

PROJECT NAME	SmartFarmer - IoT Enabled Smart Farming Application
TEAM ID	PNT2022TMID18038
COLLEGE	SNS COLLEGE OF TECHNOLOGY, COIMBATORE
TEAM LEADER	Ranjith.E
TEAM MEMBERS	Vishnu Vardhan.R Vidhusham.S Sruti.L
FACULTY MENTOR NAME	B. Vinodhini
INDUSTRY MENTOR NAME	Bharadwaj

SMART FARMER - IOT ENABLED SMART FARMING APPLICATION

1.INTRODUCTION

1.1 Project Overview:

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. So after knowing about some IoT applications in agriculture, we can say that it is definitely revolutionize the agriculture industry in a few years. IoT has been applied in several areas of agriculture. The development of SmartFarmer - IoT Enabled Smart Farming Application is day by day turning the face of agriculture production by not only enhancing it but also making it cost-effective and reducing wastage. The aim / objective of this report is to propose SmartFarmer - IoT Enabled Smart Farming Application assisting farmers in getting Live Data (Temperature, Soil Moisture, Humidity) for efficient environment monitoring which will enable them to increase their overall yield and quality of products. The IoT based Smart Farming System

being proposed via this report is integrated with Connecting IoT devices to the Watson IoT platform and exchanging the sensor data. Configuring APIs using Node-RED for communicating with a mobile application. Creating a Mobile Application through which the user interacts with the IoT device.

1.2 Purpose:

IoT plays a key role in smart agriculture. Internets of Things (IoT) sensors are used to provide necessary information about agriculture fields. The main advantage of IoT is to monitor the agriculture by using the wireless sensor networks and collect the data from different sensors which are deployed at various nodes and send by wireless protocol. By using IoT system the smart agriculture is powered by NodeMCU. It includes the humidity sensor, temperature sensor, moisture sensor and DC motor. This system starts to check the humidity and moisture level. The sensors are used to sense the level of water and if the level is below the range then the system automatically starts watering. According to the change in temperature level the sensor does its job.

IoT also shows the information of

- Humidity,
- Moisture level,
- The temperature level based on type of crops cultivated can also be adjusted.

The traditional agriculture and allied sector cannot meet the requirements of modern agriculture which requires high-yield, high quality and efficient output. Thus, it is very important to turn towards modernization of existing methods and using the information technology and data over a certain period to predict the best possible productivity and crop suitable on the very particular land. The adoptions of access to high-speed internet, mobile devices, and reliable, low-cost satellites (for imagery and positioning) are few key technologies characterizing the precision agriculture trend.

IoT has been making deep inroads into sectors such as manufacturing, health-care and automotive. When it comes to food production, transport and storage, it offers a breadth of options that can improve India's per capita food availability. Sensors that offer information on soil nutrient status, pest infestation, moisture conditions etc. which can be used to improve crop yields over time.

2.LITERATURE SURVEY

2.1 Existing problem:

In this current present system agricultural land is controlled with few sensors and microcontrollers. The following sensors are soil moisture sensor, ultrasonic sensor. In the current existing system to soil moisture sensor will detect soil moisture content that is water content in the soil and turns on water motor. But there is no automatic control off water motor. In the existing system there is no use of PIR sensor for motion detection or detecting animals. There is no pesticides spray motor control remotely using cloud.

DISADVANTAGES OF EXISTING SYSTEM

- It is not a secure system.
- There is no motion detection for protection of agriculture field.
- Automation is not available.

2.2 References:

- [1] ISSN No:-2456-2165 Volume 4, Issue 2 Feb – 2019: "Solars' Energy: - A safe and reliable, eco-friendly and sustainable Clean Energy Option for Future India: - A Review." [2] Universal Paper of advanced science and science and exploration technology.
- [2] GRD Journals- Global Research and Development Journal for Engineering | Volume 4 | Issue 3 | February (2019) ISSN: 2455-5703 "Design and Implementation of an Advanced Security System for Farm Protection from Wild Animals".
- [3] International Journal of Innovations in Engineering and Science, Impact Factor Value 4.046 e-ISSN: 2456-3463 Vol.4, No. 5, 2019 "Solar Powered Smart Fencing System for Agriculture Protection using GSM & Wireless Camera".
- [4] International Journal of Management, Technology And Engineering ISSN NO : 2249-7455 Volume 8, Issue VII, JULY/2018 "Protecting Crops From Birds, Using Sound Technology In Agriculture"
- [5] American Journal of Engineering Research (AJER) 2018 eISSN: 2320-0847 p-ISSN : 2320- 0936 Volume-7, Issue-7, pp-326-330 "Moisture Sensing Automatic Plant Watering System Using Arduino Uno".

[6] International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455- 7137 Volume – 02, Issue – 05, May – 2017, "Solar Fencing Unit and Alarm for Animal Entry Prevention".

[7] D. K. Roy, and M. H. Ansari, " Smart Irrigation Control International Journal of Environmental Research and Development, vol. 4, no 4, pp. 371-374,2014.

[8] V. L. Akubattin, A. P. Bansode, T. Ambre, A. Kachroo, and P. SaiPrasad, "Smart Irrigation System, 'In International Journal of Scientific Research in Science and Technology, vol. 2, no 5, pp. 343- 345,2016.

[9] K.. R. Kakade, A. R. Pisal, A. V. Chavanss, and S. B. Khedkar, "Smart Irrigation and Crop Suggestion Using Raspberry-P i , International Journal of Scientific Research in Science and Technology, vol. 4, no 4, pp. 235-241, 2017.

[10] M. M. Kamal, N.A.Z.M. Noar, and A.M. Sabri, "Development of Detection and Flood Monitoring via Blynk Apps, 'Indonesian Journal of Electrical Engineering and Computer Science, vol. 10, no I, pp. 361-370,2018.

2.3 Problem Statement Definition:

This project, propose an IoT based Smart Farming Agriculture Stick assisting

farmers in getting live data of

Temperature, Humidity, Soil moisture, PH, etc. for efficient environment monitoring which will enable them

to do smart farming and improve their overall yield and quality of products.

The several factors which affect the amount of water required by crops in various climatic conditions are:

- 1) Temperature.
- 2) Humidity.
- 3) Sunshine.
- 4) Wind speed.

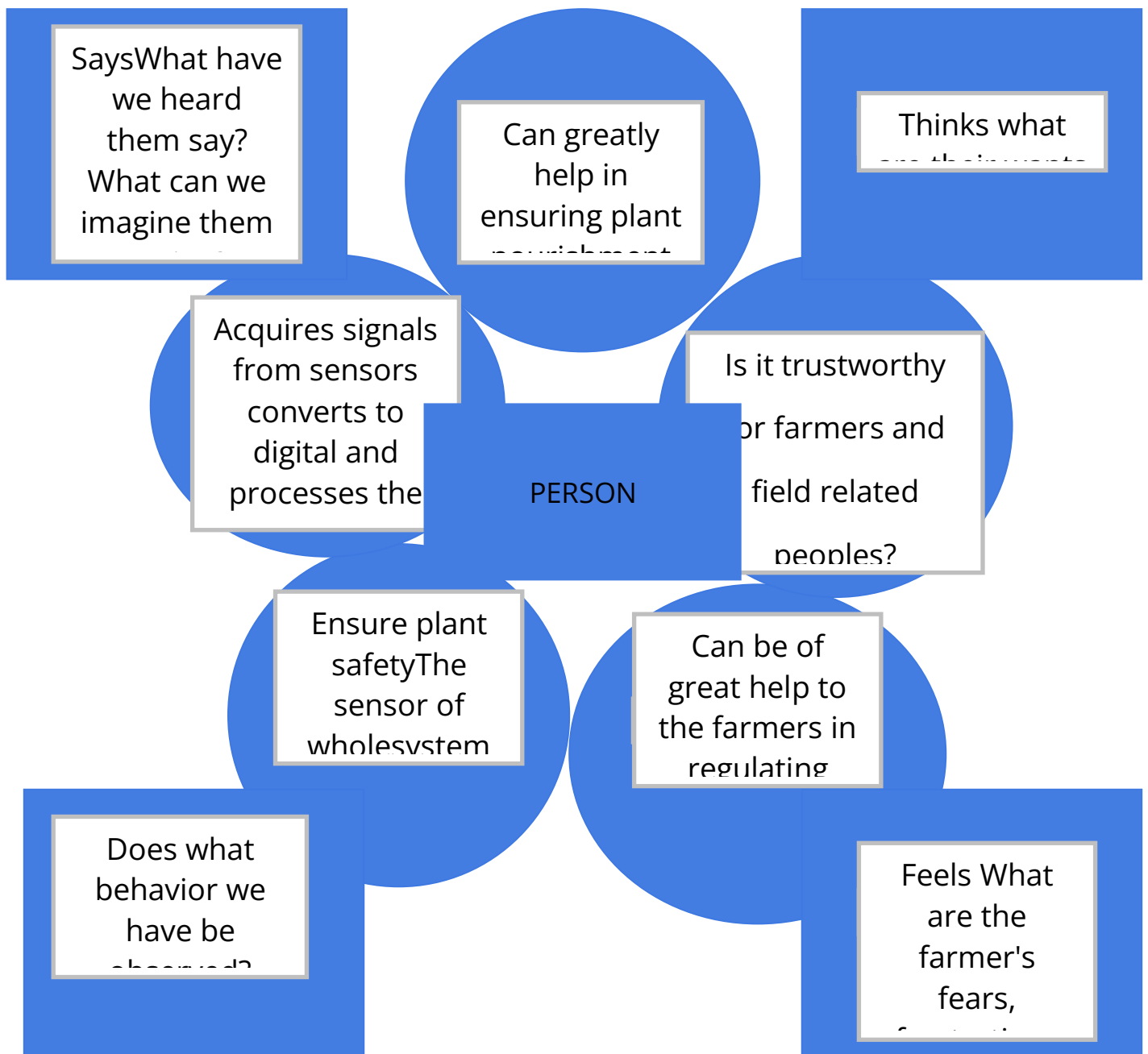
The key advantages of smart farming are:

- 1) Proper water management preventing the wastage of water.
- 2) Soil management for checking pH level and moisture in the soil.

- 3) Crop monitoring using cameras to detect infections and diseases in crop.
- 4) Weather monitoring for live monitoring and crop sowing time.


3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:



3.2 IDEATION AND BRAINSTORMING:

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Brainstorm & ideaprioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

⌚ 10 minutes to prepare
👥 1 hour to collaborate
👤 2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

⌚ 10 minutes

- Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
- Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.
- Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) →

Members of Ideation Process:

Team Leader: RANJITH E

Team Members: Vishnu Vardhan R
Sruji L
Vidhushan S

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

⌚ 5 minutes

PROBLEM

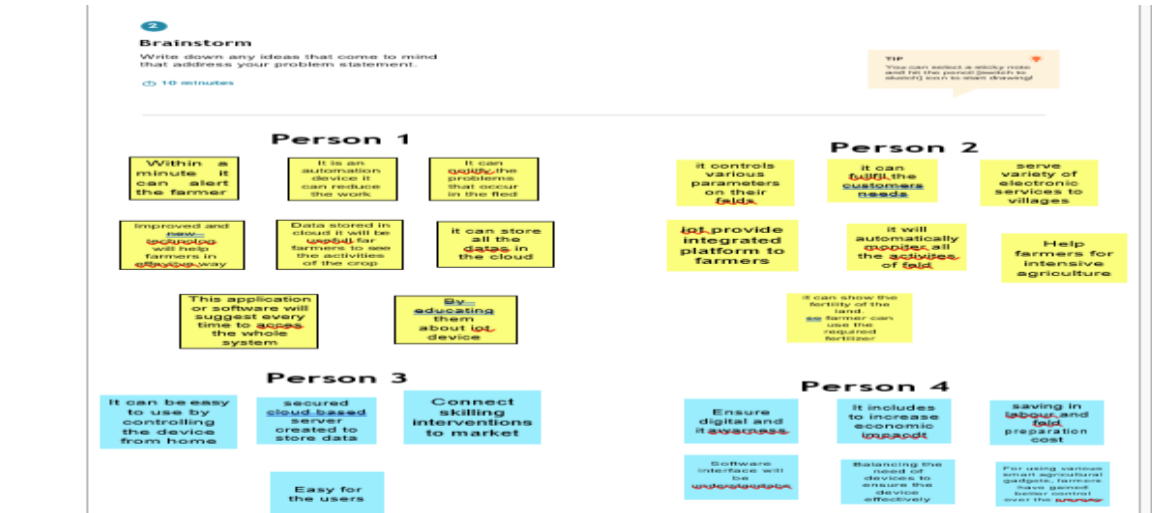
Most of the rural area people can't implement the IOT devices because of they don't know about the device uses and it is difficult to use it

Key rules of brainstorming

To run a smooth and productive session

Stay in topic.	Encourage wild ideas.
Defer judgment.	Listen to others.
Go for volume.	If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping



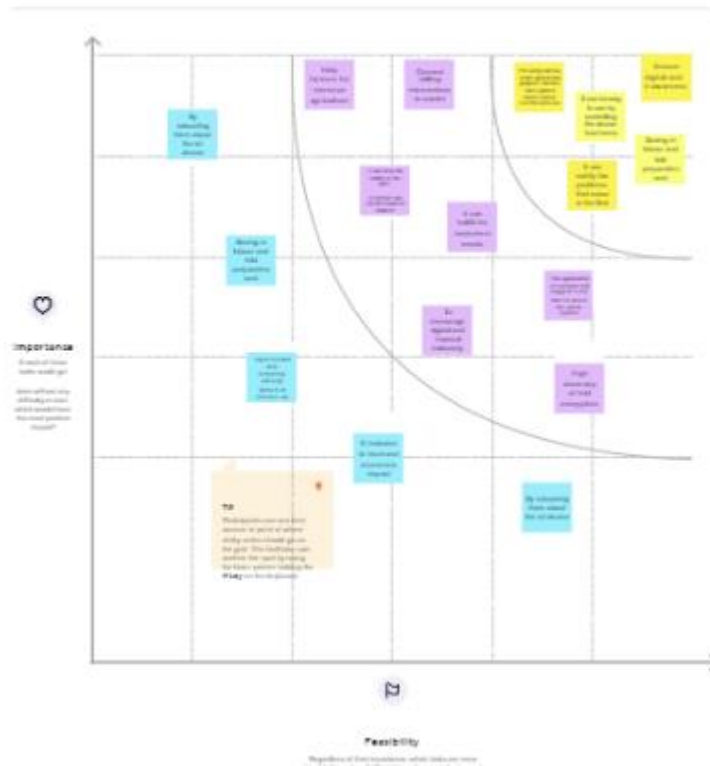
Step-3: Idea Prioritization



Prioritize

You learn about all be on the same page about what's important moving forward. Place your ideas on the grid to determine which ideas are important and which are feasible.

10 minutes



After you collaborate

You can export the mind as an image or pdf to share with members of your company who might find it helpful.

Quick actions

- Share the mind:** There is a clear line to the mind with a clear line to the mind. Share it with the mind about the mind about the mind.
- Export the mind:** Export a copy of the mind as a PDF or PNG to share with the mind about the mind about the mind.

Keep moving forward

- Strategy map:** Define the components of a strategy map. [Open the template](#)
- Customer experience journey map:** Understand customer needs, expectations, and behaviors for an experience. [Open the template](#)
- Energy, resources, opportunities & threats:** Identify strengths, weaknesses, opportunities, and threats (SWOT) for a plan. [Open the template](#)



More template options

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To make farming easier by choosing several constraints in agriculture and to overcome those constraints, to increase production quality and quantity using IOT.
2.	Idea / Solution description	Using smart techniques like monitoring farms climate, smart irrigation and soil analysis.
3.	Novelty / Uniqueness	Solar power smart irrigation system which helps you to monitor temperature, moisture ,humidity using smart sensors.
4.	Social Impact / Customer Satisfaction	It is better than the present modern irrigation system by using this method we can control soil erosion. There will be better production yield.
5.	Business Model (Revenue Model)	As the productivity increases customer satisfaction also increases and hence need for the application also increases, which increases the revenue of the business.
6.	Scalability of the Solution	It is definitely scalable we can increase the constraints when the problem arises.

3.4 Problem solution Fit:

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Who is your customer? i.e. working parents of 0-5 y.o. kids	6. CUSTOMER CONSTRAINTS CC What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.	5. AVAILABLE SOLUTIONS AS Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking	Explore AS, differentiate
	Farmers are our Customers.	The several constraints are availability of device, proper network connection and budget.	Most commonly used irrigation type is Drip irrigation the most common disadvantage is when the water is not filtered properly there will be clogs and the tubes will get affected easily. In smart farming we can use solar empowered smart irrigation system to overcome this	

Focus on J&P, lap into BE, understand RC	2. JOBS-TO-BE-DONE / PROBLEMS J&P Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.	9. PROBLEM ROOT CAUSE RC What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations.	7. BEHAVIOUR BE What does your customer do to address the problem and get the job done? i.e. Directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)	Focus on J&P, lap into BE, understand RC

3. TRIGGERS TR What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. To get correct accuracy on what to be done on the farm and to produce more crops and livestock quantitatively.	10. YOUR SOLUTION SL If you are working on an existing business, write down your current solution/first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. There will be less weed growth, Maximum use of water efficiently, Control of soil erosion and maximum crop yield	8. CHANNELS OF BEHAVIOUR CH 8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7 8.2 OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. we will reach the customer directly ask about their problems and provide effective solutions if their problems match our application and provide them knowledge about our application to make their farming even more easier In online mode will do digital marketing using advertisements.
4. EMOTIONS: BEFORE / AFTER EM How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design. As when the productivity increases farmers will be satisfied. They will not worry about the loss irrigation will be more efficient than before.		Identify strong TR & EM

4.REQUIREMENT ANALYSIS

4.1 Functional Requirements:

Following are the functional requirements of the proposed solution

FR NO.	Functional Requirement (Epic)	Sub Requirement(Story /Sub - Task)
FR - 1	User Registration/Login	Via Email Via Phone number
FR - 2	User Dashboard	Single Sample Prediction Multiple Sample Prediction View User History
FR - 3	Output Dashboard	Visual Representation Report Generation

4.2 Non-Functional requirements:

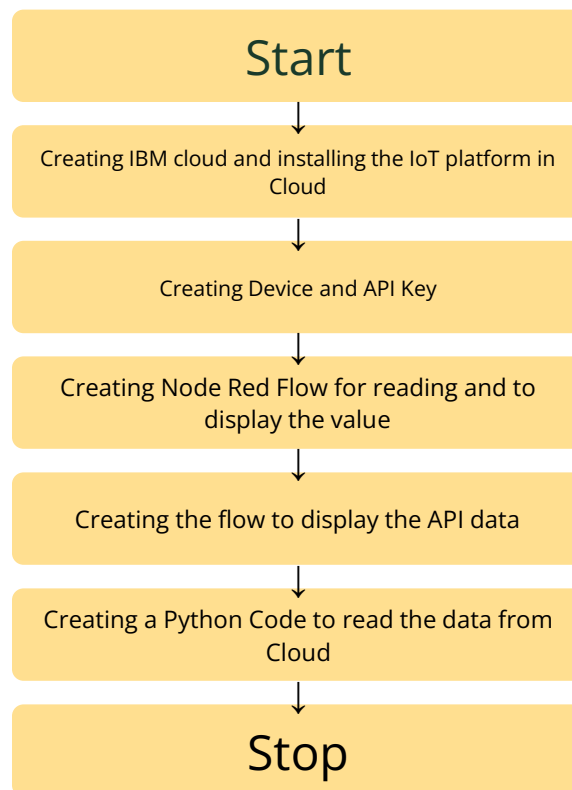
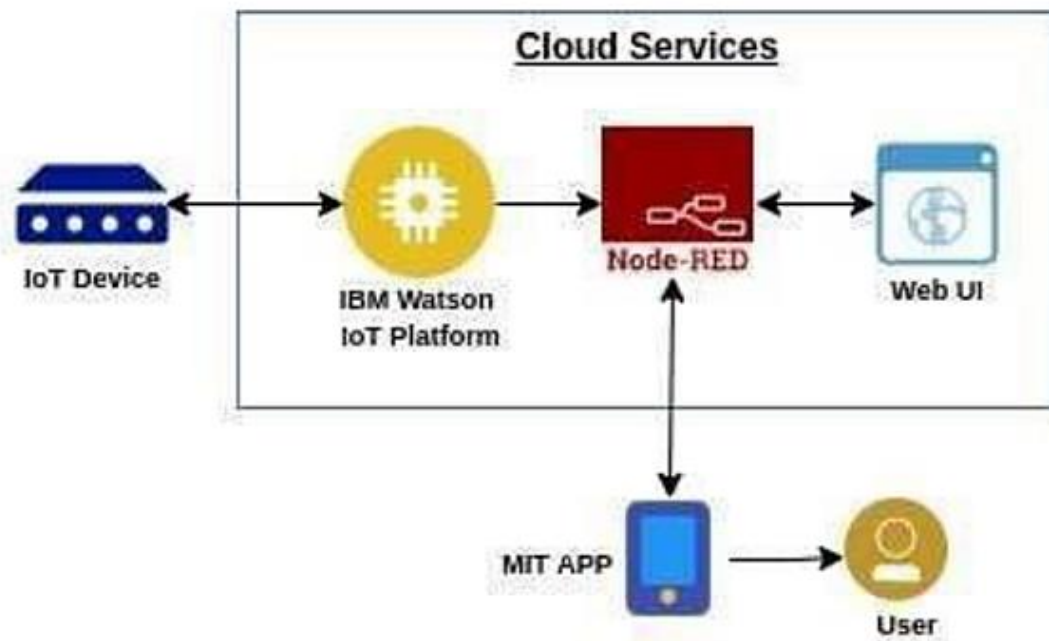
Following are the non-functional requirements of the proposed solution

FR NO.	Non-Functional Requirements	Description
NFR-1	Usability	A user-friendly interface that makes processing easier for the user Prediction are visually represented by the model.
NFR-2	Security	User authentication: To have secured access,the user might have a private dashboard.

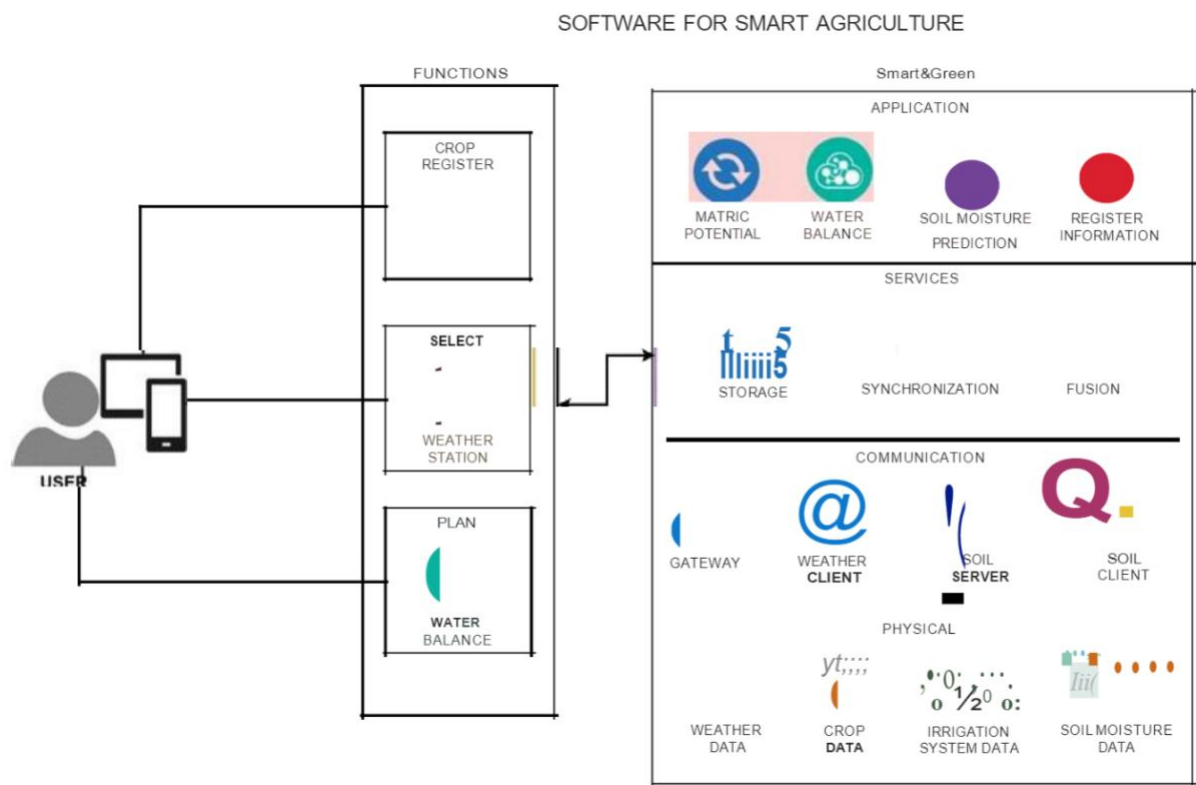
NFR-3	Reliability	The mode is capable should be able to handle massive amounts of data and run several samples at once.
NFR-4	Performance	The model's accuracy is good because it combines several ML methods.
NFR-5	Availability	The website is portable and resposive to mobile devices. To run on any device,only the most minimal configuration are needed

5.PROJECT DESIGN

5.1 Data Flow Diagrams:



5.2 Solution and Technical Architecture



5.3 User Stories:

User Type	Functional Requirements	User Story Number	User Story/Task	Acceptance Criteria	Priority	Release
Customer	Code	USN 1	Python code		High	Sprint 1

Customer	Software	USN 2	IBM Watson IOT platform,Workflows for IOT scenarious using Node- red		High	Sprint 2
Customer	MIP app	USN 3	To develop an application using MIT		High	Sprint 3
Customer	Web UI	USN 4	To make the user to interact with the software	User can access the app for the services	High	Sprint 4

6.PROJECT PLANNING & SCHEDULING

6.1Sprint Planning & Estimation:

Product Backlog,Sprint Schedule, and Estimation

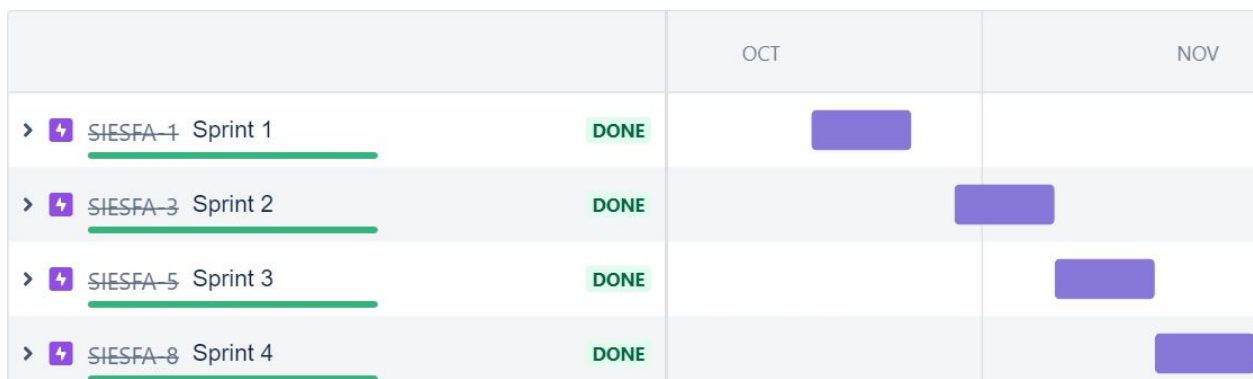
Sprint	Functional Requireme nt	User Story Number	User Story / Task	Story Poin ts	Prioirit y	Team Members
Sprint -1	Code	USN - 1	Developing a Python Code	2	High	Ranjith E VishnuVar dahn R

Sprint - 2	Software	USN - 2	Creating device in the IBM Watson IOT scenarious using Node-Red	2	High	Ranjith E VishnuVar dhan R Sruti L
Sprint - 3	MIT App Inventor	USN - 3	Develop an application for the Smart farmer project using MIT App InvSoentor	2	High	Sruti L
Sprint - 3	Dashboard	USN - 3	Design the Modules and test the app	2	High	Vidhusha m S
Sprint - 4	Web UI	USN - 4	To make the user to interact with software	2	High	Ranjith E Vishnu Vardhan R Sruti L Vidhusha m S

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date (Planned)	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	4 Days	14 Nov 2022	17 Nov 2022	20	17 Nov 2022

6.3 REPORT FROM JIRA:



7.CODING & SOLUTIONING

7.1 Feature 1:

Python Code

* Its for "Remote ON and OFF"

* Identifying Soil Moisture Temperature and Humidity

```
import wiotp.sdk.device
import time
import os
import random
myconfig = {
    "identity": {
        "orgId": "3Inltf",
        "typeId": "NodeMCU",
        "deviceId": "12345"
    },
    "auth": {
        "token": "12345678"
    }
}
client = wiotp.sdk.device.DeviceClient(config=myconfig, logHandlers=None)
client.connect()

def myCommandCallback(cmd):
    print("Messure recived from IBM ToT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']
```

```

if(m=="Motor is switched on"):
    print("Motoris switched on")
elif(m=="motoroff"):
    print("Motor is switched OFF")
    print("")
while True:
    soil=random.randint(22, 100)
    temp=random.randint(-20, 125)
    hum=random.randint(0, 100)
    myData={'soil_moisture':soil, 'temperature':temp,'humidity':hum}
    client.publishEvent(eventId="status",msgFormat="json",
data=myData,qos=0, onPublish=None )
    print ("Published data Successfully: %s",myData)
    time.sleep(2)
    client.commandCallback= myCommandCallback
client.disconnect()

```

7.2 Feature 2:

For MIT App Inventor

```

<!DOCTYPE html>

<!--                                saved                                from
url=(0048)http://ai2.appinventor.mit.edu/#5592391764279296 -->
<html style="overflow: auto;"><head><meta http-equiv="Content-Type"
content="text/html; charset=UTF-8"><style>HTML{margin:0
!important;border:none !important;}.dragdrop-handle{cursor:move;user-
select:none;-khtml-user-select:none;-moz-user-select:none;}.dragdrop-
draggable{zoom:1;}.dragdrop-dragging{zoom:normal;}.dragdrop-

```

```
positioner{border:1px dashed #1e90ff;margin:0 !important;zoom:1;z-index:100;}.dragdrop-flow-panel-positioner{color:#1e90ff;display:inline;text-align:center;vertical-align:middle;}.dragdrop-proxy{background-color:#7af;}.dragdrop-selected,.dragdrop-dragging,.dragdrop-proxy{filter:alpha(opacity \= 30);opacity:0.3;}.dragdrop-movable-panel{z-index:200;margin:0 !important;border:none !important;}</style>
```

```
<meta http-equiv="X-UA-Compatible" content="IE=10">
```

```
<!--meta name="gtwt:property" content="locale=en_US"-->
```

```
<!-- Title is set at runtime. -->
```

```
<title>MIT App Inventor</title>
```

```
<!-- Google Analytics. -->
```

```
<script type="text/javascript" async="" src="./MIT App Inventor_files/ga.js.download"></script><script type="text/javascript">
```

```
var _gaq = _gaq || [];
```

```
_gaq.push(['_setAccount', 'UA-28621056-1']);
```

```
_gaq.push(['_setDomainName', 'ai2.appinventor.mit.edu']);
```

```
_gaq.push(['_setAllowLinker', true]);
```

```
_gaq.push(['_trackPageview']);
```

```
(function() {
```

```
var ga = document.createElement('script'); ga.type = 'text/javascript'; ga.async = true;
```

```
ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') + '.google-analytics.com/ga.js';
```

```
var s = document.getElementsByTagName('script')[0]; s.parentNode.insertBefore(ga, s);
```

```
})();
```

```
</script>
```

<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/gwt.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/blockly.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/ai2blockly.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/dialog.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/hsvpalette.css">				
<link type="text/css" rel="stylesheet" href="./MIT App Inventor_files/font-awesome.min.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/leaflet.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/leaflet.toolbar.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/leaflet-vector-markers.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/Ya.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/android_holo.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/android_material.css">				
<link	type="text/css"	rel="stylesheet"	href="./MIT	App
Inventor_files/iOS.css">				

```

<link      type="text/css"      rel="stylesheet"      href="./MIT      App
Inventor_files/DarkTheme.css">
<noscript>
  <div class="floatingBox">
    <h2> App Inventor needs JavaScript enabled to run.</h2>
  </div>
</noscript>
<script      src="./MIT      App
Inventor_files/07550E3C801A777506EB0AD54C97601D.cache.js.downloa
d"></script><style type="text/css">/* Chart.js */
@keyframes      chartjs-render-
animation{from{opacity:.99}to{opacity:1}}.chartjs-render-
monitor{animation:chartjs-render-animation      1ms}.chartjs-size-
monitor,.chartjs-size-monitor-expand,.chartjs-size-monitor-
shrink{position:absolute;direction:ltr;left:0;top:0;right:0;bottom:0;overflow:hi
dden;pointer-events:none;visibility:hidden;z-index:-1}.chartjs-size-monitor-
expand>div{position:absolute;width:1000000px;height:1000000px;left:0;top
:0}.chartjs-size-monitor-
shrink>div{position:absolute;width:200%;height:200%;left:0;top:0}</style><s
tyle>.blocklyDraggable {}
.blocklySvg {
  Light {
    display: none;
  }
  .badBlock>.blocklyPath {
    stroke-width: 3px;
    stroke: #f00;

```

```
}  
.badBlock>.blocklyPathLight {  
  display: none;  
}  
.blocklyDragging>.blocklyPath,  
.blocklyDragging>.blocklyPathLight {  
  fill-opacity: .8;  
  stroke-opacity: .8;  
}  
.blocklyDragging>.blocklyPathDark {  
  display: none;  
}  
.blocklyDisabled>.blocklyPath {  
  fill-opacity: .5;  
  stroke-opacity: .5;  
}  
.blocklyDisabled>.blocklyPathLight,  
.blocklyDisabled>.blocklyPathDark {  
  display: none;  
}  
.blocklyText {  
  cursor: default;  
  fill: #fff;  
  font-family: sans-serif;  
  font-size: 11pt;  
}  
.blocklyNonEditableText>text {
```

```
pointer-events: none;
}
.blocklyNonEditableText>rect,
.blocklyEditableText>rect {
  fill: #fff;
  fill-opacity: .6;
}
.blocklyNonEditableText>text,
.blocklyEditableText>text {
  fill: #000;
}
.blocklyEditableText: hover>rect {
  stroke: #fff;
  stroke-width: 2;
}
.blocklyBubbleText {
  fill: #000;
}
.blocklyFlyout {
  position: absolute;
  z-index: 20;
}
.blocklyFlyoutButton {
  fill: #888;
  cursor: default;
}
.blocklyFlyoutButtonShadow {
```



```
fill: #666;
}
.blocklyFlyoutButton:hover {
fill: #aaa;
}
.blocklyFlyoutLabel {
cursor: default;
}
.blocklyFlyoutLabelBackground {
opacit
```

```
background: #9ab;
font-weight: bold !important;
border-color: #246 #9bd #9bd #246;
color: #fff;
}</style></head>
```

```
<!-- ODE scripts -->
```

```
<body class="gwt-bodyRob">
```

```
<div class="floatingBox" style="display:none" id="unsupported">
```

```
<h2> Your browser might not be compatible. </h2>
```

To use App Inventor for Android, you must use a compatible browser.

Currently the supported browsers are:

```
<ul>
```

```
<li> Google Chrome 29+ </li>
```

```
<li> Safari 6.1+ </li>
```

```
<li> Firefox 23+ </li>
```

```

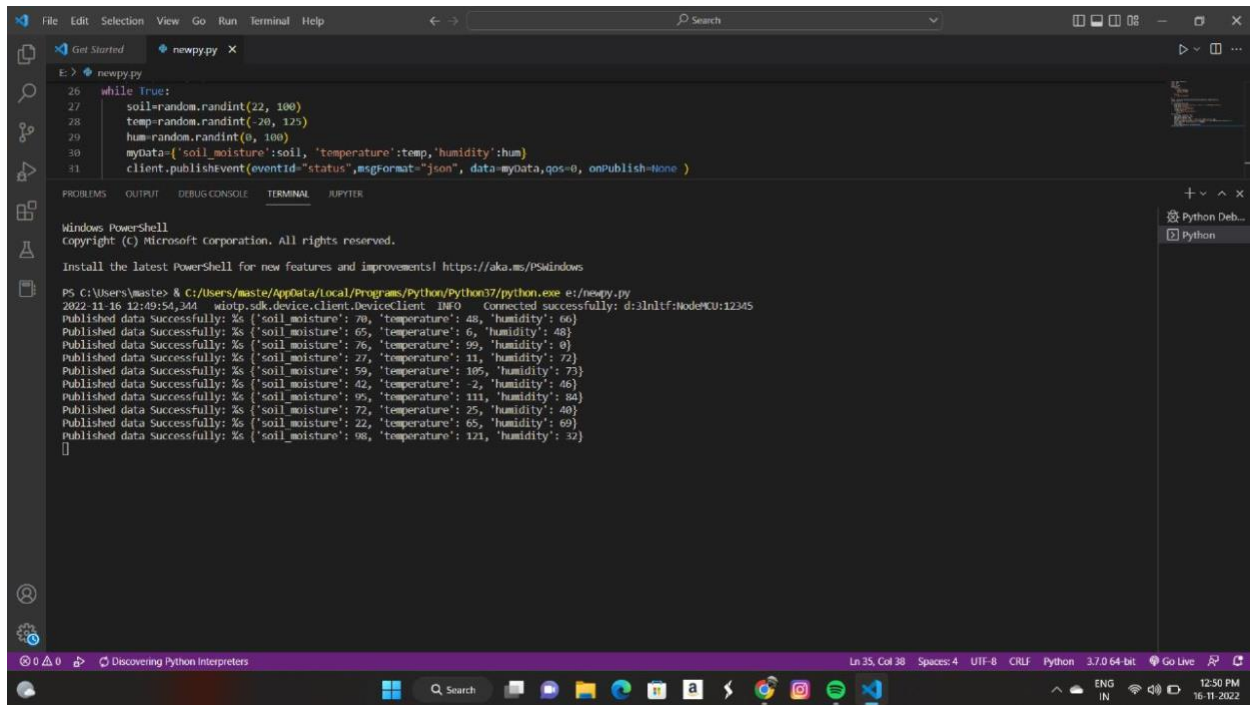
    </ul>
</div>
<script      type="text/javascript"      src="./MIT      App
Inventor_files/base.js.download"></script><script      src="./MIT      App
Inventor_files/deps.js.download"
onload="goog.Dependency.callback_(&#39;0.hoi9llmf0ng&#39;;      this)"
type="text/javascript"></script>
<script      type="text/javascript"      src="./MIT      App
Inventor_files/ode.nocache.js.download"></script>
<script src="./MIT App Inventor_files/leaflet.js.download"></script>
<script      src="./MIT      App
Inventor_files/leaflet.toolbar.js.download"></script>
<script      src="./MIT      App      Inventor_files/leaflet-vector-
markers.min.js.download"></script>
<script      src="./MIT      App      Inventor_files/leaflet-
imgicon.js.download"></script>
<script src="./MIT App Inventor_files/Path.Drag.js.download"></script>
<script      src="./MIT      App
Inventor_files/Leaflet.Editable.js.download"></script>
<script      src="./MIT      App
Inventor_files/leaflet.geometryutil.js.download"></script>
<script src="./MIT App Inventor_files/leaflet.snap.js.download"></script>
<script>
  if (window.navigator.userAgent.indexOf("MSIE") != -1){
    document.getElementById("unsupported").style.display = 'block';
  }
</script>

```

8.TESTING

8.1 Test Cases:

Command received from Node-Red:



```
E:\> newppy

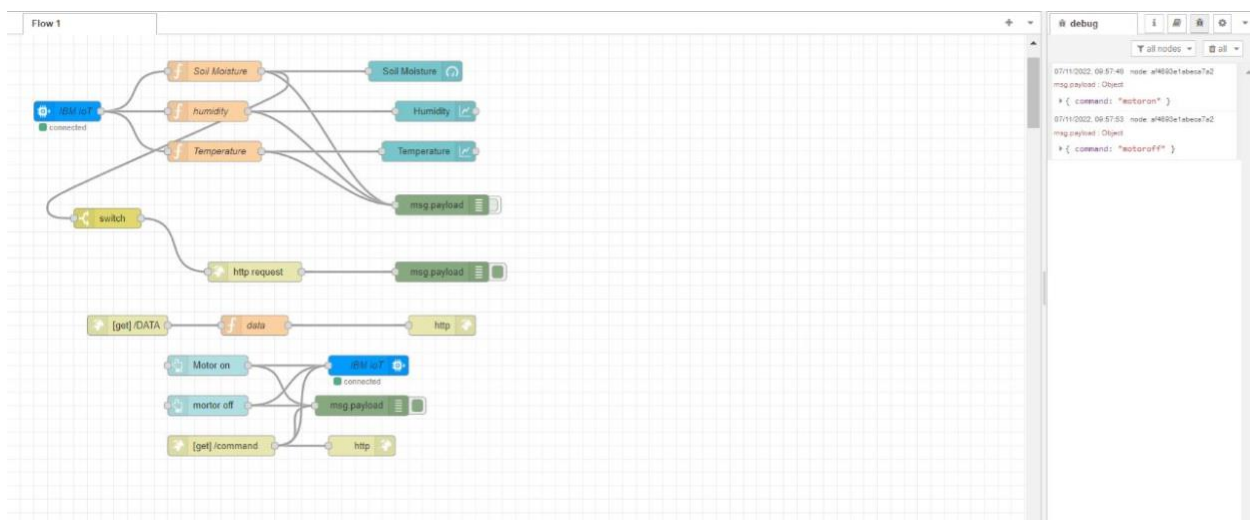
26 while True:
27     soil=random.randint(22, 100)
28     temp=random.randint(-20, 125)
29     hum=random.randint(0, 100)
30     myData={'soil_moisture':soil, 'temperature':temp, 'humidity':hum}
31     client.publish(eventId="status",msgformat="json", data=myData,qos=0, onPublish=None )

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

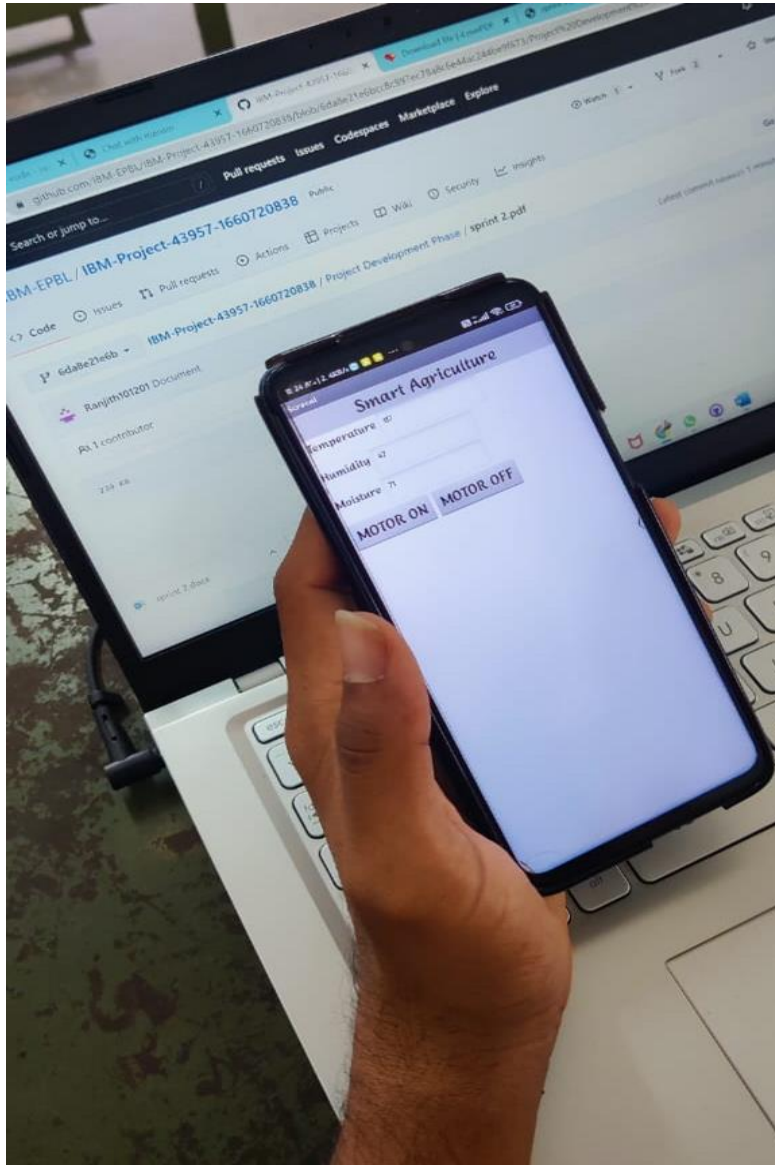
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\maste> & C:\Users\maste\AppData\Local\Programs\Python\Python37\python.exe e:/newppy.py
2022-11-16 12:49:54,344 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:3InIcf:NodeMCU:12345
Published data Successfully: %s {'soil_moisture': 70, 'temperature': 48, 'humidity': 66}
Published data Successfully: %s {'soil_moisture': 65, 'temperature': 6, 'humidity': 48}
Published data Successfully: %s {'soil_moisture': 76, 'temperature': 99, 'humidity': 0}
Published data Successfully: %s {'soil_moisture': 77, 'temperature': 11, 'humidity': 72}
Published data Successfully: %s {'soil_moisture': 59, 'temperature': 105, 'humidity': 73}
Published data Successfully: %s {'soil_moisture': 42, 'temperature': -2, 'humidity': 46}
Published data Successfully: %s {'soil_moisture': 99, 'temperature': 111, 'humidity': 84}
Published data Successfully: %s {'soil_moisture': 72, 'temperature': 25, 'humidity': 40}
Published data Successfully: %s {'soil_moisture': 22, 'temperature': 65, 'humidity': 69}
Published data Successfully: %s {'soil_moisture': 98, 'temperature': 121, 'humidity': 32}
[]
```



8.2 User Acceptance Testing: Using the App:



10:41



Search for apps on this device



redBus



Rapido



RAR



Recorder



Red Taxi



Rewards



slice



Settings



ShareMe



Sheets



Slide



Smart_Agriculture



Snapchat



Snapseed



SonyLIV



Spotify



Status, Sticker Saver



Steam



Swayam



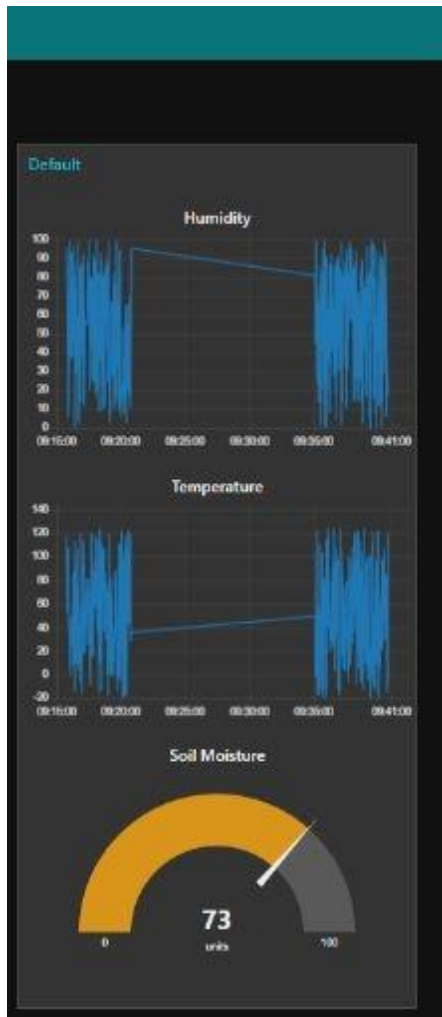
SYSTech

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y

9.RESULTS

9.1 Performance Testing:

From this we getting the values of Soil Moisture , Temperature , Humidity



10.ADVANTAGES & DISADVANTAGES

Advantages:

Excelled efficiency:

Today's agriculture is in a race. Farmers have to grow more products in deteriorating soil, declining land availability and increasing weather fluctuation. IoT-enabled agriculture allows farmers to monitor their product and conditions in real-time. They get insights fast, can predict issues before they happen and make informed decisions on how to avoid them. Additionally, IoT solutions in agriculture introduce automation, for example, demand-based irrigation, fertilizing and robot harvesting.

Expansion.:

By the time we have 9 billion people on the planet, 70% of them will live in urban areas. IoT-based greenhouses and hydroponic systems enable short food supply chains and should be able to feed the people. Smart closed-cycle agricultural systems allow growing food basically everywhere—in supermarkets, on skyscrapers' walls and rooftops, in shipping containers and, of course, in the comfort of everyone's home.

Reduced resources:

Plenty of ag IoT solutions are focused on optimizing the use of resources—water, energy, land. Precision farming using IoT relies on the data collected

from diverse sensors in the field which helps farmers accurately allocate just enough resources to within one plant.

Cleaner process:

Not only do IoT-based systems for precision farming help producers save water and energy and, thus, make farming greener, but also significantly scale down on the use of pesticides and fertilizer. This approach allows getting a cleaner and more organic final product compared to traditional agricultural methods.

Agility:

One of the benefits of using IoT in agriculture is the increased agility of the processes. Thanks to real-time monitoring and prediction systems, farmers can quickly respond to any significant change in weather, humidity, air quality as well as the health of each crop or soil in the field. In the conditions of extreme weather changes, new capabilities help agriculture professionals save the crops.

Improved product quality:

Data-driven agriculture helps both grow more and better products. Using soil and crop sensors, aerial drone monitoring and farm mapping, farmers better understand detailed dependencies between the conditions and the

quality of the crops. Using connected systems, they can recreate the best conditions and increase the nutritional value of the products.

Disadvantages:

- 1.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.
- 2.The smart farming based equipments 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.

11.CONCLUSION

Thus, the smart farming will revolutionize the world of farming and it will increase the productivity as well as improve the quality and can save lives of farmer. There is an urgent need for a system that makes the agricultural process easier and burden free from the farmer's side. With the recent advancement of technology, it has become necessary to increase the annual crop production output of our country India, an entirely Agro centric economy. The ability to conserve the natural resources as well as giving a splendid boost to the production of the crops in one of the main aims of incorporating such technology into the agricultural domain of the country. To save farmer's effort, water and time has been the most important consideration. This will also enable farmers to use IoT technology and they will be able to implement other smart farming techniques in their land to increase yield. This capstone project gave me the chance to learn new technologies and work with new tools, this was a real proof that AUI has taught us to be long-life learners and to master self-learning before teaching us other class materials. Of course, this project is a combination of what I learned from all my computer science classes, the programming languages, the database systems and the engineering process that is important in any engineering project, all together with what I learned from other disciplines and also by myself about IoT and the use of Arduino helped me to build an embedded system. In general, the project was successful and worked properly and succeeded in delivering the

prototype on due time. I am proud and happy for this achievement especially that this is my first real big theoretical and practical project. It enabled me to get concrete results and to realize that I can indeed build products that would be beneficial in real life and that I can customized upon demand as future projects.

12.FUTURE SCOPE

- In the current project we have implemented the project that can protect and maintain the the crop. In this project the farmer monitor and control the field remotely. In future we can add or update few more things to this project.
- We can create few more models of the same project ,so that the farmer can have information of a entire.
- We can update the this project by using solar power mechanism. So that the power supply from electric poles can be replaced with solar panels.
- It reduces the power line cost.
- It will be a one time investment. We can add solar fencing technology to this project.
- We can add camera feature so that the farmer can monitor his field in real time. This helps in avoiding thefts.

13.APPENDIX

Source Code:

```
import wiotp.sdk.device
import time
import os
import random

myconfig = {
    "identity": {
        "orgId":"3InItf",
        "typeId":"NodeMCU",
        "deviceId":"12345"
    },
    "auth":{
        "token":"12345678"
    }
}

client = wiotp.sdk.device.DeviceClient(config=myconfig, logHandlers=None)
client.connect()

def myCommandCallback(cmd):
    print("Messure recived from IBM ToT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']
    if(m=="Motor is switched on"):
        print("Motoris switched on")
    elif(m=="motoroff"):
        print("Motor is switched OFF")
```

```
    print("")
while True:
    soil=random.randint(22, 100)
    temp=random.randint(-20, 125)
    hum=random.randint(0, 100)
    myData={'soil_moisture':soil, 'temperature':temp,'humidity':hum}
    client.publishEvent(eventId="status",msgFormat="json",
data=myData,qos=0, onPublish=None )
    print ("Published data Successfully: %s",myData)
    time.sleep(2)
    client.commandCallback= myCommandCallback
client.disconnect()
```

GITHUB LINK :

<https://github.com/IBM-EPBL/IBM-Project-43957-1660720838>

DEMO LINK:

<https://youtu.be/6nuuPNKVe3Y>