IOT ENABLED SMART FARMING APPLICATION.

Sprint Delivery – 1

TEAMID : PNT2022TMID04037

1. Introduction

The main aim of this project is to help farmers automate their farms by providing them with a Web App through which they can monitor the parameters of the field like Temperature, soil moisture, humidity and etc and control the equipment like water motor and other devices remotely via internet without their actual presence in the field.

2. Problem Statement

Farmers are to be present at farm for its maintenance irrespective of the weather conditions. They have to ensure that the crops are well watered and the farm status is monitored by them physically. Farmer have to stay most of the time in field in order to get a good yield. In difficult times like in the presence of pandemic also they have to work hard in their fields risking their lives to provide food for the country.

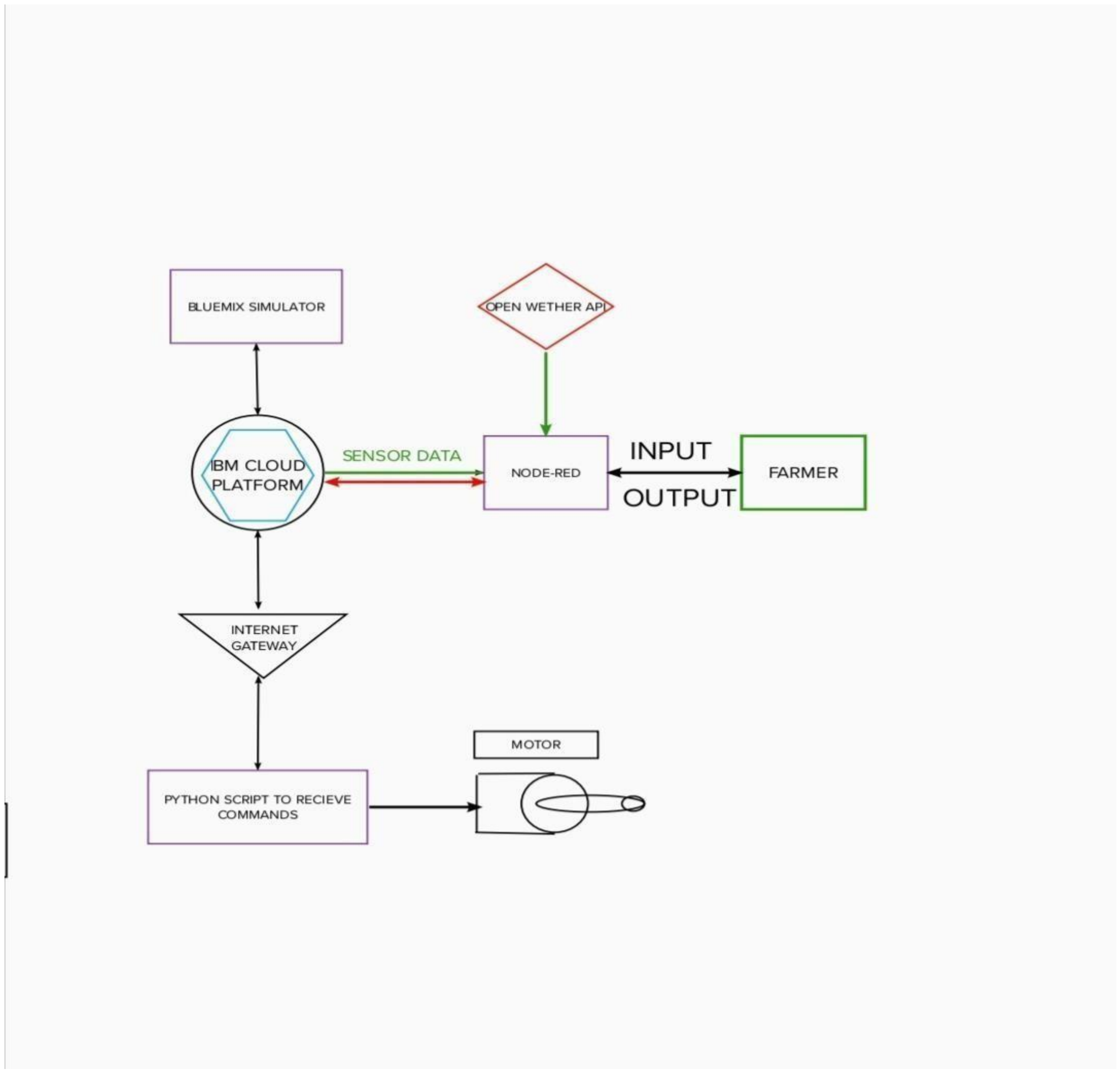
3. Proposed Solution

In order to improve the farmer’s working conditions and make them easier, we introduce IOT services to him in which we use cloud services and internet to enablefarmer to continue his work remotely via internet. He can monitor the field parameters and control the devices in farm.

4. Theoretical Analysis

Block Diagram

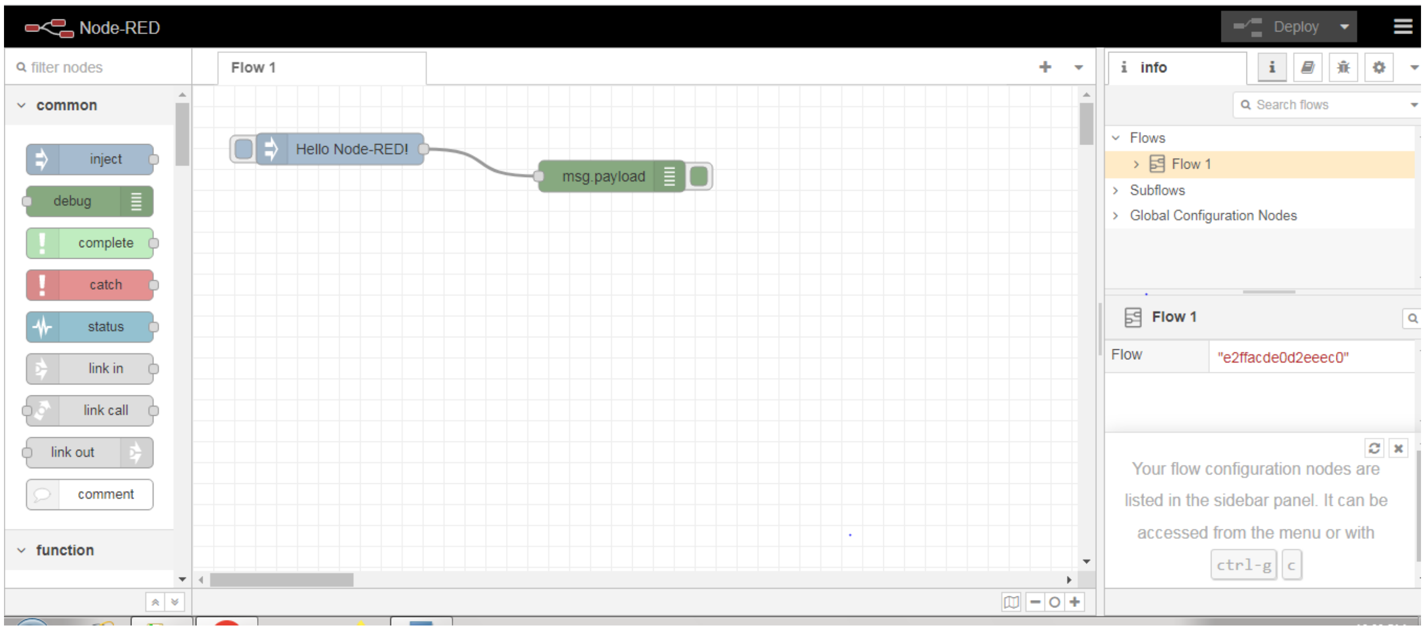
In order to implement the solution, the following approach as shown in the block diagram is used



Required Software Installation

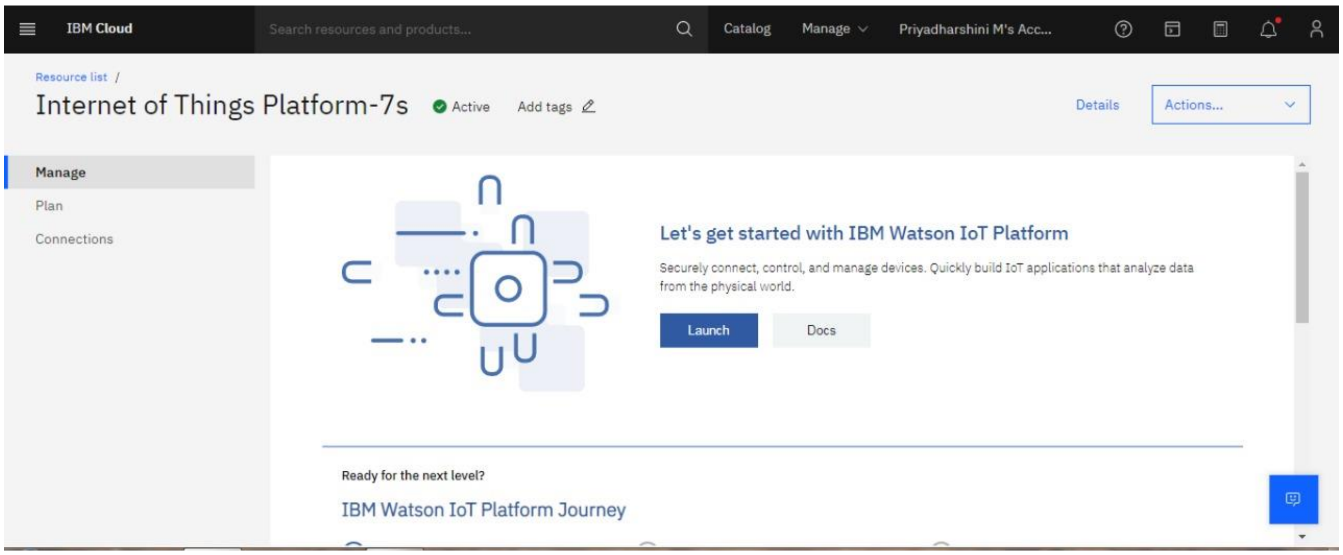
Node-Red

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things. Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions.



IBM Watson IoT Platform

A fully managed, cloud-hosted service with capabilities for device registration, connectivity, control, rapid visualization and data storage. IBM Watson IOT Platform is a managed, cloud-hosted service designed to make it simple to derive value from your IOT devices.


IBM Cloud
Search resources and products...
Catalog
Manage
Priyadharshini M's Acc...
Resource list /
Internet of Things Platform-7s
Active
Add tags
Details
Actions...
Manage
Plan
Connections
Let's get started with IBM Watson IoT Platform
Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.
Launch
Docs
Ready for the next level?
IBM Watson IoT Platform Journey

Steps to configure:

- Create an account in IBM cloud using your email ID
- Create IBM Watson Platform in services in your IBM cloud account
- Launch the IBM Watson IOT Platform
- Create a new device
- Give credentials like device type, device ID, Auth. Token
- Create API key and store API key and token elsewhere

Python IDE

Install Python3 compiler

Install any python IDE to execute python scripts, in my case I used Spyder to execute the code

Code:

```
import time import sys

import ibmiotf.application
import ibmiotf.device

import random



        #Provide your IBM Watson Device Credentials
        organization = "x0cl0i"
        deviceType = "nodemcu"
        deviceId = "sensor"
        authMethod = "use-token auth"
         authToken = "6GsCaVQ3-PfYy+J3ts"

        # Initialize GPIO
```

```
def myCommandCallback(cmd):     print("Command
received: %s" % cmd.data['command'])
status=cmd.data['command']      if status=="motoron":
print ("motor is on")     elif status == "motoroff":         print
("motor is off")     else :
       print ("please send proper command")


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:
       #Get Sensor Data from DHT11


       temp=random.randint(90,110)
       Humid=random.randint(60,100)
```

```
Mois=random.randint(20,120)

    data = { 'temp' : temp, 'Humid': Humid, 'Mois' :Mois}
    #print data        def
myOnPublishCallback():
print ("Published Temperature
= %s C" % temp, "Humidity = %s
%%" % Humid, "Moisture =%s
deg c" %Mois, "to IBM
Watson")

    success   =   deviceCli.publishEvent("IoTSensor",   "json",   data,   qos=0,
on_publish=myOnPublishCallback)  if not success:            print("Not connected
to IoTF")  time.sleep(10)

    deviceCli.commandCallback  =  myCommandCallback

# Disconnect the device and application from the cloud deviceCli.disconnect()
```

IOT Simulator

In our project in the place of sensors we are going to use IoT sensor simulator which give random readings to the connected cloud.

We need to give the credentials of the created device in IBM Watson IoT Platform to connect cloud to simulator.