Smart Agriculture system based on IoT

Contents

1. Introduction

IoT is bringing revolution to almost every aspect of our lives by changing how we do things. The use of Smart IoT devices is on the rise with all the industries heavily investing in IoT. The main aims of investing in IoT are to improve operations efficiency, improve product quality, and reduce the costs of production. The Agricultural industry is among the industries seeking to reap the benefits of IoT. The use of IoT in agriculture is commonly referred to as **Smart Farming or Smart Agriculture**. It uses various IoT sensors to send the farm's data, like humidity, temperature, soil moisture, etc. to the cloud which can be monitored and controlled from anywhere in the world.

According to the Food and Agricultural Organization (FAO), there will be eight billion people globally by the year 2025 and 9.6 billion by 2050. This means that food production must go up by 70 percent to feed that population. IoT is one of the technologies expected to assist in reaching that goal.

A. Overview

this project can monitor soil moisture and climatic conditions to grow and yield a better crop. The farmer can also get the real time weather forecasting data by using external platforms like Open Weather API. Farmer is provided a web application 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 application as per requirement without his physical presence at the crop. And based on the weather data obtained he can predict the weather and take required measures to avoid loss due to undesirable climatic conditions.

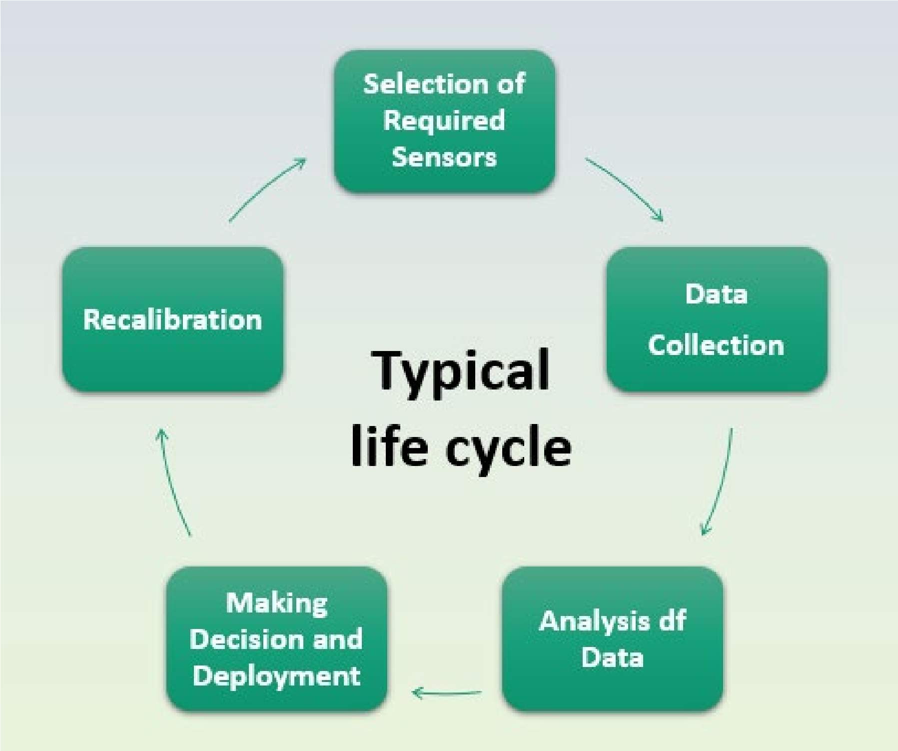
B. Purpose

Farming and agriculture form the basis of economies in several developing countries and are critical areas of development for governments. Since both these activities rely heavily on natural occurrences, environments, and conditions, predictive technologies become an essential tool for planning. The device, **Smart Agriculture system based on IoT**, can provide farmers with invaluable data about soil conditions, crop yields and rainfall, to help them predict and plan activities accordingly. Data collecting software can be synchronized with machines used in fields to completely automate the farming process, reducing the dependency on manual labour.

In this project , we will be using sensors to acquire data of soil moisture, temperature, humidity and also the weather forecast which will be helpful the farmer to get a better yield of crop.

2. Literature Survey

The Typical life cycle of an IoT Analytics based agricultural Use case



The life cycle of a typical IoT based use case, If we breakdown a sample use case of IoT analytics it consists of the following stages

- Selection of sensors : The selection of sensors differs from use case to use case, For example, the sensors required for livestock management is very different from the sensor requirements for a smart greenhouse use case.
- Data collection : Collection of data from deployed sensors and converting them to the required format.
- Making decisions and deployment : Data collected from the sensors can be used to draw insights and make automated business decisions. After getting through a proper data science life cycle models are deployed to the cloud or local servers as required.

- Recalibration of models : The results obtained from the previous processes are monitored and recalibrated based on the business KPI and deviation from the result.

A. Existing Problem

Farmers today are having to face some of the biggest challenges in the sector's history as the global population grows in size and so does the need for more and more food from fewer acres. Along with these challenges also comes the innate competition for land and water – this problem is also being enhanced by labour shortages, climate change, and increasing environmental regulation.

In the past, these challenges were solved by using mechanization as it proved to be the key to farming progress for food producers. Unfortunately, the idea of using mechanical equipment is not sufficient enough, therefore, professional growers are not only forced to embrace digital, but a whole host of innovative technologies to reduce costs and increase efficiency in this often volatile marketplace.

In 2017, the use of digital technology in agriculture resulted in 700 million dollars invested in AgTech companies, which is more than double the previous year. However, by 2015 it's estimated that the total required number of mouths to feed around the world will reach over nine and a half billion, so levels of investment will still need to rise for farming to meet the future demands for food.

So, what are our options? How can we successfully deal with this upcoming crisis? The answer for this is the Internet of Things (IoT) ,which is transforming traditional farming methods and ushering in a new era of food production.

B. Proposed Solution

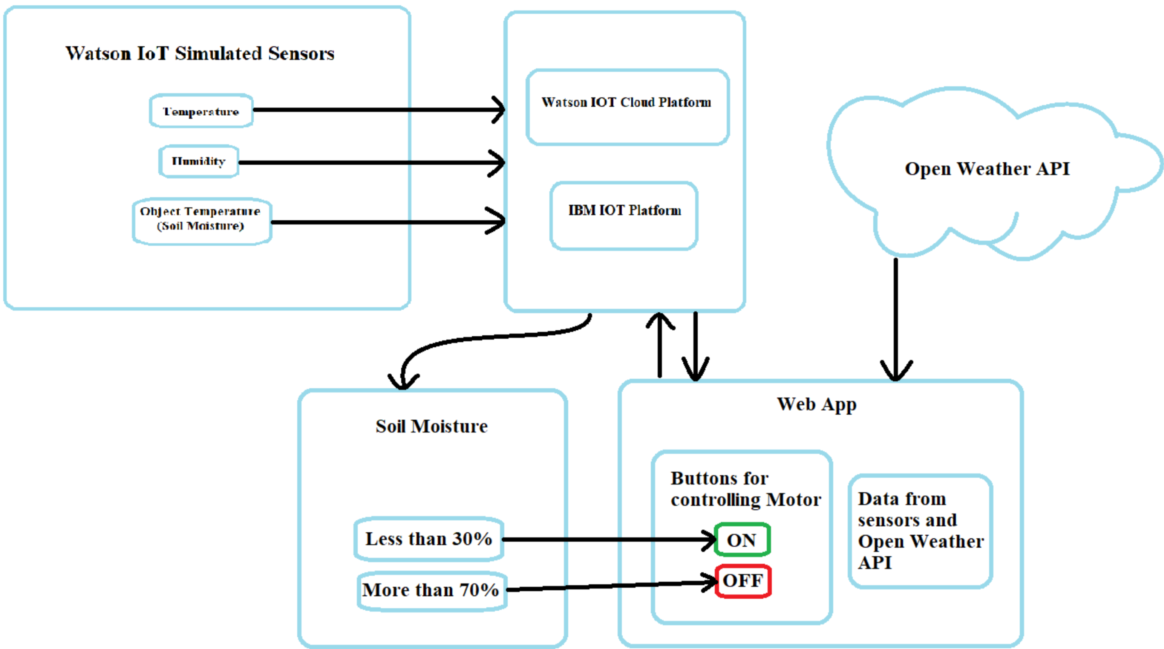
The purpose of this project is to create a web app / mobile app which can be used by farmers to get real time weather forecasting data by using external platform like Open Weather API. It has sensors to continuously monitor the environment of the crop like the soil moisture, temperature, humidity etc.

Based on the data received , farmers can control the mechanical equipments like the water pump, i.e. motor as required. If the soil moisture is not at level, the farmers can water the crop even if he is not present at the crop. The system is also programmed such that it automatically turns on the motor if moisture is below a certain level and turns off if it is above a certain level.

This project is demonstrated using IBM Watson IoT platform, Online IoT simulator and Nodered flow UI.

3. Theoretical analysis

A. Block diagram



B. Hardware/Software Designing

An ideal IoT device consists of various interfaces for making connectivity to other devices which can either be wired or wireless.
Any IoT based device consists of following components:

- I/O interface for Sensors.
- Interface for connecting to Internet.
- Interface for Memory and Storage.
- Interface for Audio/Video.

Since it is a simulated project we do not require any hardware, but if we buid a hardware prototype we may require

- Esp32 board
- Soil moisture sensor – to estimate the moisture level of the crop soil and give us time to time data so that we will be able to water accordingly
- DHT11 temperature and humidity sensor to obtain the values of temperature and humidity of surroundings of the farm

Hardware required

- A laptop with at least 4GB RAM
- A 2GB GPU

Software requirements for the simulated project

- IBM Cloud, Watson IOT platform-for simulated devices
- IoT simulator-simulated sensors(soil moisture, temperature, humidity)
- Nodered-for designing a web app for user interface
- Python IDE-programming the device
- Open Weather API-to get the weather data at a particular location

IBM cloud platform

IBM cloud computing is a set of cloud computing services for business offered by the information technology company IBM. IBM Cloud includes infrastructure as a service (IaaS), software as a service (SaaS) and platform as a service (PaaS) offered through public, private and hybrid cloud delivery models, in addition to the components that make up those clouds.

IBM offers three hardware platforms for cloud computing.[1] These platforms offer built-in support for virtualization. For virtualization IBM offers IBM Websphere application infrastructure that supports programming models and open standards for virtualization.

The management layer of the IBM cloud framework includes IBM Tivoli middleware. Management tools provide capabilities to regulate images with automated provisioning and de-provisioning, monitor operations and meter usage while tracking costs and allocating billing. The last layer of the framework provides integrated workload tools. Workloads for cloud computing are services or instances of code that can be executed to meet specific business needs. IBM offers tools for cloud based collaboration, development and test, application development, analytics, business-to-business integration, and security.

IBM Watson IoT Platform

IBM Watson IoT Platform can help you get a quick start on your next Internet of Things project. It is a fully managed, cloud-hosted service designed to make it simple to derive value from your Internet of Things devices. It provides capabilities such as device registration, connectivity, control, rapid visualization and storage of Internet of Things data.
Watson IoT Platform enabled several benefits for companies:

- Connect and register: Easily register and connect sensors and mobile devices. Remotely monitor the connectivity of devices.
- Visually assemble events: With IBM Bluemix, visually assemble events from the IoT into logic flows. Use Node-Red in IBM Bluemix for easy drag-and-drop flow assembly.
- Collect data: Collect and manage a time-series view of data and see what is happening on your devices with near-real time IoT data visualization.

- Manage devices and connections: Pay for what you use with purchase options available through IBM Bluemix. This highly scalable service allows payment through the IBM Marketplace or as a term subscription.

We can create devices in the IoT platform and these can be used for simulation. Each device is given its own credentials which can be be used for connecting it with simulated sensors and configuring with Nodered

Watson IOT Simulator

There are three simulated sensors:

- Object temperature.
- Temperature.
- Humidity.

To view the simulated sensor, go to https://watson-iot-sensor-simulator.mybluemix.net/. We have to give the required details to connect it with the device. The simulator connects automatically and starts publishing data. The simulator must remain connected to visualize the data.
We can use the simulator buttons to change the simulated sensor readings. Data is published periodically.

NodeRed

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

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range of nodes in the palette. Flows can be then deployed to the runtime in a single-click.

JavaScript functions can be created within the editor using a rich text editor.

A built-in library allows you to save useful functions, templates or flows for re-use.

The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.

With over 225,000 modules in Node's package repository, it is easy to extend the range of palette nodes to add new capabilities.

The flows created in Node-RED are stored using JSON which can be easily imported and exported for sharing with others.

An online flow library allows you to share your best flows with the world.

Node-RED is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.

To connect the IBM Watson IoT in Node-red need to install IBM IoT nodes. And install the dashboard nodes in order to create create a web app for UI

Open Weather API

OpenWeather Ltd is a British-based tech company that provides weather and satellite data worldwide. OpenWeather collects and processes raw data from a variety of sources, and gives its customers access to the data archive.

Its gives us the weather forecast data like get current weather, hourly forecast, daily forecast for 16 days, and 3-hourly forecast 5 days for your city. Historical weather data for 40 years back for any coordinate. Helpful stats, graphics, and this day in history charts are available for your reference. Interactive maps show precipitation, clouds, pressure, wind around your location.

It gives us a unique link based on our city and an API key which can be used to get the weather forecast details.

Nodered is configured to get this data and it can be visualised in the UI by using required nodes.

Python IDLE

used to write python code which gets the commands from the UI about the status of the motor, whether the motor is on or off

4. Experimental Investigations

yearly there are a lot of incidents of crop damage due to climatic conditions and also due to lack of time to time observation of the crop health. Farmers today are having to face some of the biggest challenges in the sector’s history as the global population grows in size and so does the need for more and more food from fewer acres. Along with these challenges also comes the innate competition for land and water – this problem is also being enhanced by labour shortages, climate change, and increasing environmental regulation.

The only solution is to increase the crop yield with minimal resources. We need to monitor the crop health, its environmental conditions etc. time to time we have to observe the weather forecast and have to predict the climatic conditions so that we can be able to take required measures in adverse situations to avoid or atleast minimize the damage.

We need to maintain the soil moisture levels for better health of the crop. When we have the details of soil moisture at every instant we can decide when to water the farm and when not to. Also the motor is given commands to br turned on and off it the moisture is out of certain range automatically.

Each kind of crop requires different conditions, keeping this in mind, say for example, a crop needs to be having a moisture level between 30% to 65% for it to maintain good health, then the motor commands can be configured such that if moisture is below 30% motor is automatically turned on and if it is above 65% motor is automatically turned off.

We also have to be able to control the motor for watering the field even if we are not physically present at the field if required. This can be done by using command to on and off the motor which can be triggered by button clicks in the web app UI . these commands are sent to the iot device which in turn controls the motor.

Watson IoT device

Two devices are created in the Watson IoT platform. One is connected to the IOT simulated sensor , which has simulated sensors of temperature, humidity, object temperature. When the sensors are connected to the device, we get the data from the sensors periodically so that the situations can be analysed. These data are used by the Watson iot platform , we can create cards which can be gauges, charts etc. for better visualization of this data.

Each device created is given its own credentials which can be used to configure them with Nodered, IoT simulated sensors and any other interfaces if required.

Simulated IoT sensors

It has 3 sensors to measure the various parameters like humidity, temperature and object temperature(for soil moisture). Each parameter has the provision to be increased and decreased so that their values are passed to all the configured interfaces for visualization.

The sensor asks for device credentials in order to be connected to a device in IBM IoT platform.

Open Weather API

When an account is created, we will be given a unique API key in Open Weather API. After selecting the place of which we need the weather forcast details, a unique link is generated using the place name and the API key given to us. Uaing this link we can get the weather forcast details. These details are displayed on the UI by configuring in Nodered by HTTP request.

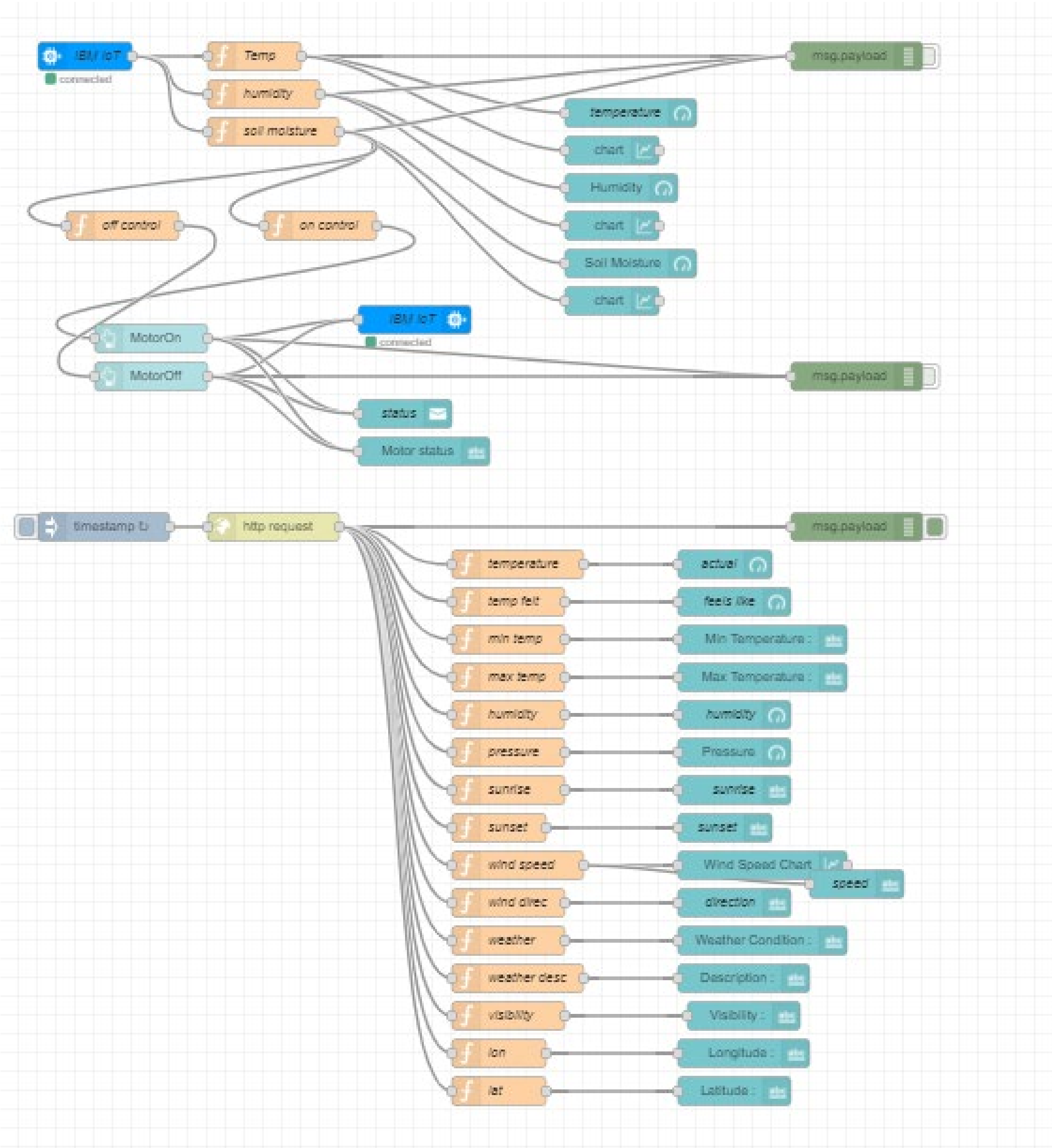
Nodered

Required nodes are installed in Nodered like the IBM IoT, Dashboard, etc apart from which that are already present for configuring it with Watson IoT platform devices, Open Weather API so that the data, commands can be published and received for visualization in the created User Interface i.e. the web app created using the Node red.

- **Ibmiot in** node is used to get data from the devices in IBM Watson IOT platform, here we connect it to the device which receives data from the simulated IOT sensors.
- The temperature, humidity and object temperature data received is as a single object by the input node, for visualizing them in the web app , we use the **function** node to separate them.
- The **debug** node is used to observe the received data in the debug window within the nodered.
- The nodes **gauge, charts, text,** etc from dashboard section are used to display the received data in a more graphical manner.A guage is used to display the value of a particular parameter at that instant, a chart is used to displau the values of the parameter for a certain duration of time
- **ibmiot out** node is used to publish data or commands to the device in the Watson IoT platform. Any data received by this node is published in the Watson IoT platform device to which it is configured to.
- **Button** nodes are used to send commands preconfigured when they are clicked, we can use the button nodes to send on off commands to the device in Watson IOT platform i.e. the one which corresponds to the motor. Not onlu on being clicked , a button can be triggered when it receives a message from a connected node. So we can use it such that it is automatically triggered when soil moisture is higher or lower than a required threshold range.
- **Notification** node is used to send notifications popups in the web app i.e. UI, here we can use it to indicate whenever the status of motor is changed.
- **Inject** node Injects a message into a flow either manually or at regular intervals. The message payload can be a variety of types, including strings, JavaScript objects or the current time.here we can use it to trigger the **http request** node to get the data from a given url.
- **http request** Sends HTTP requests and returns the response. Here it can be used to get data from the Open Weather API in order to display required weather forecast data in th UI in a more readable manner. The data from the link is again separated using the **function** nodes and given to the **guage** or **charts** for better visualization.

In this manner the Nodered is configured with the simulated sensors, devices in Watson IoT platform, Open Weather API and the data and controls are deployed in a web app for better user interface.

Nodered Flow for the UI



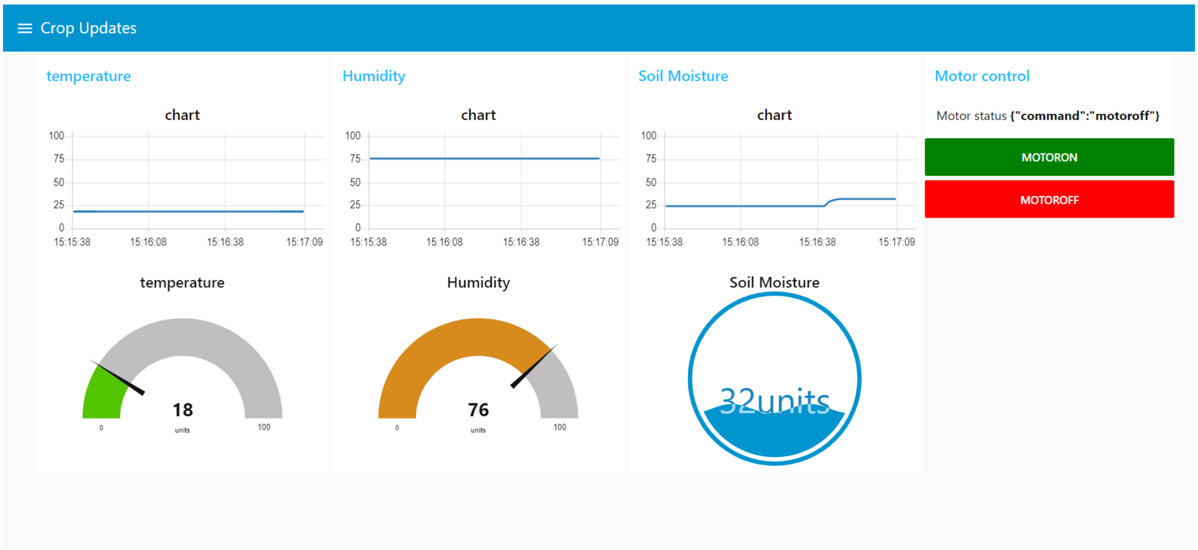
Web App for UI

Using the Nodered we can create a UI which gives us all the required data and provision to provide commands in a more readable and easily understandable manner. So that it becomes easy foe the farmer to access the data from anywhere and he will be able to take decisions accordingly.

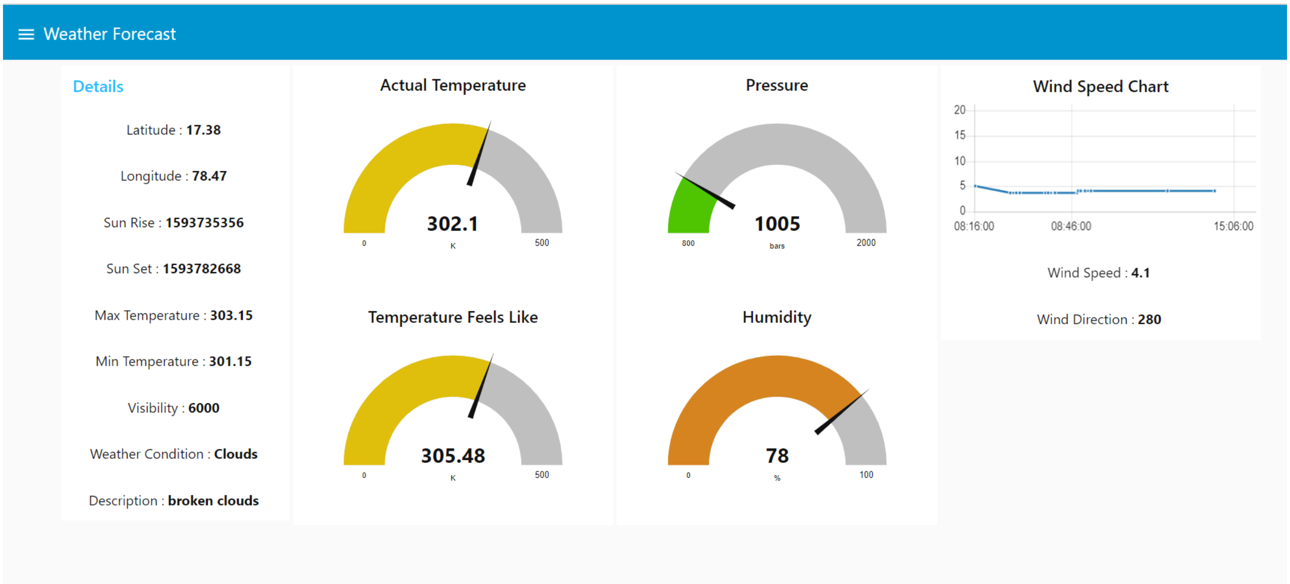
The UI dashboard of the web app can be accessed using the link http://127.0.0.1:1880/ui/

The created Web App with the above mentioned flow will be having 2 tabs in the dashboard.

- One will be having the data from the simulated sensors and the buttons to control the motor. It has a Motor status indicator which displays the recent command passed to the motor, And every time the status of the motor is changed a pop up notification is received which tells us the same.

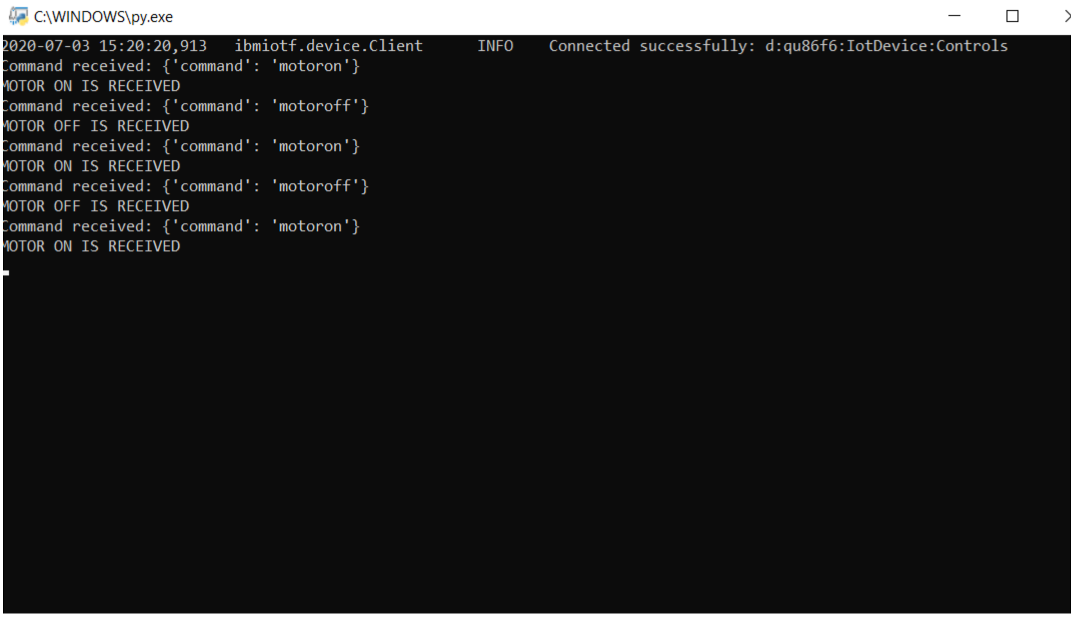


- Second tab will have the data retrieved using HTTP request from the Open Weather API. Details like the position, sunrise and sunset timings, wind speed and direction, temperature, humidity, pressure, visibility, weather condition etc are obtained and displayed time to time in this tab so that the farmer can monitor and take necessary decisions and precautions in undesired conditions.

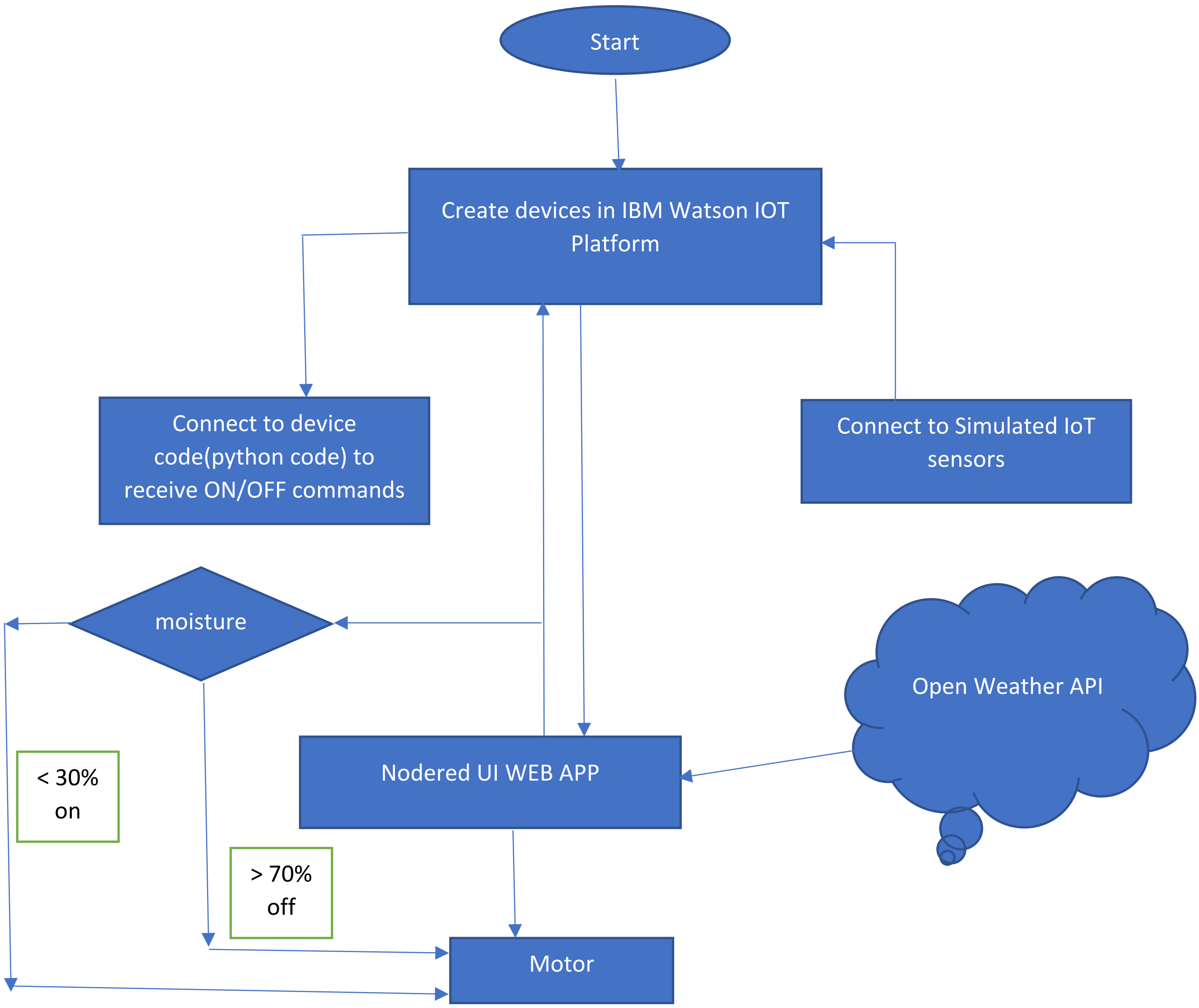


Python code

A python code is written to subscribe to IBM Watson IOT platform and get the commands of the motor being on or off by using the buttons in the UI.


C:\WINDOWS\py.exe
2020-07-03 15:20:20,913 ibmiotf.device.Client INFO Connected successfully: d:qu86f6:IotDevice:Controls
Command received: {'command': 'motoron'}
MOTOR ON IS RECEIVED
Command received: {'command': 'motoroff'}
MOTOR OFF IS RECEIVED
Command received: {'command': 'motoron'}
MOTOR ON IS RECEIVED
Command received: {'command': 'motoroff'}
MOTOR OFF IS RECEIVED
Command received: {'command': 'motoron'}
MOTOR ON IS RECEIVED

5. Flowchart


Start
Create devices in IBM Watson IOT Platform
Connect to device code(python code) to receive ON/OFF commands
Connect to Simulated IoT sensors
moisture
Open Weather API
Nodered UI WEB APP
< 30% on
> 70% off
Motor

6. Result

A Smart Agriculture system based on IoT is being done which can be used to 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 web 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 web 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.

7. Advantages and Disadvantages

A. Advantages

There are many reasons to implement a smart agriculture solution into commercial and local farming. In a world where the internet of things is accelerating adoption of data gathering and automation, an important industry such as agriculture can surely benefit.

Monitoring and collecting data for soil moisture, air temperature, air humidity and sunlight intensity across multiple fields will improve efficiency of water usage and crop yield of large and local farms. As the world population increases, farming and food production will have to increase with it. Low cost sensors, data insights and IoT

Platforms will enable this increase in efficiency and production. Here are some benefits of implementing a smart agriculture solution:

- **Increased Production** : Optimized crop treatment such as accurate planting, watering, pesticide application and harvesting directly affects production rates.
- **Water Conservation** : Weather predictions and soil moisture sensors allow for water use only when and where needed.
- **Real-Time Data and Production Insight** : Farmers can visualize production levels, soil moisture, sunlight intensity and more in real time and remotely to accelerate decision making process.
- **Lowered Operation Costs** : Automating processes can reduce resource consumption, human error and overall cost.
- **Increased Quality of Production** : Analysing production quality and results in correlation to treatment can teach farmers to adjust processes to increase quality of the product.

- **Accurate Farm and Field Evaluation** : Accurately tracking production rates by field over time allows for detailed predicting of future crop yield and value of a farm.
- **Reduced Environmental Footprint** : All conservation efforts such as water usage and increased production per land unit directly affect the environmental footprint positively.
- **Remote Monitoring** : Local and commercial farmers can monitor multiple fields in multiple locations around the globe from an internet connection. Decisions can be made in real-time and from anywhere.

The Internet of Things has truly enhanced many industries by providing data collection, real-time insight and process automation through low cost sensors and IoT platform implementation. As seen in the above benefits, the farming and agriculture industry overall can really benefit from implementation of such an IoT solution or platform. Not only is a smart agriculture solution the innovative farming method of today, it is the key solution to the growing concern of the global population’s food consumption and environmental footprint.

B. **Disadvantages**

- The smart agriculture needs availability of internet continuously. Rural part of most of the developing countries do not fulfil this requirement. Moreover internet connection is slower.
- The smart farming based equipment require farmers to understand and learn the use of technology. This is major challenge in adopting smart agriculture farming at large scale across the countries.
- There could be wrong Analysis of Weather Conditions. In the case of agriculture, most of the process is dependent on weather conditions. It is a natural phenomenon which in spite of the updated technology can become unpredictable. There is no force which can change or control the weather conditions such as rain, sunlight, drought etc. Even when the smart systems are in place, the importance of natural occurrences cannot be changed.
- Since technology involves a lot of machines, there are chances where the data might get wrong at times. If there are faulty data processing equipment or sensors then it will lead to the situation where the wrong decisions are taken. This will lead to the overuse of resources like fertilizers or water. It might even lead to the over-application of fertilizers or pesticides on crops. This excessive use of chemicals might destroy the crop and reduce the richness of the land.
- The Cost Involved in Smart Agriculture, while the use of smart technology in agriculture is impressive, it does incur a lot of costs if the devices are to be altered according to the level of the farmers i.e. to make as simple as possible such that anyone can understand to operate it, it will involve a lot of money to transform these types of equipment.

  This, on the other hand, means that the process will cost huge money. Since the farming industry does not see higher profits, huge investments in this space are unlikely. Even after the altering of machines, there are chances where the farmers might tend to operate the machines wrongly causing it to damage or send it to repair. Since these pieces of equipment are already costly, repairing it or replacing it will again cost a lot of money. The cost of maintenance becomes high whether there is a repair or not.

8. Applications

- **Real-time crop monitoring** : Smart sensors enable farmers to get the real-time data of their farms to monitor the quality of their products and optimize resource management.
- **IoT analytics in agriculture** : Data from smart sensors can be analysed for predictive analysis and automated decision making. It helps farmers with smart automated decision making instead of classic rule-based systems or manual procedures. Predictive analysis and machine learning can help farmers to cope up with extreme weather conditions like floods, drought, etc. The main advantage of IoT analytics is the inclusion of soil quality, temperature, humidity as a parameter.
- **Smart greenhouse solutions** : Greenhouses are conventionally used to maintain the necessary atmosphere for plants and this process demands continuous monitoring and manual intervention. Industrial IoT solutions can be incorporated to automate this process. Data collected by smart sensors can be analysed automatically and deep learning-based systems can be deployed to make decisions and creating certain climate automatically. With these smart sensors, climate variables and water consumption can be monitored via SMS or Wi-Fi-based systems.
- **Smart waste management :** One of the main differences between agriculture and other industry is the value of industrial waste. Biological wastes produced from farming can be reused for making fertilizers, IoT solutions can help to manage the process remotely in a smart way. Smart sensors can be used to measure the presence of toxic chemicals in wastes and manage the proper agricultural lifecycle.

9. Conclusion

Smart Agriculture system based on IoT for Live Monitoring of Temperature, humidity and Soil Moisture has been proposed using IBM cloud platform, Watson IoT platform, IoT simulated sensors and weather forecast details from Open Weather API. The System has high efficiency and accuracy in fetching the live data of temperature, humidity and soil moisture. 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, humidity and soil moisture with accurate results and also forecast from Open Weather API is helpful. It has the provision to control the motor via a web app so that the physical presence of the farmer is not needed for controlling the motor. Also the motor is automatically turned on and off if soil moisture is out of desired range.

10. 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.

its ability to innovate the landscape of current farming methods is absolutely ground breaking. IoT sensors capable of providing farmers with information about crop yields, rainfall, pest infestation, and soil nutrition are invaluable to production and offer precise data which can be used to improve farming techniques over time. New hardware, like the corn-tending **Rowbot**, is making strides by pairing data-collecting software with robotics to fertilize the corn, apply seed cover-crops, and collect information in order to both maximize yields and minimize waste.

Another direction in which smart farming is headed involves intensively controlled indoor growing methods. The **OpenAG** Initiative at MIT Media Lab uses "personal food computers" (small indoor farming environments that monitor/administrate specific growing environments) and an open source platform to collect and share data. The collected data is termed a "climate recipe" which can be downloaded to other personal food computers and used to reproduce climate variables such as carbon dioxide, air temperature, humidity, dissolved oxygen, potential hydrogen, electrical conductivity, and root-zone temperature. This allows users very precise control to document, share, or recreate a specific environment for growing and removes the element of poor weather conditions and human error. It could also potentially allow farmers to induce drought or other abnormal conditions producing desirable traits in specific crops that wouldn't typically occur in nature.

With a future of efficient, data-driven, highly-precise farming methods, it is definitely safe to call this type of farming smart. We can expect IoT will forever change the way we grow food.

11. Bibliography

[1] https://www.researchgate.net/publication/334131097_A_Project_Report_On_IoT_based_SMART_FARMING_SYSTEM_CERTIFICATE_OF_APPROVAL_Countersigned_by
[2] https://www.educba.com/iot-in-agriculture/
[3] https://www.iotdesignpro.com/articles/smart-farming-iot-applications-in-agriculture
[4] https://www.c2m.net/blog/10-benefits-of-a-smart-agriculture-solution
[5] https://www.ibm.com/cloud/get-started
[6] https://nodered.org/
[7] https://www.iotone.com/software/ibm-watson-iot-platform/s62
[8] https://www.1001artificialplants.com/2019/06/06/the-advantages-and-disadvantages-of-smart-agriculture-using-iot/
[9] https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-Smart-Agriculture-Farming.html

12. Appendix

Source Code

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

#Provide your IBM Watson Device Credentials
organization = "qu86f6" #replace the ORG ID
deviceType = "IotDevice"#replace the Device type wi
deviceId = "Controls"#replace Device ID
authMethod = "token"
authToken = "controls" #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()
```