PROJECT REPORT

PROJECT NAME : Smart Agriculture System Based on IoT

PROJECT ID : SPS_PRO_101

CLIENT : Farmer

PROJECT MANAGER : G.Vigneshwaran

DATE : 13-06-2020

INTRODUCTION :

- Agriculture is the main backbone of Indian economical growth. The most important barrier that arises in traditional farming is climate change.
- The number of effects of climate change includes heavy rainfall most intense storm and heat waves, less rainfall etc. due to these the productivity decrease to the major extent.
- Climate change also raises the environmental consequences such as the seasonal change in the life cycle of the plant. To boost the productivity and minimize the barrier in agriculture field there is need to use innovative technology and technique called Internet of things.
- The technological advances in their areas gather increasing momentum and this means that maintaining as the overview.

OVERVIEW :

The highlighting features of this project include smart irrigation with smart control based on real time fied data. monitor the temperature, humidity and soil moisture parameters along with weather forecasting details. And finally the recommendation to farmer for smart agriculture.

PURPOSE :

There are many reasons to implement a smart agriculture solution into commercial and local farming as well as in different agriculture related institutions and organizations. In a world where the Internet of Things (IOT) is accelerating adoption of automation and data gathering, an important industry such as agriculture can surely be benefited and our project of making agriculture in a smarter way will definitely help in the growth of this industry.

Existing problem :

- Less Accurancy in weather data.
- Farmer will require certain skill sets in particular in order to understand and operate the equipment.
- For the smart agriculture, Internet of Things is essential which will require artificial intelligence and computer-based intelligence. This cannot be balanced here.
- There is no force which can change or control the weather conditions such as rain, sunlight, drought etc.

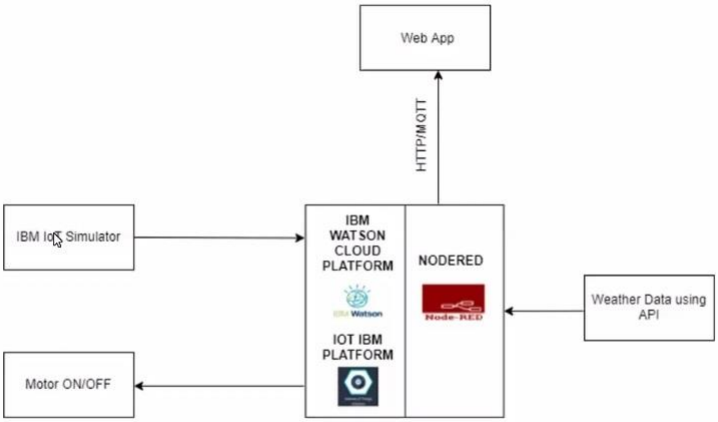
- While the use of smart technology in agriculture is impressive, it does incur a lot of costs.

Proposed solution :

- Real-Time Data and Production Insight
- Controlling the motors using the mobile application from anywhere.
- Monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.
- Based on all the parameters he can water his crop by controlling the motors using the mobile application.
- The Online IoT simulator for getting the Temperature, Humidity and Soil Moisture values.

THEORITICAL ANALYSIS :

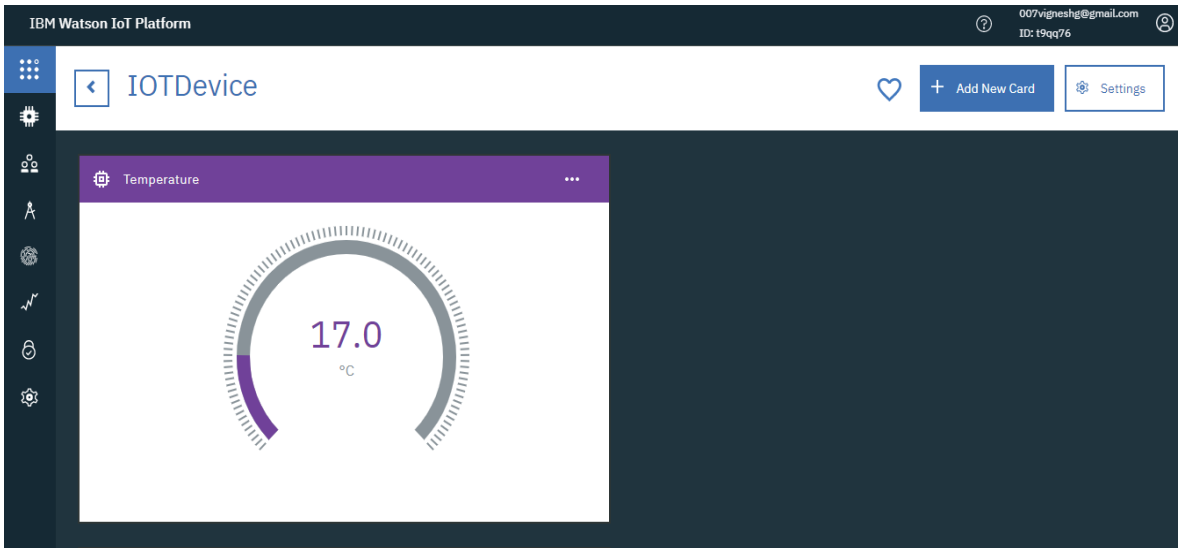
Block Diagram :

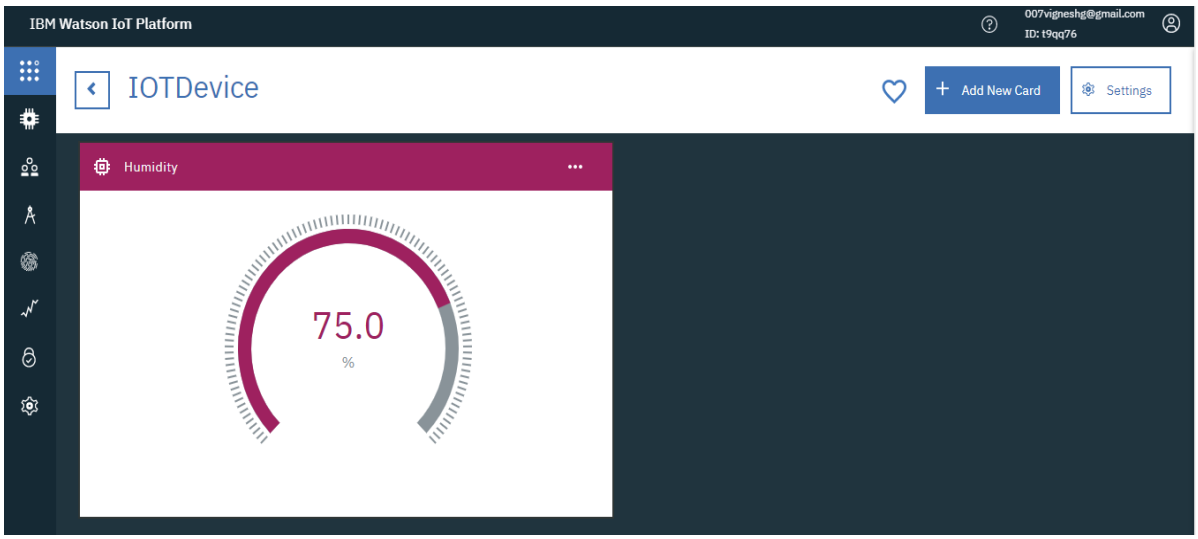


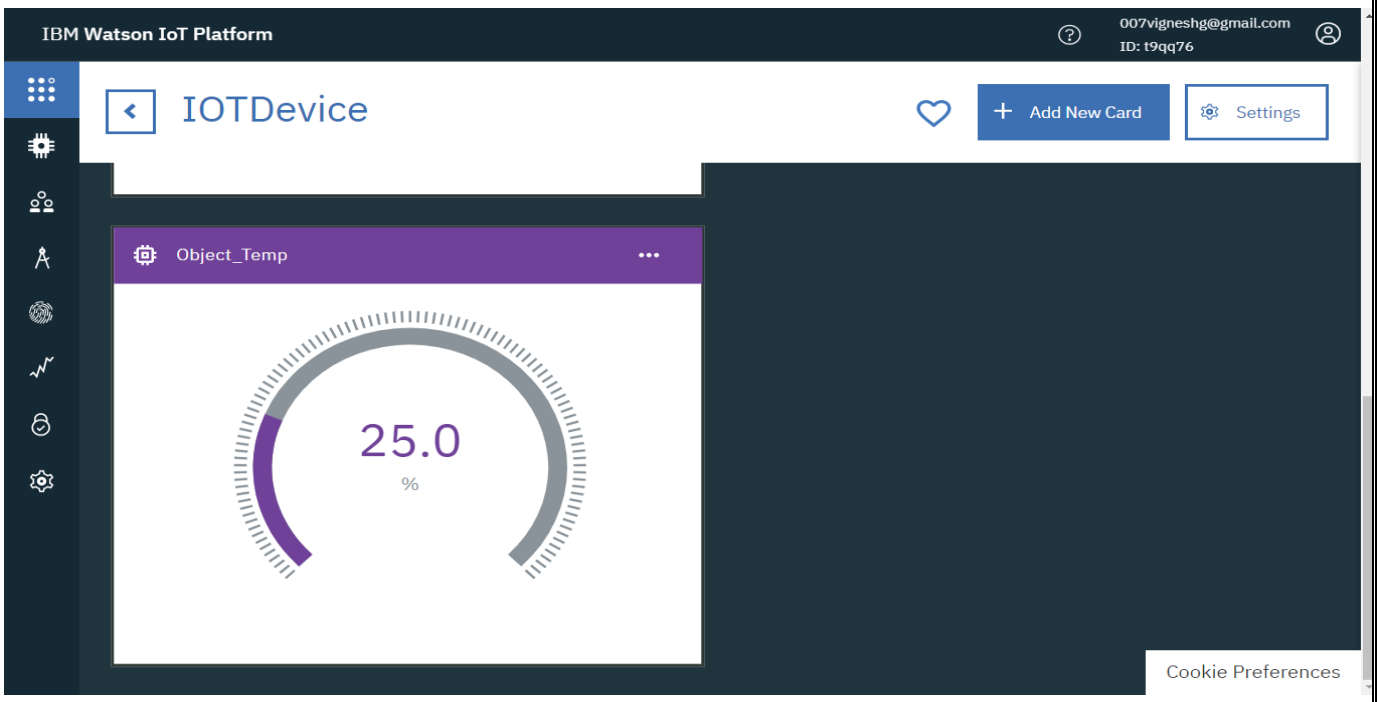
EXPERIMENTAL INVESTIGATIONS :

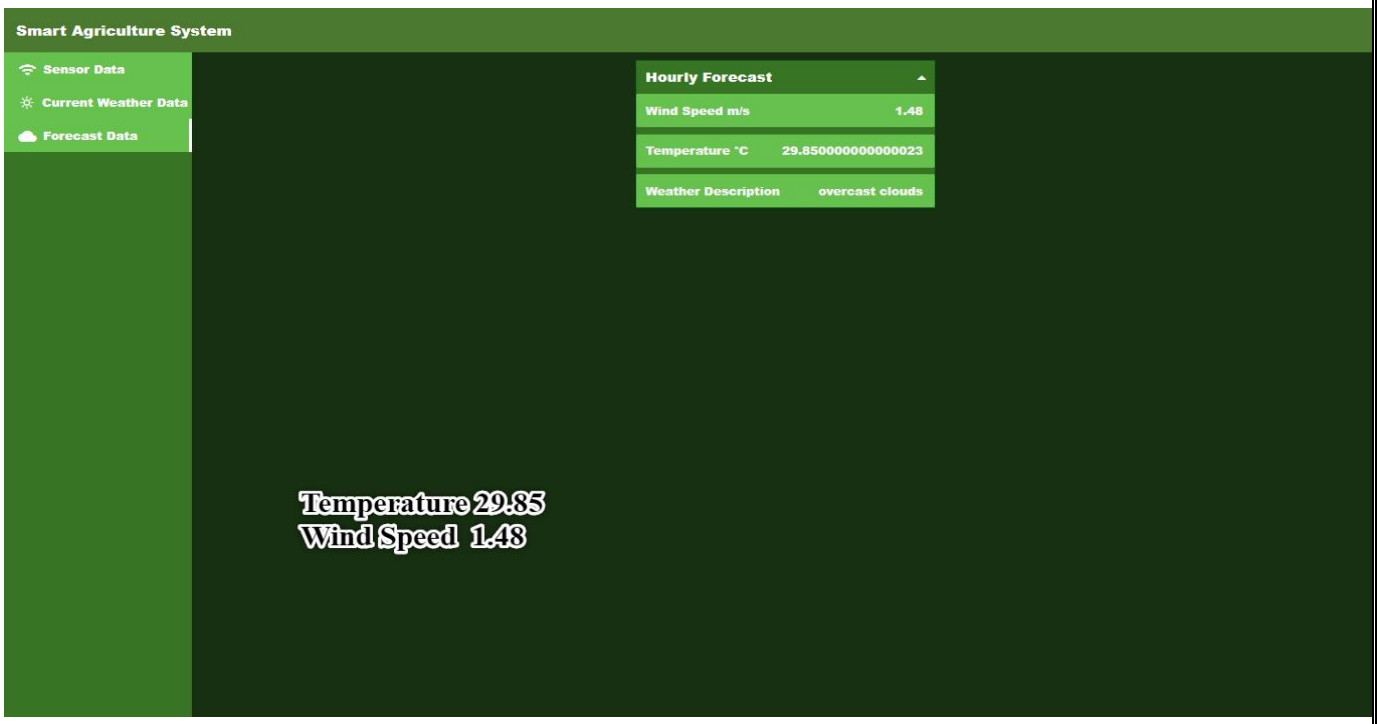
An experiment was conducted using the soil testing module to investigate some essential factors in our test field. Measurements were performed by the system and the variation of temperature; soil moisture level, Wind speed and humidity were observed over an approximately hourly forecast time period.

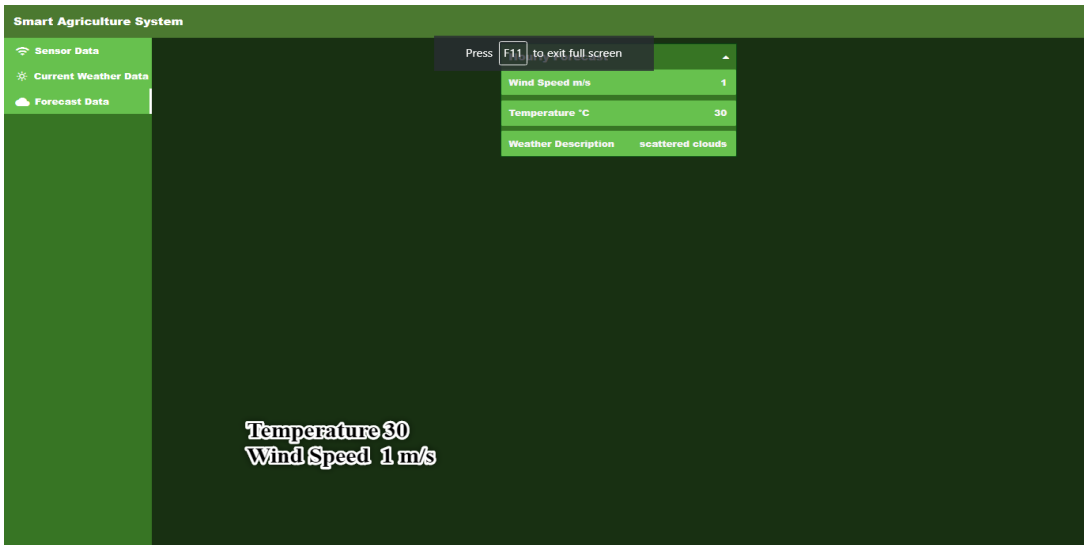
The following figure shows





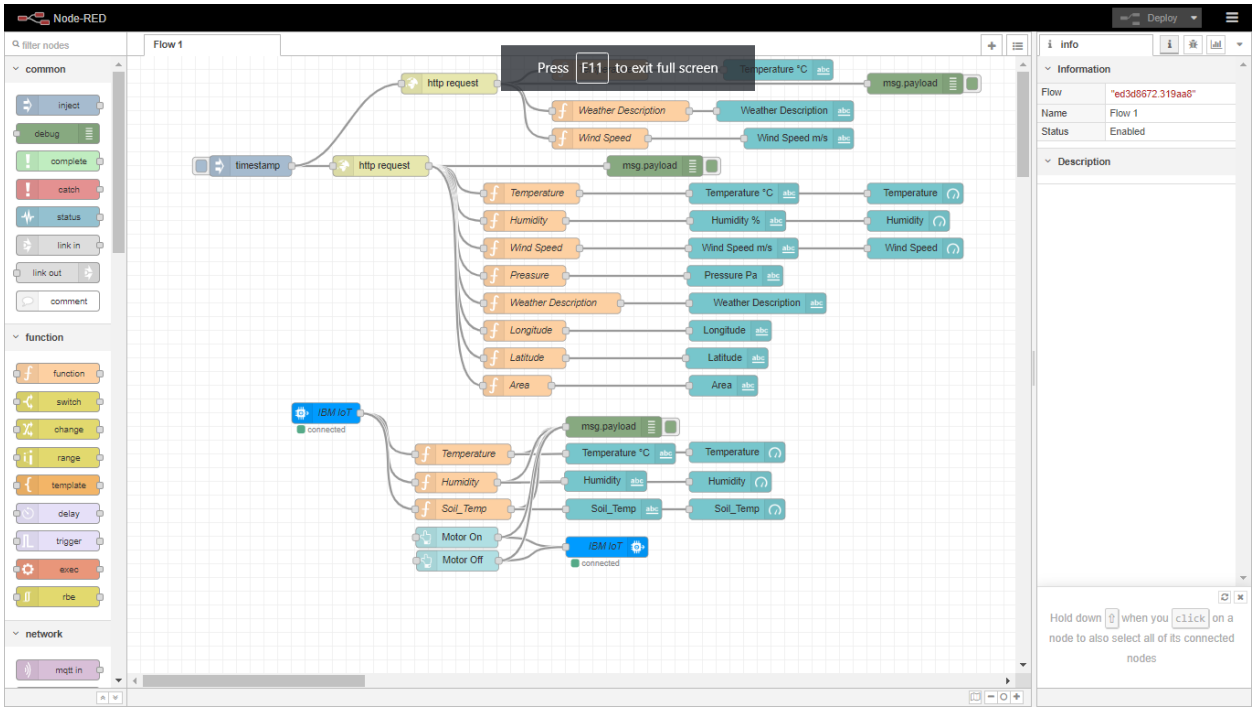

IBM Watson IoT Platform
007vigneshg@gmail.com
ID: t9qq76
IOTDevice
Add New Card
Settings
Object_Temp
25.0
%
Cookie Preferences


Smart Agriculture System
Sensor Data
Current Weather Data
Forecast Data
Hourly Forecast
Wind Speed m/s 1.48
Temperature °C 29.850000000000023
Weather Description overcast clouds
Temperature 29.85
Wind Speed 1.48

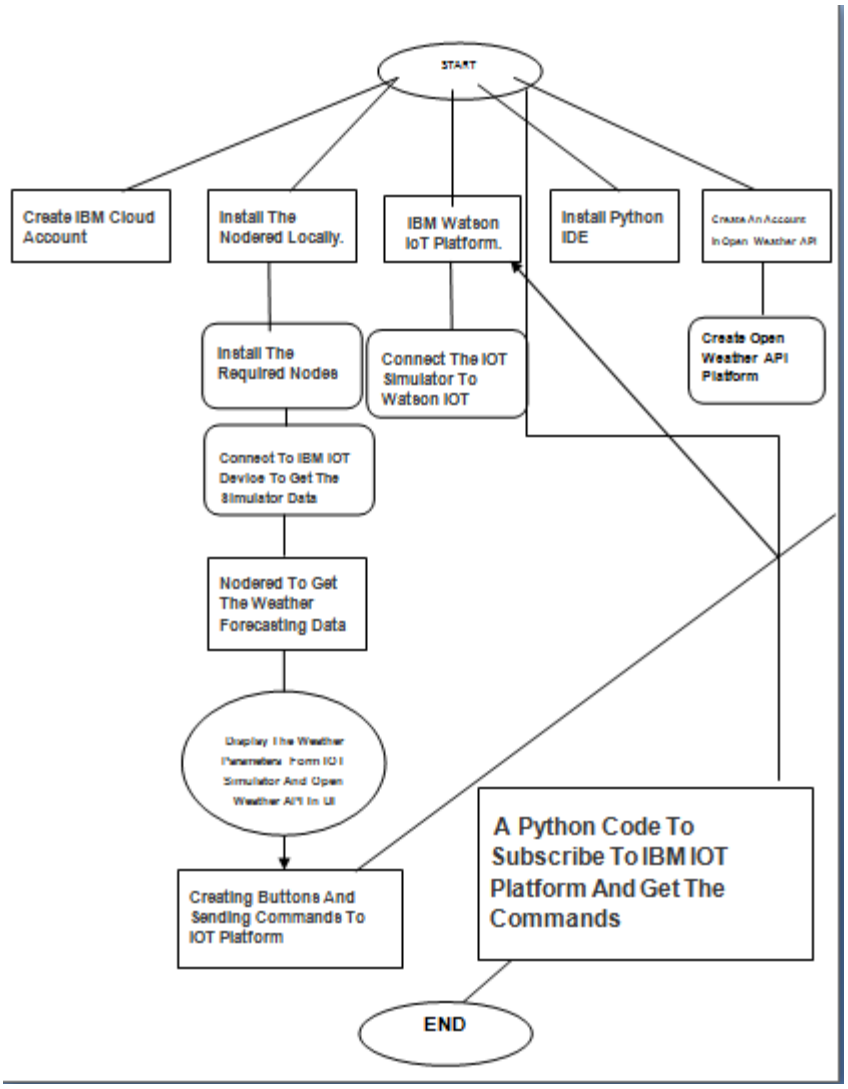


NODE RED FLOWS :

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.



FLOWCHART :


START
Create IBM Cloud Account
Install The Nodered Locally.
IBM Watson IoT Platform.
Install Python IDE
Create An Account In Open Weather API
Install The Required Nodes
Connect The IOT Simulator To Watson IOT
Create Open Weather API Platform
Connect To IBM IOT Device To Get The Simulator Data
Nodered To Get The Weather Forecasting Data
Display The Weather Parameters From IOT Simulator And Open Weather API In UI
Creating Buttons And Sending Commands To IOT Platform
A Python Code To Subscribe To IBM IOT Platform And Get The Commands
END

RESULT :

By testing the soil with our module, farmers and gardeners will have an accurate data on the condition of the soil i.e. the temperature, pressure, wind speed, moisture level and also the humidity of the surrounding.

ADVANTAGES & DISADVANTAGES :

ADVANTAGES :

- Accurency and Real-Time Data in weather data.
- Monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.
- Based on all the parameters he can water his crop by controlling the motors using the mobile application.
- It is cost effective method.
- Fast response.
- User friendly.

DISADVANTAGES :

- The smart agriculture needs availability of internet continuously.
- Rural part of most of the developing countries do not fulfill this requirement. Moreover internet connection is slower.
- The smart farming based require farmers to understand and learn the use of technology.

- This is major challange in adopting smart agriculture farming at large scale across the countries.

APPLICATIONS :

In This Project I used the following Applications,

- IBM Cloud Platform.
- IBM IOT Platform.
- Node Red.
- Python 3 IDLE.

These all are the required applications for this project.

CONCLUSION :

In a world where the Internet of Things (IOT) is accelerating adoption of automation and data gathering, an important industry such as agriculture can surely be benefited and our project of making agriculture in a smarter way will definitely help in the growth of this industry.

FUTURE SCOPE :

- For the fore coming days, In Smart Agriculture System I have an idea to monitor the water pressure level with flow level and above discussed details can be displayed.

BIBILOGRAPHY :

- IBM Cloud Platform.
- IBM IOT Platform.
- Node Red.
- Python 3 IDLE.
- Open Weather API.
- Node Js.

APPENDIX :

IBM Cloud :

The IBM® cloud platform combines platform as a service (PaaS) with infrastructure as a service (IaaS) to provide an integrated experience. The platform scales and supports both small development teams and organizations, and large enterprise businesses. Globally deployed across data centers around the world, the solution you build on IBM Cloud™ spins up fast and performs reliably in a tested and supported environment.

IBM Watson IOT Platform :

IBM Watson IoT Platform is a managed, cloud-hosted service designed to make it simple to derive value from your IoT devices. Watson IoT Platform and its additional add on services - Blockchain service and analytic service - enable organizations to capture and explore data for devices, equipment, and machines, and discover insights that can drive better decision-making.

Source code

Python Code to Retrieve Command From IBM IoT.

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device
#Provide your IBM Watson Device Credentials
organization = "t9qq76" #replace the ORG ID
deviceType = "MotorDevice"#replace the Device type wi
deviceId = "123456"#replace Device ID
authMethod = "token"
authToken = "123456789" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
print("Command received: %s" % cmd.data)
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 information: '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": deviceId, "auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....


except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()
```

Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times

deviceCli.connect()

while True:

deviceCli.commandCallback = myCommandCallback

Disconnect the device and application from the cloud

deviceCli.disconnect()

THANKING YOU!