# SmartFarmer - IoT Enabled Smart Farming Application

## A PROJECT REPORT

***Submitted by***


| | |
|---|---|
| **ABINAYA S** | **732119205002** |
| **AGALYA C** | **732119205003** |
| **BOOPALAN G** | **732119205011** |
| **POOJITHA S** | **732119205039** |


**TEAM ID : PNT2022TMID32243**


**NANDHA COLLEGE OF TECHNOLOGY, ERODE**

vaikalmedu, perundhurai - 638 052

**TABLE OF CONTENTS :**

# 1 . INTRODUCTION

## 1.1 : Project Overview

Agriculture is the root to country's economic development. In recent times, huge scientific advancement has been implemented in various agricultural fields for the betterment of the future. Despite of various researches, proper assessment and productivity couldn't be reached. The Agriculture Parameters are utilizing an IOT Technology and system availability that draw in these objects to assemble and deal information. The IOT enables things selected recognized or potentially forced remotely crosswise over completed the process of existing configuration, manufacture open gateways for all the additional obvious merge of the substantial earth into PC based frameworks, in addition to acknowledging overhauled capacity, precision and cash interconnected favored stance. Precisely when IOT is extended with sensors and actuators, the improvement modifies into an occasion of the all the extra wide category of electronic physical structures, which in like manner incorporates headways, for instance, clever grids, splendid homes, canny moving and smart urban groups.

## 1.2 : Purpose

We have tried to focus on different scientific applications which could be put together in agricultural field for better accuracy with better productivity using less manpower. Moreover, we include a method for monitoring the agricultural fields from any remote location and assess the basic condition of the field.

This is the project from the motivation of the farmers working in the farmlands are solely dependent on the rains and bore wells for irrigation of their land. In recent times, the farmers have been using irrigation technique through the manual control in which the farmers irrigate the land at regular intervals by turning the water-pump ON/OFF when required.

# 2 . LITERATURE SURVEY

## 2.1 : Existing Probelm

Horticulture is the foundation of our Nation. In long time past days agriculturists used to figure the ripeness of soil and influenced presumptions to develop which to kind of product. They didn't think about the dampness, level of water and especially climate condition which horrible an agriculturist more. They utilize pesticides in view of a few suspicions which made lead a genuine impact to the yield if the supposition isn't right. The profitability relies upon the last phase of the harvest on which agriculturist depends.

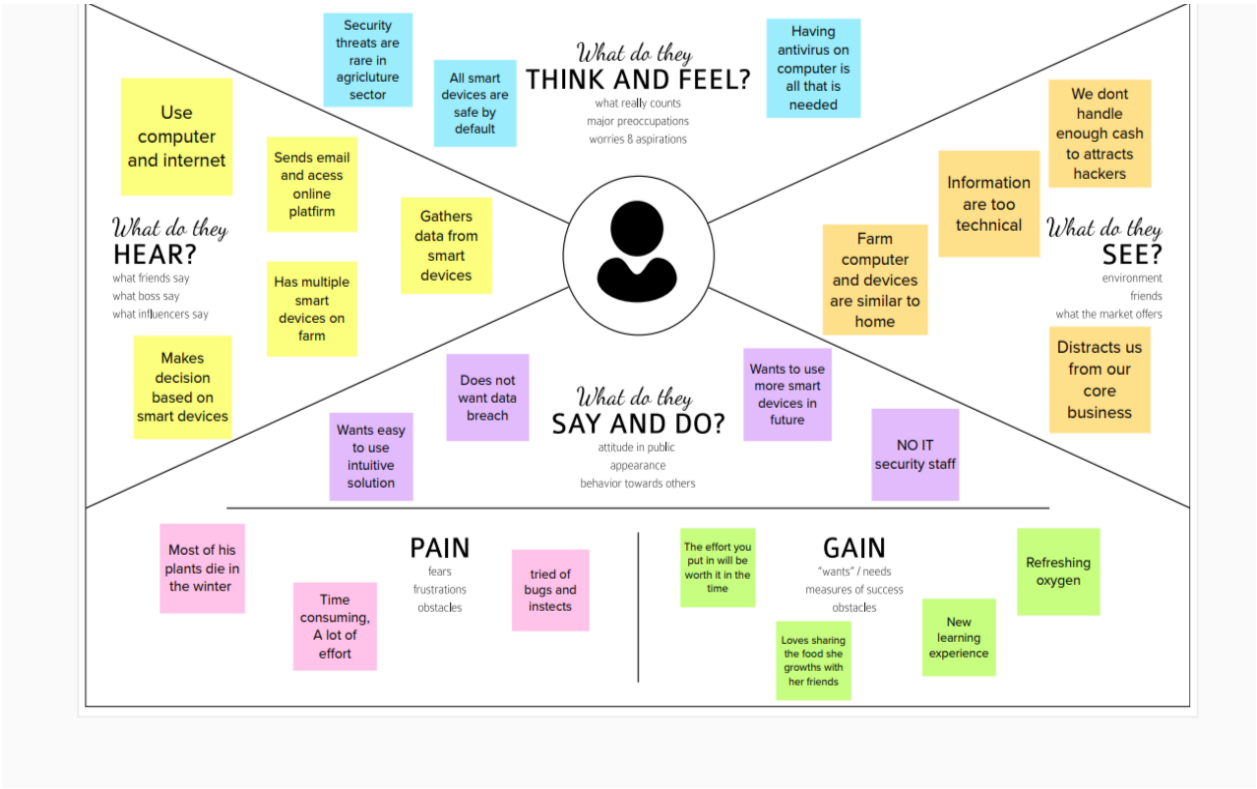
## 2.2 : Problem Statement

- IoT-based agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, temperature, and humidity using some sensors.
- Farmers can monitor all the sensor parameters by using a web or mobile application even if the farmer is not near his field. Watering the crop is one of the important tasks for the farmers.
- They can make the decision whether to water the crop or postpone it by

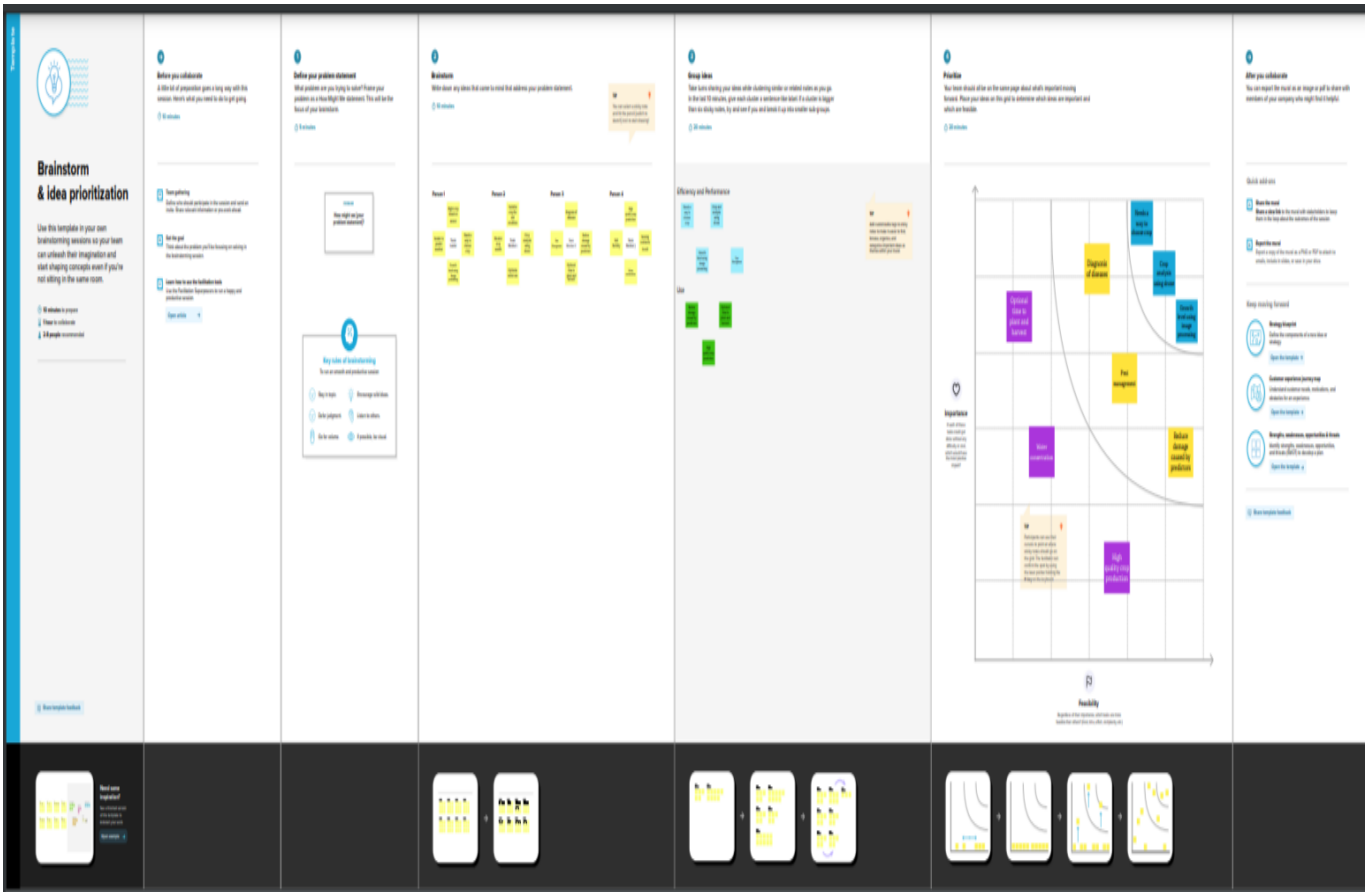
monitoring the sensor parameters and controlling the motor pumps from the mobile application itself.

# 3. IDEATION & PROPOSED SOLUTION

## 3.1 : Empathy Map Canvas



# 3.2 : Ideation and Brainstorming



# 3.3 : Proposed Solution

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | To provide efficient decision support system using wireless sensor network which handle different activities of farm and gives useful information related to farm. Information related to Soil moisture, Temperature and Humidity content. Due to the weather condition, water level increasing Farmers get lot of distractions which is not good for Agriculture. |
| 2. | Idea / Solution description | Smart Agricultural System solutions provide an integrated IoT platform in agriculture that allows farmers to leverage sensors, smart gateways and monitoring systems to collect information, control various parameters on their farms and analyze real-time data in order to make informed decisions. |
| 3. | Novelty / Uniqueness | Various eminent researchers have been making efforts for smart farming by using IoT concepts in agriculture. But, a bouquet of unfolded challenges is still in a queue for their effective solution. This study makes some efforts to discuss past research and open challenges in IoT based agriculture. |

| | | |
|---|---|---|
| 4. | Social Impact / Customer Satisfaction | Reduces the wages for labors who work in the agricultural field. It saves a lot of time. IoT can help improve customer relationships by enhancing the customer's overall experience. |

# 3.4 : Problem Solution Fit

| | | |
|---|---|---|
| **1.Customer segments:-** t<br><br>Generally farmers are customers, there are differen types of farmers such as marginal farmers, small farmers and large farmers. Mostly Large farmers requires smart farming, because they are not able to monitor entire field and they need to Cope with climate change, soil erosion and biodiversity loss. | **6.Customer constraints:-**<br><br>Lack of proper irrigation facilities, production machinery, and access to institutional credit, difficulties procuring inputs and storing products, and negative impacts of climate. | 5. Available solutions<br><br>Smart Farming has enabled farmers to reduce waste and enhance productivity with the help of sensors (light, humidity, temperature, soil moisture, etc.) and automation of irrigation systems. Further with the help of these sensors, farmers can monitor the field conditions from anywhere. |
| 2. Jobs to be done :-<br><br>The rapid changes in climate, soil erosion, improper usage of pesticides are solved by Internet of Things via smart farming | 9. Problem route cause:-<br><br>The main problems faced by the farmers are Cope with climate change, soil erosion and biodiversity loss.<br>Satisfy consumers' changing tastes and expectations.<br>Meet rising demand for more food of higher quality.<br>Invest in farm productivity. | 7. Behavior:-<br><br>Predict the climate change in advance and prevent from biodiversity loss is always a difficult task for an customer such as farmer |
| 3. Triggers:-<br><br>Some of the triggers in smart farming are advertising in television and create awareness about smart farming. | 0. Solution:-<br><br>'o overcome the all types of problems faced by ustomers | 8. Channels of behavior:- |

# 4.REQUIREMENT ANALYSIS

## 4.1 : Functional requirements

| FR NO | FUNCTIONAL REQUIREMENTS | SUB REQUIREMENTS |
|---|---|---|
| **FR-1** | Check the rainfall level | View rainfall percentage compared to corresponding week of last year<br>View the percentage of rainfall in the selected week to the long term average |
| **FR-2** | Area Progress Report | View the Sown Area Progress Report View the water availability Report View the Reservoir Capacity Report |
| **FR-3** | Fodder and Cattle | Fodder and Cattle Report on Damage of Fodder & Cattle due to Drought |
| **FR-4** | Farmer employability | Manage the information of number of farmers & area where crop loss is >50%. |
| **FR-5** | Drought level | Manage the Drought Declaration Certificate Report on Damage of Agriculture due to Drought |
| **FR-6** | Pest report | Check the pests occurs in our plant that causes a damage<br>And reduce the level of pests using pesticides |
| **FR-7** | Grievance | Monitor Grievance Check Grievance Status Resolve Grievance report grievance |
| **FR-8** | Capacity report | Manage the Information of Reservoir Capacity and their details for moisturizing |

# 4.2: Non-Functional requirements

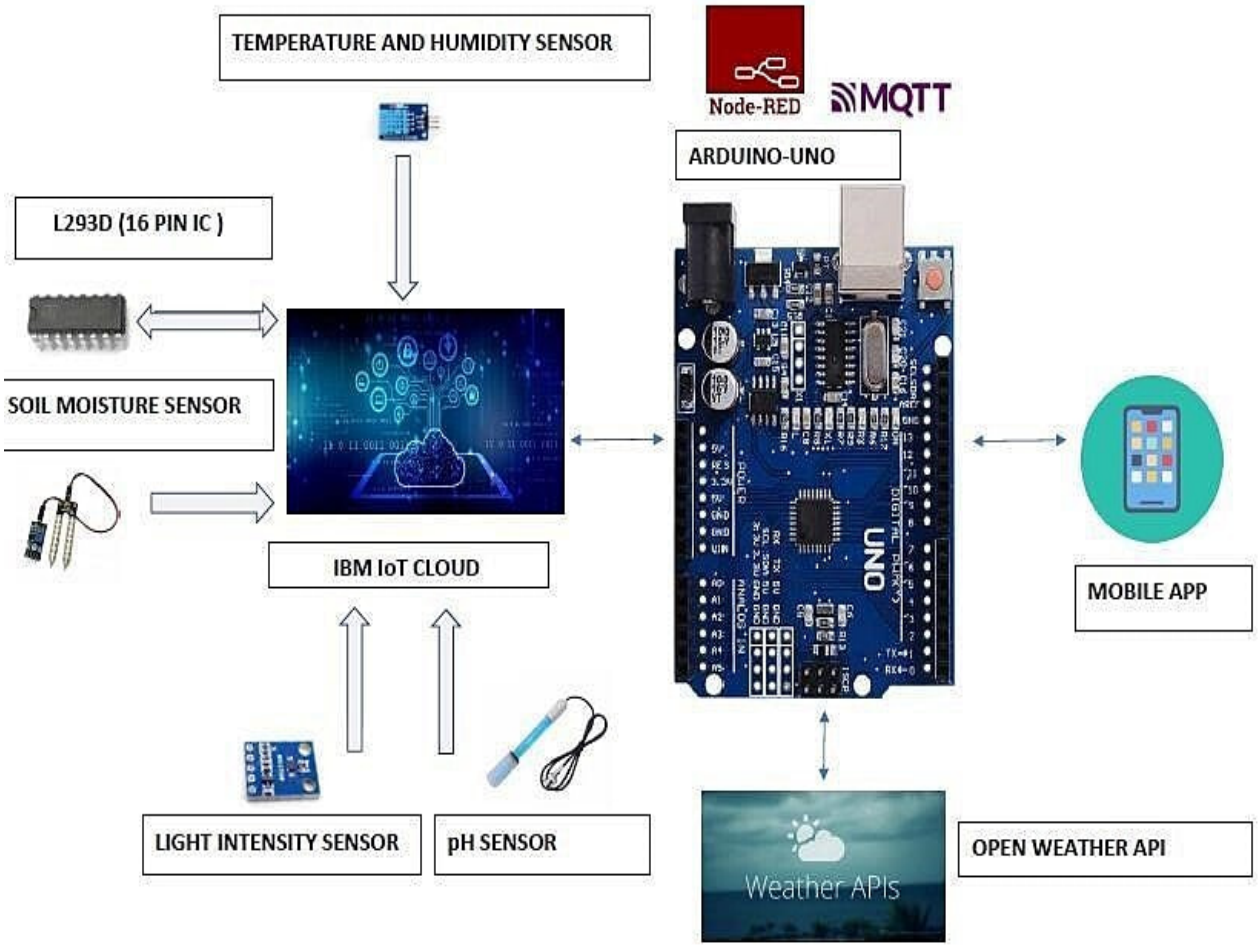
## Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

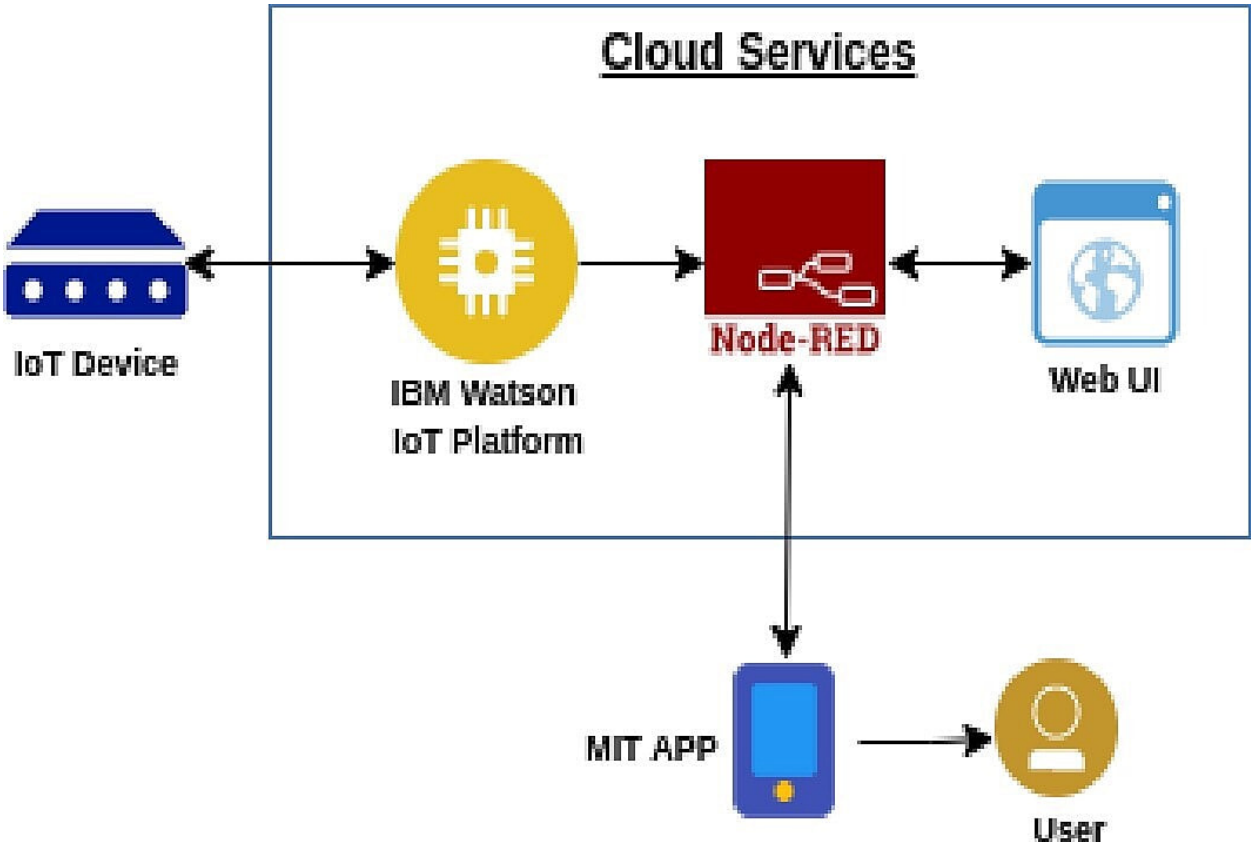
| NFR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | i. A logical interface is essential to make easy use of system, speeding up common tasks.<br>ii. The product could be used by two categories of people mainly administrator category and other users. |
| NFR-2 | **Security** | Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below:<br>i. Keep specific log or history data sets.<br>ii. Utilize certain cryptographic techniques. |

# 5 . PROJECT DESIGN

## 5.1 : Data Flow Diagrams



## 5.2 : Solution &Technical Architecture




## 5.3 : User Stories

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria |
|---|---|---|---|---|
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard |
| | | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm |

| | | | | |
|---|---|---|---|---|
| | | USN-3 | As a user, I can register for the application through Facebook | I can register & access the dashboard with Facebook Login |
| | | USN-4 | As a user, I can register for the application through Gmail | I can register & access the dashboard with Gr Login |
| | Login | USN-5 | As a user, I can log into the application by entering email & password | I can access dashboard with email login |
| | Dashboard | USN-6 | As a user I can enter into dashboard by using navigation panel | I can access the dashboard by using navigatio panel |
| Customer (Web user) | Registration | USN-1 | As a user, I can register for the web application by entering my email, password, and confirming my password. | I can access my account / dashboard |
| | | USN-2 | As a user, I will receive confirmation email once I have registered for the web application | I can receive confirmation email & click confirr |
| | Login | USN-3 | As a user, I can log into the web application by entering email & password | I can access dashboard with email login |
| | Dashboard | USN-4 | As a user I can enter into web dashboard by using navigation panel | I can access into dashboard by using navigati panel |
| Customer Care Executive | Registration | USN-1 | As a user I can contact the customer care service through phone or mail medium | I can receive confirmation SMS or email |

| | | | | |
|---|---|---|---|---|
| | | USN-2 | As a user I want customer care to answer the questions related to product and services | I can get the problem solved within a day |
| | | USN-3 | As a user I want customer care to register my complaints | I can receive a confirmation message stating my complaint is registered |
| | | USN-4 | As a user I want customer care to collect and analyze consumer feedback | I can get the status of my feedback |
| | | USN-5 | As a user I want customer care to troubleshoot technical problems | I can get the problem solved within a day |
| Administrator | | USN-1 | As a user I want the administrator to use good working hardware | I can get a guarantee and warranty card |
| | | USN-2 | As a user I want the administrator to sell the product in a reasonable rate | I can get the cost of bill of materials |
| | | USN-3 | As a user I want the administrator to refund my amount if I am not satisfied with the product | I can get an assurance stating I will get amount back |

# 6 . PROJECT PLANNING & SCHEDULING

## 6.1 : Sprint Planning & Estimation

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Simulation creation | USN-1 | Connect Sensors and Arduino with python code | 2 | High | Agalya |
| Sprint-1 | | USN-2 | Connect the sensor with the python code | 2 | Medium | Boopalan |
| Sprint-1 | | USN-3 | Arduino connection with the python code | 1 | Medium | Abinaya |
| Sprint-2 | Software | USN-4 | Creating device in the IBM Watson IoT platform, workflow for IoT scenarios using Node-Red | 1 | High | Poojitha |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-2 | | USN-5 | Creating device in Watson iot | 2 | High | Agalya |

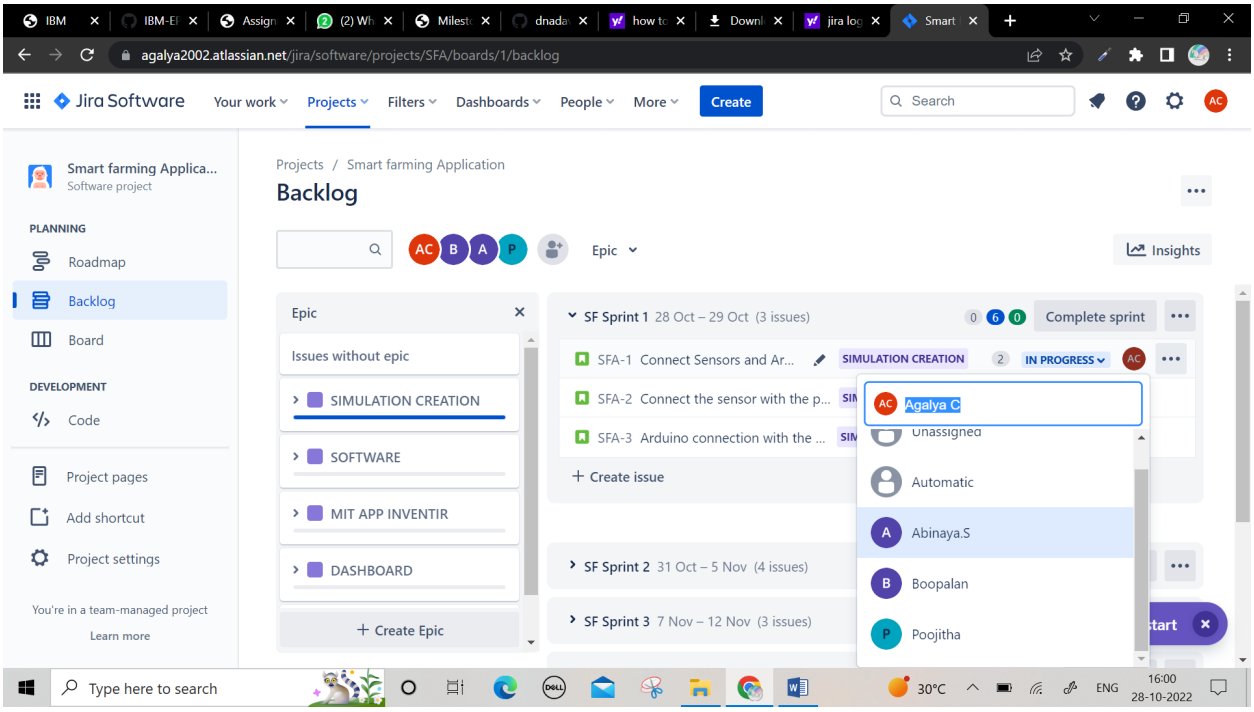
| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-2 | | USN-6 | Create Node red service | 1 | High | Boopalan |
| Sprint-2 | | USN-7 | workflow for IoT scenarios using Node-Red | 2 | High | Abinaya |
| Sprint-3 | MIT App Inventor | USN-8 | Develop an application for the Smart farmer project using MIT App Inventor | 2 | High | Poojitha |
| Sprint-3 | | USN-9 | Create MIT app inventor in smart farming | 1 | Low | Agalya |
| Sprint-3 | | USN-10 | Create an application using MIT app inventor | 1 | HIgh | Boopalan |

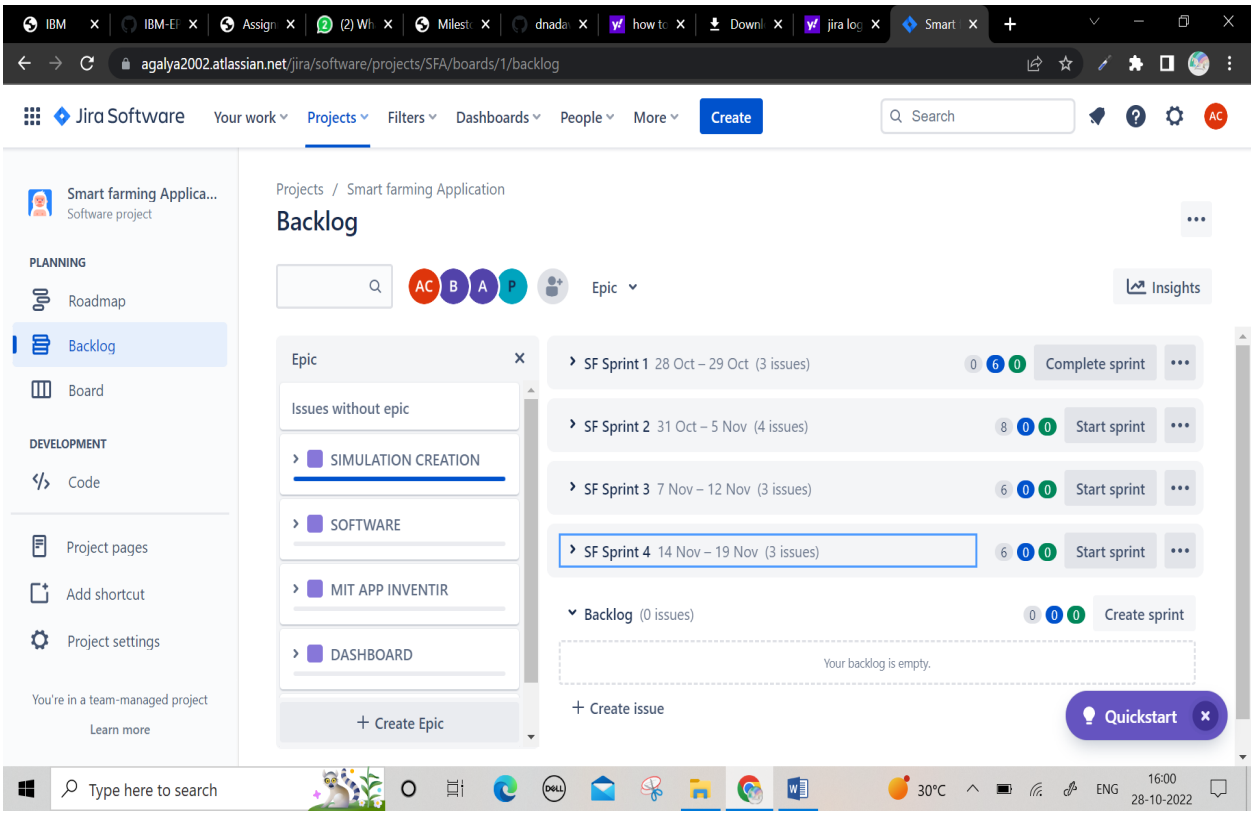
| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-4 | Dashboard | USN-11 | Design the Modules | 1 | High | Abinaya |
| Sprint-4 | | USN-12 | Test the app | 1 | High | Poojitha |
| Sprint-4 | Web UI | USN-13 | To make the user to interact with software. | 2 | High | Agalya |

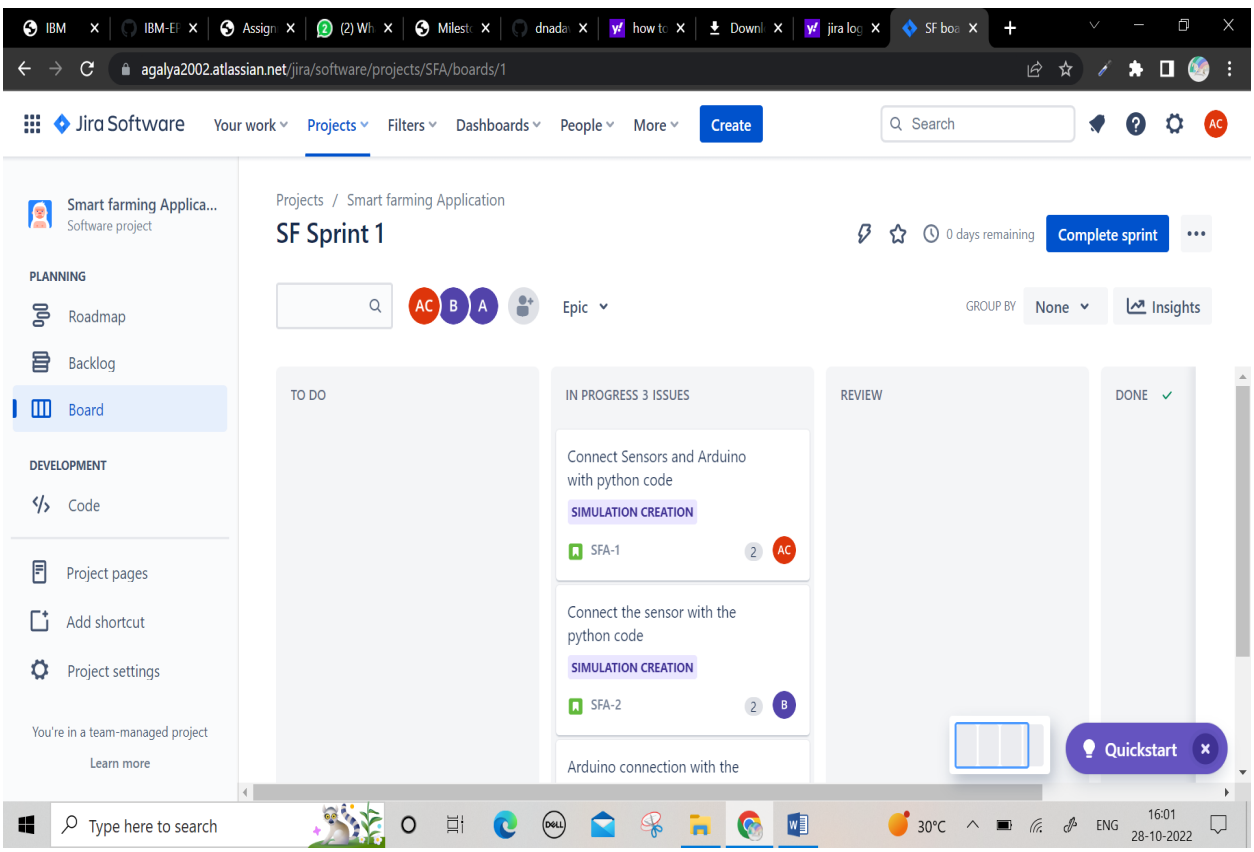
## 6.2 : Sprint Delivery Schedule

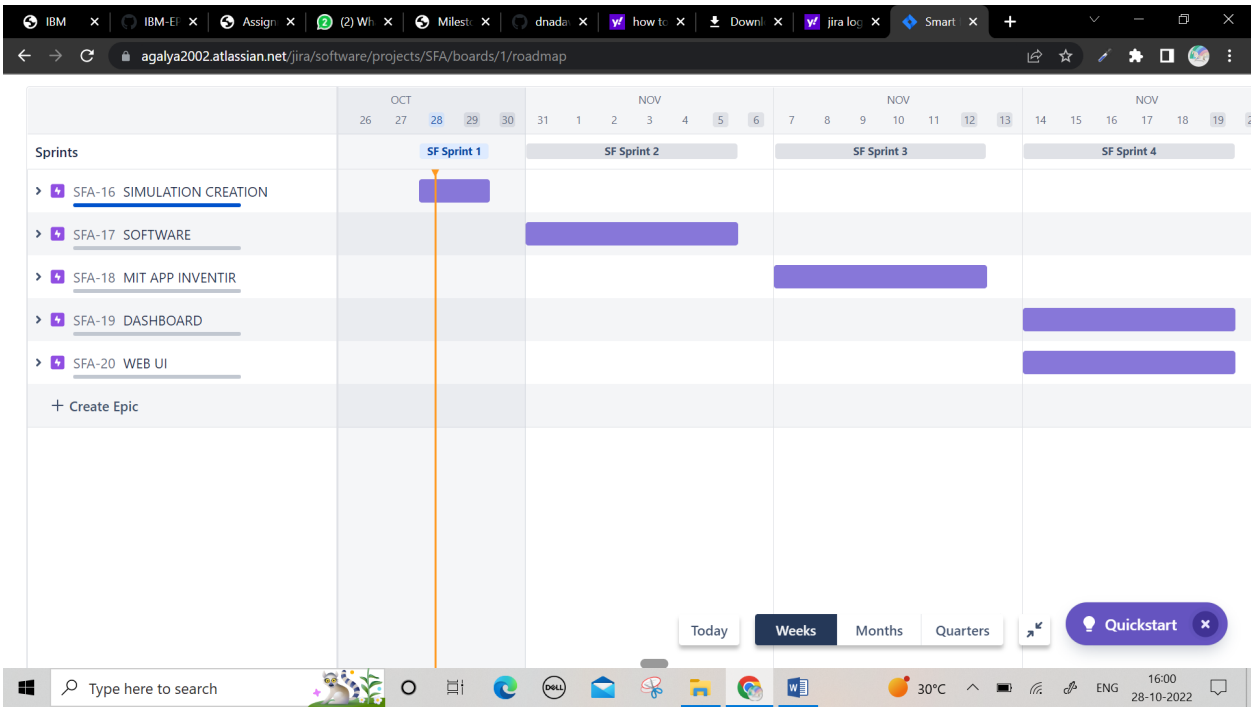
| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint Release Date (Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 20 | 05 Nov 2022 |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 | 12 Nov 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 20 | 19 Nov 2022 |

# 6.3 : Reports From JIRA




agalya2002.atlassian.net/jira/software/projects/SFA/boards/1/backlog
Jira Software
Your work
Projects
Filters
Dashboards
People
More
Create
Search
Smart farming Applica...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / Smart farming Application
Backlog
Epic
Insights
Issues without epic
SIMULATION CREATION
SOFTWARE
MIT APP INVENTIR
DASHBOARD
+ Create Epic
SF Sprint 1 28 Oct – 29 Oct (3 issues)
Complete sprint
SF Sprint 2 31 Oct – 5 Nov (4 issues)
Start sprint
SF Sprint 3 7 Nov – 12 Nov (3 issues)
SF Sprint 4 14 Nov – 19 Nov (3 issues)
Backlog (0 issues)
Create sprint
Your backlog is empty.
+ Create issue
Quickstart
Type here to search
30°C
ENG
16:00
28-10-2022


agalya2002.atlassian.net/jira/software/projects/SFA/boards/1
Jira Software
Your work
Projects
Filters
Dashboards
People
More
Create
Search
Smart farming Applica...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / Smart farming Application
SF Sprint 1
0 days remaining
Complete sprint
Epic
GROUP BY
None
Insights
TO DO
IN PROGRESS 3 ISSUES
REVIEW
DONE
Connect Sensors and Arduino with python code
SIMULATION CREATION
SFA-1
Connect the sensor with the python code
SIMULATION CREATION
SFA-2
Arduino connection with the
Quickstart
Type here to search
30°C
ENG
16:01
28-10-2022




# 7 . CODING & SOLUTIONING

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload,
```

```
unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "y3vxbr"//IBM ORGANITION ID
#define DEVICE_TYPE "newdev"//Device type
mentioned in ibm watson IOT Platform
#define DEVICE_ID "newdevid"//Device ID mentioned
in ibm watson IOT Platform
#define TOKEN "oyYEZ1V9ZSDPcSDcj-"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG
".messaging.internetofthings.ibmcloud.com";// Server
Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic
name and type of event perform and format in which data
to be send
char subscribetopic[] = "iot-
```

```
2/cmd/command/fmt/String";// cmd  REPRESENT
command type AND COMMAND IS TEST OF
FORMAT STRING
char authMethod[] = "use-token-auth";// authentication
method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":"
DEVICE_ID;//client id


//--------------------------------------
WiFiClient wifiClient; // creating the instance for
wificlient
PubSubClient client(server, 1883, callback ,wifiClient);
//calling the predefined client id by passing parameter like
server id,portand wificredential
void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
```

```
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  Serial.print("temperature:");
  Serial.println(t);
  Serial.print("humidity:");
  Serial.println(h);

  PublishData(t, h);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}
```

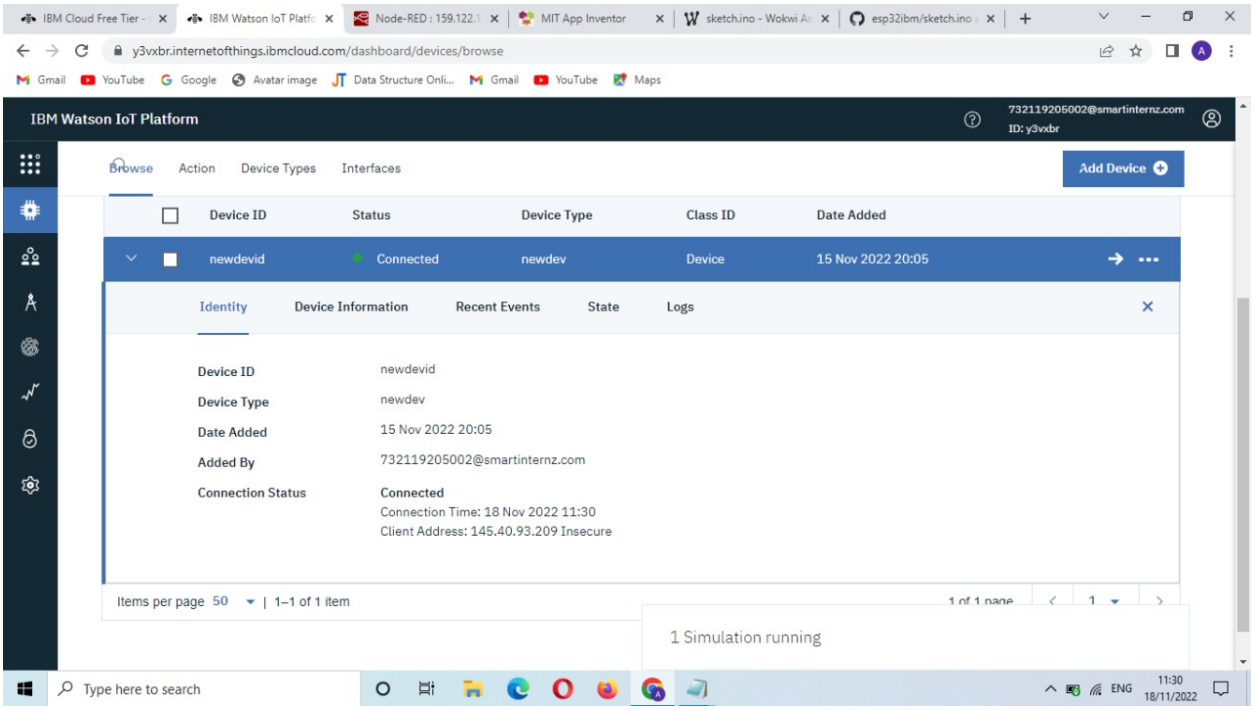
```
/*.....................................retrieving to
Cloud.............................*/

void PublishData(float temp, float humid) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data
to ibm cloud
  */
  String payload = "{\"Temperature\":";
  payload += temp;
  payload += "," "\"Humidity\":";
  payload += humid;
  payload += "}";


  Serial.print("Sending payload: ");
  Serial.println(payload);
```

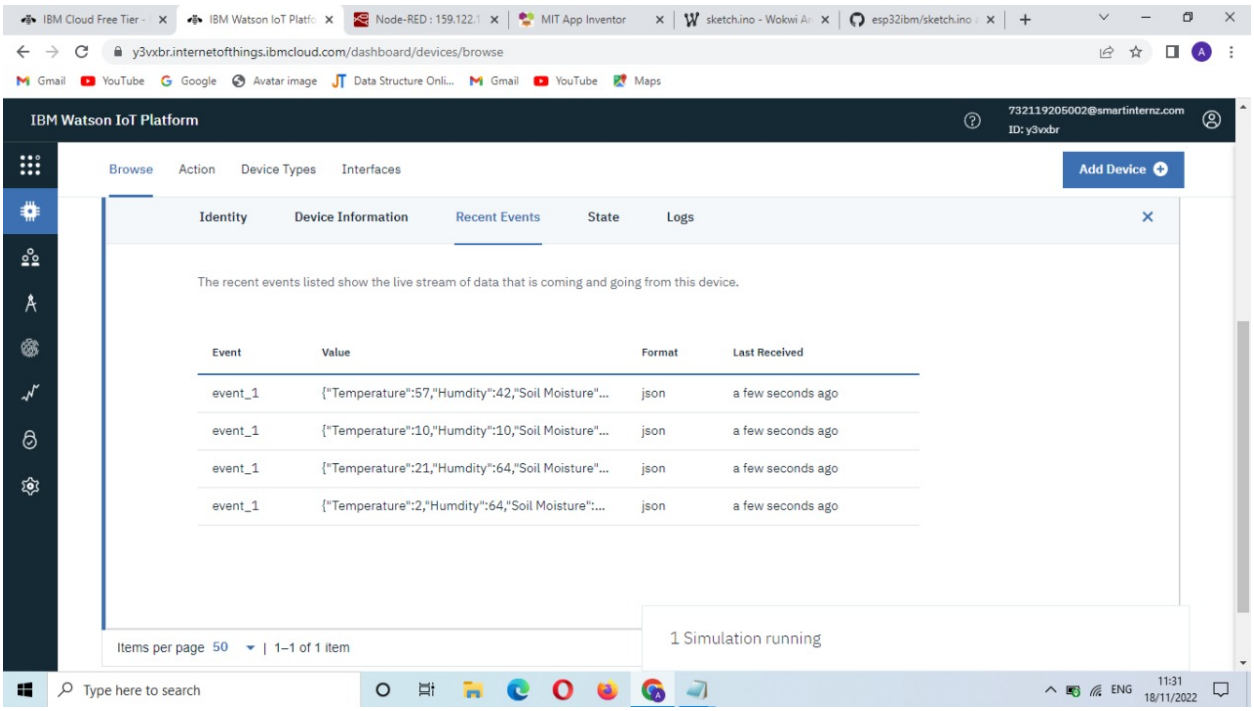
```
  if (client.publish(publishTopic, (char*) payload.c_str()))
{
    Serial.println("Publish ok");// if it sucessfully upload
data on the cloud then it will print publish ok in Serial
monitor or else it will print publish failed
  } else {
    Serial.println("Publish failed");
  }


}
void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!!!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }

     initManagedDevice();
     Serial.println();
  }
```

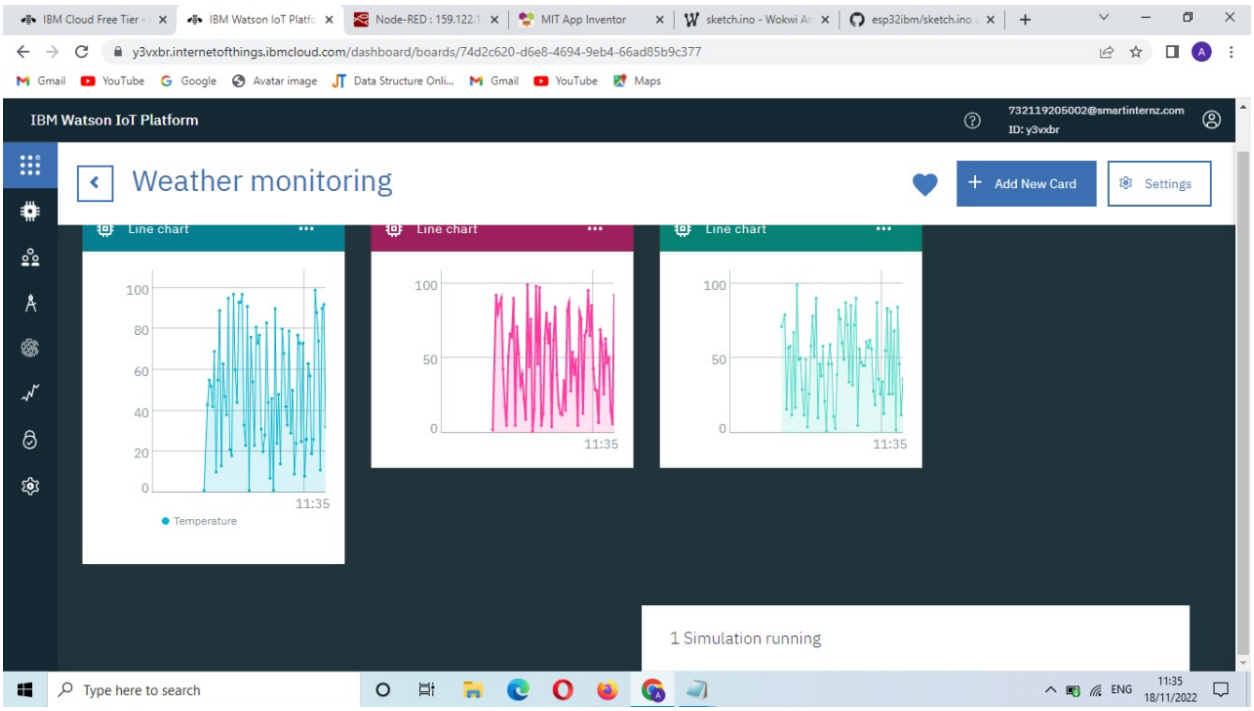
```
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi
credentials to establish the connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
```

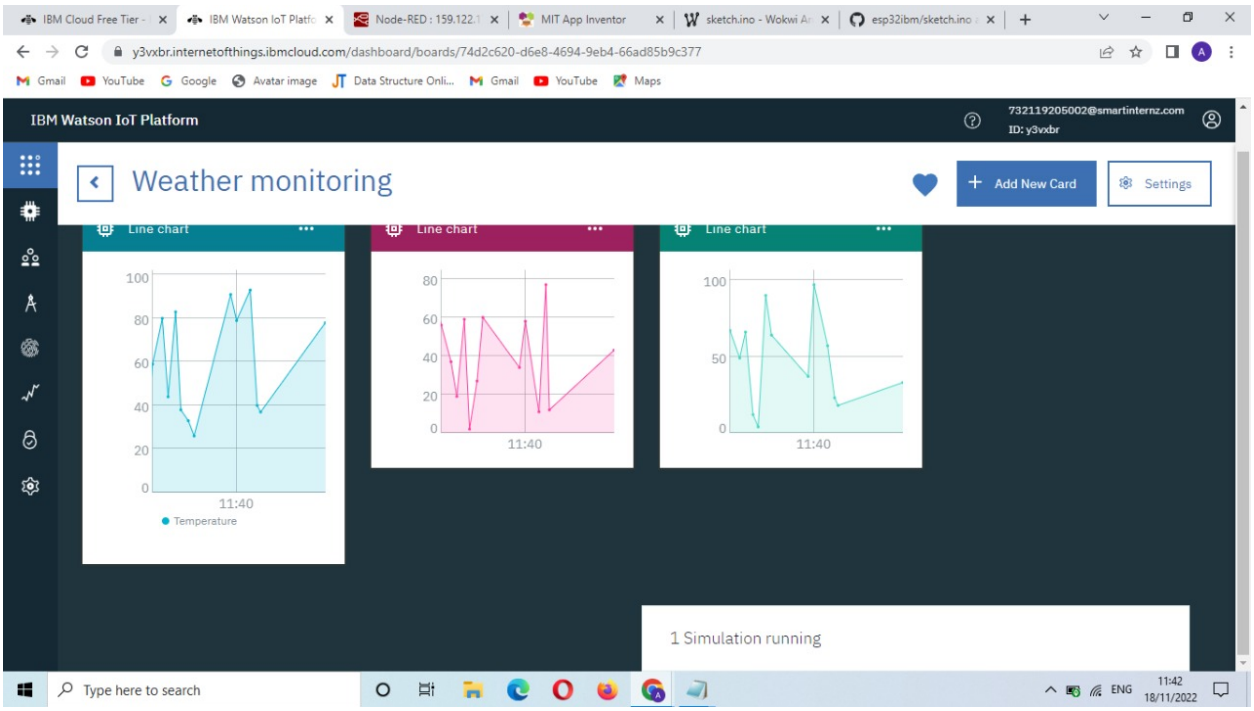
```
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}

void callback(char* subscribetopic, byte* payload,
unsigned int payloadLength)
{

  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {
    //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }

  Serial.println("data: "+ data3);
  if(data3=="lighton")
  {
Serial.println(data3);
digitalWrite(LED,HIGH);
```

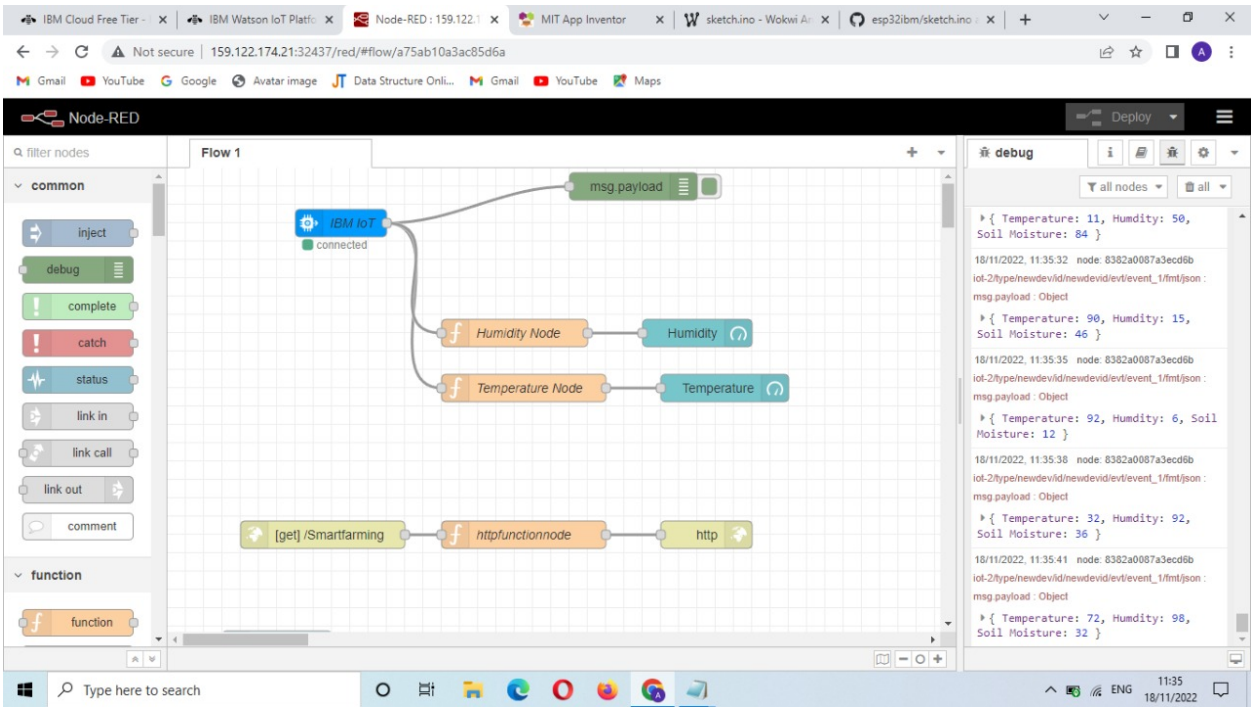
```
  }

  else
  {
Serial.println(data3);
digitalWrite(LED,LOW);

  }
data3="";


}
```

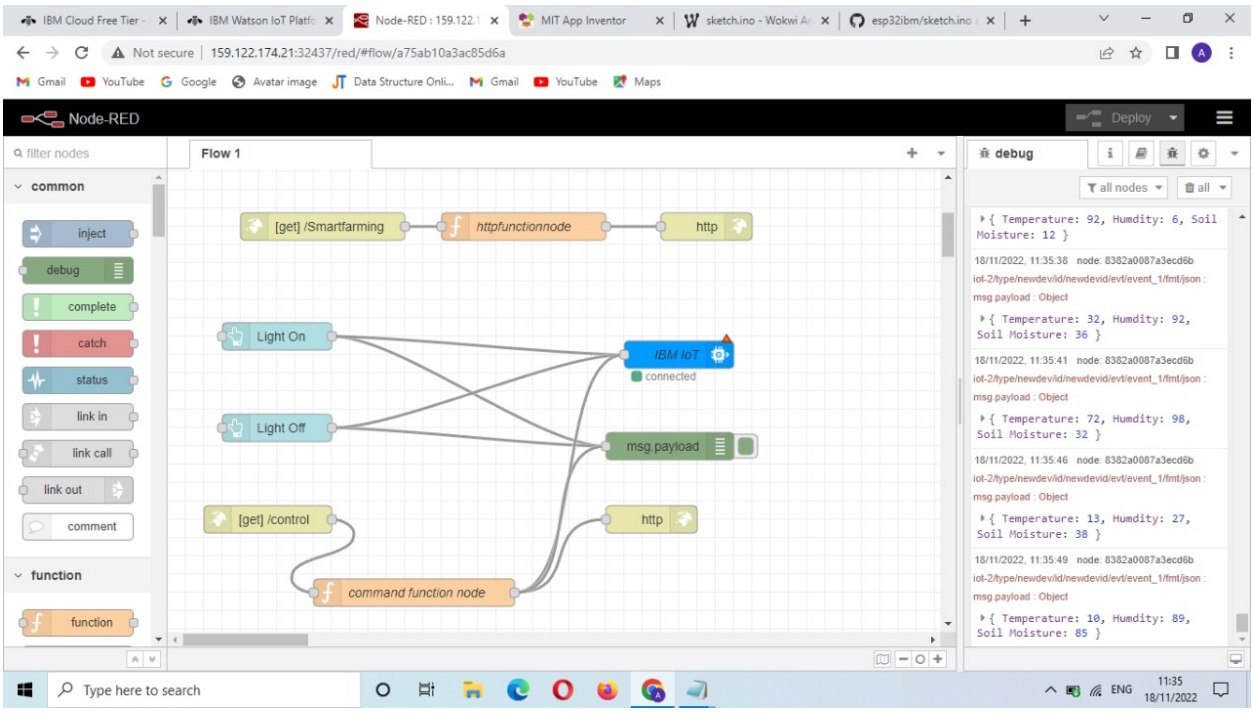


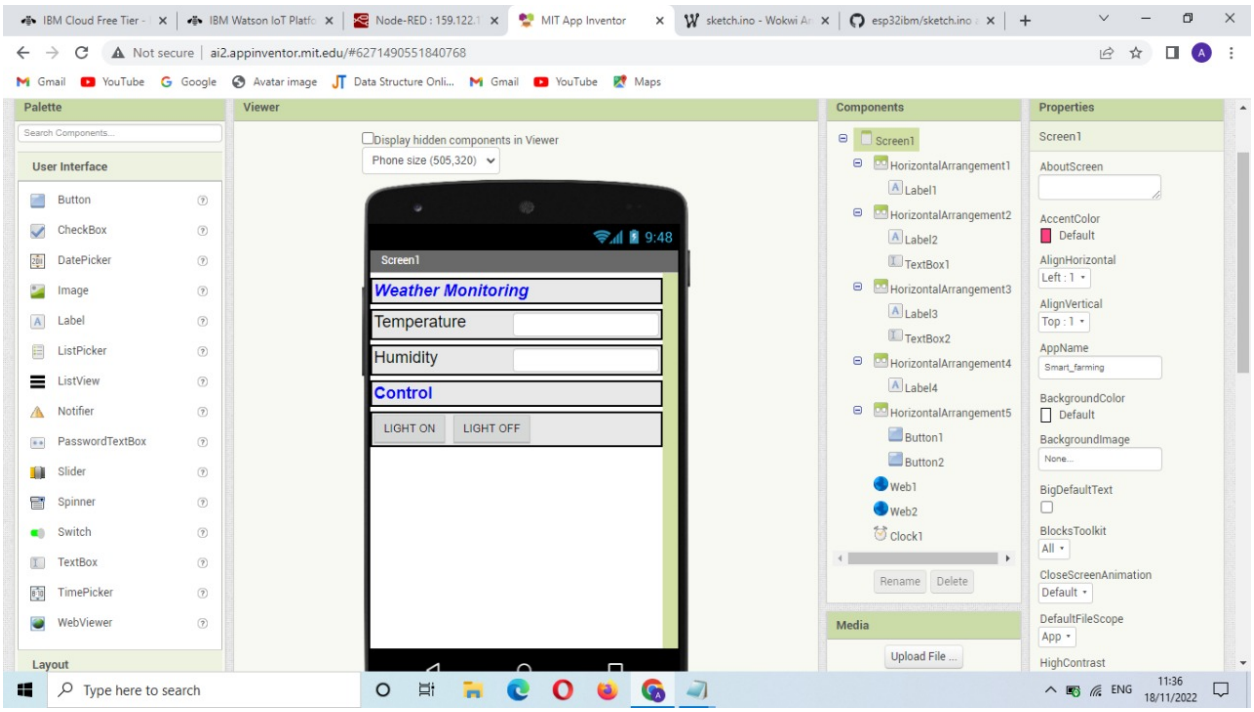

IBM Watson IoT Platform
732119205002@smartinternz.com
ID: y3vxbr
Browse
Action
Device Types
Interfaces
Add Device
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
event_1
{"Temperature":57,"Humdity":42,"Soil Moisture"...
json
a few seconds ago
event_1
{"Temperature":10,"Humdity":10,"Soil Moisture"...
json
a few seconds ago
event_1
{"Temperature":21,"Humdity":64,"Soil Moisture"...
json
a few seconds ago
event_1
{"Temperature":2,"Humdity":64,"Soil Moisture":...
json
a few seconds ago
Items per page 50
1–1 of 1 item
1 Simulation running
Type here to search
ENG
11:31
18/11/2022

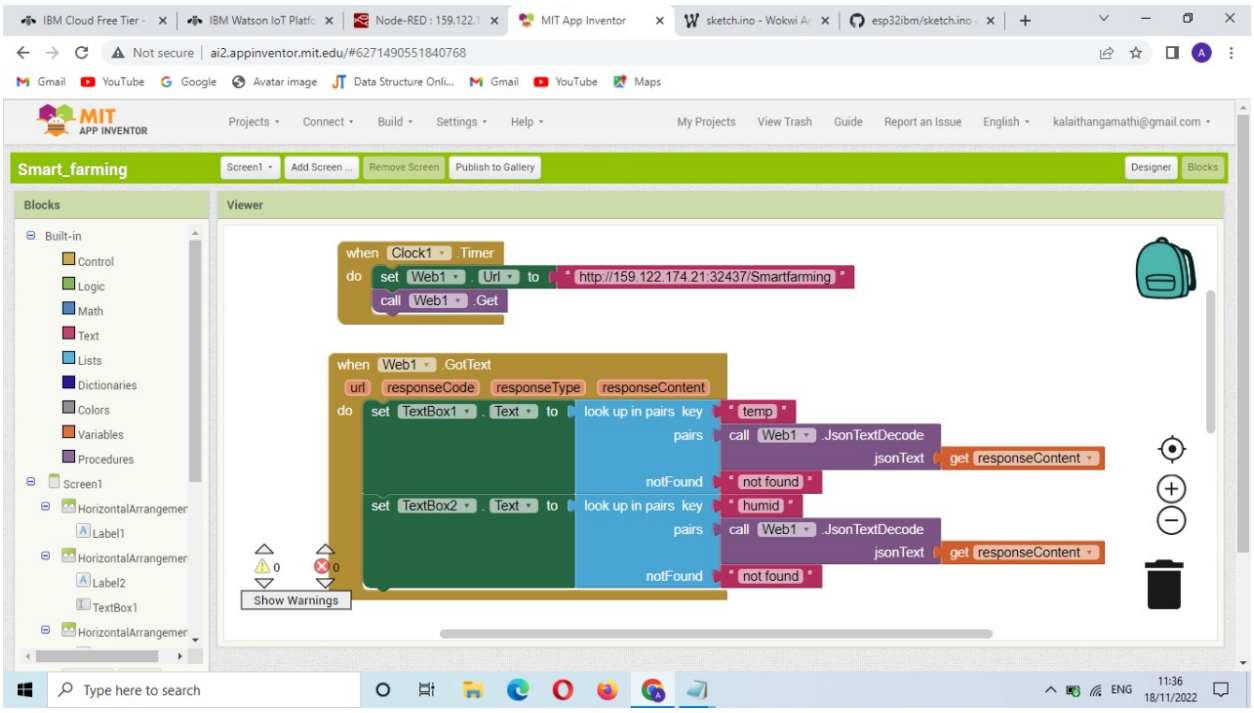

IBM Watson IoT Platform
732119205002@smartinternz.com
ID: y3vxbr
Weather monitoring
Add New Card
Settings
Line chart
Line chart
Line chart
100
80
60
40
20
0
11:35
Temperature
100
50
0
11:35
100
50
0
11:35
1 Simulation running
Type here to search
ENG
11:35
18/11/2022

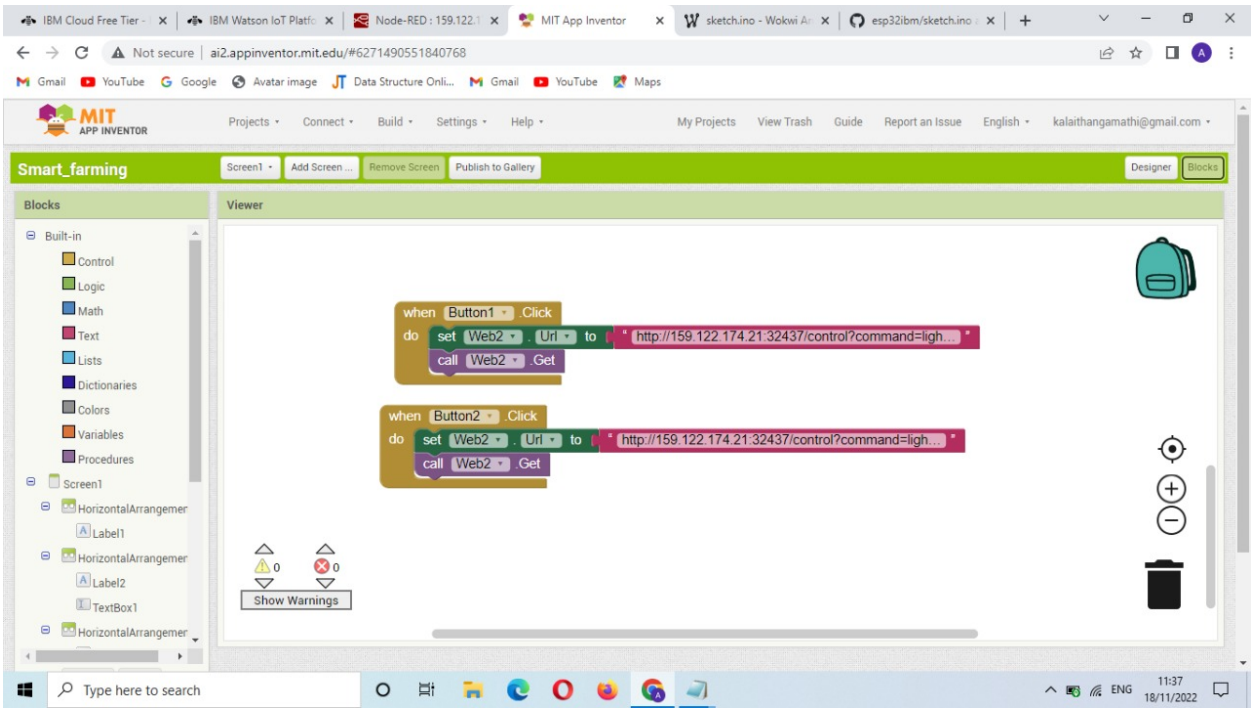

IBM Watson IoT Platform
Weather monitoring
Add New Card
Settings
Line chart
Temperature
1 Simulation running

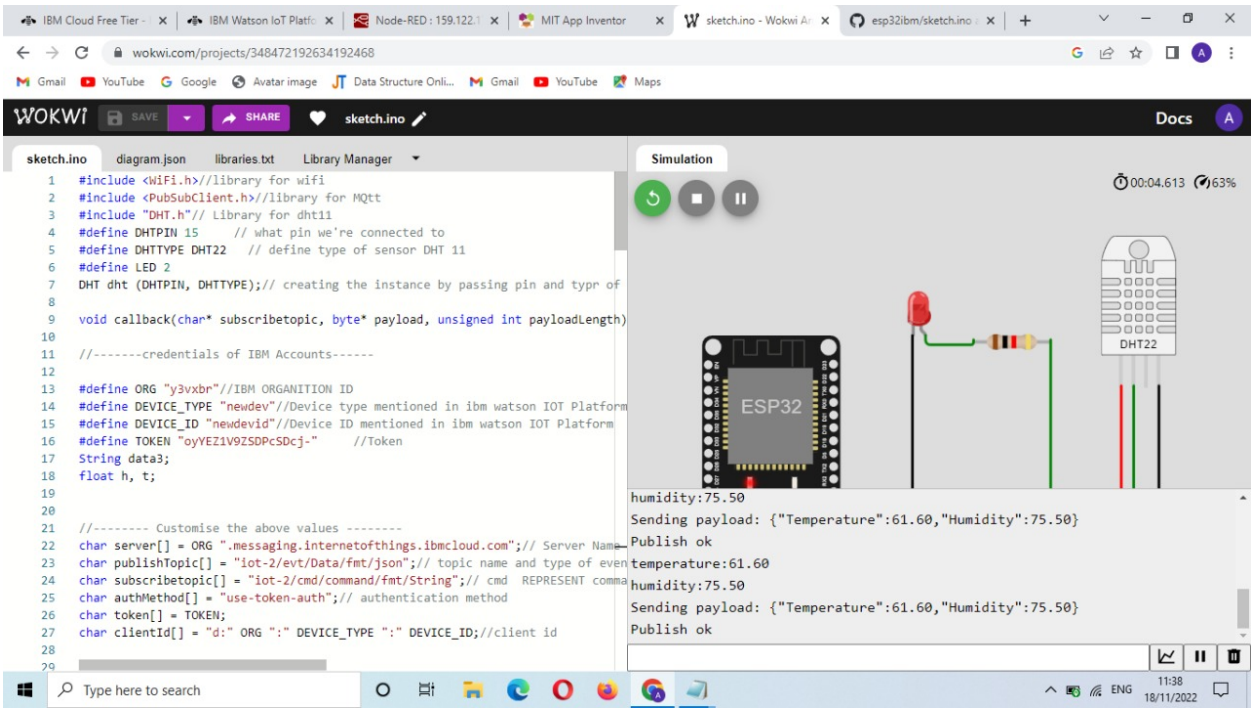

Node-RED
Deploy
Flow 1
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
msg.payload
IBM IoT
connected
Humidity Node
Humidity
Temperature Node
Temperature
[get] /Smartfarming
httpfunctionnode
http
debug
all nodes
{ Temperature: 11, Humdity: 50, Soil Moisture: 84 }
{ Temperature: 90, Humdity: 15, Soil Moisture: 46 }
{ Temperature: 92, Humdity: 6, Soil Moisture: 12 }
{ Temperature: 32, Humdity: 92, Soil Moisture: 36 }
{ Temperature: 72, Humdity: 98, Soil Moisture: 32 }

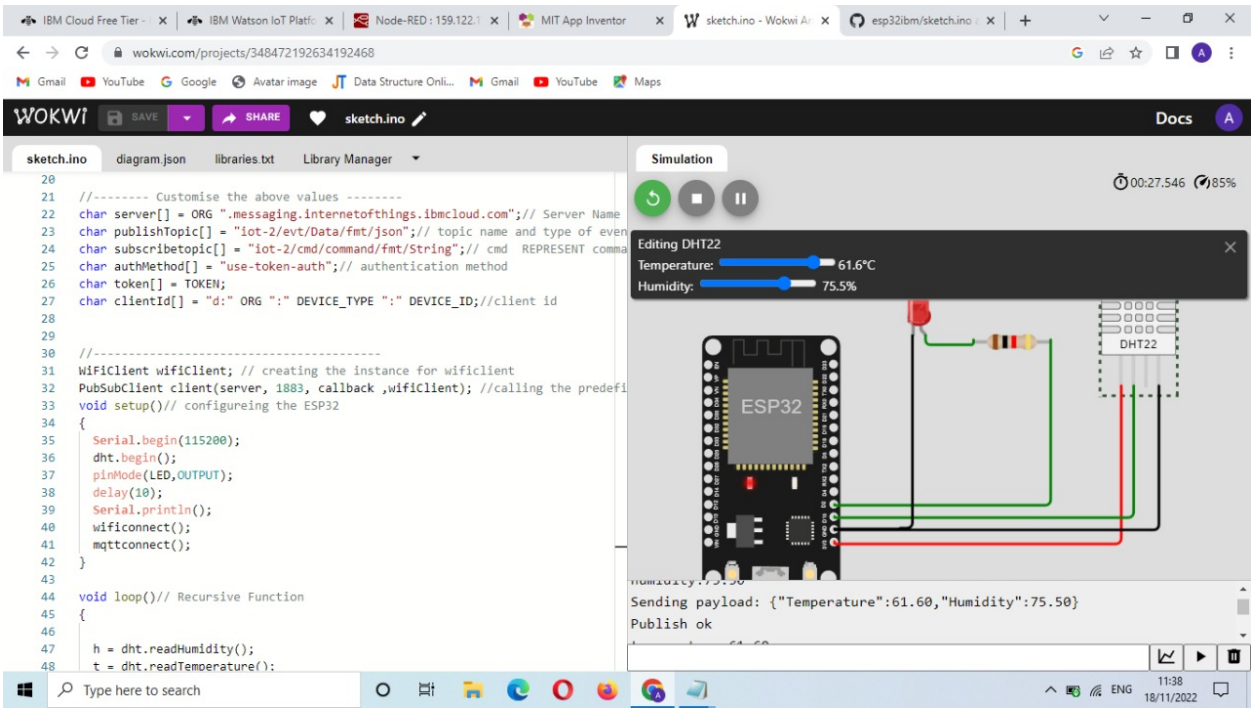

Node-RED
Deploy
Flow 1
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
[get] /Smartfarming
httpfunctionnode
http
Light On
Light Off
IBM IoT
connected
msg.payload
[get] /control
http
command function node
debug
all nodes
{ Temperature: 92, Humdity: 6, Soil Moisture: 12 }
18/11/2022, 11:35:38 node: 8382a0087a3ecd6b
iot-2/type/newdev/id/newdevid/evt/event_1/fmt/json : msg.payload : Object
{ Temperature: 32, Humdity: 92, Soil Moisture: 36 }
18/11/2022, 11:35:41 node: 8382a0087a3ecd6b
iot-2/type/newdev/id/newdevid/evt/event_1/fmt/json : msg.payload : Object
{ Temperature: 72, Humdity: 98, Soil Moisture: 32 }
18/11/2022, 11:35:46 node: 8382a0087a3ecd6b
iot-2/type/newdev/id/newdevid/evt/event_1/fmt/json : msg.payload : Object
{ Temperature: 13, Humdity: 27, Soil Moisture: 38 }
18/11/2022, 11:35:49 node: 8382a0087a3ecd6b
iot-2/type/newdev/id/newdevid/evt/event_1/fmt/json : msg.payload : Object
{ Temperature: 10, Humdity: 89, Soil Moisture: 85 }
Not secure | 159.122.174.21:32437/red/#flow/a75ab10a3ac85d6a
Type here to search
ENG
11:35
18/11/2022

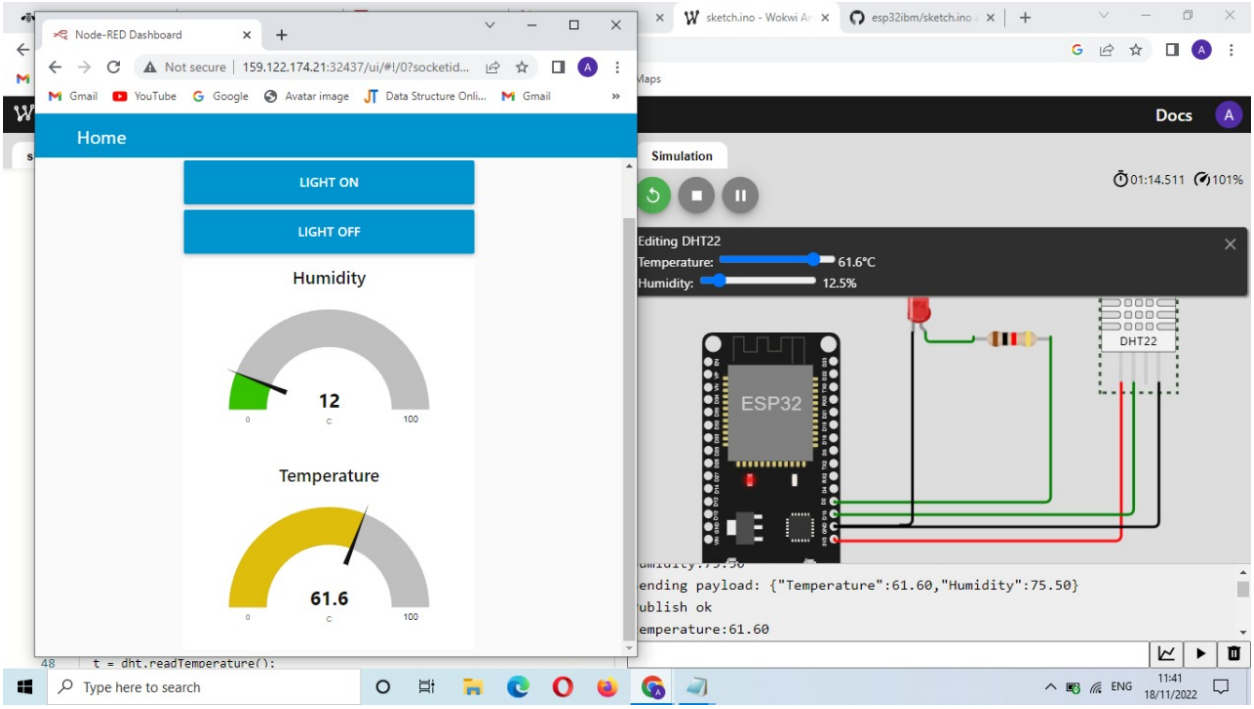

Palette
Search Components...
User Interface
Button
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Viewer
Display hidden components in Viewer
Phone size (505,320)
9:48
Screen1
Weather Monitoring
Temperature
Humidity
Control
LIGHT ON
LIGHT OFF
Components
Screen1
HorizontalArrangement1
Label1
HorizontalArrangement2
Label2
TextBox1
HorizontalArrangement3
Label3
TextBox2
HorizontalArrangement4
Label4
HorizontalArrangement5
Button1
Button2
Web1
Web2
Clock1
Rename
Delete
Media
Upload File ...
Properties
Screen1
AboutScreen
AccentColor
Default
AlignHorizontal
Left : 1
AlignVertical
Top : 1
AppName
Smart_farming
BackgroundColor
Default
BackgroundImage
None...
BigDefaultText
BlocksToolkit
All
CloseScreenAnimation
Default
DefaultFileScope
App
HighContrast
Not secure | ai2.appinventor.mit.edu/#6271490551840768
Type here to search
ENG
11:36
18/11/2022

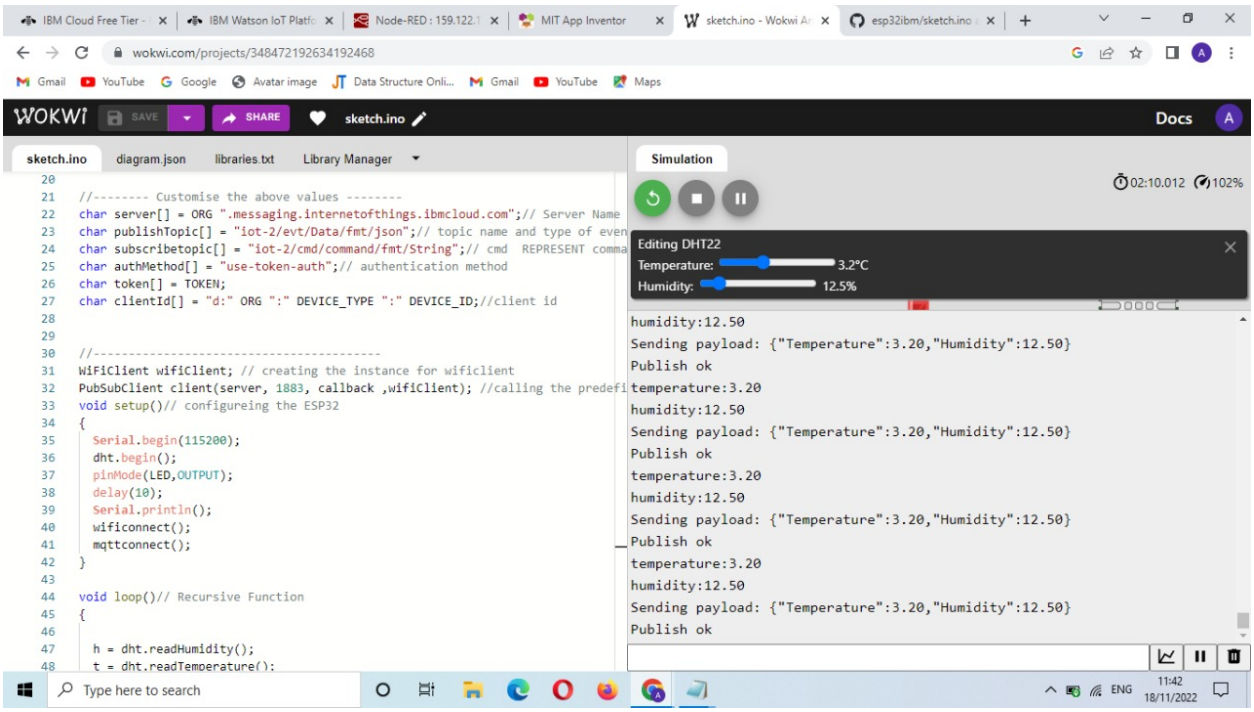

MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
kalaithangamathi@gmail.com
Smart_farming
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen1
HorizontalArrangemer
Label1
HorizontalArrangemer
Label2
TextBox1
HorizontalArrangemer
Show Warnings
when Clock1 .Timer
do set Web1 . Url to " http://159.122.174.21:32437/Smartfarming "
call Web1 .Get
when Web1 .GotText
url responseCode responseType responseContent
do set TextBox1 . Text to look up in pairs key " temp "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " not found "
set TextBox2 . Text to look up in pairs key " humid "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " not found "
Type here to search
ENG
11:36
18/11/2022

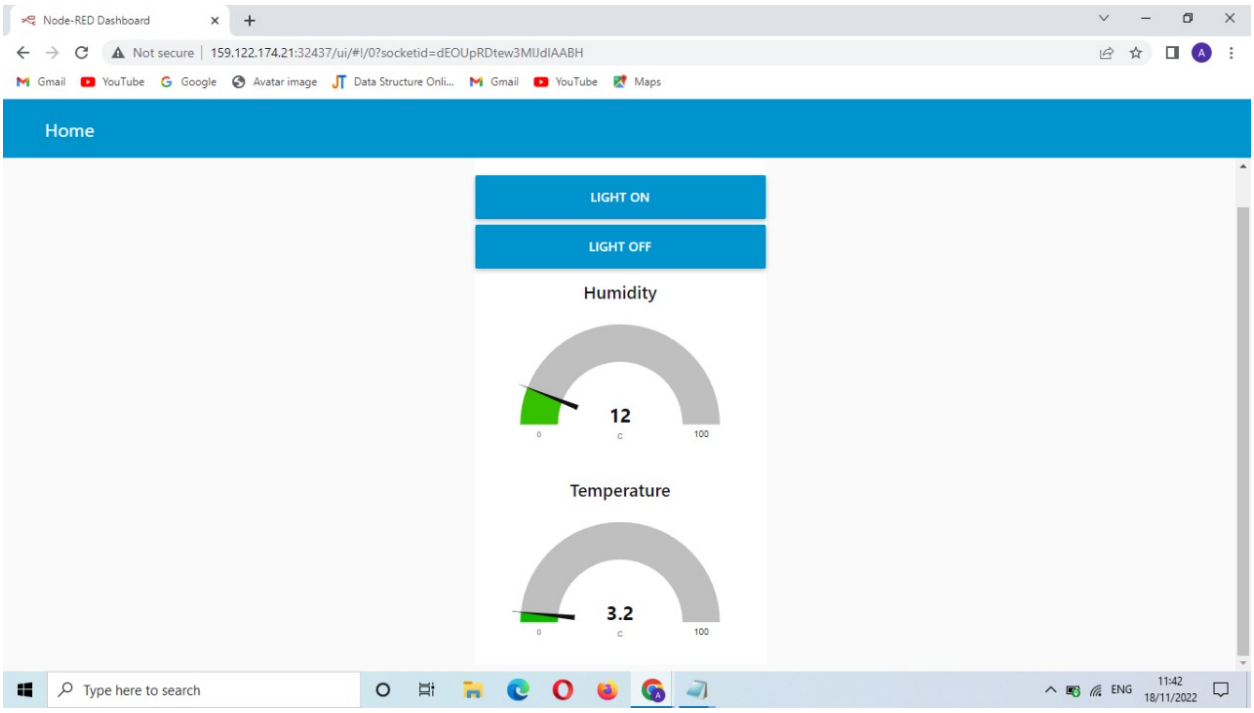

MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
kalaithangamathi@gmail.com
Smart_farming
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Designer
Blocks
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen1
HorizontalArrangemer
Label1
HorizontalArrangemer
Label2
TextBox1
HorizontalArrangemer
Show Warnings
when Button1 .Click
do set Web2 . Url to " http://159.122.174.21:32437/control?command=ligh... "
call Web2 .Get
when Button2 .Click
do set Web2 . Url to " http://159.122.174.21:32437/control?command=ligh... "
call Web2 .Get
Type here to search
ENG
11:37
18/11/2022


wokwi.com/projects/348472192634192468
WOKWI
SAVE
SHARE
sketch.ino
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15 // what pin we're connected to
#define DHTTYPE DHT22 // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
//-------credentials of IBM Accounts------
#define ORG "y3vxbr"//IBM ORGANITION ID
#define DEVICE_TYPE "newdev"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "newdevid"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "oyYEZ1V9ZSDPcSDcj-" //Token
String data3;
float h, t;
//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of even
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT comma
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
Simulation
00:04.613
63%
ESP32
DHT22
humidity:75.50
Sending payload: {"Temperature":61.60,"Humidity":75.50}
Publish ok
temperature:61.60
humidity:75.50
Sending payload: {"Temperature":61.60,"Humidity":75.50}
Publish ok
Type here to search
ENG
11:38
18/11/2022


wokwi.com/projects/348472192634192468
WOKWI
SAVE
SHARE
sketch.ino
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of even
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT comma
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
//----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefi
void setup()// configureing the ESP32
{
Serial.begin(115200);
dht.begin();
pinMode(LED,OUTPUT);
delay(10);
Serial.println();
wificonnect();
mqttconnect();
}
void loop()// Recursive Function
{
h = dht.readHumidity();
t = dht.readTemperature();
Simulation
00:27.546
85%
Editing DHT22
Temperature:
61.6°C
Humidity:
75.5%
ESP32
DHT22
Sending payload: {"Temperature":61.60,"Humidity":75.50}
Publish ok
Type here to search
ENG
11:38
18/11/2022


Node-RED Dashboard
Not secure 159.122.174.21:32437/ui/#!/0?socketid...
Home
LIGHT ON
LIGHT OFF
Humidity
12
Temperature
61.6
sketch.ino - Wokwi
Simulation
01:14.511
101%
Editing DHT22
Temperature: 61.6°C
Humidity: 12.5%
ESP32
DHT22
ending payload: {"Temperature":61.60,"Humidity":75.50}
ublish ok
emperature:61.60
Type here to search
ENG
11:41
18/11/2022


wokwi.com/projects/348472192634192468
WOKWI
SAVE
SHARE
sketch.ino
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of even
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT comma
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefi
void setup()// configureing the ESP32
{
Serial.begin(115200);
dht.begin();
pinMode(LED,OUTPUT);
delay(10);
Serial.println();
wificonnect();
mqttconnect();
}
void loop()// Recursive Function
{
h = dht.readHumidity();
t = dht.readTemperature();
Simulation
02:10.012
102%
Editing DHT22
Temperature: 3.2°C
Humidity: 12.5%
humidity:12.50
Sending payload: {"Temperature":3.20,"Humidity":12.50}
Publish ok
temperature:3.20
Type here to search
ENG
11:42
18/11/2022




# 8.Testing

## 8.1 Performance Testing

| S.No. | Parameter | Values | Screenshot |
|---|---|---|---|
| 1. | Metrics | **Regression Model:** Logistic Regression<br>**Classification Model:**<br>• Confusion Matrix - [[ 0 5]<br>[ 1 74]]<br>• Accuracy Score - 92.50000<br>• Recall Score - 98.666667<br>• ROC Score - 49.333333<br>• Classification Report :<br>precision - 0.88<br>support - 80<br>f1-score - 0.90<br>recall - 0.93 |  |
| 2. | Tune The Model | Hyper parameter Tuning:{GridSearchCV}<br>clf.best_score_ - 0.921875<br>Validation Method – GridSearchCV (estimator=SVC()) |  |

# 8.2 User Acceptance Testing

### 1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Exploratory Analysis Of Rainfall Data In India For Agriculture project at the time of the release to User Acceptance Testing (UAT).

### 2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|---|---|---|---|---|---|
| By Design | 10 | 4 | 2 | 3 | 20 |
| Duplicate | 1 | 0 | 3 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 11 | 2 | 4 | 20 | 37 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 0 | 1 | 1 | 2 |
| Won't Fix | 0 | 5 | 2 | 1 | 8 |
| Totals | 24 | 14 | 13 | 26 | 77 |

### 3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| Print Engine | 7 | 0 | 0 | 7 |
| Client Application | 30 | 0 | 0 | 30 |
| Security | 2 | 0 | 0 | 2 |

# 9. RESULTS

# 9.1 Performance Metrics

Software quality is a measurement of something intangible, “how good” a software product really is. Some of the aspects of so ftware
quality taken are

a. Scalability

b. Speed

c. Stability

d. Reliability

e. Security
f. Maintainability and code quality

# LOAD TEST

| | |
|---|---|
| Scenario Name | Load Test –SmartFarmer - IoT Enabled Smart Farming Application |
| Scenario Type | Load Test – Duration 1 hour |
| Scenario Objective | To Simulate the peak load and to monitor the performance of the Website |
| Steps | The online load will be maintained at steady state |
| Entry Criteria | The online load will be maintained at steady state |
| Exit Criteria | The online load will be maintained at steady state |

# STRESS TEST

| | |
|---|---|
| Scenario Name | Stress Test - SmartFarmer - IoT Enabled Smart Farming Application |
| Scenario Type | Stress Test |
| Scenario Objective | Objective is to verify that the application can handle the projected growth and to discover the breaking point |
| Steps | Ramp up to 150% of peak volume and continuously increase load until breaking point |
| Entry Criteria | All the monitors are in place Test Data is set up Peak load test completed successfully |
| Exit Criteria | Test completion report is agreed upon as per expectation |

# ENDURANCE / SOAK TEST:

| | |
|---|---|
| Scenario Name | Soak Test - Exploratory Analysis of Rainfall Data in India for Agriculture |
| Scenario Type | Endurance – Duration 8 hours |
| Scenario Objective | To discover memory issues and bottlenecks that might occur under daily usage of the application |

| | |
|---|---|
| Steps | Steady state is maintained for 8 hours with half of the peak load |
| Entry Criteria | All the monitors are in place<br>Test Data is set up<br>Peak load test completed successfully |
| Exit Criteria | Test completion report is agreed upon as per expectation |

# 10.ADVANTAGE

- Easy to acess in rural areas.
- It helps to acces a sensor by using an single url.

# DISADAVATAGE

- Lack of internet
- Low investment
- Lack of connectivity in rural areas

# 11.CONCLUSION

IoT based SMART AGRICULTURE SYSTEM for Live Monitoring of Temperature and Soil Moisture and to control motor and light remotely has been proposed using Node Red and IBM Cloud Platform. The System has high efficiency and accuracy in fetching the live data of temperature and soil moisture. The IoT based smart farming System being proposed via this project 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 with more than 99% accurate results. Therefore, the project proposes a thought of consolidating the most recent innovation into the agrarian field to turn the customary techniques for water system to current strategies in this way making simple profitable and temperate trimming.

# 12.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 fullfledged Agriculture Precision ready product.

# 13.APPENDIX

## Source code

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected
```

```
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "y3vxbr"//IBM ORGANITION ID
#define DEVICE_TYPE "newdev"//Device type mentioned in ibm watson IOT
Platform
#define DEVICE_ID "newdevid"//Device ID mentioned in ibm watson IOT
Platform
#define TOKEN "oyYEZ1V9ZSDPcSDcj-"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT
command type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined
```

```
client id by passing parameter like server id,portand wificredential
void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  Serial.print("temperature:");
  Serial.println(t);
  Serial.print("humidity:");
  Serial.println(h);

  PublishData(t, h);
  delay(1000);
  if (!client.loop()) {
```

```
    mqttconnect();
  }
}
```

## Github link

**https://github.com/IBM-EPBL/IBM-Project-13824-1659532447**

**DEMO VIDEO LINK**

**https://drive.google.com/file/d/1MlrIanB1F_0sb21aLgjGKq5AwVE359YW/view?usp=share_link**