A Project Report

On

Smart Agriculture System Based on IOT

by

CHALLA SAI SASHANK

(saisashank2000@gmail.com)

as an intern at

smartinternz.com@rsip2020

on

Internet of Things

INTRODUCTION

Overview

The objective of this report is to propose IoT based Smart Agriculture System which aids the farmer in controlling the motors in his field remotely by checking the weather as well as soil conditions of the field through a Web App. This Web App provides the weather forecast data of a certain location as well as the soil conditions of the field with the help of sensors placed in the field. This web app also includes a switch that enables the farmer to control the motor placed in his field.

This Project Report consists of sequential activities that are to be performed in order to obtain the User Interface for Smart Agriculture System.

Purpose

The scenario of decreasing water tables, drying up of rivers and tanks, unpredictable environment present an urgent need of proper utilization of water. To cope up with this a modern technique called Internet of Things is implemented, where a farmer can perform agriculture in a smart way. This IoT based Smart Agriculture System uses temperature and moisture sensors at suitable locations for monitoring of crops and controlling the motors in the field.

SCOPE

- Smart Agriculture System based on IoT can monitor soil moisture and climatic conditions to grow and yield a good crop.
- The farmer can also get the real-time weather forecasting data by using external platforms like Open Weather API.
- Farmer is provided a mobile app using which he can 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.
- Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.

Here we are using the Online IoT simulator for getting the Temperature, Humidity and Soil Moisture values.

Project Deliverables

The main objective of the project is to help the farmer to monitor the weather conditions, soil temperature, humidity and motor controls by the means of an external hardware device placed at the field. The farmer is provided with a mobile app which is configured to control the hardware device.

For the development of the web app, the below steps are to be followed:

1) Create a platform on the IBM Cloud to use the services in order to create the Web Application.
2) Create a device on the cloud to set the temperature and humidity by connecting the IoT Simulator to the Watson IoT platform.
3) Configure Node-Red to retrieve data from the IBM IoT platform and Open Weather API.
4) Create a Web App to preview the node-red code.
5) Configure your device to retrieve information from the web app and control your motors.
6) Check whether the motor buttons work.

The farmer can know the current weather conditions provided by the Open Weather API through the web app.

Project Team: CHALLA SAI SASHANK

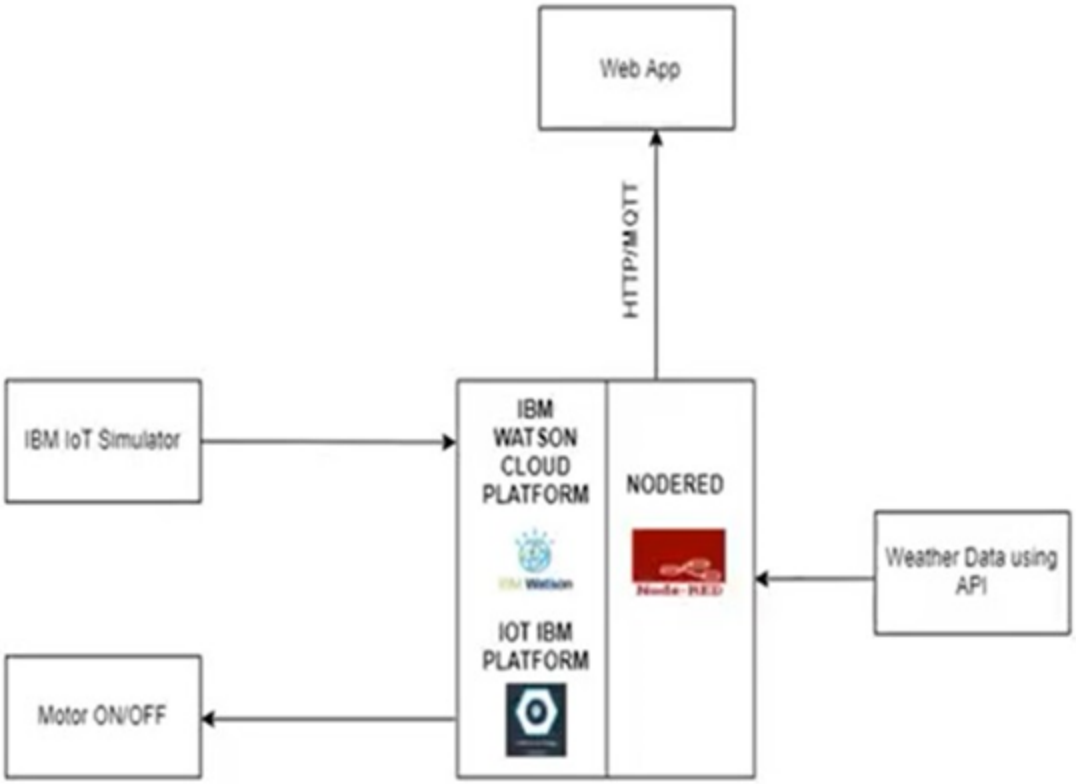
LITERATURE SURVEY

The major problems occurring in the existing agriculture system is decreasing water tables, drying up of rivers and tanks, unpredictable weather and an urgent need of effective utilization of water. These are the major problems that are devastatingly affecting the current agriculture system from yielding good results.

In recent times, to overcome these problems, a new technique called Internet of Things is implemented where a farmer can perform agriculture in a smart way. In this technique, various sensors applications are used to monitor the soil and weather conditions of a certain location. This data is updated in a web app that is used by the farmer to check the soil and weather conditions of the location and control his motors in the field remotely. This process is referred as IoT based Smart Agriculture System.

THEORITICAL ANALYSIS

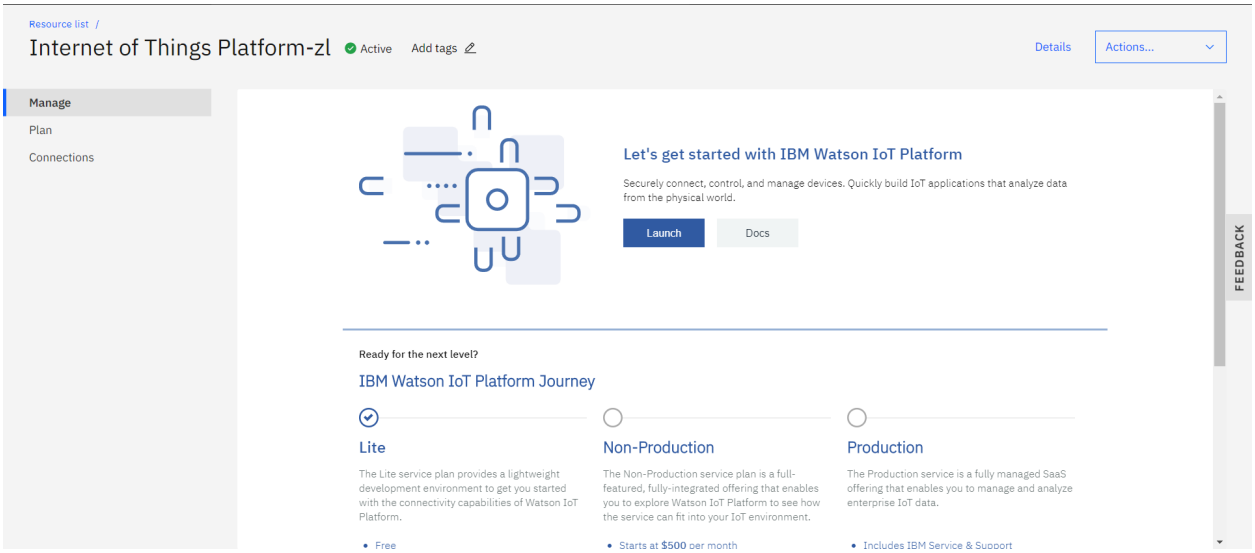
The Block Diagram for the IoT based Smart Agriculture System can be seen below:



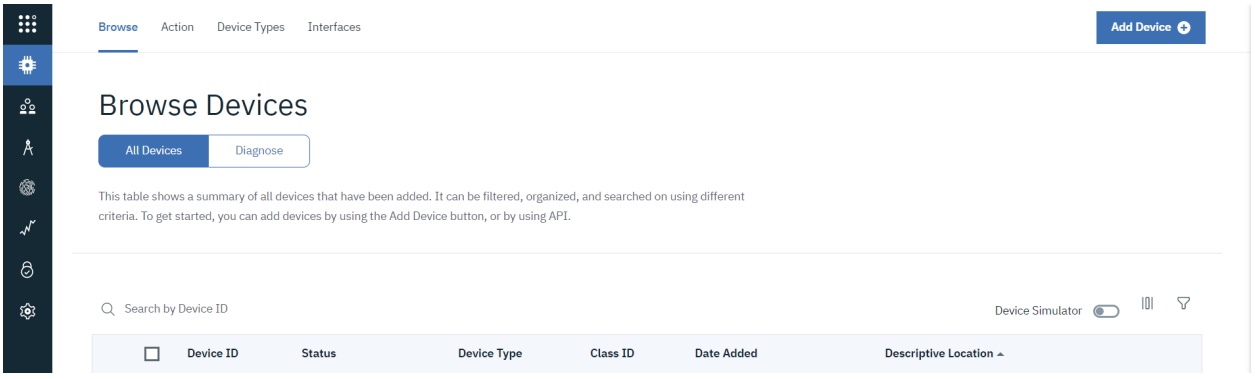
Designing Procedure

- Sign-up for IBM Academic Initiative Account through the link https://my15.digitalexperience.ibm.com/b73a5759-c6a6-4033-ab6b-d9d4f9a6d65b/dxsites/151914d1-03d2-48fe-97d9-d21166848e65/academic/home.

1) Sign-in to your IBM cloud account from the link https://cloud.ibm.com/login. There, go to Catalog and search for IoT in the search bar. Then select Internet of Things platform and subscribe for the desired plan and click create. Now, in the menu, go to Resource List --> Services --> Internet of Things Platform and then click Launch, as shown below:

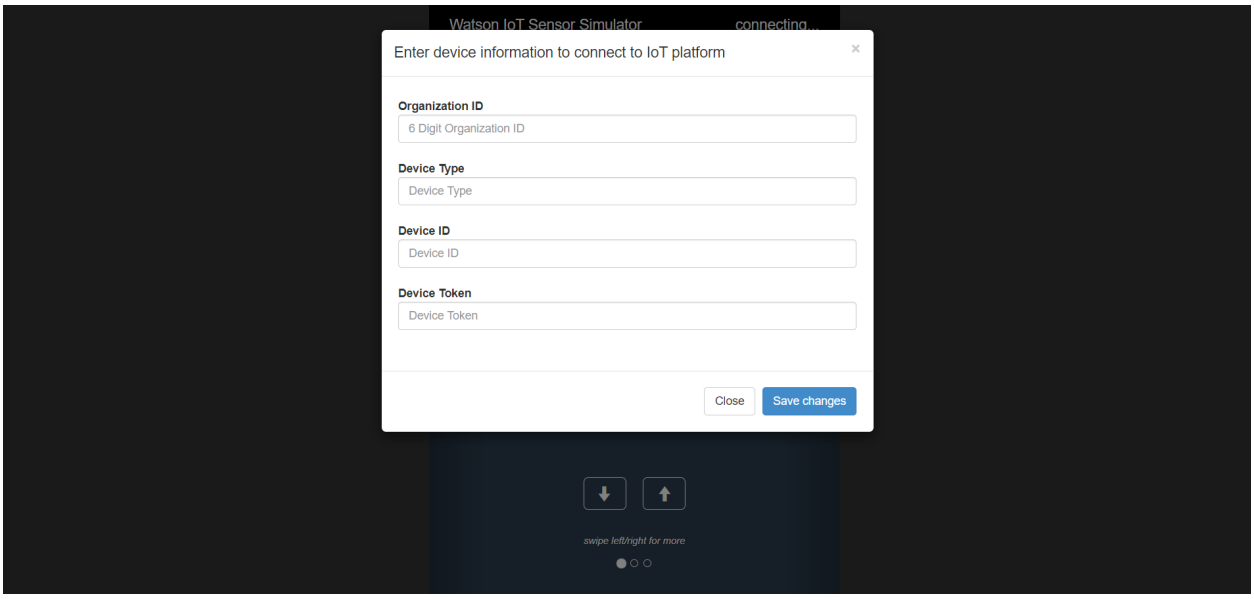


2) Now in the Watson IoT platform, click on the Add Device button at the top right corner, as shown below, to create a device to get the soil conditions from the sensor (simulator).



Make a note of the device credentials given during the device creation for further uses.

3) Now go to http://watson-iot-sensor-simulator.mybluemix.net/ to use the IoT sensor Simulator to generate sensor data to be uploaded to the cloud. It redirects to the screen similar to the following image. Here, give the device credentials that are saved earlier while creating the device.



Now, we can view the simulator data in the Watson Iot Platform by creating cards in the Boards section at the left.

4) Now, once the sensor data is received by the cloud, we use a special tool called Node-Red, a low-code programming tool for event-driven applications, to build a Web-App.

To install Node-Red on windows, go to https://nodered.org/docs/getting-started/windows.
(For further details on how to use Node-Red, visit https://nodered.org/docs/user-guide/)
Now, to build the web app for the Smart Agriculture System using Node-Red, a minimum of three flows would be required:

Flow-1: To take the weather forecast data from Open Weather API.
Flow-2: To take the sensor data from the IBM cloud.

Flow-3: To control the motor by passing commands to the cloud.

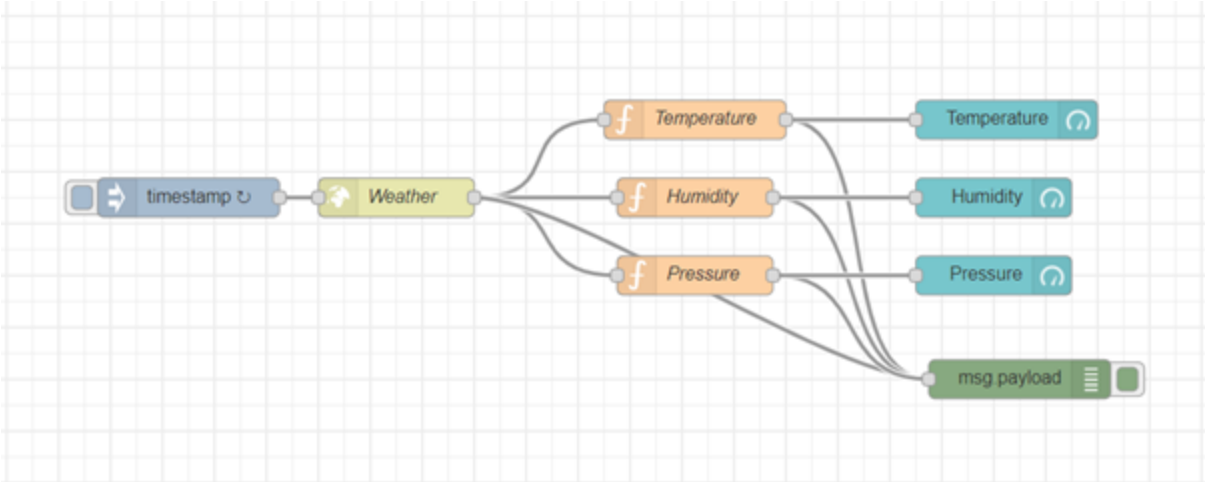
Flow-1:-

To obtain the weather forecast details from the Open Weather API, first create an account in the Open Weather through the link https://home.openweathermap.org/users/sign_up.

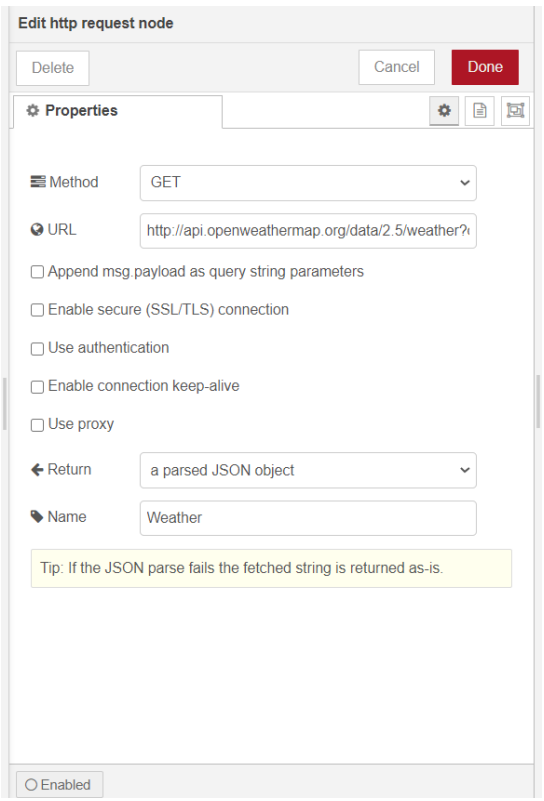
Then, go to the API Keys tab and generate a key and make a note of the key.

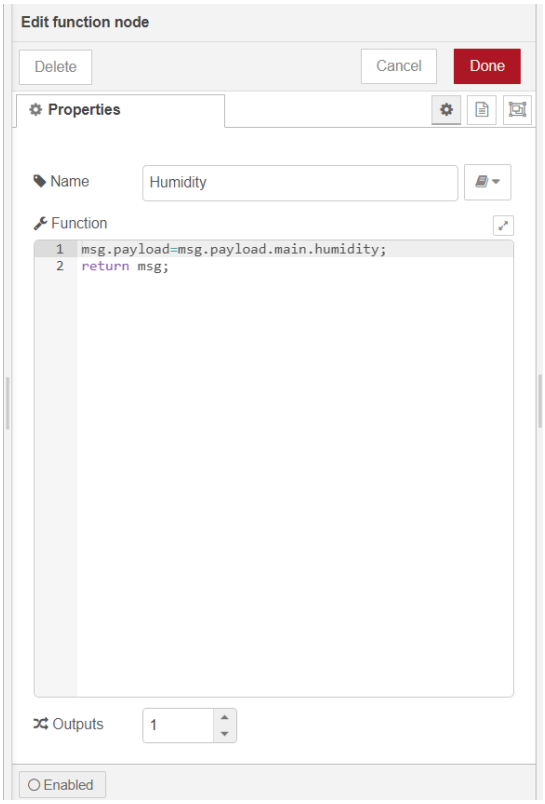
Now, in the API tab, select the API Docs under Current Weather Data and make a note of the API call accordingly.

Now, the following flow is implemented to get the weather forecast data from the Open Weather API:



In the above flow, the node properties of ‘http request’ and ‘function’ nodes are as follows:



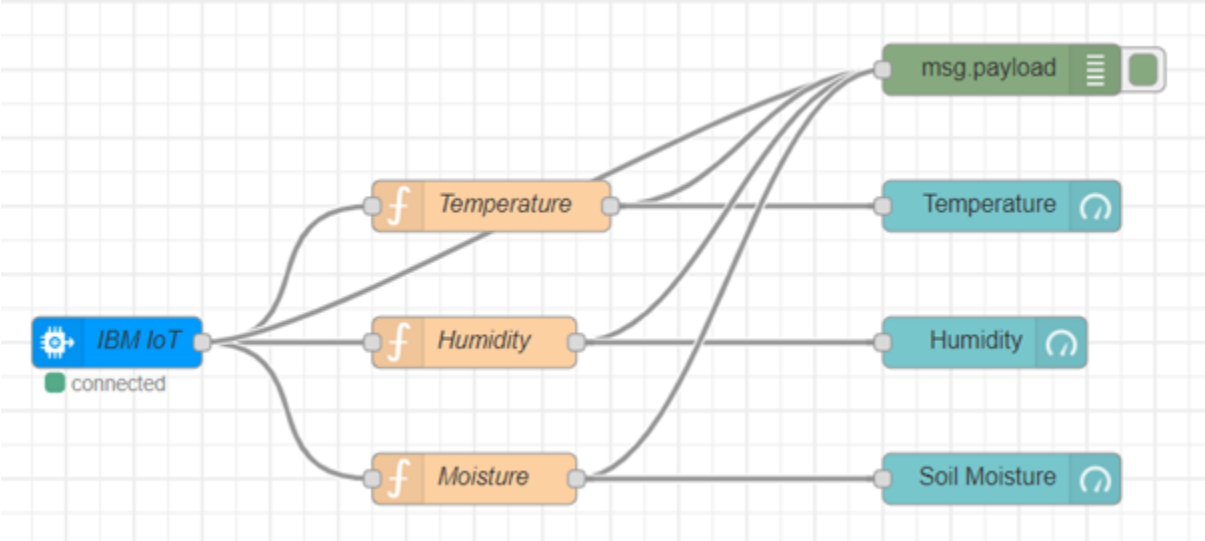


The URL property in the http request node is the API Call obtained from the Open Weather, replaced with appropriate ‘city name’ and ‘api key’.

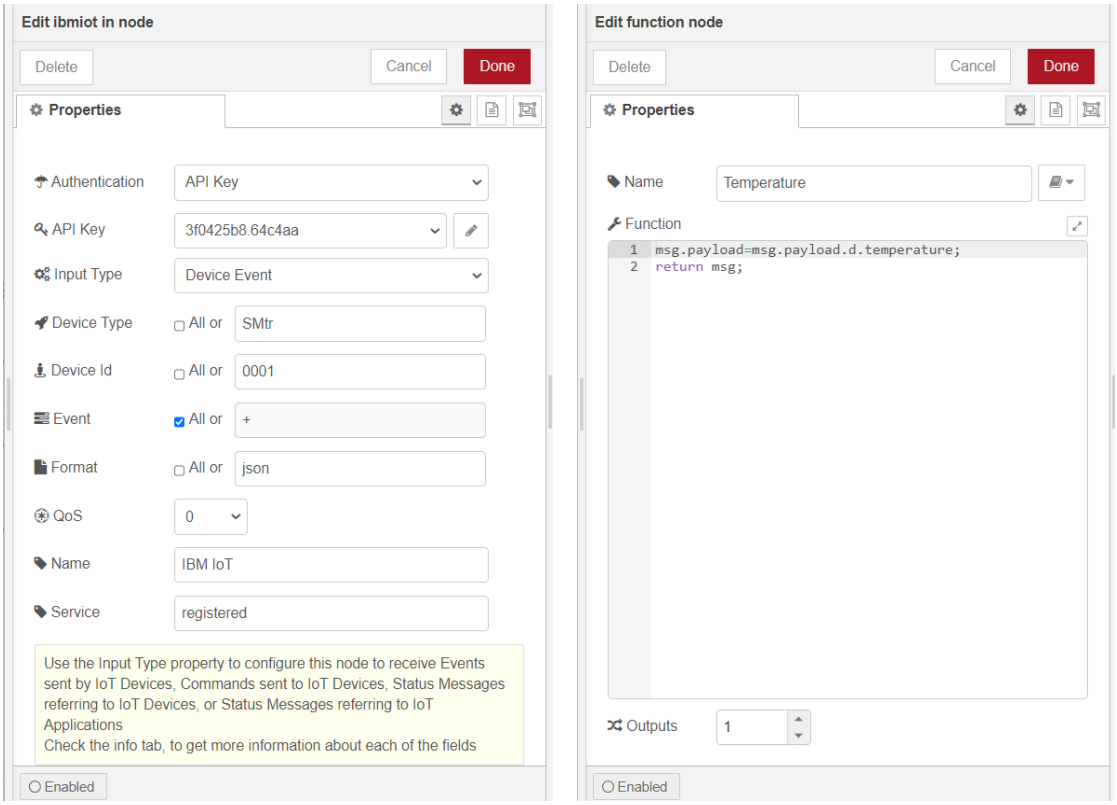
Flow-2:-

In order to get the sensor data from the cloud, IBM IoT nodes are to be installed externally in Node-Red using "node-red-contrib-scx-ibmiotapp" in the Manage Palette inside Node-Red.

Then, the following flow is implemented to get the sensor data from the cloud:

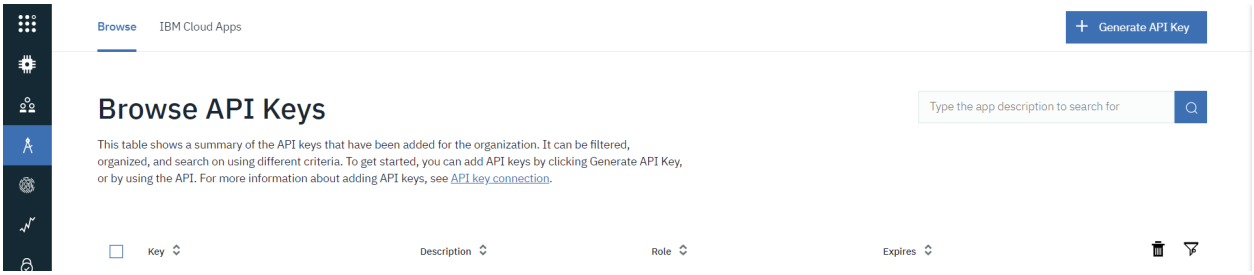


The configuration of the ‘IBM IoT in’ and ‘function’ nodes in the above flows are given as:



The API Key property in the ‘ibmiot in’ node refers to the API key of the cloud.

This API can be generated in the Apps tab in the Watson IoT platform by clicking on the ‘Generate API Key’ at the top right corner, as shown in the image:

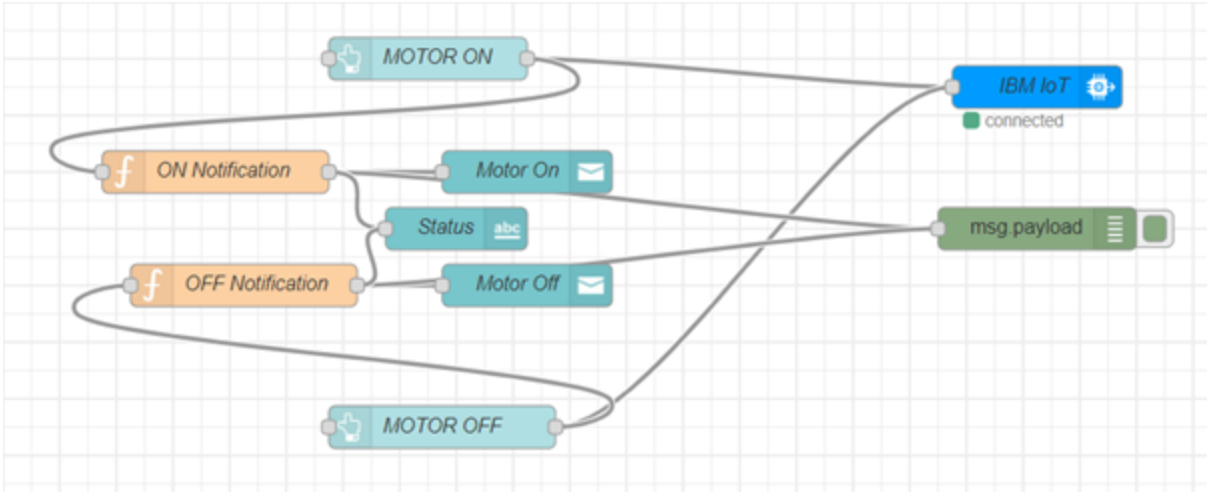


Make a note of the credentials while generating the API Key.

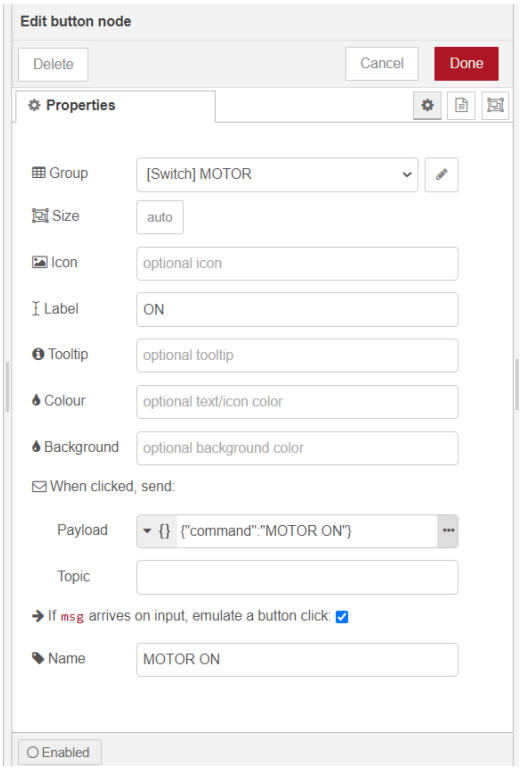
Flow-3:-

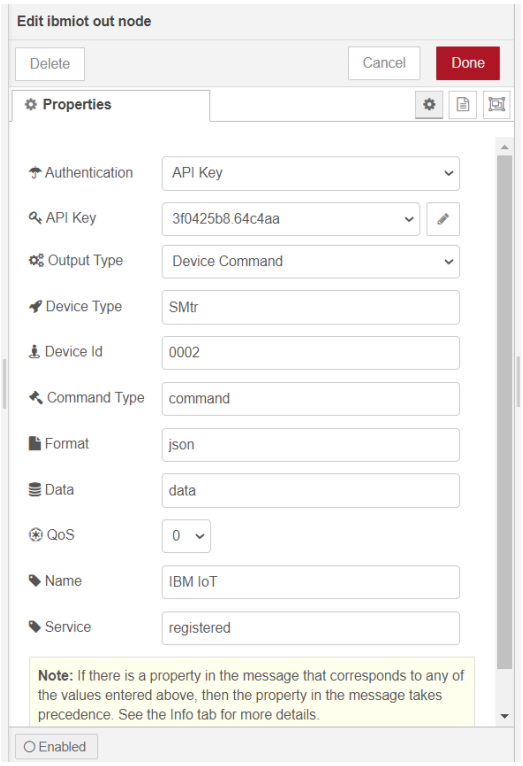
To control the motor by passing commands to the cloud, create another device in the Watson IoT Platform following the same procedure as the first one.

The following flow is to be implemented in order to control the motor by passing commands to the cloud:



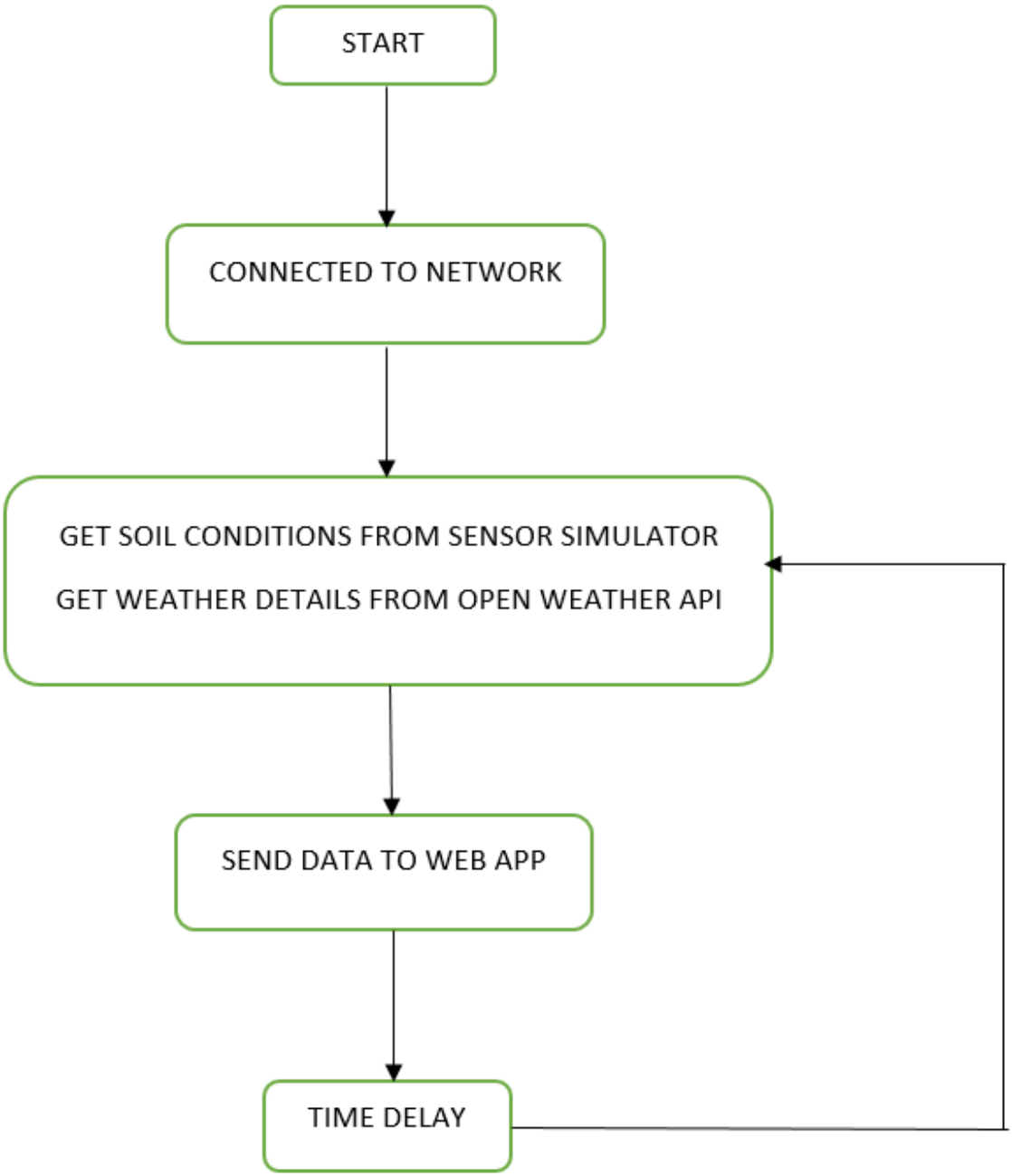
The configuration of ‘button’ and ‘IBM IoT out’ nodes are given as:





(Additional nodes can be used for the customization of the Web App)

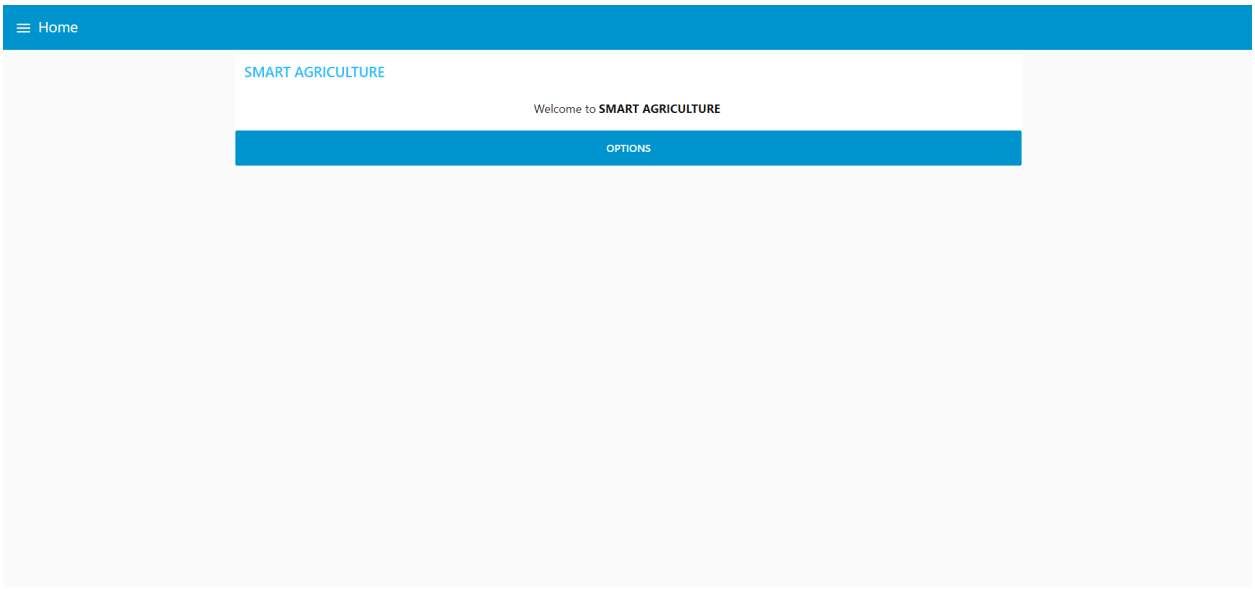
FLOW CHART DESCRIBING THE WORKING OF IOT BASED SMART AGRICULTURE SYSTEM

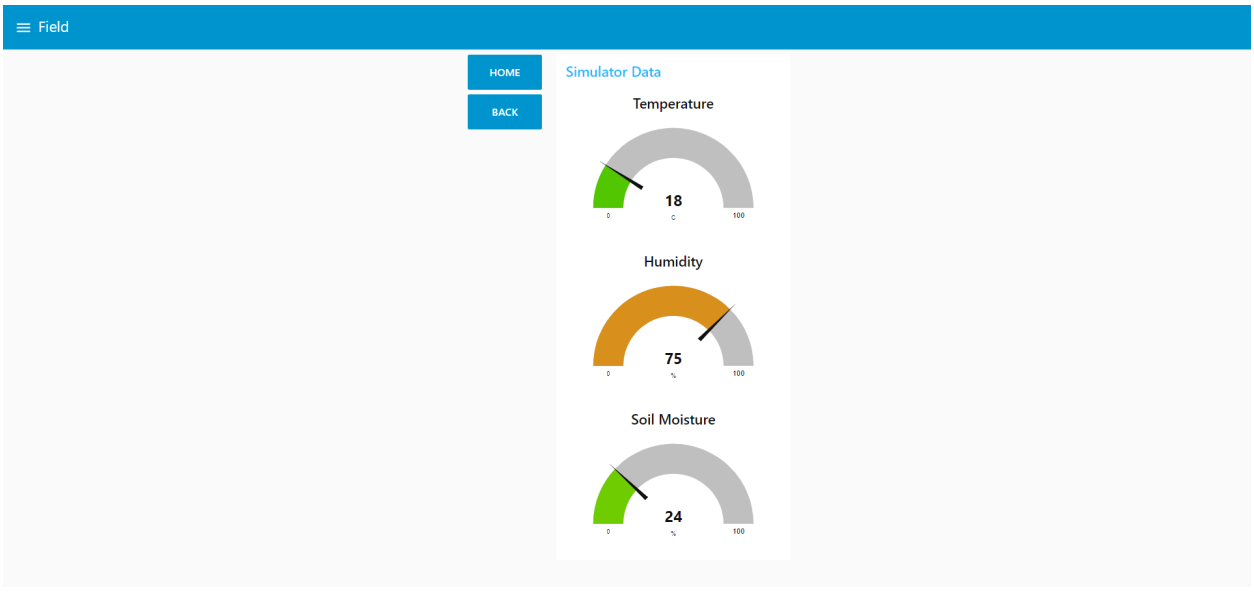


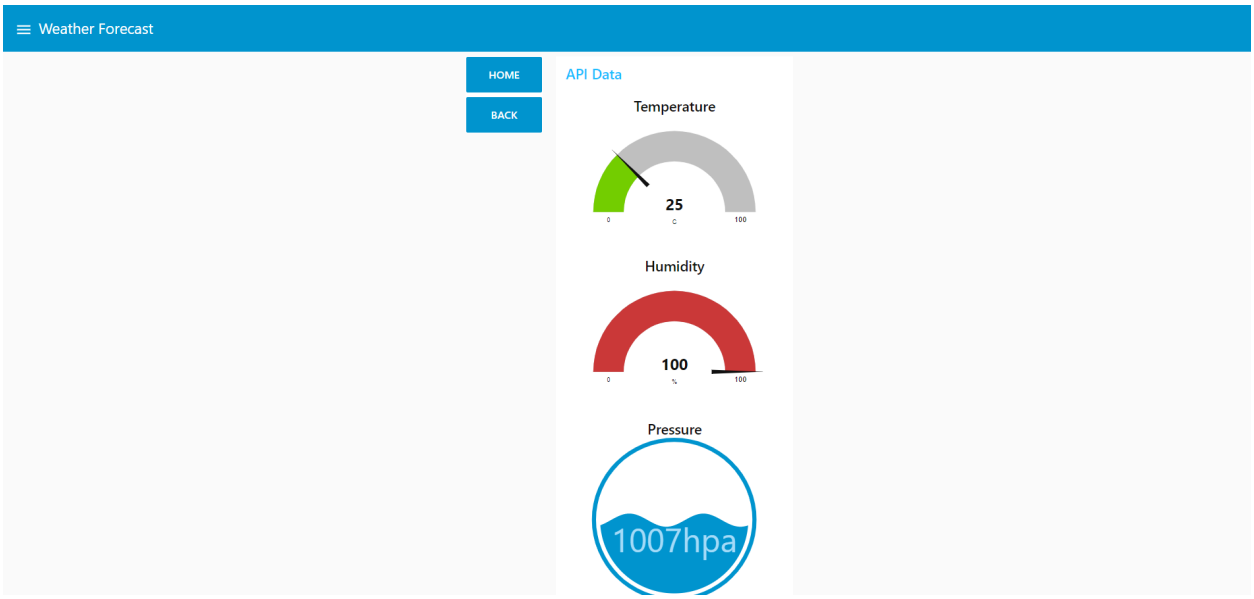
RESULT

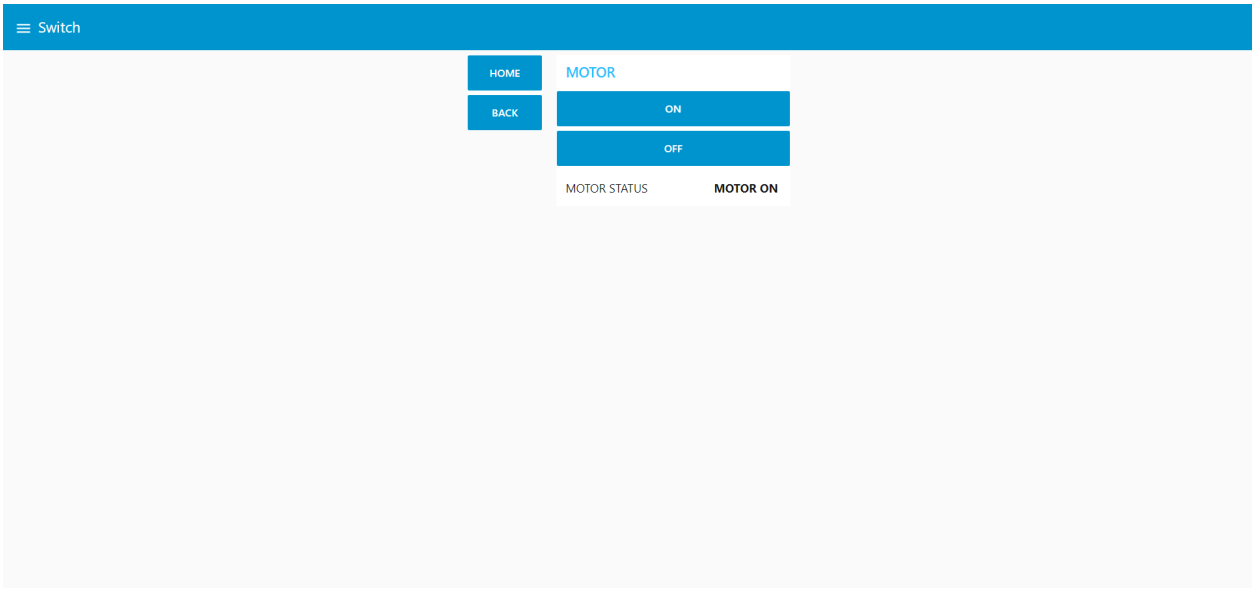
Following the above designing procedure results in a Web Application that is used by the farmer to perform agriculture in a smart way.

The Web Application generated by the above designing procedure is as follows:






Weather Forecast
HOME
BACK
API Data
Temperature
25
0
C
100
Humidity
100
%
Pressure
1007hpa


Switch
HOME
BACK
MOTOR
ON
OFF
MOTOR STATUS
MOTOR ON

Python code to retrieve commands from IBM Watson IoT Platform:-

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "fhfvz4" #replace the ORG ID
deviceType = "SMtr"#replace the Device type wi
deviceId = "0002"#replace Device ID
authMethod = "token"
authToken = "Abcdefgh" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
        print("Command received: %s" % cmd.data)
        if cmd.data['command']=='MOTOR ON':
                print("MOTOR ON IS RECEIVED")

        elif cmd.data['command']=='MOTOR OFF':
                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()
```

Output of the python code on receiving commands:-

CONCLUSION

IoT based SMART FARMING SYSTEM for Live Monitoring of Temperature and Soil Moisture has been proposed using IoT sensor simulator and Cloud Computing.The IoT based smart farming System being proposed via this report will assist farmers in increasing the agriculture yield and take efficient care of food production as the System will always provide helping hand to farmers for getting accurate live feed of environmental temperature and soil moisture.

FUTURE SCOPE

Future work would be focused more on increasing sensors on this system to fetch more data especially with regard to Pest Control and by also integrating GPS module in this system to enhance this Agriculture IoT Technology to full-fledged Agriculture Precision ready product.