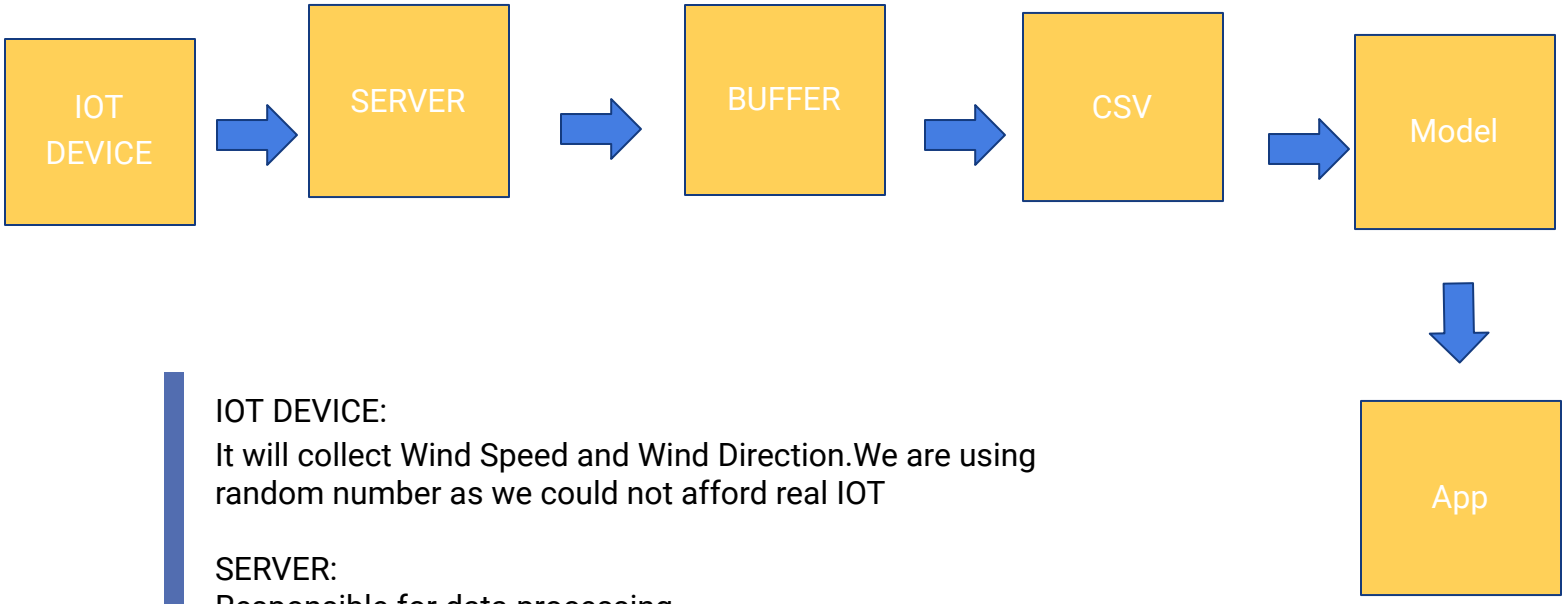
# Details and Project Model Structure and Basic Working

We have written an Web-API for data storage and retrieval along with other features.  You can find project directory structure has been made available in github.  It has notebook containing current LSTM model with dataset..




IOT DEVICE:
It will collect Wind Speed and Wind Direction.We are using random number as we could not afford real IOT

SERVER:
Responsible for data processing.

BUFFER::
Stores the data from IOT for to be used as input to model.

CSV:
It will contain a part of formated data to be send to model.

MODEL:
it is the deployed LSTM model.

# Advanced Working Explanation of the LSTM model

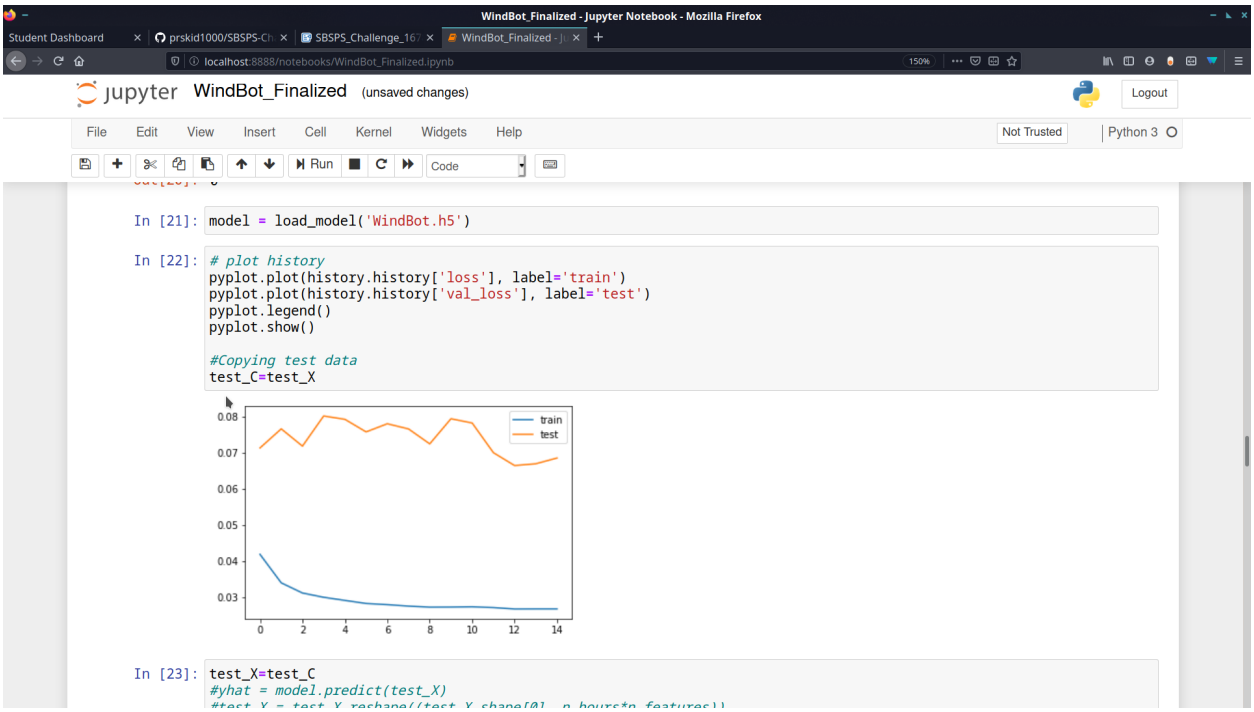
*On opening our App.It will automatically detect the time and send it to the node red server*
*The server will fetch last 24 hours data r(t-24) to r(t) to the deployed model.*
*The deployed model return predicted energy output of the next hour.*

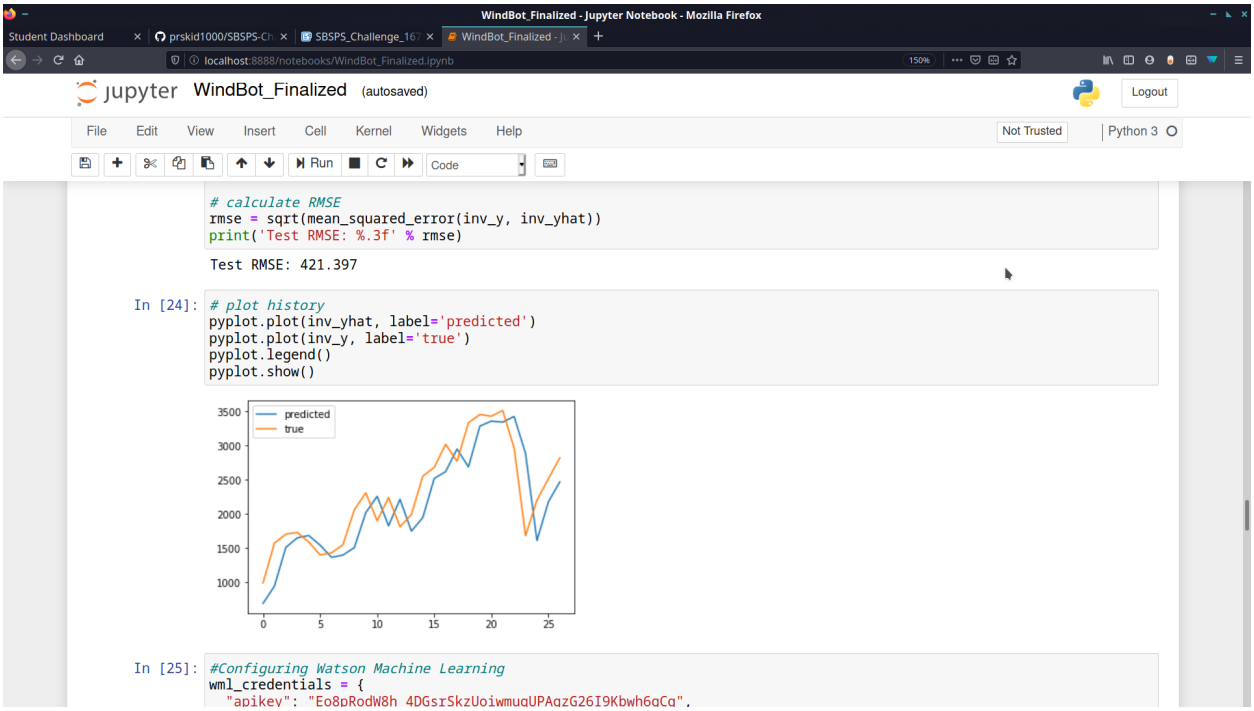
*in the next phase our app will get the wind direction and speed from database for the next hour.Then it will combine predicted energy along above data to generate a temporary row p(t+1).*

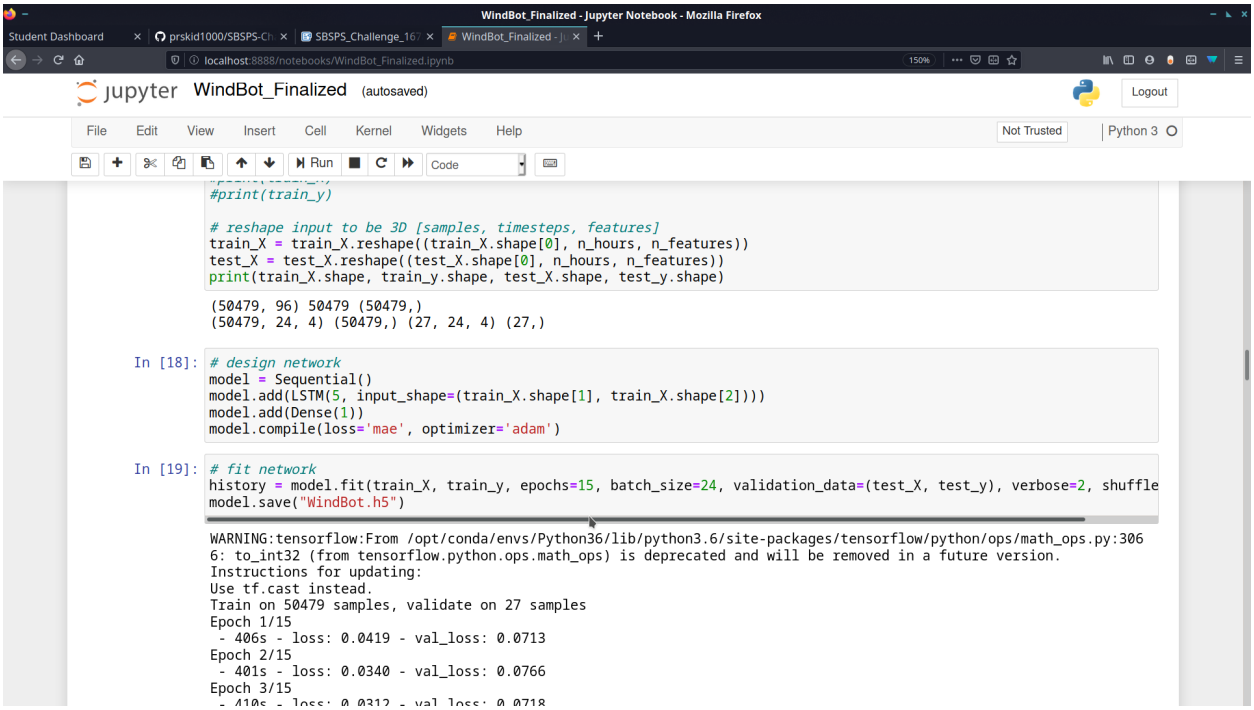
*Then we will again send it r(t-23) to r(t) along with p(t+1) to predict energy output of p(t+2).*
*It will continue until we get all p(t+72) rows.We will calculate best hour to use the enegry.*

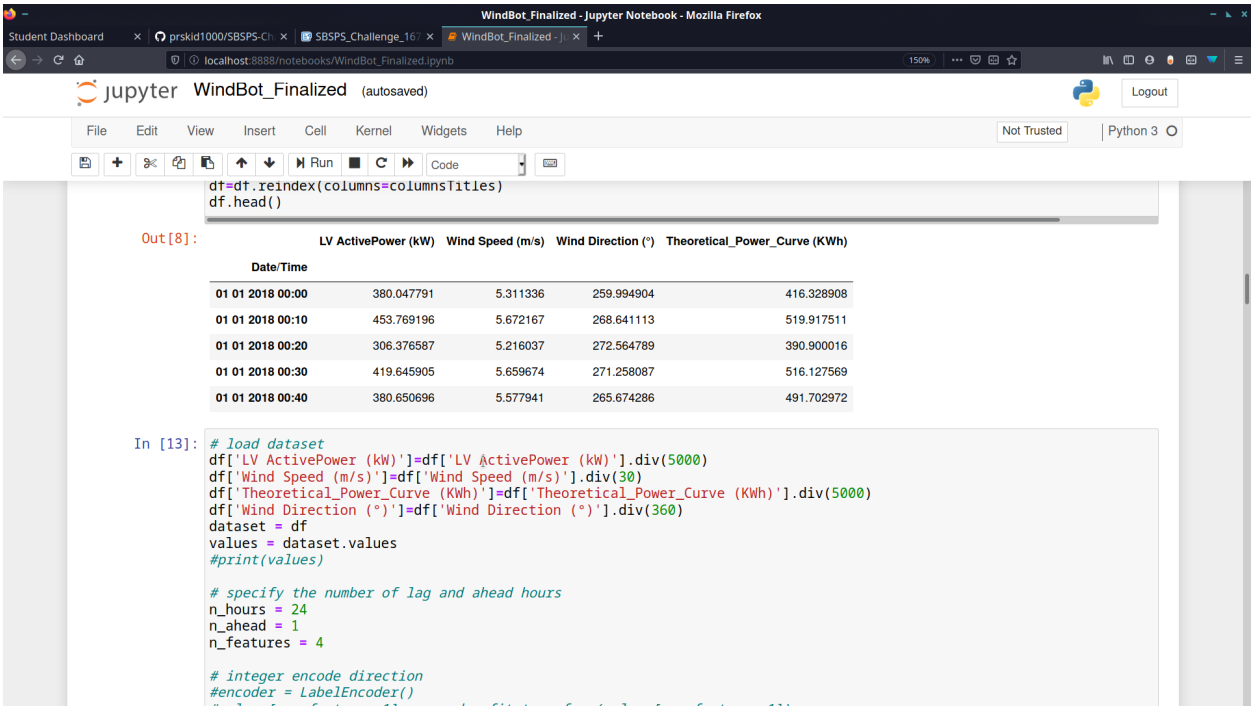
*Finally Node-Red Server will return all the data required to plot a graph of energy prediction of the next 72 hours.*
*It will also display best hour to use the energy..*

# Technical Details regarding the model




WindBot_Finalized - Jupyter Notebook - Mozilla Firefox
Student Dashboard
prskid1000/SBSPS-Cl
SBSPS_Challenge_167
WindBot_Finalized - Ju
localhost:8888/notebooks/WindBot_Finalized.ipynb
jupyter WindBot_Finalized (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
#print(train_y)
# reshape input to be 3D [samples, timesteps, features]
train_X = train_X.reshape((train_X.shape[0], n_hours, n_features))
test_X = test_X.reshape((test_X.shape[0], n_hours, n_features))
print(train_X.shape, train_y.shape, test_X.shape, test_y.shape)
(50479, 96) 50479 (50479,)
(50479, 24, 4) (50479,) (27, 24, 4) (27,)
In [18]:
# design network
model = Sequential()
model.add(LSTM(5, input_shape=(train_X.shape[1], train_X.shape[2])))
model.add(Dense(1))
model.compile(loss='mae', optimizer='adam')
In [19]:
# fit network
history = model.fit(train_X, train_y, epochs=15, batch_size=24, validation_data=(test_X, test_y), verbose=2, shuffle
model.save("WindBot.h5")
WARNING:tensorflow:From /opt/conda/envs/Python36/lib/python3.6/site-packages/tensorflow/python/ops/math_ops.py:306
6: to_int32 (from tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Train on 50479 samples, validate on 27 samples
Epoch 1/15
- 406s - loss: 0.0419 - val_loss: 0.0713
Epoch 2/15
- 401s - loss: 0.0340 - val_loss: 0.0766
Epoch 3/15


WindBot_Finalized - Jupyter Notebook - Mozilla Firefox
Student Dashboard
prskid1000/SBSPS-Cl
SBSPS_Challenge_167
WindBot_Finalized - Ju
localhost:8888/notebooks/WindBot_Finalized.ipynb
jupyter WindBot_Finalized (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
df=df.reindex(columns=columnsTitles)
df.head()
Out[8]:
LV ActivePower (kW) Wind Speed (m/s) Wind Direction (°) Theoretical_Power_Curve (KWh)
Date/Time
01 01 2018 00:00 380.047791 5.311336 259.994904 416.328908
01 01 2018 00:10 453.769196 5.672167 268.641113 519.917511
01 01 2018 00:20 306.376587 5.216037 272.564789 390.900016
01 01 2018 00:30 419.645905 5.659674 271.258087 516.127569
01 01 2018 00:40 380.650696 5.577941 265.674286 491.702972
In [13]:
# load dataset
df['LV ActivePower (kW)']=df['LV ActivePower (kW)'].div(5000)
df['Wind Speed (m/s)']=df['Wind Speed (m/s)'].div(30)
df['Theoretical_Power_Curve (KWh)']=df['Theoretical_Power_Curve (KWh)'].div(5000)
df['Wind Direction (°)']=df['Wind Direction (°)'].div(360)
dataset = df
values = dataset.values
#print(values)
# specify the number of lag and ahead hours
n_hours = 24
n_ahead = 1
n_features = 4
# integer encode direction
#encoder = LabelEncoder()

## Working of Nodered backend

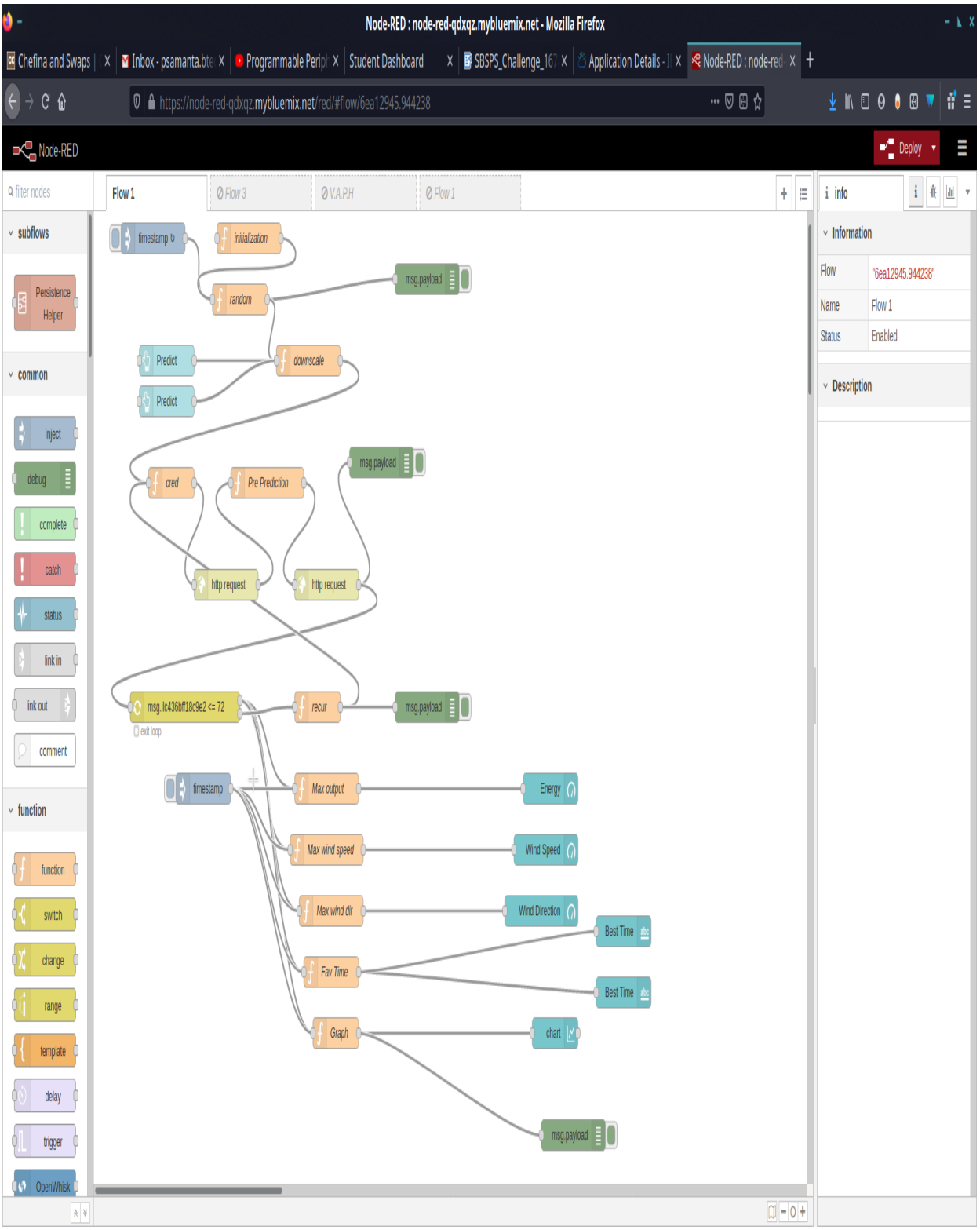
We initialize some global variables which can be used any time in our code.

We record the wind speed, wind direction, actual power output, and theoretical power output of the last 24 hours. We could have done so using the IoT Device but the option for its customization was not available for us. So, we analyzed the dataset and generated random input data based on the most common range given in the provided dataset. The data will keep getting updated every hour and we will have the data of the last 24 hours which we will use for prediction.

Then we downscale our data and supply it to the trained machine learning model.
We obtain the data for the next hour and update our input data with it and supply it again to the trained model. Thus we continue this process 72 times and obtain the predictions for the next 72 hours

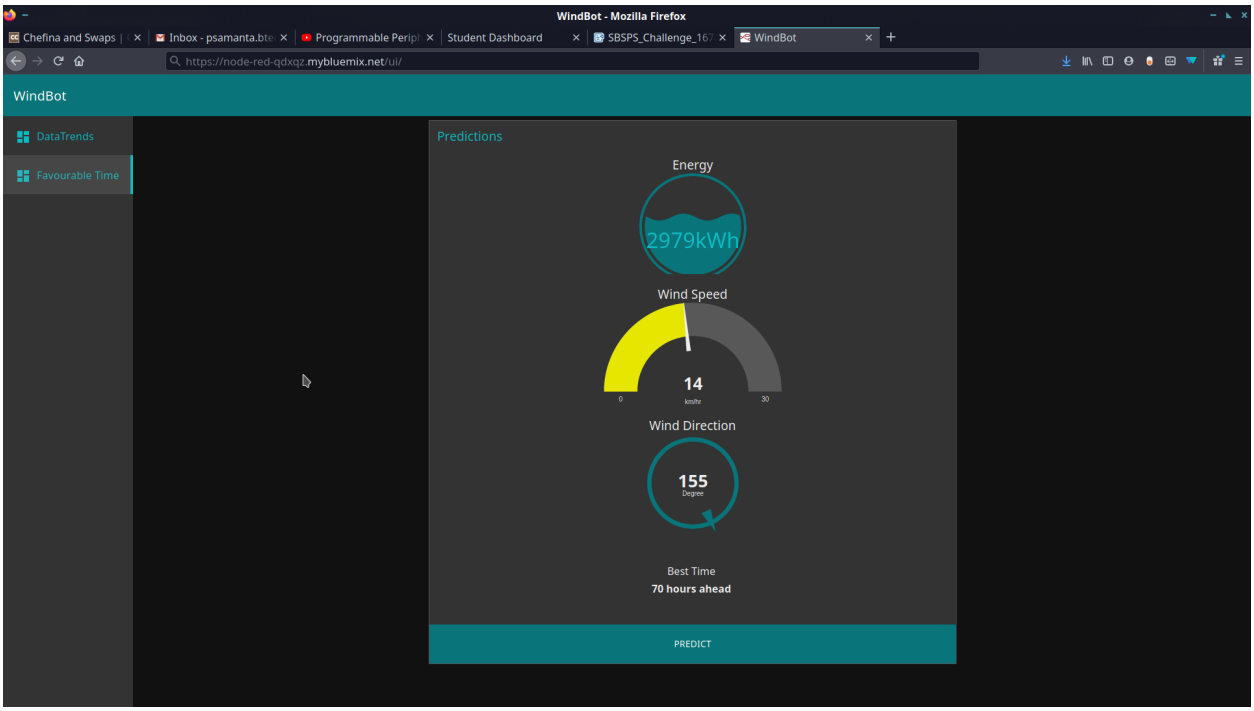
Then we upscale our output and supply it to the frontend and display the output graph, maximum output, the wind speed, and wind direction at that output. We will also display the time at which we will obtain the maximum output power.

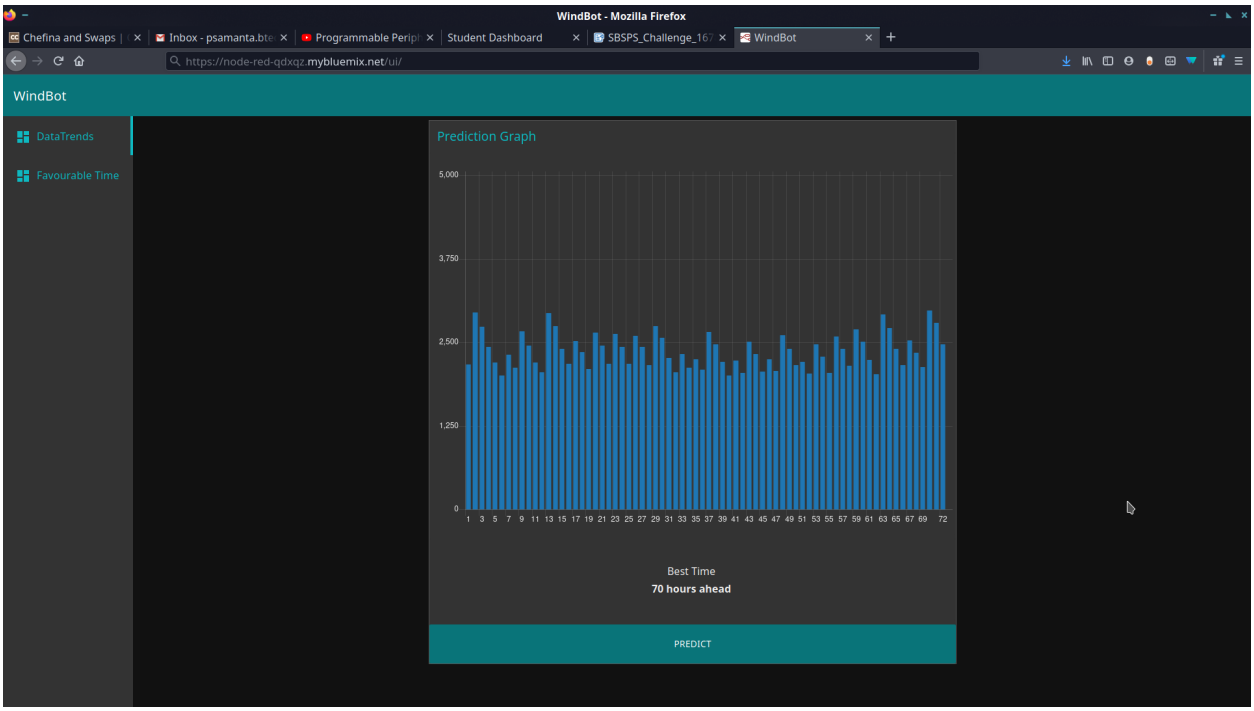
# Flow-View


Node-RED : node-red-qdxqz.mybluemix.net - Mozilla Firefox
https://node-red-qdxqz.mybluemix.net/red/#flow/6ea12945.944238
Node-RED
Deploy
Flow 1
Flow 3
V.A.P.H
Flow 1
subflows
Persistence Helper
common
inject
debug
complete
catch
status
link in
link out
comment
function
function
switch
change
range
template
delay
trigger
OpenWhisk
timestamp
initialization
random
msg.payload
Predict
Predict
downscale
cred
Pre Prediction
msg.payload
http request
http request
msg.ilc436bff18c9e2 <= 72
exit loop
recur
msg.payload
timestamp
Max output
Energy
Max wind speed
Wind Speed
Max wind dir
Wind Direction
Best Time
Fav Time
Best Time
Graph
chart
msg.payload
info
Information
Flow
"6ea12945.944238"
Name
Flow 1
Status
Enabled
Description

# Application Details

## Link: https://node-red-qdxqz.mybluemix.net/ui/



## Video Link

https://drive.google.com/file/d/1RteSeDfWqTvcNk-IeViG_BKggu1xGZIx/view?usp=drivesdk