system, given the uncertainty associated	Overall accurate wind power prediction reduces the financial and technical risk of uncertainty of wind power production for all electricity market participants.
				with the wind farm power output.	

1.4 PROBLEM STATEMENT DEFINITION :

I am	I'm trying to	But	Because	Which makes me feel
Manufacturer	Produce more amount of wind energy	Don't know where to place the windmills	no analysis of the weather conditions	concerned
A customer	Should have no powe cuts and reliable energy	I don't know whether wind energy is effective	no knowledge about the wind patterns in my region	worried
organisantion	production of wind energy	I face overproduction and high cost issues	accurately can't able to predict the wind energy	sad
a engineer	predicte the energy output from the wind turbine	it takes a long time	their is often error in the manual calculation	Frustrated

3.IDEATION & PROPOSED SOLUTION 3.1

EMPATHY MAP CANVAS:

3.2 IDEATION& BRAINSTORMING:

BRAINSTROM & IDEA PRIORITIZATION:

Use this template in your own brain storming sessions so your team can unleash their imagination   and start shaping concepts even if you're not sitting in the same room.

Before you collaborate A little bit of preparation goes along way with this session. Here's

what you need to do to get going.

Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement.This will be the focus of your brainstorm.

Brainstorm
Write down any ideas that come to mind that address your problem statement.

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing.

Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes ,try and see if you and break it up into smaller sub-groups.

Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

After you collaborate
You can export the mural as an image or pdf to share with members of your company who might find it helpful.

_A *Team gathering*

Define ho should participate in the session and send an invite. Share relevant information or pre work ahead.

_B *Set the goal*

Think about the problem you'll be focusing on solving in the brainstorming session.

_C *Learn how to use the facilitation tools*

Use the Facilitation Super powers to run a happy.

3.3 PROPOSED SOLUTION:

Proposed :

Project team shall fill the following information in proposed solution template.

S .No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Our time –tested kombi Box produce gives a higher priority to those forecasts having the lowest prediction error in respective weather situation.
2.	Idea/Solution description	Our aim is to map weather data to energy production
3.	Novelty/ Uniqueness	A good overview on the different methods that were recently applied in forecasting of wind power generation can be found in.
4.	Social Impact/ Customer Satisfaction	If there's a lot of wind, you get more energy output than if there's less wind, which means you will likely want to do maintenance when the winds are low to minimize downtime.
5.	Business Model(Revenue Model)	Real time projections for solar power, including behind-the-meter generation Grid-oriented forecasts
6.	Scalability of the Solution	The model prediction is then showcased on user interface to predict the energy output of wind turbine

3.4 PROBLEM SOLUTION FIT:

1. CUSTOMERSEGMENT(S):

Wind flows over the blades creating lift (similar to the effect on air plane wings), which causes the blades to turn.

The blades are connected to a drive shaft that turns an electric generator, which produces (generates).

1. JOBS-TO-BE-DONE/PROBLEMS:

Turbines produce noise and alter visual aesthetics

Wind farms have different impacts on the environment compared to conventional power plants ,but similar concerns exist over both the noise produced by the turbine blades and the visual impacts on the landscape.

3. TRIGGERS:

The wind speed is always fluctuating and thus the energy content of the wind is also always changing. Exactly how large the variation is depends both on the weather and
On local surface conditions and obstacles.

4. EMOTIONS:

BEFORE / AFTER

Before: who live in close proximity to wind turbines say they experience sleep disturbances, headaches and concentration problems.
After: Between working long hours, climbing turbines multiple times a day ,and dealing with extreme heat in the summer and cold in the winter.

5. AVAILABLE SOLUTIONS:

These residential windmills can generate electricity by churning the wind through its blades, which in turn rotates the turbine and generates power, which can meet the needs of a small family. The energy that is yielded from these wind turbines is clean, renewable, and are also cost-effective.

6. CUSTOMER CONSTRAINTS:

- Intermittent
- Low operating costs **Noise** and visual pollution Efficient use of land space
- Some adverse environmental impact
- Wind energy is a job creator

7. BEHAVIOUR:

The wind speed is always fluctuating and thus the energy content of the wind is also always changing.

Exactly how large the variation is depends both on the weather and on local surface conditions and obstacles.

8. CHANNELS OF BEHAVIOUR:

Online:
An efficient automated approach to wind farm operation monitoring is presented.
Offline:
Product is available for offline usage.

9. PROBLEM ROOT CAUSE:

The degradation, weakening and de bonding of the adhesive layers (in the trailing or leading edges or on the spar/shell joint) is one of the main processes leading to wind turbine blade failure.

10 .YOUR SOLUTION:

Larger rotor diameters allow wind turbines to sweep more area, capture more wind, and produce more electricity. A turbine with longer blades will be able to capture more of the available wind than shorter blades—even in areas with relatively less wind.

4. REQUIREMENT ANALYSIS:

4.1. FUNCTIONAL REQUIREMENTS:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement(Epic)	Sub Requirement(Story/Sub-Task)
FR-1	User Registration and logging in by entering their user name and password.	Registration through Form.
FR-2	User Confirmation by validating the user name with respect to the password	Confirmation via pop-up Message.
FR-3	Displaying the further information about the application.	By selecting the about button the details of the application will be displayed.
FR-4	Validating the city name.	System checks whether the city entered by the user is present or not. If present it will collect the further details else it will display the pop up message as error in the city.
FR-5	Checking the data type of the value.	System checks for the data type of the value entered by the user.
FR-6	Validating all required fields.	Before predicting the output the system checks whether all the values are entered by the user and checks whether all values are correct.
FR-7	Displaying weather Conditions for a given city.	It displays the weather of the city which have Been selected.
FR-8	Displaying predicted Energy output power.	The predicted output will be displayed as amount of wind energy power generated.

4.2. NON-FUNCTIONAL REQUIREMENTS:
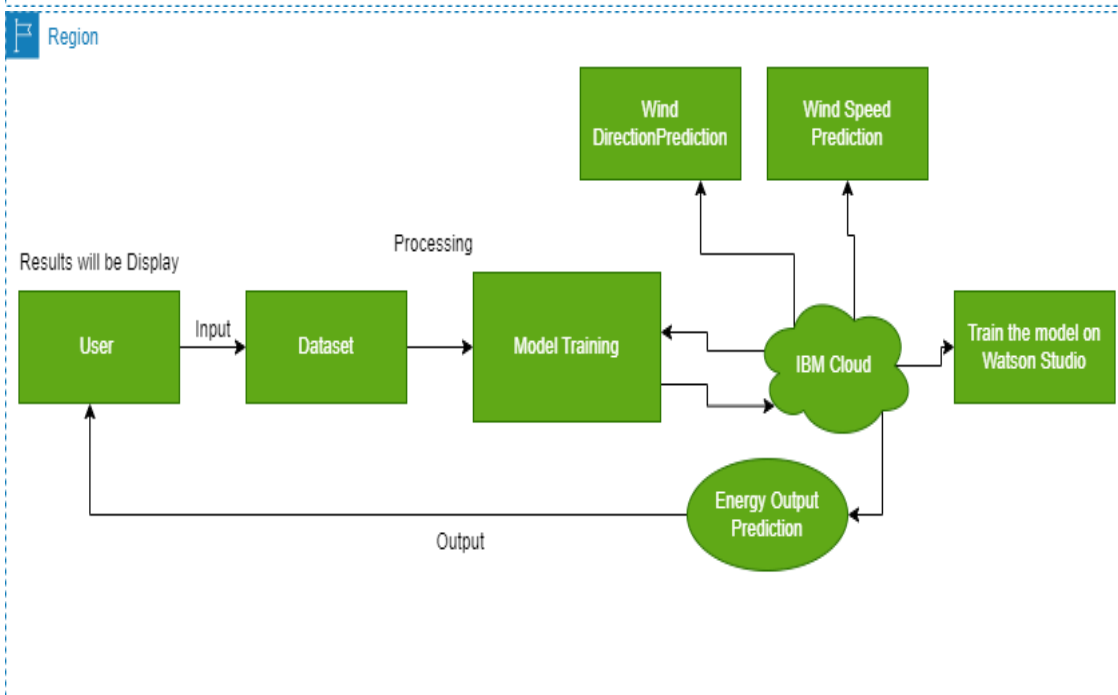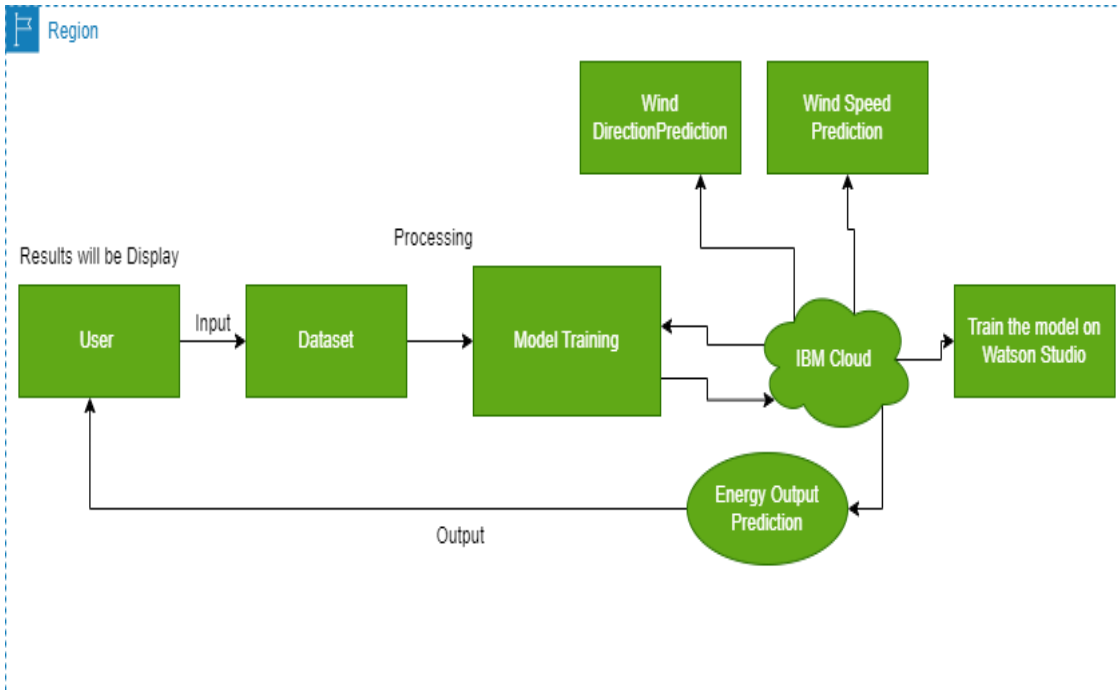
Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system satisfies the user goals and the application is easy to use.
NFR-2	Security	The data provided to system will be protected from Attacks and unauthorized access

NFR-3	Reliability	The system will provide the consistency in output without producing an error.
NFR-4	Performance	The performance will never degrade even the work load is increased.
NFR-5	Availability	The application is available for24*7
NFR-6	Scalability	The system can be used as web application as well as mobile application with a sufficient internet availability.

5. PROJECT DESIGN:

5.1. DATA FLOW DIAGRAMS:

Region

Wind DirectionPrediction

Wind Speed Prediction

Processing

Results will be Display

User

Input

Dataset

Model Training

IBM Cloud

Train the model on Watson Studio

Energy Output Prediction

Output
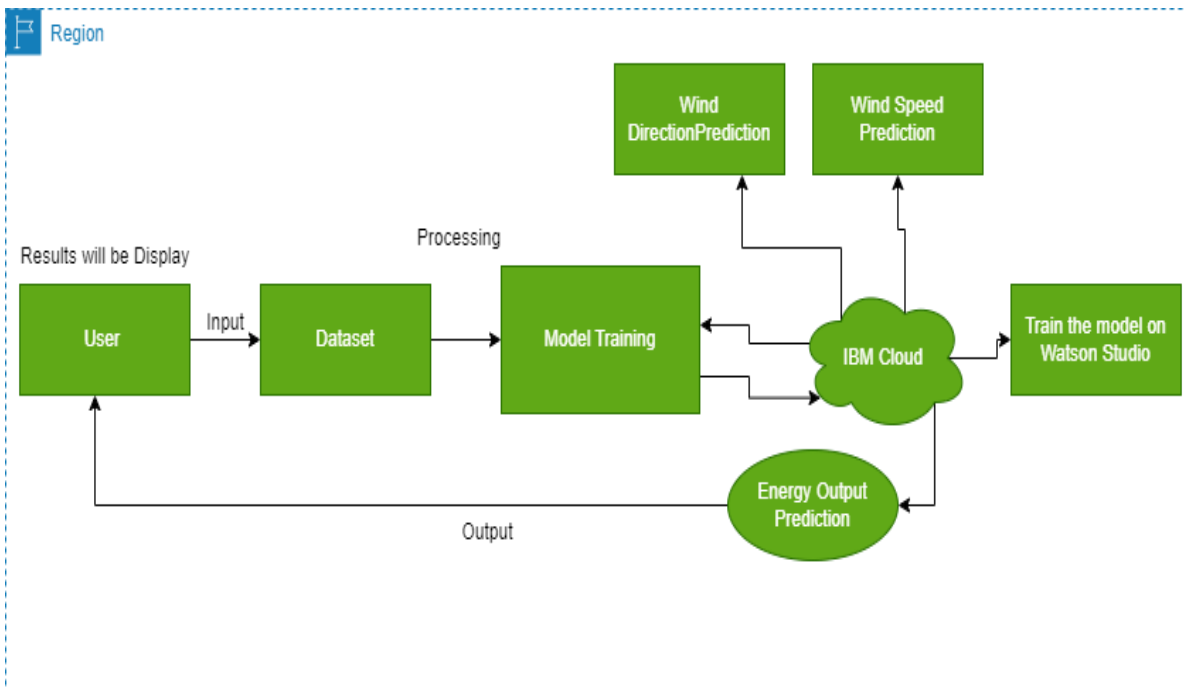
1. User enter input weather data.
2. Send data to the central server.
3. Train the model on Watson studio.
4. W ind direction and speed is predicted.
5. Result will be displayed.

5.2. SOLUTION & TECHNICAL ARCHITECTURE:

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Wind power generation differs from conventional thermal generation due to the stochastic nature of wind.
- Thus wind power forecasting plays a key role in dealing with the challenges of balancing supply and demand in any electricity system, given the uncertainty associated with the wind farm power output.
- The inlet condition of the wind farm is forecasted by the auto regressive model.
- We report on the correlation of the different variables for the energy output.

Example - Solution Architecture Diagram:
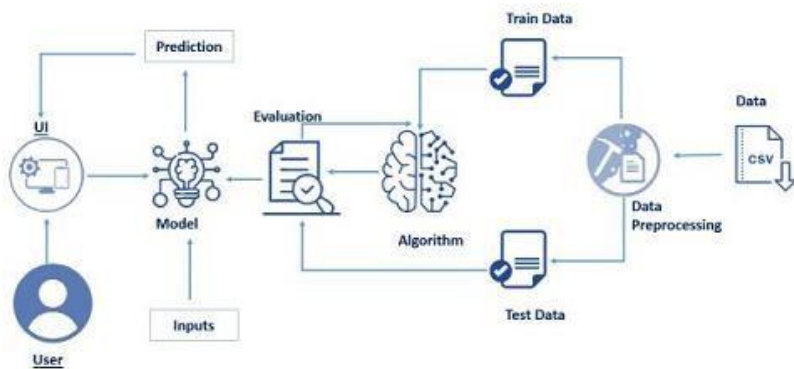
Figure 1: Architecture and data flow

Reference: https://github.com/SmartPracticeschool/llSPS-INT-3437-Predicting-the-Energy-Output-of-WindTurbine-Based-on-Weather-Conditions-Watson-Auto-
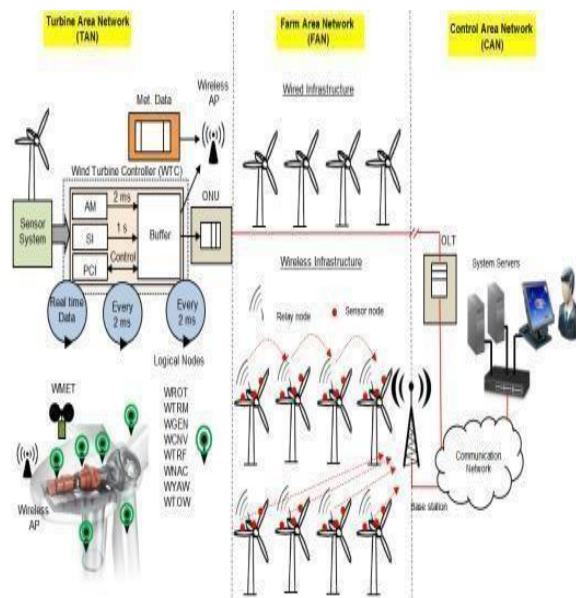
Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1&table2

Example:Predictingtheenergyoutputofwindfarmbasedonweatherconditions.
Reference: https://www.mdpi.com/1996-1073/7/6/3900

Guidelines:

The proposed communication network architecture for the Smart-WPF consists of three networks :the turbine area network (TAN), the farm area network(FAN), and the control area network (CAN).
It consists of hierarchical architectures whereLevel1isasensor network in a single wind turbine, Level2 is the wind turbine-to-wind turbine interaction in the WPF, Level 3 is the local control center to wind turbine interaction, and Level 4 is the farm-to-farm interaction to optimize grid operation.

In order to implement hierarchical network architecture, a hybrid communication solution is considered. EPON-based architecture represents a wired solution, while ZigBee-
Pro is considered for the wireless solution. In this work, Levels1 and 2 are explained in more detail, while Levels 3 and 4 are out the scope of this work

6. PROJECT PLANNING & SCHEDULING:
6.1 SPRINT PLANNING & ESTIMATION:

Sprint	Sprint Start Date	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)
Sprint-1	01 Nov2022	14 Nov2022	20
Sprint-2	07 Nov2022	14Nov2022	20
Sprint-3	07Nov2022	15Nov2022	20

Sprint				
Sprint-4	09 Nov2022	15Nov2022	20	

6.2 SPRINT DELIVERY SCHEDULE:

Sprint	SprintStartDate	SprintEndDate(Planned)	Story PointsCompleted (ason PlannedEndDate)	SprintReleaseDate(Actual)	
Sprint-1	01 Nov2022	14 Nov2022	20	14 Nov 2022	
Sprint-2	07 Nov2022	14Nov2022	20	14 Nov2022	
Sprint-3	07Nov2022	15Nov2022	20	15 Nov2022	
Sprint-4	09 Nov2022	15Nov2022	20	15 Nov2022	

7. CODING & SOLUTIONING:

7.1 FEATURE 1:

Saving Wind Dataset.csv to Wind Dataset.csv

In [2]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import joblib
```

Data Preprocessing

In [3]:
```python
path = "Wind Dataset.csv"
df = pd.read_csv(path)
df.rename(columns={"Date/Time":"Time", "LV ActivePower (kW)":"ActivePower(KW
                   "Wind Direction(°)":"Wind_Direction"},
df.head()            inplace=True)
```

In [4]:

		ActivePower(KW)	WindSpeed(m/s)	Theoretical_Power_Curve (KWh)	Wind Direction (°)
0	01.01 2018 00:00	380.047791	5.311336	416.328908	259.994904
1	01.01 2018 00:10	453.769196	5.672167	519.917511	268.641113
2	01.01 2018 00:20	306.376587	5.216037	390.900016	272.564789
3	01.01 2018 00:30	419.645904	5.659674	516.127569	271.258087
4	01.01 2018 00:40	380.650696	5.577941	491.702972	265.674286

```
ActivePower(KW)                    0
WindSpeed(m/s)                     0
Theoretical_Power_Curve (KWh)      0
Wind Direction (°)                 0
dtype: int64
```

Out[4]: sns.pairplot(df)

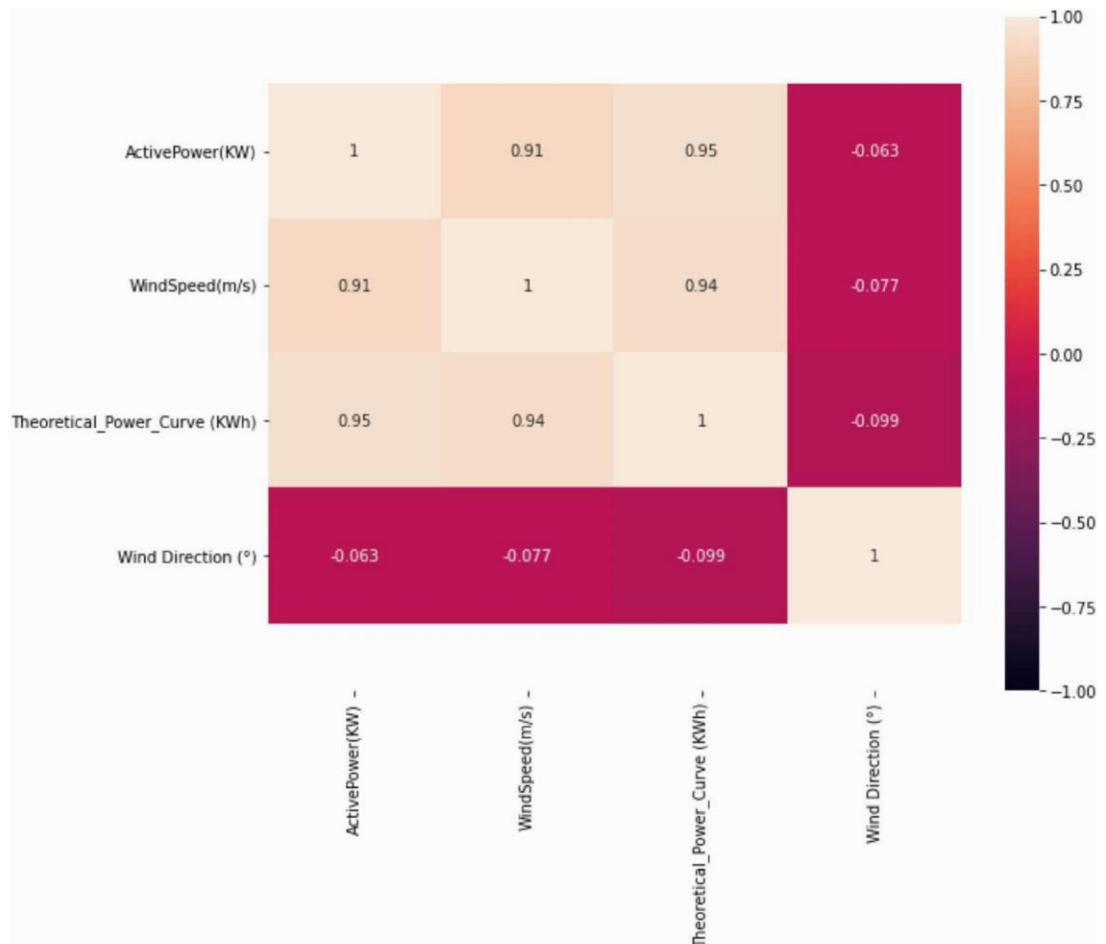|	Wind Direction (°)
ActivePower(KW)	-0.062702
WindSpeed(m/s)	-0.077188
Theoretical_Power_Curve (KWh)	-0.099076
Wind Direction (°)	1.000000

Out[6]:

In [5]:

Out[5]:

In [7]:
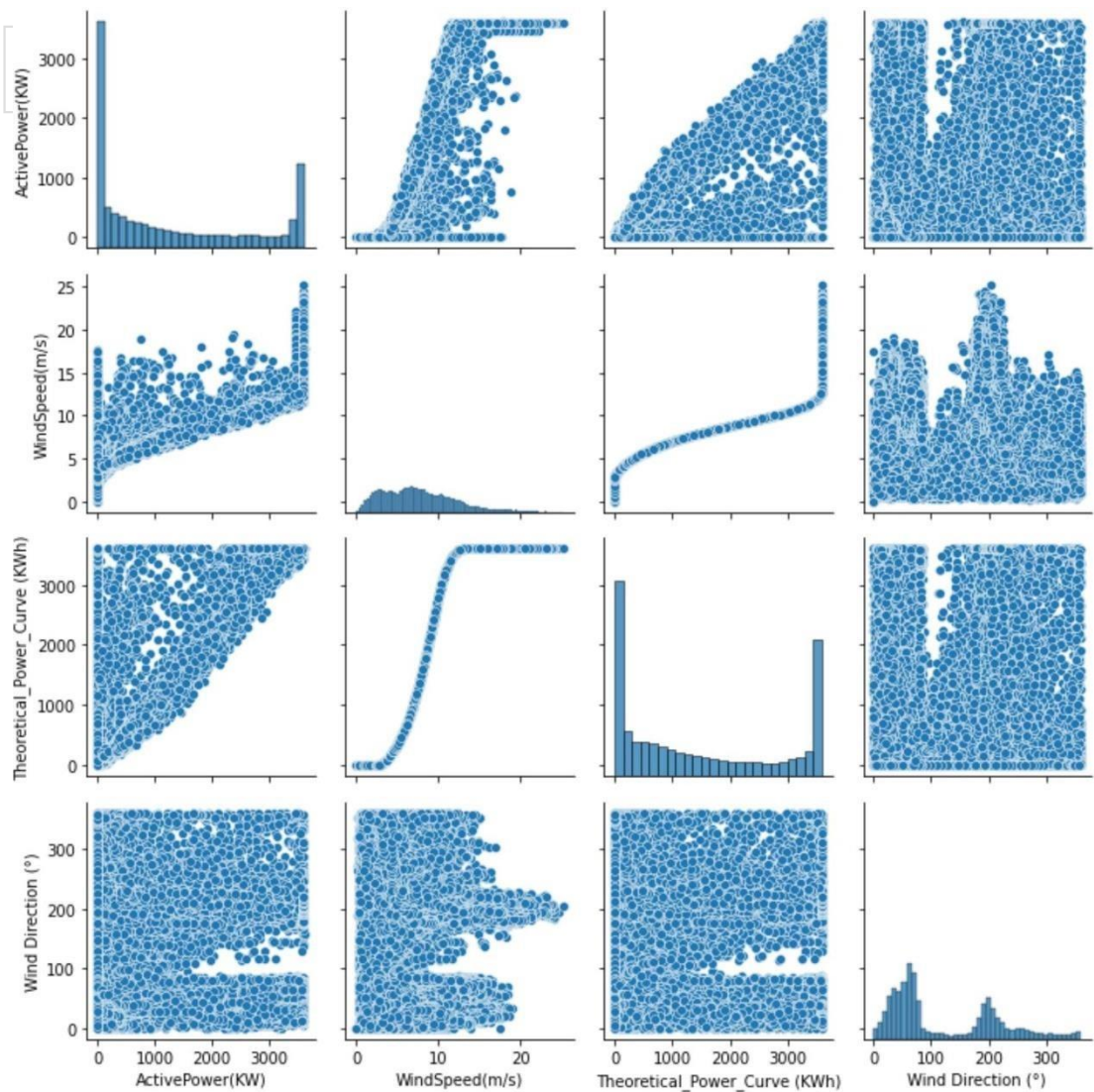
```python
plt.figure(figsize=(10, 8))
corr = df.corr()
ax = sns.heatmap(corr, vmin = -1,vmax = 1,annot = True)
bottom, top = ax.get_ylim()
ax.set_ylim(bottom + 0.5, top - 0.5)

print(corr)
```

```python
df["Time"] = pd.to_datetime(df["Time"], format = "%d %m %Y %H %M", errors =
```

Splitting the data to Train and Test

In [9]:
```python
y = df["ActivePower(KW)"]
X = df[["Theoretical_Power_Curve (KWh)", "WindSpeed(m/s)"]]

from sklearn.model_selection import train_test_split
train_X, val_X, train_y, val_y = train_test_split(X, y, random_state=0)
```

Model Building

In [10]:
```python
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error,r2_score

forest_model = RandomForestRegressor(n_estimators = 750, max_depth = 4, max_

forest_model.fit(train_X, train_y)
```

Out[10]:
```
RandomForestRegressor(max_depth=4, max_leaf_nodes=500, n_estimators=750,
                      random_state=1)
```

In [11]:
```python
power_preds = forest_model.predict(val_X)

print(mean_absolute_error(val_y, power_preds))
print(r2_score(val_y, power_preds))
```

```
0.911349642890175
```

```python
joblib.dump(forest_model, "power_prediction.sav")
```

`['power_prediction.sav']`

```python
df
```

	Time	ActivePower(KW)	WindSpeed(m/s)	Theoretical_Power_Curve (KWh)	Wind Direction (°)
0	NaT	380.047791	5.311336	416.328908	259.994904
1	NaT	453.769196	5.672167	519.917511	268.641113
2	NaT	306.376587	5.216037	390.900016	272.564789
3	NaT	419.645904	5.659674	516.127569	271.258087
4	NaT	380.650696	5.577941	491.702972	265.674286
...	...	...	...	...	...
50525	NaT	2963.980957	11.404030	3397.190793	80.502724
50526	NaT	1684.353027	7.332648	1173.055771	84.062599
50527	NaT	2201.106934	8.435358	1788.284755	84.742500
50528	NaT	2515.694092	9.421366	2418.382503	84.297913
50529	NaT	2820.466064	9.979332	2779.184096	82.274620

50530 rows × 5 columns

8. TESTING:

8.1 TEST CASES:

TestingNo:1	Testing Type	Function Result
	Functionality testing	Yes
	Usability testing	Yes
	Interface testing	Yes
	Performance testing	Yes(medium)
	Security testing	Yes

TestingNo:2	Testing Type	Function Result
	Functionality testing	Yes
	Usability testing	Yes
	Interface testing	Yes
	Performance testing	Yes(medium)
	Security testing	Yes

TestingNo:3	Testing Type	Function Result
	Functionality testing	Yes
	Usability testing	Yes
	Interface testing	Yes
	Performance testing	Yes
	Security testing	Yes

TestingNo:4	Testing Type	Function Result
	Functionality testing	Yes
	Usability testing	Yes
	Interface testing	Yes
	Performance testing	Yes
	Security testing	Yes

9. RESULT:

9.1 PERFORMANCE METRIES:

DATA PREPROCESSING:

Out[4]:

	01 01			
0	2018 00:00	380.047791	5.311336	416.328908
259.994904				
1	01 01 2018 00:10	453.769196	5.672167	519.917511
268.641113				
2	01 01 2018 00:20	306.376587	5.216037	390.900016
272.564789				
3	01 01 2018 00:30	419.645904	5.659674	516.127569
271.258087				
4	01 01 2018 265.674286	380.650696	5.577941	491.702972

MODEL BUILDING:

```
Out[10]:    RandomForestRegressor(max_depth=4, max_leaf_nodes=500, n_estimators=750,
                                  random_state=1)
```

FINAL RESULT:

```
1    import numpy as np
2    from flask import Flask, request, jsonify, render_template
3    import joblib
4    import requests
5
6    # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
7    API_KEY = "z5w_7QoNuv-PziHyHPjrv_fKcKhYzVDd_vhj4IWS2dli"
8    token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
9      API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
10   mltoken = token_response.json()["access_token"]
11
12   header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
13
```

```
* Serving Flask app 'app_IBM' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Activate Windows
Go to Settings to activate Windows.

10. ADVANTAGES& DISADVANTAGES:

ADVANTAGES:

• Weather Underground Services provide very accurate Historical Weather Data which increased the accuracy of model.

• Website is more convenient to use due to zero storage.

 • With Choosing city, Website can accurately predict power output using weather condition.

DISADVANTAGES:

•Weather API is paid and the free version provide limited API requests per day.

• Android Website can be deployed on IBM Cloud.

 • No free server available on IBM Cloud for deploying Backend.

11. CONCLUSION:

lusions In this study we showed that wind energy output can be predicted from publicly available weather data w accuracy at best 80% R2 on the training range and at best 85, 5% on the unseen test data.

 We identified the smallest space of input variables (windGust2 and dewPoint), where reported accuracy can be achieved, and provided clear trade-offs of prediction accuracy for decreasing the input space to the windGust2 variable.

We demonstrated that an off-the-shelf data modeling and variable selection tool can be used with mostly default settings to run the symbolic regression experiments as well as variable importance, variable contribution analysis, ensemble selection and validation.

12. FUTURE SCOPE:

Most wind power forecasting models study 'regular' wind conditions.

The EU funded project called 'Safe wind' aims to improve wind power prediction over challenging and extreme weather periods and at different temporal and spatial scales.

Development activities are on-going to reduce error in the wind power prediction, to improve regionalized wind power forecasting for on - shore windfarms and to derive methods for wind power prediction for offshore wind farms.

It is possible that use of ensemble and combined weather prediction methods together may enhance forecasting.

If the error in wind power forecasting and prediction is reduced then electricity markets can trade with more certainty. Contract errors as a function of time in electricity markets can be as high as 39% for a forecasting lead time of 4 present a new tool called the WILMAR and ANEMOS scheduling Methodology (WALT) to reduce the number of thermal generators on stand-by or in reserve using the probability of generation outages and load shedding are system reliability criteria instead of generation adequacy based solely on generation outage.

The wind and load forecast errors are modelled using a Gaussian stochastic variable approach.

However, in another study it was found that the prediction errors do not satisfy the Kolmogorov Smirnov test for normal distribution.

In Carta , it was shown that, the use of auto correlated (and thus not independent) successive hourly mean wind speeds, though invalidating all of the usual statistical tests, has no appreciable effect on the shape of the pdf estimated from the data.

13. APPENDIX:

13.1 SOURCE CODE:

> ### app.py

```python
import numpy as np
from flask import Flask, request, jsonify, render_template
import joblib
```

```python
import requests

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')
    apikey = "a802b0f626c637d04185e582b5ad0d58"
    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.ppredict(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)
```

➤ **predict.html**

```html
<html>
    <head>
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-
scale=1.0" />
        <meta http-equiv="X-UA-Compatible" content="ie=edge" />
        <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.7.2/css/all.css"
            integrity="sha384-
fnmOCqbTlWIlj8LyTjo7mOUStjsKC4pOpQbqyi7RrhN7udi9RwhKkMHpvLbHG9Sr"
crossorigin="anonymous" />
        <link href="https://fonts.googleapis.com/css?family=Dosis"
rel="stylesheet" />
        <link rel="stylesheet" href="static/css/main.css"/>
        <link rel="stylesheet" href="static/css/media.css"/>
        <link rel="stylesheet" href="static/css/items_grid.css"/>


    <title>Wind Energy Prediction</title>
    <style>
#page {
  max-width: 80%;
  margin: auto;
}
body {
    background-image: url(https://images2.alphacoders.com/753/753985.jpg);
    width: 100%;
    height: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: cover;
    overflow: hidden;

}
table {
      width: 100%;
      border-collapse: collapse;
}

.card {
        margin-right: auto;
        margin-left: 15%;
        width: 300px;
```

```css
        box-shadow: 0 15px 25px rgba(129, 124, 124, 0.2);

        border-radius: 5px;
        backdrop-filter: blur(14px);
        background-color: rgb(180, 180, 180);
        padding: 15px;
        text-align: center;
}
.head {
        top:0px;
        margin:0px;
        left: 0px;
        right: 0px;
        position: fixed;
        background: #aeb90f;
        color: white;
        overflow: hidden;
        padding-bottom: 30px;
        font-size: 2.25vw;
        width: 100%;
        padding-left:0px;
        text-align: center;
        padding-top:20px;
}
.second{
        top:80px;
        bottom:0px;
        margin:0px;
        left: 0px;
        right: 0px;
        position: fixed;
        padding: 0px;
        width: 100%;

        font-family:Georgia, serif;
        color:black;
        font-size:20px;
}
.inside{
        top:80px;
        bottom:0px;
        margin:0px;
        left: 51%;
        right: 0%;
        position: fixed;
        padding-left: 40px;
        padding-top:8%;
```

```css
            padding-right:40px;

            font-family:Georgia, serif;
            color:#96f400;
            font-size:20px;
            text-align:justify;

        }
        .myButton{
            border: none;
            text-align: center;
            cursor: pointer;
            text-transform: uppercase;
            outline: none;
            overflow: hidden;
            color: #fff;
            font-weight: 700;
            font-size: 12px;
            background-color: #183a1d;
            padding: 10px 15px;
            margin: 0 auto;
            box-shadow: 0 5px 15px rgba(0,0,0,0.20);
            margin-left:17%;
        }
    input {
    width:50%;
    margin-bottom: 10px;
    background: #e1eedd;
    border: none;
    outline: none;
    padding: 10px;
    font-size: 13px;
    color: #6c493a;
    text-shadow: white;
    border: #6c493a;
    border-radius: 4px;
    box-shadow: white;
}
::placeholder {
  color: black;
  opacity: 1;
}

.left{
        top:80px;
        bottom:0px;
        margin:0px;
```

```css
            left: 0%;
            right: 45.5%;
            position: fixed;
            padding-left: 10%;
            padding-top:5%;
            padding-right:40px;

            font-family:bold,Georgia, serif;
            color:rgb(238, 255, 0);;
            font-size:25px;

}

    select {
    width:50%;
    margin-bottom: 10px;
    background: white;
    border: none;
    outline: none;
    padding: 10px;
    font-size: 13px;
    color: #183a1d;
    text-shadow: white;
    border: #6c493a;
    border-radius: 40px;
    box-shadow: white;
}

input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px
1px rgba(255,255,255,0.2); }



        table, th, td {
  border: 1px solid rgb(86, 72, 128);
  border-collapse: collapse;
  color: #3f00ff;
}

@media screen and (max-width: 500px) {

  .left,
  .second,
  .third {
      width: 70%;
  }
}
```

```html
    </style>
    </head>
    <body>


        <header id="head">
        <div class="head">Predicting The Energy Output Of Wind Turbine
Based On Weather Condition</div>
        </header>


        <div class="second">
        <div class="left">

            <p style="padding: 8px; border: 1px solid rgb(0, 110, 255);
width: 575px;"> GIVE YOUR CITY NAME TO KNOW THE WEATHER CONDITIONS</p>

        <div style="margin-left:10%">
        <form action="{{ url_for('windapi')}}"method="post" >
                <select name="city"  required >
                    <option value="" selected>select City</option>
                    <option value ="Ariyalur"    >    Ariyalur    </option>
                    <option value ="Andimadam"   >    Andimadam   </option>
                    <option value ="Coimbatore" >    Coimbatore  </option>
                    <option value
="Chengalpattu"    >    Chengalpattu    </option>
                    <option value ="Cuddalore"   >    Cuddalore   </option>
                    <option value ="Chennai"    >    Chennai </option>
                    <option value ="Dindigul"    >    Dindigul    </option>
                    <option value ="Dharmapuri" >    Dharmapuri </option>
                    <option value ="Erode"   >    Erode    </option>
                    <option value ="Karur"   >    Karur    </option>
                    <option value
="Kancheepuram"    >    Kancheepuram    </option>
                    <option value ="Krishnagiri"     >    Krishnagiri
</option>

                    <option value
="Kallakurichi"   >    Kallakurichi    </option>
                    <option value ="Madurai"     >    Madurai </option>
                    <option value ="Mayiladuthurai"
>    Mayiladuthurai  </option>
                    <option value
="Nagapattinam"   >    Nagapattinam    </option>
                    <option value ="Kanyakumari"     >    Kanyakumari
</option>
```

```html
                    <option value ="Namakkal"    >   Namakkal    </option>
                    <option value ="Perambalur" >   Perambalur </option>
                    <option value ="Pudukottai" >   Pudukottai </option>
                    <option value ="Ramanathapuram"
>   Ramanathapuram    </option>
                    <option value ="Ranipet"    >   Ranipet </option>
                    <option value ="Salem"    >   Salem    </option>
                    <option value ="Sivagangai" >   Sivagangai </option>
                    <option value ="Tenkasi"    >   Tenkasi </option>
                    <option value ="Thanjavur"  >   Thanjavur    </option>
                    <option value ="Theni"    >   Theni    </option>
                    <option value ="Thiruvallur"    >   Thiruvallur
</option>
                    <option value ="Thiruvarur" >   Thiruvarur </option>
                    <option value ="Tuticorin"  >   Tuticorin    </option>
                    <option value
="Trichirapalli"  >   Trichirapalli   </option>
                    <option value
="Thirunelveli"   >   Thirunelveli    </option>
                    <option value ="Tirupathur" >   Tirupathur </option>
                    <option value ="Tiruppur"    >   Tiruppur    </option>
                    <option value ="Tiruvannamalai"
>   Tiruvannamalai </option>
                    <option value ="The Nilgiris"   >   The
Nilgiris    </option>
                    <option value ="Vellore"    >   Vellore </option>
                    <option value ="Viluppuram" >   Viluppuram   </option>
                    <option value
="Virudhunagar"   >   Virudhunagar    </option>


            </select><br><br>
              <div style="margin-left:-15%"><button type="submit"
class="myButton" >Check the Weather Conditions</button></div>

        </form>
        </div>
        <br>

        <div class="card">
        <table style="margin-left:2%; text-align:center; border-
spacing:20px;">
            <tr>
                <td colspan="2" style="font-size:25px;">The weather
conditions of the city are</td>
            </tr>
            <tr>
```

```html
            <td>Temperature</td><td>{{temp}}</td>
        </tr>
        <tr>
            <td>Humidity</td><td>{{humid}}</td>
        </tr>
        <tr>
            <td>Pressure</td><td>{{pressure}}</td>
        </tr>
        <tr>
            <td>Wind Speed</td><td>{{speed}}</td>
        </tr>
    </table>
    </div>
</div>
    <div class="inside">
    <div style="font-size:23px;font-weight:bold;">Predict the Wind
Energy!!</div>
    <br><br>
    <form action="{{ url_for('y_predict')}}"method="post">
        <input type="text" name="theo" placeholder="Theoretical
Power in KWh" required="required"  />
        <input type="text" name="wind" placeholder="Wind Speed in
m/s" required="required"  /><br><br>
        <button type="submit" class="myButton" >Predict</button>

     </form>

    <br>
    <br>
    {{ prediction_text }}

</div>



</div>


</body>

</html>
```

Intro.html

```html
<html>
    <head>
    <title>Wind Energy Prediction</title>
```

```
<style>

    .header {
        top:0px;
        margin:0px;
        left: 0px;
        right: 0px;
        position: fixed;
        background: #a4a717;
        color: rgb(255, 255, 255);
        overflow: hidden;
        padding-bottom: 30px;
        font-family:Georgia, 'Times New Roman', Times, serif, serif;
        font-size: 2.5vw;
        width: 100%;
        padding-left:0px;
        text-align: center;
        padding-top:20px;
    }
    .second{
        top:90px;
        bottom:0px;
        margin:0px;
        left: 0px;
        right: 0px;
        position: fixed;
        padding: 0px;
        width: 100%;
        background-
image:url(https://i.pinimg.com/originals/c4/d2/f9/c4d2f98e88a85b702f8ff257
d74714d8.gif);
        background-repeat:no-repeat;
        background-size: contain;
    }
    .inside{
        top:90px;
        bottom:0px;
        margin:0px;
        left: 35%;
        right: 0%;
        position: fixed;
        padding-left: 40px;
        padding-top:15%;
        padding-right:40px;
        background-color:#f5e3c5;
        opacity: 100%;
        font-family:Georgia, serif;
```

```
        color:black;
        font-size:20px;
        text-align:justify;


    }
    .myButton{
        border: none;
        text-align: center;
        cursor: pointer;
        text-transform: uppercase;
        outline: none;
        overflow: hidden;
        color: #fff;
        font-weight: 700;
        font-size: 15px;
        background-color: #6c493a;
        padding: 10px 15px;
        margin: 0 auto;
        box-shadow: 0 5px 15px rgba(0,0,0,0.20);
    }
    </style>
    </head>
    <body>


        <div class="header">Predicting The Energy Output Of Wind Turbine
Based On Weather Condition</div>
        <div class="second">
            <div class="inside">A wind turbine turns wind energy into
electricity using the aerodynamic force from the rotor blades, which work
like an airplane wing or helicopter rotor blade. <br><br>
                The amount of electricity generated by wind increased by
almost 273 TWh in 2021 (up 17%), 45% higher growth than that achieved in
2020 and the largest of all power generation technologies. Wind remains
the leading non-hydro renewable technology, generating 1 870 TWh in 2021,
almost as much as all the others combined.
        <br><br><br>
        <a href="{{url_for('predict')}}"><button type="button"
class="myButton" >Click Here To Predict The wind Energy!</button></a>
        </div>


        </div>
    </body>
</html>
```