PROJECT REPORT

IOT Application Developer

SMART AGRICULTURE SYSTEM BASE IN IOT

Name : M.Ramya
Title : Smart Agriculture System Based On IOT
Application ID : SPS_APL_20200001282
Project ID :SPS_PRO_101

Internship At smartinternz.com@2020

By,

Mudhigonda Ramya
(ramyamudhigonda123@gmail.com)

Table Of Contents

INTRODUCTION:

1.1 OVERVIEW

Agriculture is major source of income for the largest development of India and is the major contributor to the India economy.In past decade it is observed that there are not much crop development in agriculture sector.There are number of factors which is responsible for this it may be due to water waste,low soil fertility,fertilizer abuse,climate change or diseases etc.,It is very essential to make effective the intervention in agriculture and the solution is IOT in integration with wireless sensor network.IOT is a method of connecting everything to the internet,this project uses IOT technology in agriculture,gathering crops growth environmental parameters in a fixed place to help farmers find problems in time.Agriculture experts give guidelines with specific information to increase the farmer's income and help them in prevention and control of crop diseases.Through the customer development of mobile phone apps,it has been implemented with agriculture technology promotion and expert online FAQ.

1.2 PURPOSE

The main purpose of the IOT is ensuring delivery of right information to right people at right time.simple and easy to install and configure.

Need of smart agriculture irrigation:

- Saving energy and resources,so that it can be utilized in proper way.
- Farmers would be able to smear to right amount of water at the right time by automatic irrigation.
- Automated irrigation system uses values to turn motor ON and OFF.motors can be automated easily by using controllers and no need of labor to turn motor ON and OFF.
- it measures accurate soil moisture,temperature,humidity.
- It is time saving,human error elimination in adjusting soil moisture levels.
- High reliability,security,compatibility of technical requirement,scalability.

LITERATURE SURVEY:

2.1 EXISTING PROBLEM

Experts have analyzed collected data for finding correlation between environment work and yield for the standard work.they are concentrated on crop monitoring,information of temperature and rainfall is collected as initial spatial data and analyzed to reduce the crop losses and to improve the crop production.[1]
An IOT based agriculture monitoring system explains to monitor a crop field.A system is developed by using sensors and according to the decision from a server based on sensed data.[2]By using the wireless transmission the sensed data forwarded to web server database.the user can monitor and control the system remotely with the help of application which provides the web user[3].IOT based agriculture monitoring system develops various features like GPS remote controlled monitoring,temperature,humidity, moisture sensing.

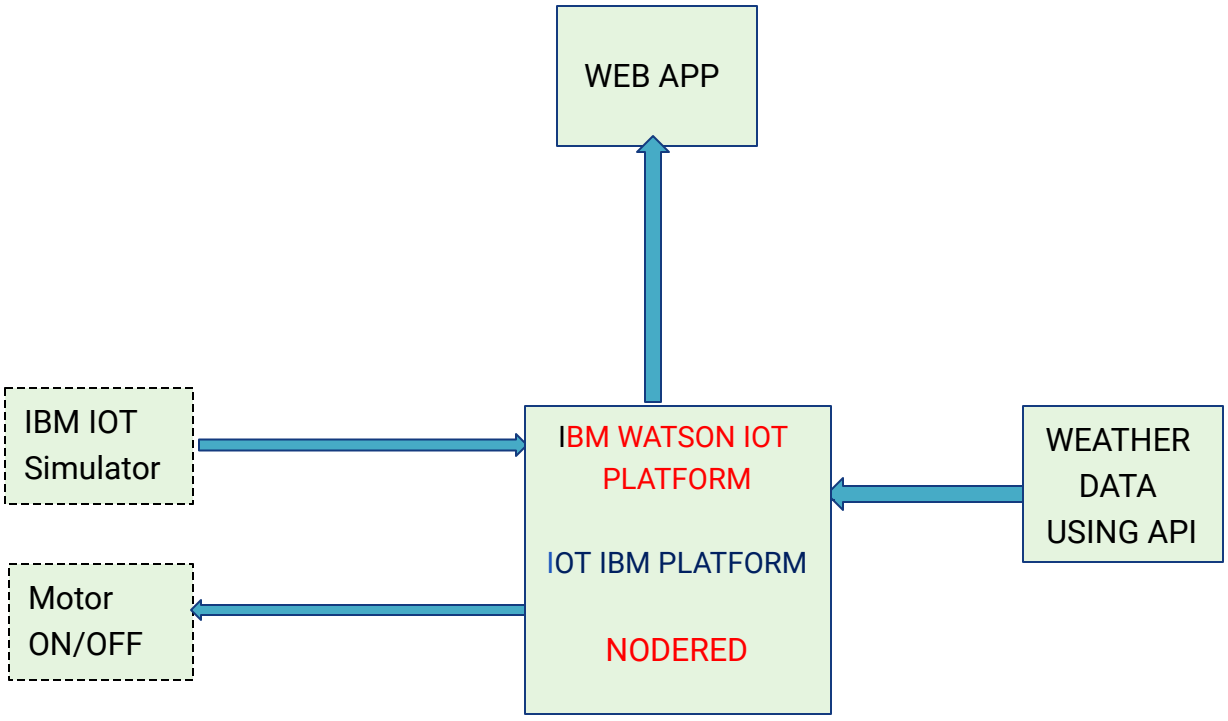
2.2 PROPOSED SOLUTION

The solution is that using IOT for monitoring the agriculture through the webserver.the system development composes parts: the server,PC client,open weather API,sensor, IBM cloud platform.the mobile application will be developed in android .it helps to monitor an controlled field from anywhere.

- The supply of water can becontrolled from anywhere by controlling the motor state(ON/OFF),using web application.
- real time weather conditions can also be cheked using theopen weatherAPI's from different websites and it is displayed on our web app.
- soil moisure and other parameters can be checked by using the watson iot sensor and send the data to cloud services& to the web app.
- our surrounding temperature can be checked by the sensor &controlled easily

THEORITICAL ANALYSIS:

3.1 BLOCK DIAGRAM:



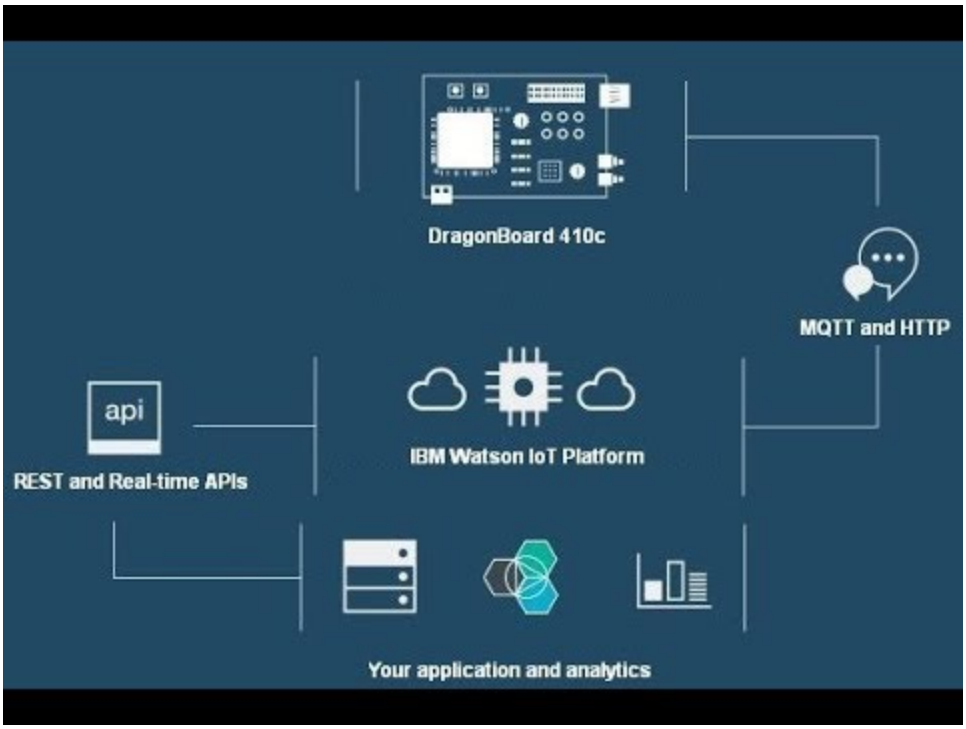
3.2 HARDWARE&SOFTWARE DESIGNING

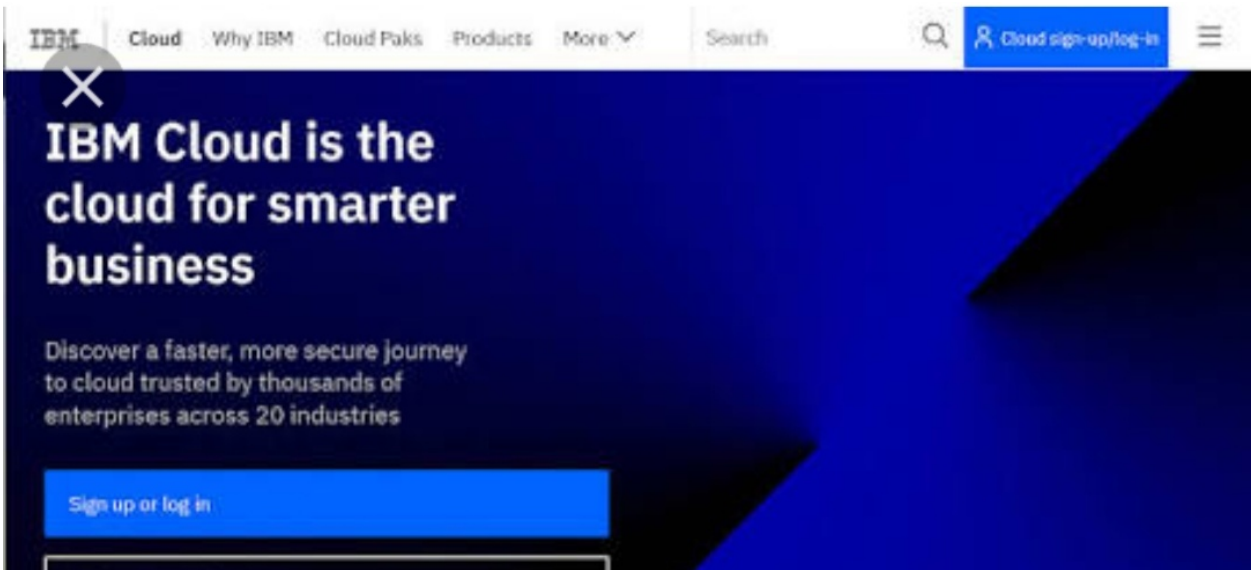
1. A Device is created tn IBM cloud.
2. connect that device to the watson iot sensor in order to get weather conditions.
3. create the Node-red flow to build a web application to display the weather conditions and to controll the devices
4. Get the real time weather condition data from open weather API and configure it to the Node-red.
5. A python code should write to retrieve the commands from web app in IBM platform. In order to controll the motor(ON/OFF) from anwhere.

SOFTWARE REQUIREMENTS:

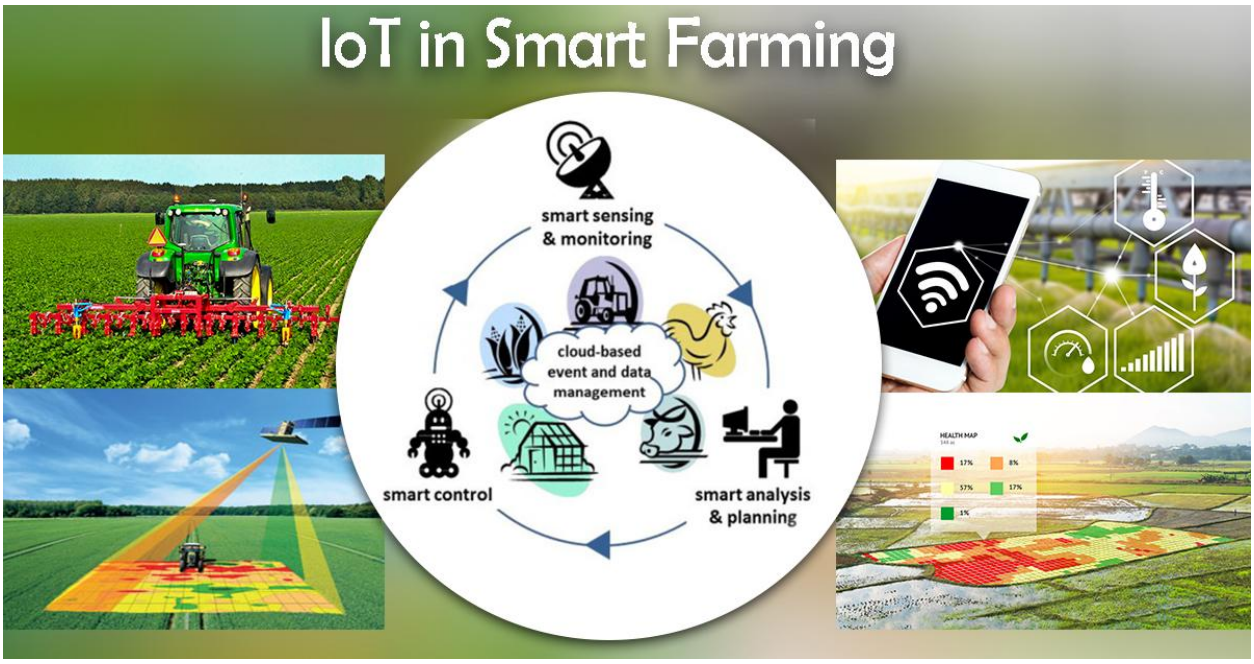
1.IBM CLOUD
2.IBM WATSON IOT PLATFORM
3.NODE-RED
4.OPEN WEATHER API
5.PYTHON 3 IDLE
6.WATSON IOT SENSOR SIMULATOR

WATSON IOT PLATFORM





EXPERIMENTAL INVESTIGATIONS:



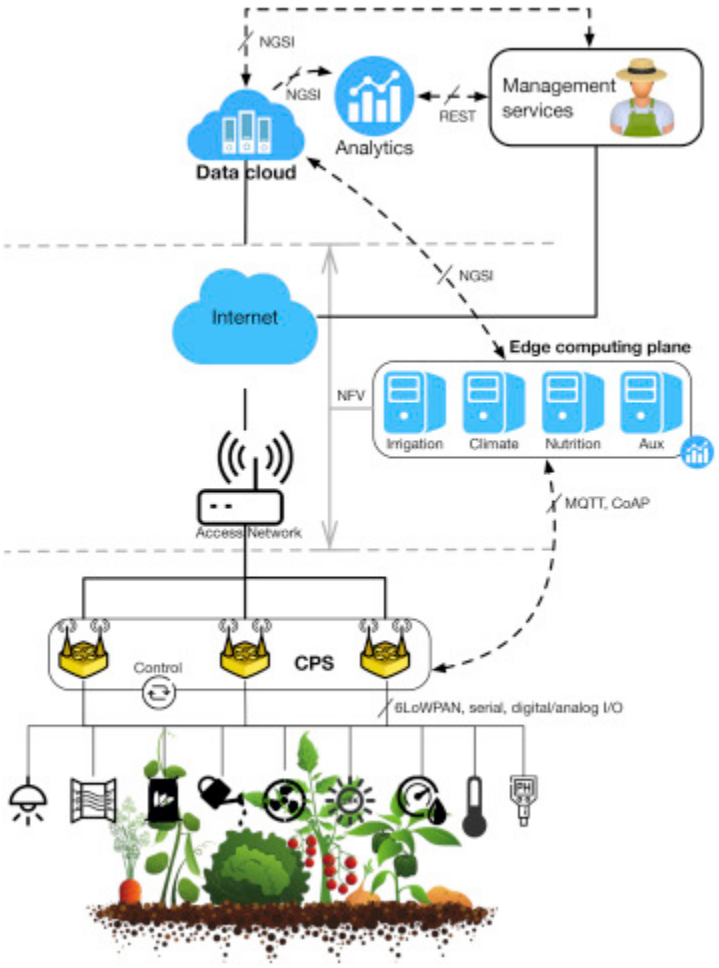
Monitor the farming through the cloud and Iot from anywhere in mobile phone app.

FLOWCHARTS:

NODE-RED FLOW

Hear we are used node-red programming tool in order to connect the devices.
NODE-RED is a flow based development tool for visual programming developed by IBM.
For wiring together hardware devices,API's and online services as part of internet of things(IOT).
Node-RED provides a web browser-based flow editor,which can be used to create Javascripts functions.

THIS IS THE GENERAL FLOWDIAGRAM OF SMART AGRICULTURE SYSTEM BASED ON IOT



In this flow first sensors will take the information from the farm like temperature,humidity,soil moisture,water level,sunlight etc.,the sensor is that IBM Iot simulator.

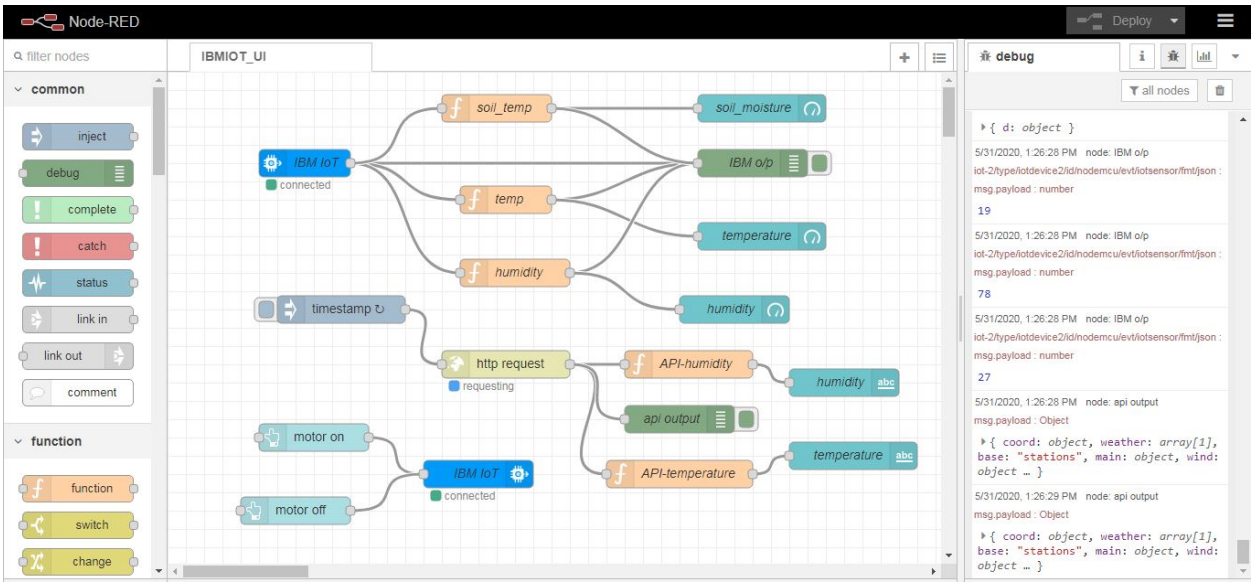
After it will send the information to he IBM Watson IOT platform means cloud which access through the internet,in that we have node-red flow programming developing tool,there flow will be created and debugs. there we will get the output ,which connects through the weather data using API f current city.

By seeing the all data we can control and access the data from anywhere through the web app using http request.Managemnt will control the farm by using motor ON/OFF commands from mobile phone app .this reduce the labor work,saves the time.

The nodes used in the web application and open weather map

1. IBM IOT :IN & OUT.
2. Function nodes.
3. Gauge nodes.
4. chart nodes.
5. Debug nodes.
6. Button nodes.
7. Timestamp node.
8. http request node.
9. Text nodes.

NODE-RED FLOW



RESULT:

WATSON IOT SENSOR OUTPUT

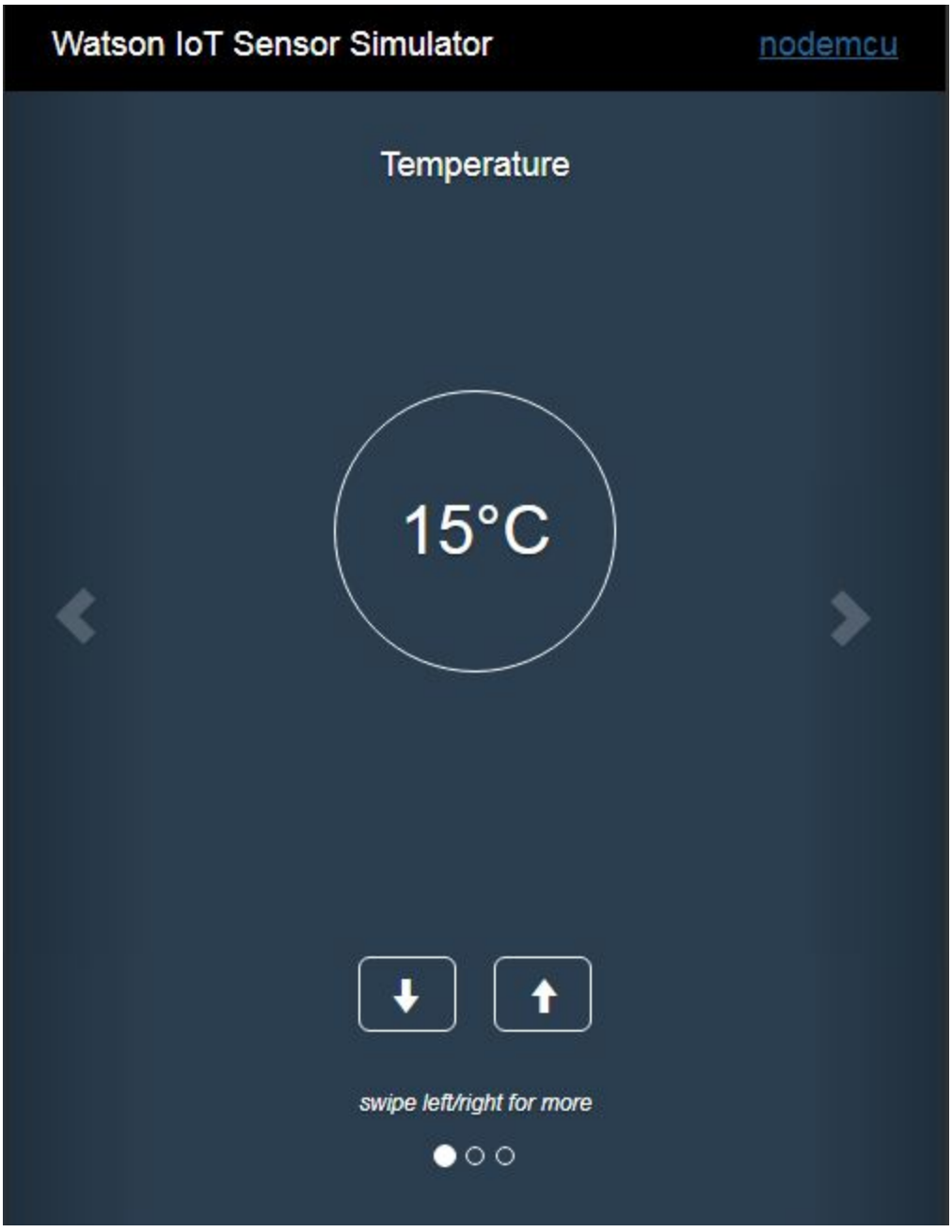


Fig:Readings of temperature in sensor

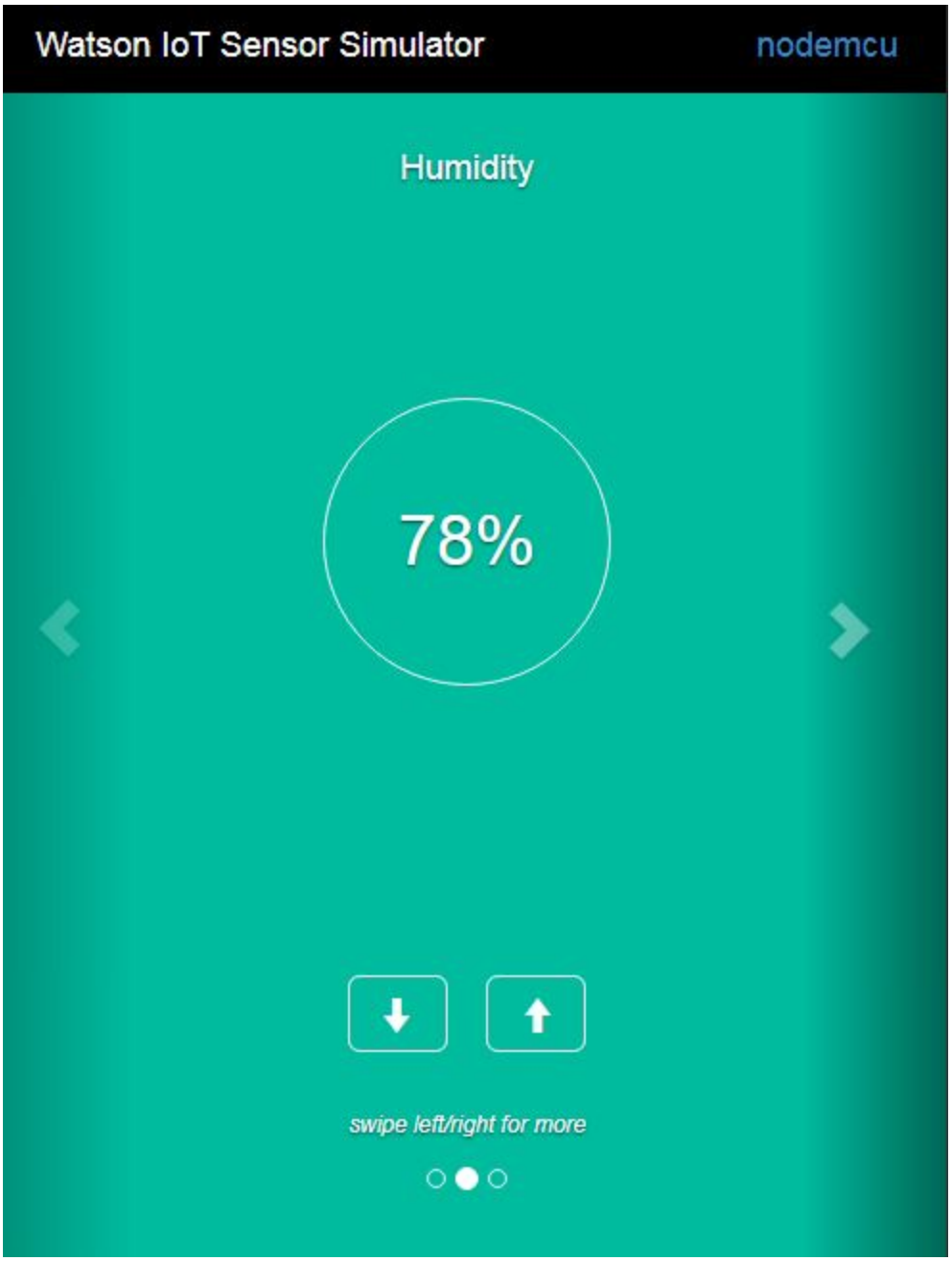


Fig:Readings of humidity in sensor

Fig:Readings of soil moisture in sensor

NODE-RED FLOW OUTPUT

This is the output of node-red flow,when we debug the node -red we will get this output and we can use motor on and motor off buttons to get the result

in the form of python code.

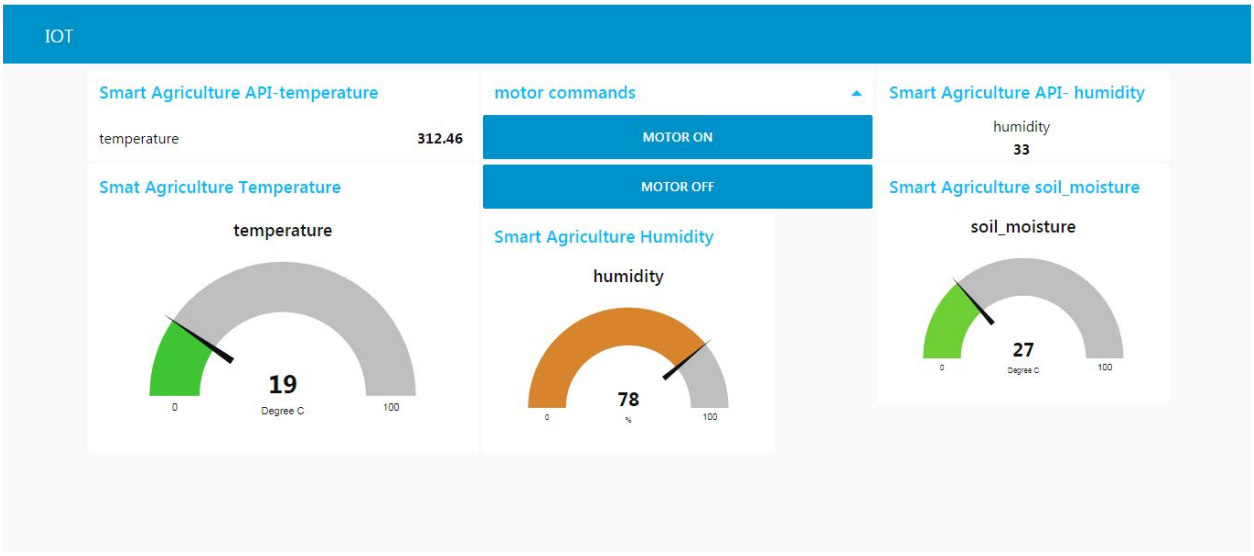
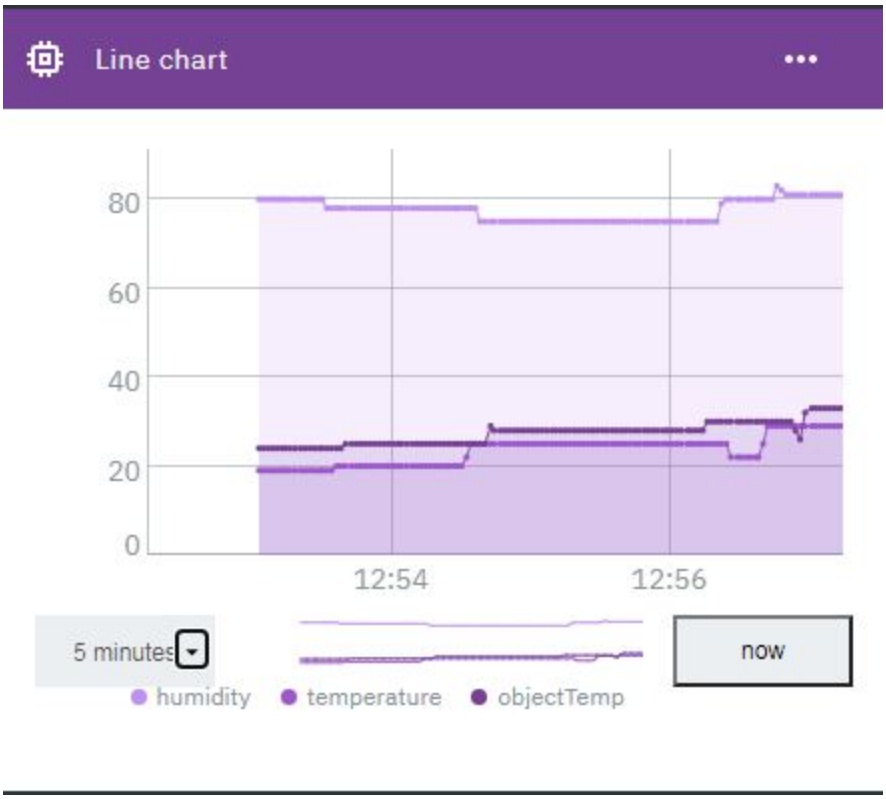
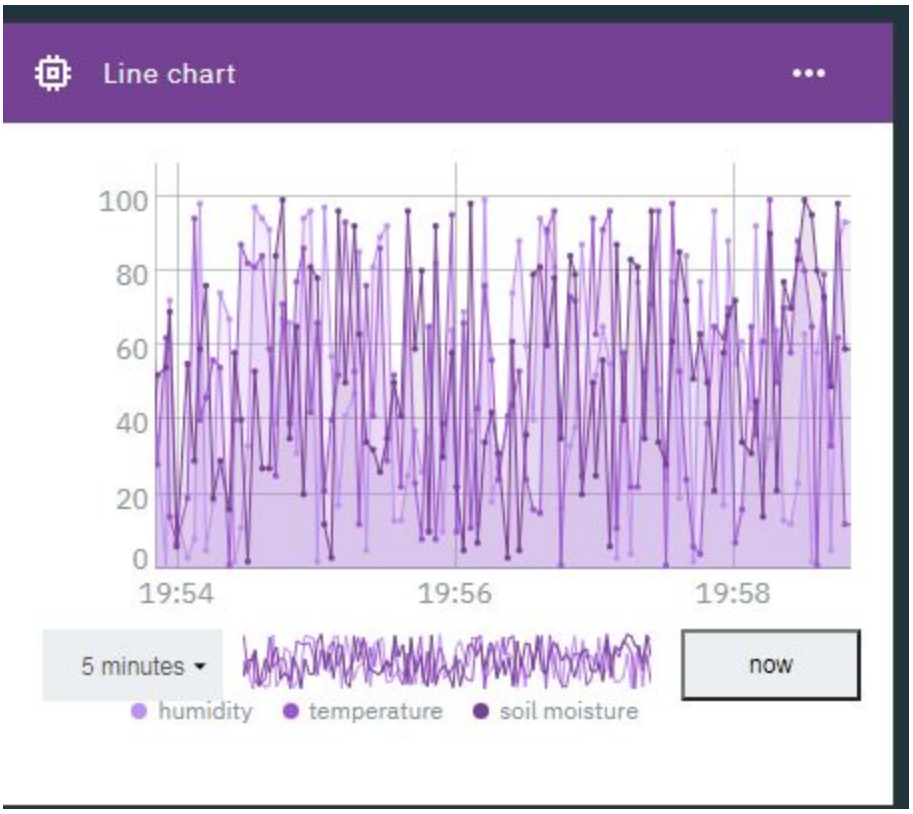


Fig:Readings in web app

GRAPH OF IOT SIMULATOR OUTPUT



DEVICE SIMULATOR OUTPUT



ADVANTAGES:

- Utilization of resources efficiently.
- Minimization of human efforts.
- Time saving.
- Increase data collection.

DISADVANTAGES:

- Security.
- Privacy.
- Complexity.

APPLICATIONS:

- Precision farming
- Agriculture drones
- Livestock monitoring

CONCLUSION:

The proposed model explores the use of IOT in agriculture sector.This model aims at increasing the crop yield by helping in predicting better crop sequence for a particular soil.Data on cloud helps the agriculturists in improving the yields.This system is cost effective and feasible.this system will sense all environmental parameters and send the data to the user via cloud.It leads to higher crop yield,better quality,and less use of protective chemicals.

FUTURE SCOPE:

The future scope of this project could be including variety of soil sensors like pH sensor, rain sensor and then collecting and storing the data on server.This would make the predicting and analyzing processes more accurate.

BIBLIOGRAPHY:

- IBM Cloud:
  https://cloud.ibm.com/docs/overview?topic=overview-whatis-platform
- Watson IOT:
  https://www.iotone.com/software/ibm-watson-iot-platform/s62
- Node-RED:
  https://nodered.org/docs/getting-started/windows#3-run-node-red
  https://www.youtube.com/watch?v=cicTw4SEdxk
- Openweathermap:
  https://openweathermap.org/
- Github:
  https://github.com/rachuriharish23/ibmsubscribe
- Watson iot simulator :
  https://watson-iot-sensor-simulator.mybluemix.net

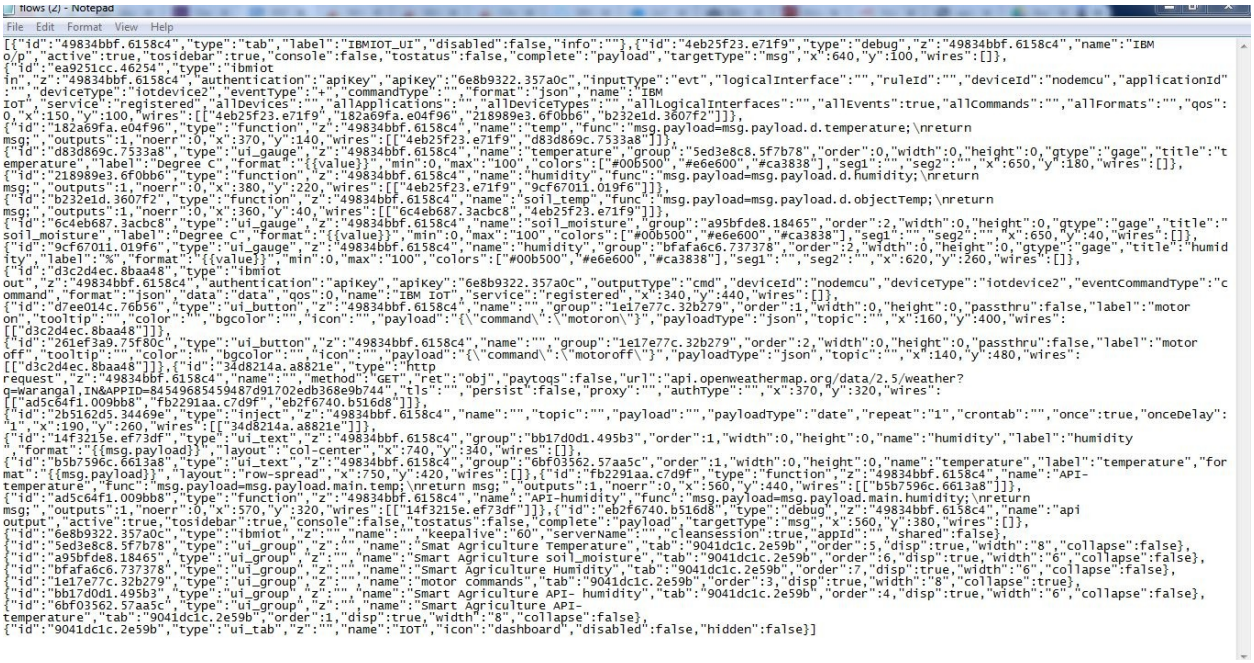
APPENDIX:

A. Source Code

```
import ime
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device
# provide your IBM Watson Device Credentials
organization="ORG ID" # replace the ORG ID
devicetype="Device type"# replacethe Device type
deviceId="Device ID "# replace Device ID
authMethod="token"
authToken="authtoken" # Replace the authtoken
def myCommandCallback(cmd): # function for callback
if cmd.data['command']=='motoron':
print("MOTOR ON IS RECEIVED")
elif cmd.data['command']=='motoroff':
print("MOTOR OFF IS RECEIVED")
if cmd.command=="setInterval":
if'interval' not in cmd.data:
print("Error-command is missing required
informaton:'interval'")
else:
interval=cmd.data['interval']
elif cmd.command=="print":
if 'message' not in cmd.data:
print("Error-command is missing required
information:'message'")
else:
output=cmd.data['message']
print(output)
try:
deviceOptions={"org":organization,"type":deviceType,"id":de
viceId,"auth-method":authMethod,"auth-token":authToken}
devicecli=ibmiotf.device.client(deviceoptions)
#.....................
except Exception as e:
print("Caught eception connecting device:%s"%str(e))
```

```
sys.exit()
#Connect and send a datapoint "hello"with the value "world"into the cloud as an event of type "greeting"10 times
deviceCli.connect()
#Disconnect the device and application from the cloud
deviceCli.disconnect
```

NODE-RED Flow:




```
[{"id":"49834bbf.6158c4","type":"tab","label":"IBMIOT_UI","disabled":false,"info":""},{"id":"4eb25f23.e71f9","type":"debug","z":"49834bbf.6158c4","name":"IBM
o/p","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":640,"y":100,"wires":[]},
{"id":"ea9251cc.46254","type":"ibmiot
in","z":"49834bbf.6158c4","authentication":"apiKey","apiKey":"6e8b9322.357a0c","inputType":"evt","logicalInterface":"","ruleId":"","deviceId":"nodemcu","applicationId"
:"","deviceType":"iotdevice2","eventType":"+","commandType":"","format":"json","name":"IBM
IoT","service":"registered","allDevices":"","allApplications":"","allDeviceTypes":"","allLogicalInterfaces":"","allEvents":true,"allCommands":"","allFormats":"","qos":
0,"x":150,"y":100,"wires":[["4eb25f23.e71f9","182a69fa.e04f96","218989e3.6f0bb6","b232e1d.3607f2"]]},
{"id":"182a69fa.e04f96","type":"function","z":"49834bbf.6158c4","name":"temp","func":"msg.payload=msg.payload.d.temperature;\nreturn
msg;","outputs":1,"noerr":0,"x":370,"y":140,"wires":[["4eb25f23.e71f9","d83d869c.7533a8"]]},
{"id":"d83d869c.7533a8","type":"ui_gauge","z":"49834bbf.6158c4","name":"temperature","group":"5ed3e8c8.5f7b78","order":0,"width":0,"height":0,"gtype":"gage","title":"t
emperature","label":"Degree C","format":"{{value}}","min":0,"max":"100","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":650,"y":180,"wires":[]},
{"id":"218989e3.6f0bb6","type":"function","z":"49834bbf.6158c4","name":"humidity","func":"msg.payload=msg.payload.d.humidity;\nreturn
msg;","outputs":1,"noerr":0,"x":380,"y":220,"wires":[["4eb25f23.e71f9","9cf67011.019f6"]]},
{"id":"b232e1d.3607f2","type":"function","z":"49834bbf.6158c4","name":"soil_temp","func":"msg.payload=msg.payload.d.objectTemp;\nreturn
msg;","outputs":1,"noerr":0,"x":360,"y":40,"wires":[["6c4eb687.3acbc8","4eb25f23.e71f9"]]},
{"id":"6c4eb687.3acbc8","type":"ui_gauge","z":"49834bbf.6158c4","name":"soil_moisture","group":"a95bfde8.18465","order":2,"width":0,"height":0,"gtype":"gage","title":"
soil_moisture","label":"Degree C","format":"{{value}}","min":0,"max":"100","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":650,"y":40,"wires":[]},
{"id":"9cf67011.019f6","type":"ui_gauge","z":"49834bbf.6158c4","name":"humidity","group":"bfafa6c6.737378","order":2,"width":0,"height":0,"gtype":"gage","title":"humid
ity","label":"%","format":"{{value}}","min":0,"max":"100","colors":["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":620,"y":260,"wires":[]},
{"id":"d3c2d4ec.8baa48","type":"ibmiot
out","z":"49834bbf.6158c4","authentication":"apiKey","apiKey":"6e8b9322.357a0c","outputType":"cmd","deviceId":"nodemcu","deviceType":"iotdevice2","eventCommandType":"c
ommand","format":"json","data":"data","qos":0,"name":"IBM IoT","service":"registered","x":340,"y":440,"wires":[]},
{"id":"d7ee014c.76b56","type":"ui_button","z":"49834bbf.6158c4","name":"","group":"1e17e77c.32b279","order":1,"width":0,"height":0,"passthru":false,"label":"motor
on","tooltip":"","color":"","bgcolor":"","icon":"","payload":"{\"command\":\"motoron\"}","payloadType":"json","topic":"","x":160,"y":400,"wires":
[["d3c2d4ec.8baa48"]]},
{"id":"261ef3a9.75f80c","type":"ui_button","z":"49834bbf.6158c4","name":"","group":"1e17e77c.32b279","order":2,"width":0,"height":0,"passthru":false,"label":"motor
off","tooltip":"","color":"","bgcolor":"","icon":"","payload":"{\"command\":\"motoroff\"}","payloadType":"json","topic":"","x":140,"y":480,"wires":
[["d3c2d4ec.8baa48"]]},{"id":"34d8214a.a8821e","type":"http
request","z":"49834bbf.6158c4","name":"","method":"GET","ret":"obj","paytoqs":false,"url":"api.openweathermap.org/data/2.5/weather?
q=warangal,IN&APPID=845496854594870d91702edb368e9b744","tls":"","persist":false,"proxy":"","authType":"","x":370,"y":320,"wires":
[["ad5c64f1.009bb8","fb2291aa.c7d9f","eb2f6740.b516d8"]]},
{"id":"2b5162d5.34469e","type":"inject","z":"49834bbf.6158c4","name":"","topic":"","payload":"","payloadType":"date","repeat":"1","crontab":"","once":true,"onceDelay":
"1","x":190,"y":260,"wires":[["34d8214a.a8821e"]]},
{"id":"14f3215e.ef73df","type":"ui_text","z":"49834bbf.6158c4","group":"bb17d0d1.495b3","order":1,"width":0,"height":0,"name":"humidity","label":"humidity
","format":"{{msg.payload}}","layout":"col-center","x":740,"y":340,"wires":[]},
{"id":"b5b7596c.6613a8","type":"ui_text","z":"49834bbf.6158c4","group":"6bf03562.57aa5c","order":1,"width":0,"height":0,"name":"temperature","label":"temperature","for
mat":"{{msg.payload}}","layout":"row-spread","x":750,"y":420,"wires":[]},{"id":"fb2291aa.c7d9f","type":"function","z":"49834bbf.6158c4","name":"API-
temperature","func":"msg.payload=msg.payload.main.temp;\nreturn msg;","outputs":1,"noerr":0,"x":560,"y":440,"wires":[["b5b7596c.6613a8"]]},
{"id":"ad5c64f1.009bb8","type":"function","z":"49834bbf.6158c4","name":"API-humidity","func":"msg.payload=msg.payload.main.humidity;\nreturn
msg;","outputs":1,"noerr":0,"x":570,"y":320,"wires":[["14f3215e.ef73df"]]},{"id":"eb2f6740.b516d8","type":"debug","z":"49834bbf.6158c4","name":"api
output","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":560,"y":380,"wires":[]},
{"id":"6e8b9322.357a0c","type":"ibmiot","z":"","name":"","keepalive":"60","serverName":"","cleansession":true,"appId":"","shared":false},
{"id":"5ed3e8c8.5f7b78","type":"ui_group","z":"","name":"Smat Agriculture Temperature","tab":"9041dc1c.2e59b","order":5,"disp":true,"width":"8","collapse":false},
{"id":"a95bfde8.18465","type":"ui_group","z":"","name":"Smart Agriculture soil_moisture","tab":"9041dc1c.2e59b","order":6,"disp":true,"width":"6","collapse":false},
{"id":"bfafa6c6.737378","type":"ui_group","z":"","name":"Smart Agriculture Humidity","tab":"9041dc1c.2e59b","order":7,"disp":true,"width":"6","collapse":false},
{"id":"1e17e77c.32b279","type":"ui_group","z":"","name":"motor commands","tab":"9041dc1c.2e59b","order":3,"disp":true,"width":"8","collapse":true},
{"id":"bb17d0d1.495b3","type":"ui_group","z":"","name":"Smart Agriculture API- humidity","tab":"9041dc1c.2e59b","order":4,"disp":true,"width":"6","collapse":false},
{"id":"6bf03562.57aa5c","type":"ui_group","z":"","name":"Smart Agriculture API-
temperature","tab":"9041dc1c.2e59b","order":1,"disp":true,"width":"8","collapse":false},
{"id":"9041dc1c.2e59b","type":"ui_tab","z":"","name":"IOT","icon":"dashboard","disabled":false,"hidden":false}]
```


GITHUB REPOSITORY:

[LINK OF GITHUB REPOSITORY]