### **Predictive Maintenance of Generators**

T. Uamesh

T Omasree

S Naga Phani

T Priyanka

TADEPALLI UTTEJ NAGA VENKATA MARUTHI RAO

**INTRODUCTION:**

Predictive maintenance (PDM) relies on condition-monitoring equipment to assess the performance of assets in real-time. By combining condition-based diagnostics with predictive formulas and with a little help from the Internet of Things (IoT), PDM creates an accurate tool for collecting and analyzing asset data.

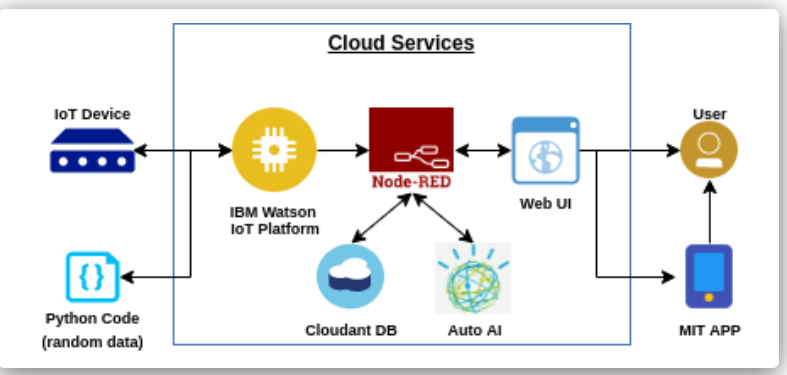
**LITERATURE SURVEY:**

1. **EXISTING PROBLEMS:**

Detect emerging issues, simplify root cause analysis, reduce warranty risk. Use machine learning to identify issuesfor improved quality, reduced financial impact. Machine Equipment Failure. Service Pack Optimization. IOT Data Analytics. Manufacturing Analytics.

**THEORITICAL ANALYSIS**

**BLOCK DIAGRAM:**



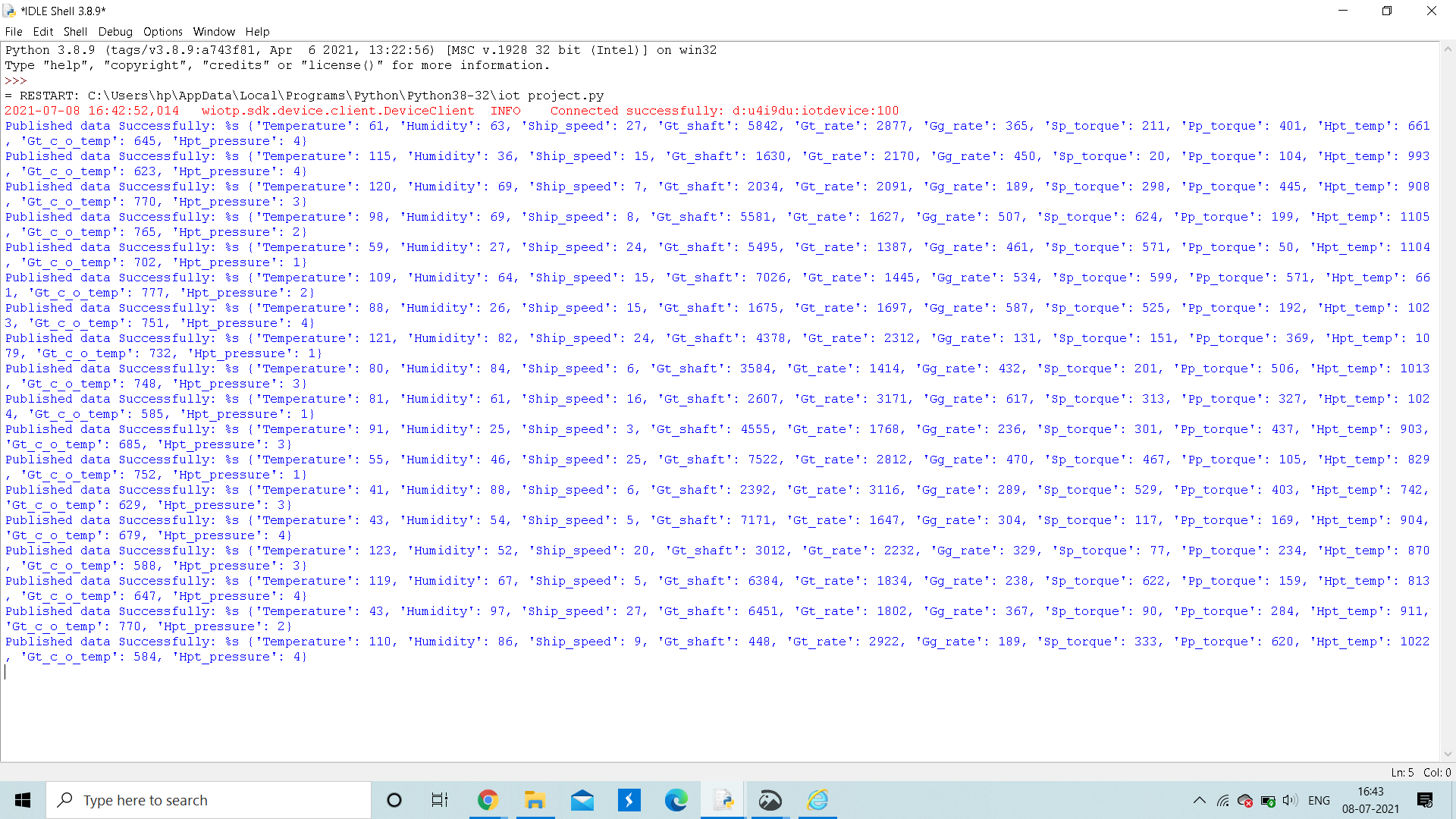
This is the block diagram of predictive maintenance of industrial generators

1. **HARDWARE/SOFTWARE DESIGNING**

* In this we have used python code
* Nodered platform to store data
* Mit APP inventor to create mobile application

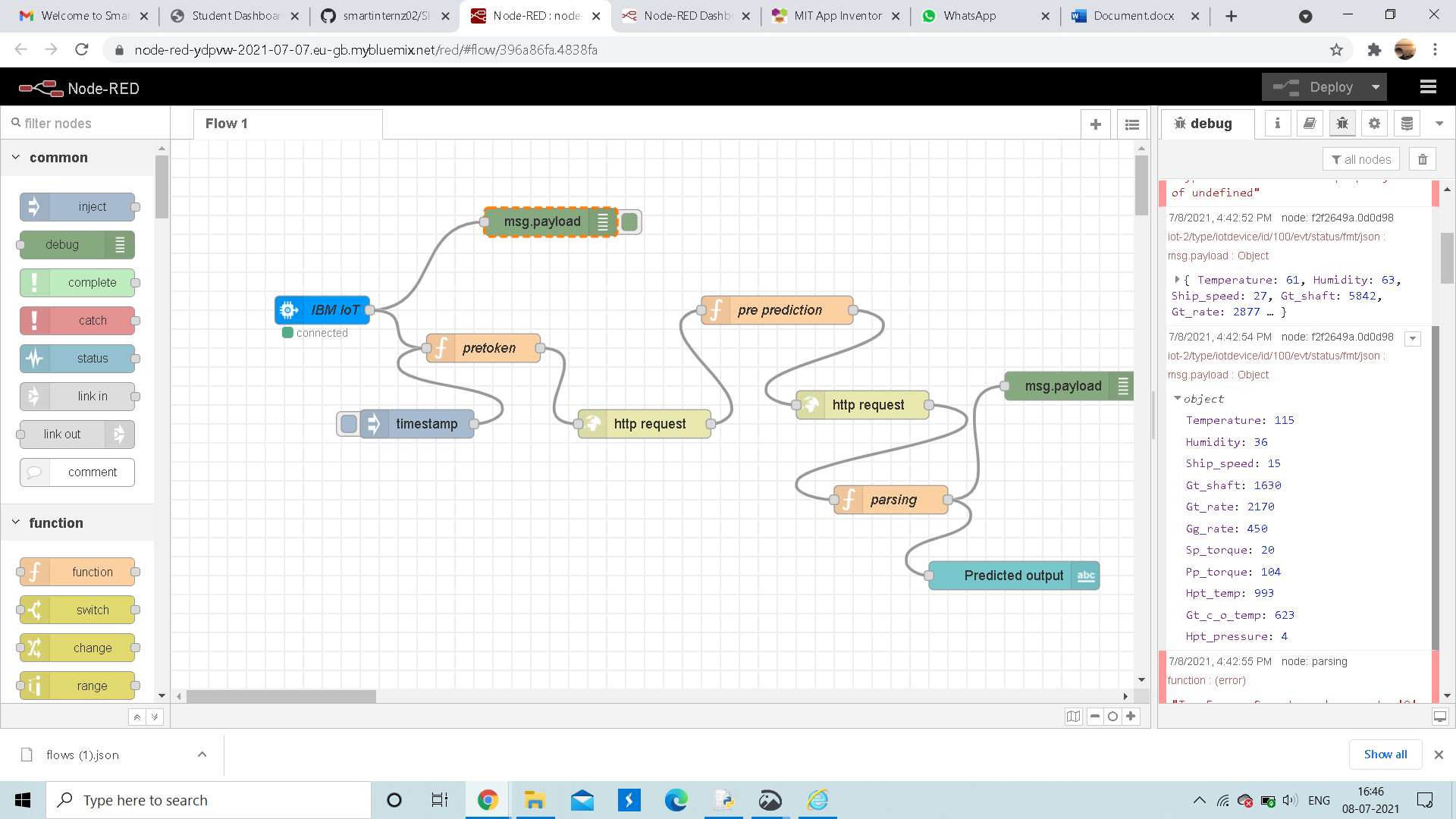
**EXPERIMENTAL INVESTIGATIONS**

In this project we first we connected device with device credentials and then generated random values of required parameters like temperature, humidity,ship\_speed, Gt\_shaft, Gt\_rate, no suggestions by using random function in Python code.



Through the code the data is sent to IBM cloud input node. To observe the Ui dashboard we need function node it is connected to IBM cloud input. Debug node which is connected to function node is to deploy the message in debug panel.

This is our node red connections in which we have connected a parsing function node which is used to set the parameters values and an http request node and a pre prediction node to get the values of the parameters, and we have connected an http request node in which we have kept an URL which was obtained when creating an AI Model, and finally a template node is used to display the values in UI.

.

**ADVANTAGES & DISADVANTAGES:**

**A: ADVANTAGES:**

* Reduction in maintenance costs
* Reduction in machine failures
* Reduced downtime for repairs
* Reduced stock of spare parts
* Increased service life of parts
* Increased production
* Improved operator safety.
* Less risk factor- Because the equipment and your building are being regularly checked, they are at less risk to breaking down without notice. Therefore, creating a safer working environment for employees.
* Follows a schedule- By following a schedule, you are able to keep to a budget while maintaining your building. Also, you will be able to keep track of all your equipment and pin point times when you will need to replace your equipment.
* Longer equipment/building life- When equipment is being checked and maintained, it will be kept in its best shape, therefore extending its lifetime. With routine check-ups on building parts such as pipes, boilers, and roofing you’ll extend the life of your building as well.
* Money saving-Over time, you will see that less money is being spend because you will not have to replace equipment as much, as well as dealing with last minute break downs. While there still may be some unplanned maintenance needed, the likely hood will go down when the building and equipment are regularly checked. Property wise, you’ll be able to catch roof leaks before they escalate and quickly repair them before mould and debris occur.
* Less energy wasting- In general when equipment is not kept in the best conditions possible, it will drain more energy, hiking up your utilities bill. With properly maintained equipment, it will be saving you energy and money. While regularly kept lighting and cooling/heating systems will also help reduce the energy bill.
* Less disruptions- With regular checks, you won’t be surprised when something goes wrong. It will be a quick fix because you will know what needs to be done. There will not be problems when it comes to closing down your property and disrupting your workers, if a large problem were to occur.

**B: DISADVANTAGES:**

* Data can be misinterpreted, leading to false maintenance requests,
* It's costly to establish a complete IoT system with sensors, transmission costs and analysis,
* Predictive analysis may not take contextual information into account, such as equipment age or weather.
* More money upfront- When initially starting a preventative maintenance plan, it will cost you more to regularly maintain equipment and the building, than it would be if you waited for things to simply break down.
* Over maintenance- Because there is a regular plan, sometimes items may not need to be checked as often as planned. If this is the case, you can change your maintenance plan to checking the specific equipment or areas less often, while still maintaining a schedule.
* More workers- Preventative maintenance require more workers because regular checks are a must. When compared to reactive maintenance, you simply need to call someone in for a onetime fix. Instead, this method requires workers to always be on site and perform daily works.

**APPLICATIONS:**

* Monitoring generators
* Alerting system

**Conclusion:**

* Generators should be maintained and serviced regularly, for this IoT will keep on tracking the crucial parameters while the generator is working.
* All the monitored parameters are stored in the node red on which we implement machine learning algorithms to predict the failures.
* The data is visualized in the Web App

**FUTURE SCOPE:**

* It can be used for predictive maintenance of motors

**BIBILOGRAPHY:**

<https://github.com/gnaneshwarbandari/IOT/blob/main/ibm_code.py>

<https://flows.nodered.org/node/node-red-dashboard>

**APPENDIX:**

**A: SOURCE CODE:**

import wiotp sdk .device

import time

import random

myConfig = {

"identity": {

"orgId": "s65wg1",

"typeId": "new",

"deviceId":"100"

},

"auth": {

"token": "12345678"

}

}

def myCommandCallback(cmd):

print("Message received from IBM IoT Platform: %s" % cmd.data['command'])

print(type(cmd.data['command']))

m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect()

while True:

Temp=random.randint(20,125)

Hum=random.randint(25,100)

ship=random.randint(3,27)

gt=random.randint(289,8379)

gt\_rat=random.randint(1364,3560)

gg\_rate=random.randint(92,644)

sp\_trq=random.randint(7,644)

pp\_trq=random.randint(7,644)

hpt\_temp=random.randint(635,1114)

gt\_c\_o\_temp=random.randint(559,788)

hpt\_prs=random.randint(1,4)

myData={'Temperature':Temp, 'Humidity':Hum, 'Ship\_speed':ship, 'Gt\_shaft':gt, 'Gt\_rate':gt\_rat, 'Gg\_rate':gg\_rate, 'Sp\_torque':sp\_trq, 'Pp\_torque':pp\_trq, 'Hpt\_temp':hpt\_temp, 'Gt\_c\_o\_temp':gt\_c\_o\_temp,'Hpt\_pressure':hpt\_prs}

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)

print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

time.sleep(2)

client.disconnect()’

‘

**B:UI OUTPUT SCEENSHOT:**

