Report of

Smart Farmer – IOT ENABLED

SMART FARMING APPLICATION

TEAM ID : PNT2022TMID52019

CONTENTS

1.Introduction:

1.1.Project Overview :

Today, everyone is familiar with mobile devices. It also includes the farmers and country side people. Mobile plays vital role in daily life of farmers as well as other people. The farmers, who were dependent on clouds for rain, now are looking into the Cloud Computing for their solutions towards cultivation of superior crops in today’s modern agricultural world. The traditional methods used by the farmers, peculiarly in India, are very slow and understandable. Although most people can see the benefits of using more precious approach to manage crops with additional information, the tools provided for farming and other information technologies have not yet moved into mainstream agriculture management. This will effectively help farmers to sell their product in global market and earn remarkable profit. Hence, this frame work uses MC, which in effect, puts power into a farmer’s hand. The experimental setup uses tools like Android SDK, IoT Watson Platform Server etc

1.2.Purpose:

In India most of the population is dependent on agriculture. However, there is also need to review and revitalize the mechanism for updating the technology. In the upcoming years agriculture will see major changes. Unlike the earlier 'green revolution' which had a foundation of advanced pesticides and fertilizers, now the agriculture will be revolutionized with the help of technology. Every developing economy has agriculture sector as irreplaceable pillar and so does India. In India the agriculture sector contributes close to 20% of GDP (Gross Domestic Product). Either directly or indirectly, 60% of total population of India depends on agriculture.

The vast majority of Indian farmers, which includes small scale producers, are often unable to access the information and technological resources that could increase the yield and lead to better prices for their crops and products. The wide spread network of mobile phones could be the game changer in this problem. It will put agriculture field to its zenith such project is to develop a mobile phone based solution that helps in farm management, leads to agricultural yield improvement and helps in care/maintenance of the farms. Smart farming methods increase the production in almost every sector. Using modern agriculture and farming is a must because modern farming, methods can increase production and can feed the world. The economy of some country is depend upon agriculture and farming related business. A major part of the population is directly or indirectly involved with agriculture and farming business. Income source of people are limited.

2.1.Literature Survey:

2.1.Existing Problem:

- The IoT related equipment allows the farmer to understand the use of technology and to learn.
  Given any security measures, the system offers little power and can lead to various kinds of network attacks.
- Most of the high yielding seeds are for irrigated lands but 4% of our sown area is dry. Nearly 90% farmers are small and Marginal.
- The modules used here has to withstand several climate changes.the Wi-Fi module used here needs internet connection which is hard to implement in large agricultural fields. Shock might occur due to water as it is a good conductor.
- The farmers need to understand the technology and learn it.
- Interference may be caused by different network systems such as same spectrum based such as zigbee, sigfox.
- Increased privacy concerns.
- Increased unemployment rates.
- Highly dependent on the internet.
- Lack of mental and physical activity by humans and leading to health issues.

- Complex system for maintenance.
- Lack of security.Absence of international standards for better communication.

2.2.References:

- Internet of Things (IoT), Agriculture, Soil moisture sensor, Arduino-UNO ATmega328p, IR sensor, Smart farming.
- Smart Agriculture System using IoT Technology, September 2020 by Adithya Vadapalli,Swapna Peravali & Venkata Rao Dadi.
- International journal of Research in science & Technology, January to March 2021 by Garigipati Vijay Kumar.
- Internet of Things (IoT), Internet of Lighting, Fertilization, Agriculture, Remote, Communication V Dankan Gowda et al 2021.
- , IoT based Agriculture (IoTA): Architecture, Cyber Attack, Cyber Crime and DigitalForensics Challenges. 2022, Research Square.
- IoT for Smart Precision Agricultural and Farming in Rural areas - Nuzaman Ahamed-2018.

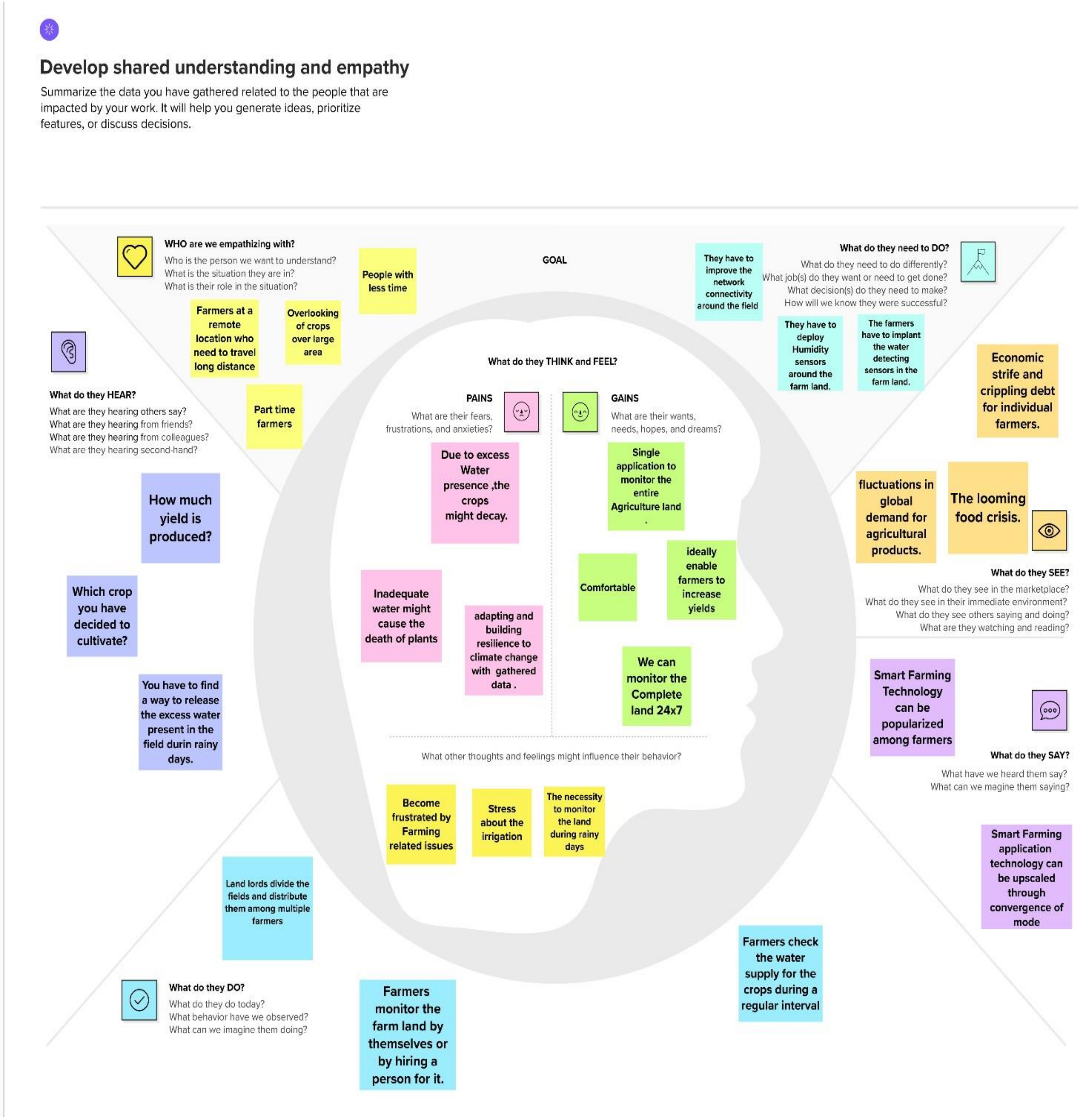
2.3.Problem Statement Definition:

- To provide efficient decision support system using wireless sensor natural which handle different activities of farm and gives useful information related to agriculture soil moisture, temperature, soil humidity control. The previous proposed systems have a drawback of network issues. Which causes delays in many operations.
- There is a problem of excess water supply or lack of water supply which makes the crops die. Because for rice, sugarcane, coconut crops require more water for the cultivation but in case of crops like pumpkin, ladies finger, carrot require water in drops so the requirement of water depends on the crops.
- There is a possibility of hackers to control the water supply by intruding into the server.
- After rain, there is no automated facility to alert the farmers about the presence of excess water in the field.

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map canvas is a more in-depth version of the original empathy map, which helps identify and describe user’s needs and pain points. And this is valuable information for improving the user experience.



Develop shared understanding and empathy

Summarize the data you have gathered related to the people that are impacted by your work. It will help you generate ideas, prioritize features, or discuss decisions.

WHO are we empathizing with?

Who is the person we want to understand?
What is the situation they are in?
What is their role in the situation?

- People with less time
- Farmers at a remote location who need to travel long distance
- Overlooking of crops over large area
- Part time farmers

GOAL

What do they need to DO?

What do they need to do differently?
What job(s) do they want or need to get done?
What decision(s) do they need to make?
How will we know they were successful?

- They have to improve the network connectivity around the field
- They have to deploy Humidity sensors around the farm land.
- The farmers have to implant the water detecting sensors in the farm land.

What do they HEAR?

What are they hearing others say?
What are they hearing from friends?
What are they hearing from colleagues?
What are they hearing second-hand?

- How much yield is produced?
- Which crop you have decided to cultivate?
- You have to find a way to release the excess water present in the field durin rainy days.

What do they THINK and FEEL?

PAINS

What are their fears, frustrations, and anxieties?

- Due to excess Water presence ,the crops might decay.
- Inadequate water might cause the death of plants
- adapting and building resilience to climate change with gathered data .

GAINS

What are their wants, needs, hopes, and dreams?

- Single application to monitor the entire Agriculture land .
- Comfortable
- ideally enable farmers to increase yields
- We can monitor the Complete land 24x7

What other thoughts and feelings might influence their behavior?

- Become frustrated by Farming related issues
- Stress about the irrigation
- The necessity to monitor the land during rainy days

What do they SEE?

What do they see in the marketplace?
What do they see in their immediate environment?
What do they see others saying and doing?
What are they watching and reading?

- Economic strife and crippling debt for individual farmers.
- fluctuations in global demand for agricultural products.
- The looming food crisis.

What do they SAY?

What have we heard them say?
What can we magine them saying?

- Smart Farming Technology can be popularized among farmers
- Smart Farming application technology can be upscaled through convergence of mode

What do they DO?

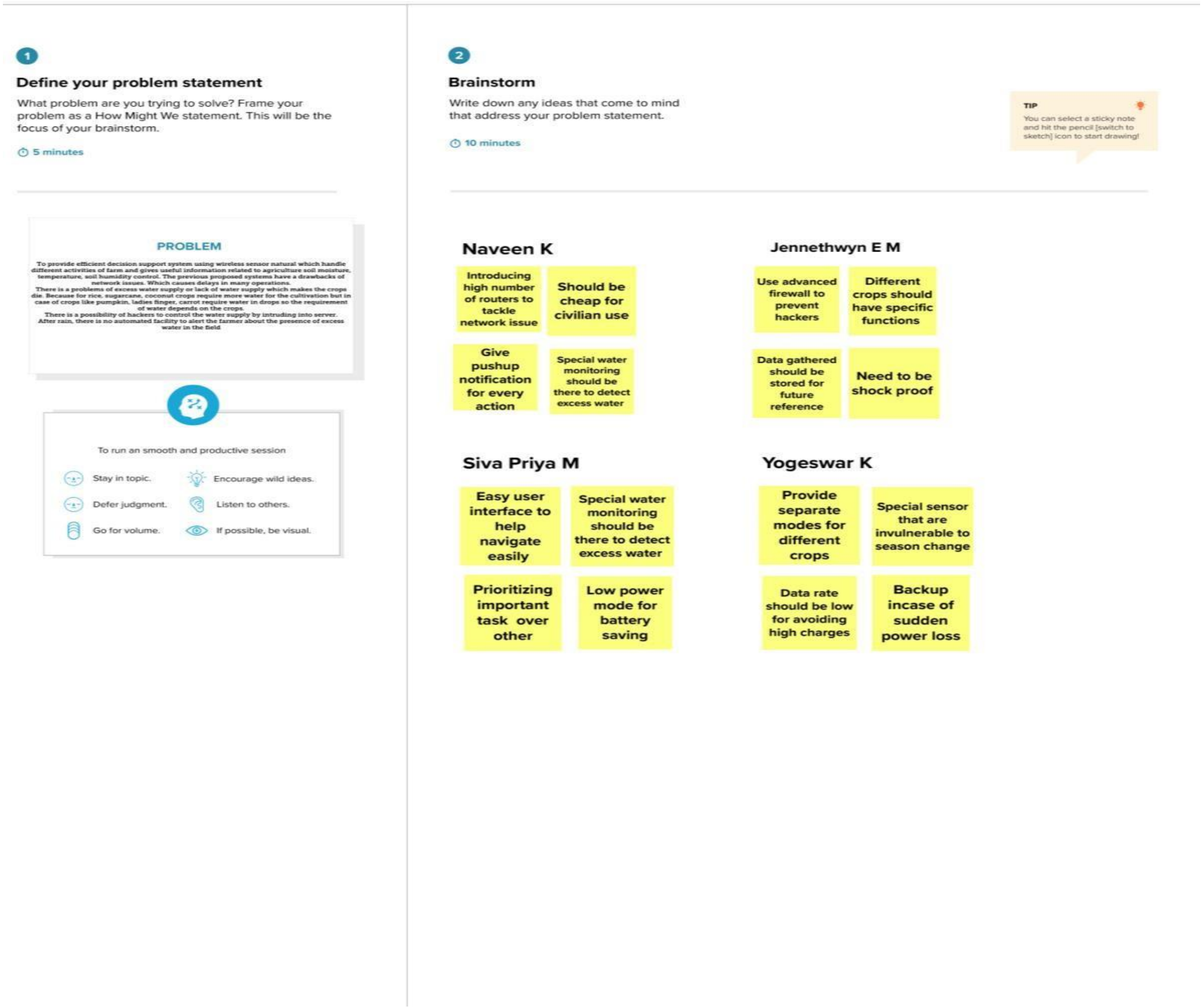
What do they do today?
What behavior have we observed?
What can we imagine them doing?

- Land lords divide the fields and distribute them among multiple farmers
- Farmers monitor the farm land by themselves or by hiring a person for it.
- Farmers check the water supply for the crops during a regular interval

3.2 Ideation & Brainstorming

Ideation is often closely related to the practice of brainstorming, a specific technique that is utilized to generate new ideas. A principal difference between ideation and brainstorming is that **ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity**

STEP 1: *Team Gathering, Collaboration and Select the Problem Statement*



1

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

To provide efficient decision support system using wireless sensor natural which handle different activities of farm and gives useful information related to agriculture soil moisture, temperature, soil humidity control. The previous proposed systems have a drawbacks of network issues. Which causes delays in many operations.

There is a problems of excess water supply or lack of water supply which makes the crops die. Because for rice, sugarcane, coconut crops require more water for the cultivation but in case of crops like pumpkin, ladies finger, carrot require water in drops so the requirement of water depends on the crops.

There is a possibility of hackers to control the water supply by intruding into server.

After rain, there is no automated facility to alert the farmer about the presence of excess water in the field

To run an smooth and productive session

Stay in topic. Encourage wild ideas.

Defer judgment. Listen to others.

Go for volume. If possible, be visual.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Naveen K

- Introducing high number of routers to tackle network issue
- Should be cheap for civilian use
- Give pushup notification for every action
- Special water monitoring should be there to detect excess water

Jennethwyn E M

- Use advanced firewall to prevent hackers
- Different crops should have specific functions
- Data gathered should be stored for future reference
- Need to be shock proof

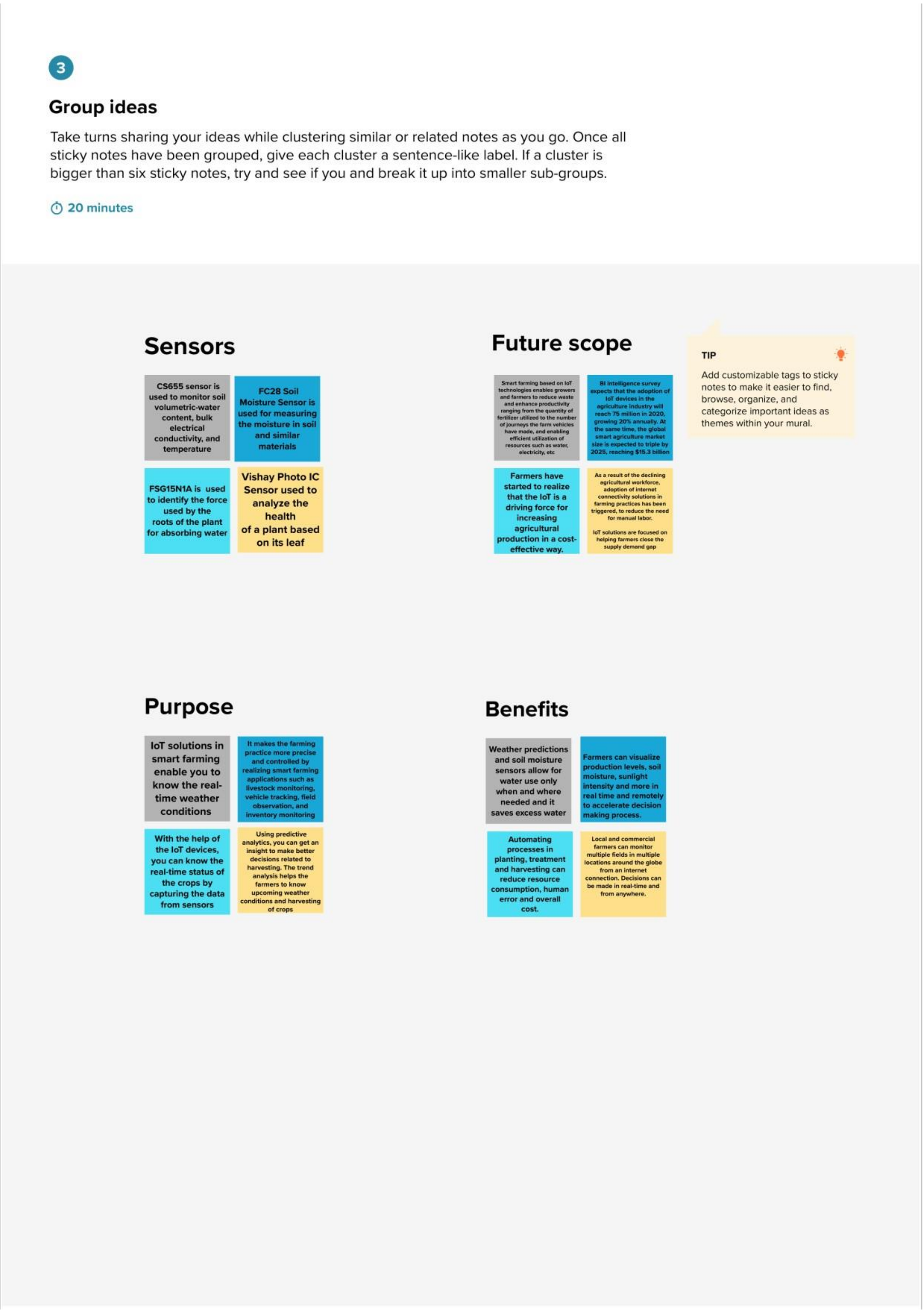
Siva Priya M

- Easy user interface to help navigate easily
- Special water monitoring should be there to detect excess water
- Prioritizing important task over other
- Low power mode for battery saving

Yogeswar K

- Provide separate modes for different crops
- Special sensor that are invulnerable to season change
- Data rate should be low for avoiding high charges
- Backup incase of sudden power loss

STEP 2: *Brainstorm, Idea Listing and Grouping*



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Sensors

CS655 sensor is used to monitor soil volumetric-water content, bulk electrical conductivity, and temperature

FC28 Soil Moisture Sensor is used for measuring the moisture in soil and similar materials

FSG15N1A is used to identify the force used by the roots of the plant for absorbing water

Vishay Photo IC Sensor used to analyze the health of a plant based on its leaf

Future scope

Smart farming based on IoT technologies enables growers and farmers to reduce waste and enhance productivity ranging from the quantity of fertilizer utilized to the number of journeys the farm vehicles have made, and enabling efficient utilization of resources such as water, electricity, etc

BI Intelligence survey expects that the adoption of IoT devices in the agriculture industry will reach 75 million in 2020, growing 20% annually. At the same time, the global smart agriculture market size is expected to triple by 2025, reaching $15.3 billion

Farmers have started to realize that the IoT is a driving force for increasing agricultural production in a cost-effective way.

As a result of the declining agricultural workforce, adoption of internet connectivity solutions in farming practices has been triggered, to reduce the need for manual labor.

IoT solutions are focused on helping farmers close the supply demand gap

Purpose

IoT solutions in smart farming enable you to know the real-time weather conditions

It makes the farming practice more precise and controlled by realizing smart farming applications such as livestock monitoring, vehicle tracking, field observation, and inventory monitoring

With the help of the IoT devices, you can know the real-time status of the crops by capturing the data from sensors

Using predictive analytics, you can get an insight to make better decisions related to harvesting. The trend analysis helps the farmers to know upcoming weather conditions and harvesting of crops

Benefits

Weather predictions and soil moisture sensors allow for water use only when and where needed and it saves excess water

Farmers can visualize production levels, soil moisture, sunlight intensity and more in real time and remotely to accelerate decision making process.

Automating processes in planting, treatment and harvesting can reduce resource consumption, human error and overall cost.

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.

3.3 Proposed Solution

1. Problem Statement (Problem to be solved) - To provide efficient decision support system using wireless sensor natural which handle different activities of farm and gives useful information related to agriculture soil moisture, temperature, soil humidity control. There is a problem of excess water supply or lack of water supply which makes the crops die. There is a possibility of hackers to control the water supply by intruding into the server. After rain, there is no automated facility to alert the farmers about the presence of excess water in the field.

2. **Idea / Solution description** - The network issue problem can be over rid by providing high number of routers around the farm in the mesh network form. We can provide certain modes for different crop. This different mode can be given a range of humidity level for different crops. To measure the soil humidity level CS655, VTU933 sensors were used. But cs655 sensors are vulnerable to surface disturbances. Hence, we can come with FC28 sensor which can be used in both analog and digital mode and also it can be suitable for automatic control applications. The VTU933 sensor is also a time-consuming sensor and not suitable for automatic control system-- By using smart water sensors, the excess water presence can be identified and the alert will be given to the farmers through the mobile application

3. **Novelty / Uniqueness** - Our project system has customized function for different types of crops. Each mode comes with its own set of features depending on the crop. This system monitors some parameters of the crop using the sensors and determine the soil moisture, humidity and depending on the need, it automates water supply. The sensors used here is less suspected to environmental conditions therefore giving long lasting performance. Moisture sensor probes are coated with an immersion gold that protects the Nickel probes from oxidation. Several sensors can be used to identify the presence of excess water in the field. This feature is not present in many of the recent Farming applications.

4. **Social Impact / Customer Satisfaction** - Smart farming increases the production, efficiency and protection of crops. This enables the farmers to manage their fields remotely via smart gadgets, which will save lot of resources and time. With the help of automation and sensor technology, benefits the society by conservation of water, resource management and better crop yield. Realtime monitoring allows the user to know the present situation with just an application. The data collected can help the farmers to predict the weather in the upcoming years. The application is very user friendly and so anyone can learn how to use it in very short duration.

5. Business Model (Revenue Model) –

- **Channels** - Direct contact with engineers and Farm owners.
- **Key Activities:** Online Portal: - Farm registration - Monitoring services - Models, dataset, maps and services.
- **Customer Relationships** - Open offices - Regular training sessions - Engineers follow up - social media.
- **Supply chain** - Machinery consolidation - Transportation and packaging - Buying farm products - Bidding and auctions.
- **Customer Relationships** - Open offices - Regular training sessions - Engineers follow up - social media.

6. Scalability of the Solution - Smart farmer platform dramatically reduces time to market and efforts to create smart-farming solutions. Save up to 90% of development time for your smart-farming solution by utilizing the following platform benefits:

- Reliable and fault-tolerant data collection from your IoT devices and sensors to monitor facilities state, crop growth characteristics, humidity level, etc.
- Powerful rule engine to process produces alarms and valuable insights.
- Optimize resource consumption with automatic remote control of actuators.
- Advanced and flexible visualization for real-time and historical data.
- Customizable end-user dashboards to share farm monitoring results.

3.4 Problem Solution Fit

1. CUSTOMER SEGMENT(S)

The main customer for our project is:
i)Farmers who wants to improve the yield of their crops.

ii)Farmers who wants to know the condition of their crops and its environmental conditions so they could take the necessary methods immediately.

2. JOBS-TO-BE-DONE / PROBLEMS

i)The farmers will initially find it hard to use the device as they have to get familiar with the technologies.

ii) They must be with their phone/laptop always so that they would be alarmed when they get the message/mail.

3. TRIGGERS

Customers get triggered mainly because to save their crops and to prevent them from the damage as they feel depressed when they face the losses and it indirectly affects their family too. This device is also a budget friendly device.

4. EMOTIONS: BEFORE / AFTER

Before: Depressed, loss of time, Facing more losses

After: Confident, gets chance to spend time efficiently,95%.

5. AVAILABLE SOLUTIONS

For smart farming, lot of IoT based solutions are there. But, one huge disadvantage of smart farming is that it requires an unlimited or continuous internet connection to be successful. This means that in rural communities, especially in the developing countries where we have mass crop production, it is completely impossible to operate this farming method. In places where internet connections are frustratingly slow, smart farming will be an impossibility.

6. CUSTOMER CONSTRAINTS

i)The availability of device, proper Network facilities and budget are several constraints, knowledge about the application.

ii) Network connectivity would be the main constraint as we use Wi- Fi which has major limitations like in coverage, scalability and power consumption.

7. BEHAVIOUR

i) The customers will reach us when they don’t have idea on how to analyse the soil and to improve the current irrigation system.

ii) As in the case of weather condition monitoring, sensors for crop monitoring also collect all information like crop health, humidity, precipitation temperature, and other parameters.

8.CHANNELS of BEHAVIOUR

Offline:

The IoT-based smart farming not only helps in modernizing the conventional farming methods but also targets other agriculture methods like organic farming, family farming (complex or small spaces, particular cattle and/or cultures, preservation of particular or high-quality varieties, etc.), and enhances highly transparent farming.

Online:

IoT-based smart farming is also beneficial in terms of environmental issues. It can help the farmers to efficiently use water, optimize the inputs and treatments

9. PROBLEM ROOT CAUSE

Technologies keep developing but still the farmers are not able to achieve their goals ie due to the presence of excess water in the field, varying climatic conditions etc. which affects the crop. So, in order to avoid this there is a need for smart farming which helps to improve the time efficiency, crop monitoring, soil management etc.

10. YOUR SOLUTION

1)To provide an alternate (i.e) to avoid the network problems we are also going to introduce the manual mode where the farmers can stop the water flow /provide limited amount of water flow into the field., Make it more user friendly (like appoint the help center team to guide them whenever they are facing any trouble with our app).Additional features like create an awareness about where to get agricultural loans, government agriculture schemes and get the feedback of every farmers on every month end.

2)There will be less weed growth, Maximum use of water efficiently, Control of soil erosion and maximum crop yield.

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	Registration through Gmail
FR-2	User Confirmation	Confirmation through registered Email via OTP

FR-3	Log in to system	Check credentials and allow user to enter the main page with regarding their roles
FR-4	Software	Web UI, Node-red, IBM Watson, MIT app
FR-5	IOT devices	Sensors, Relays and Wi-fi module

4.2.Non-functional Requirements:

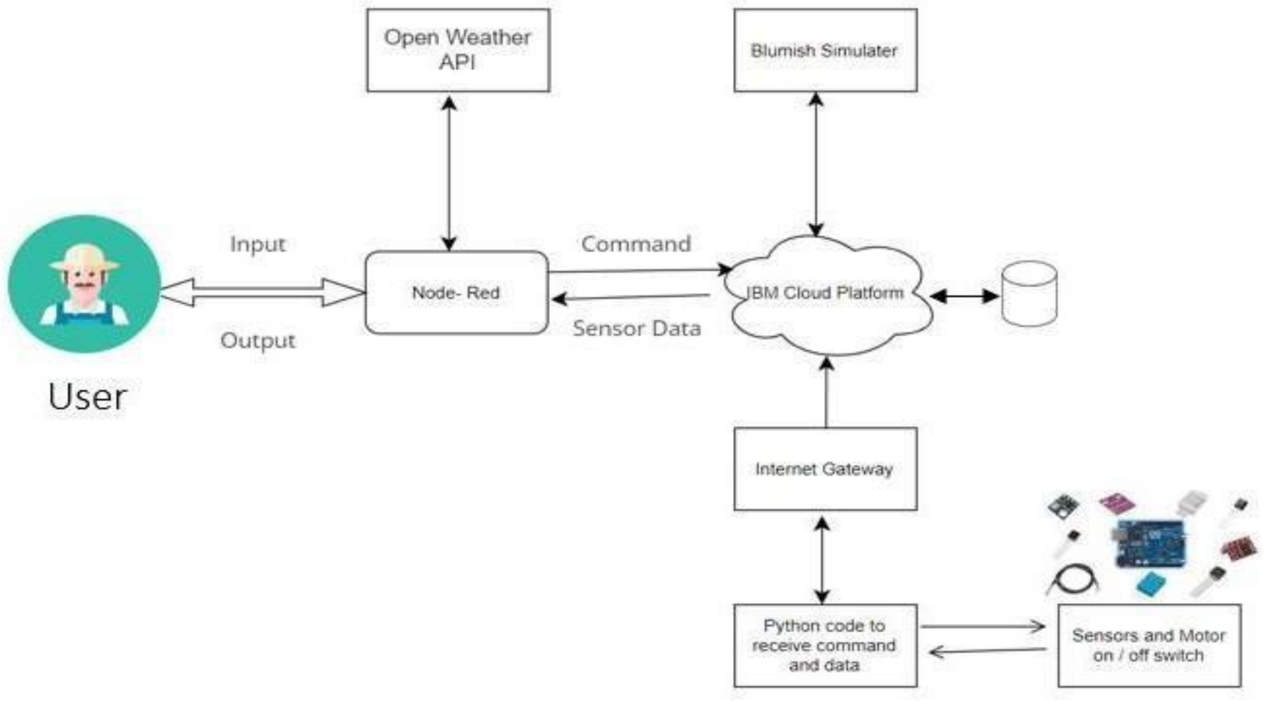
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Includes easy learn ability, efficiency in use, remember ability, lack of errors in operation and subjective pleasure
NFR-2	**Security**	Sensitive and private data must be protected
NFR-3	**Reliability**	Farmer can get a field task accomplished in a timely manner, it also reduces lost time and productivity.
NFR-4	**Performance**	Smart farms are accurate farms oriented to increase efficiency
NFR-5	**Availability**	With permitted network connectivity the application is accessible. The availability of low-cost cloud services encourages farmers to adopt smart farming
NFR-6	**Scalability**	It is perfectly scalable many new constraints can be added

5.Project Design

5.1.Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



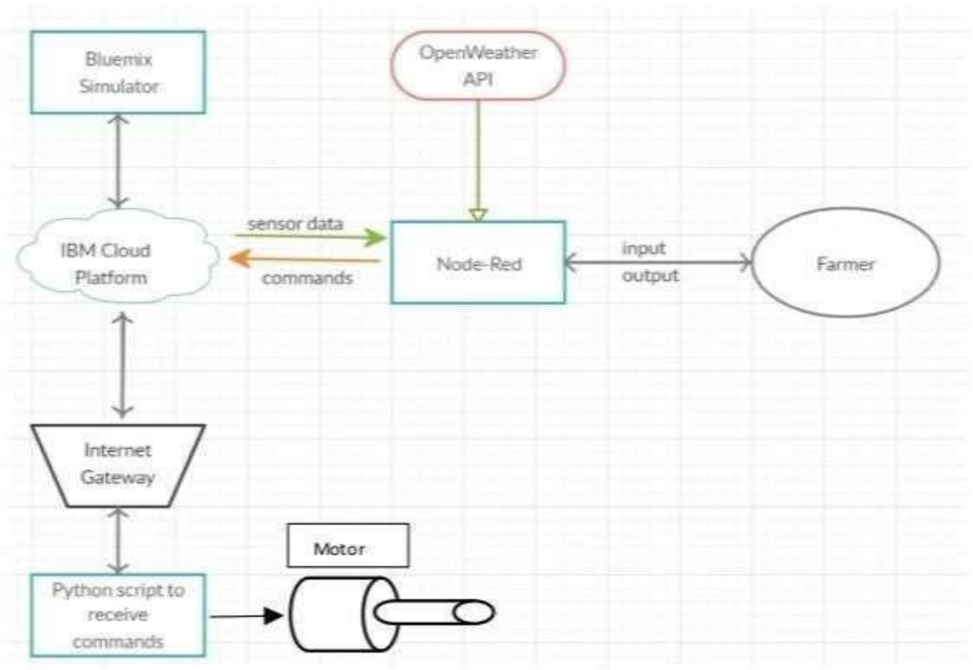
5.2.Solutions & Technical Architecture:

Guidelines:

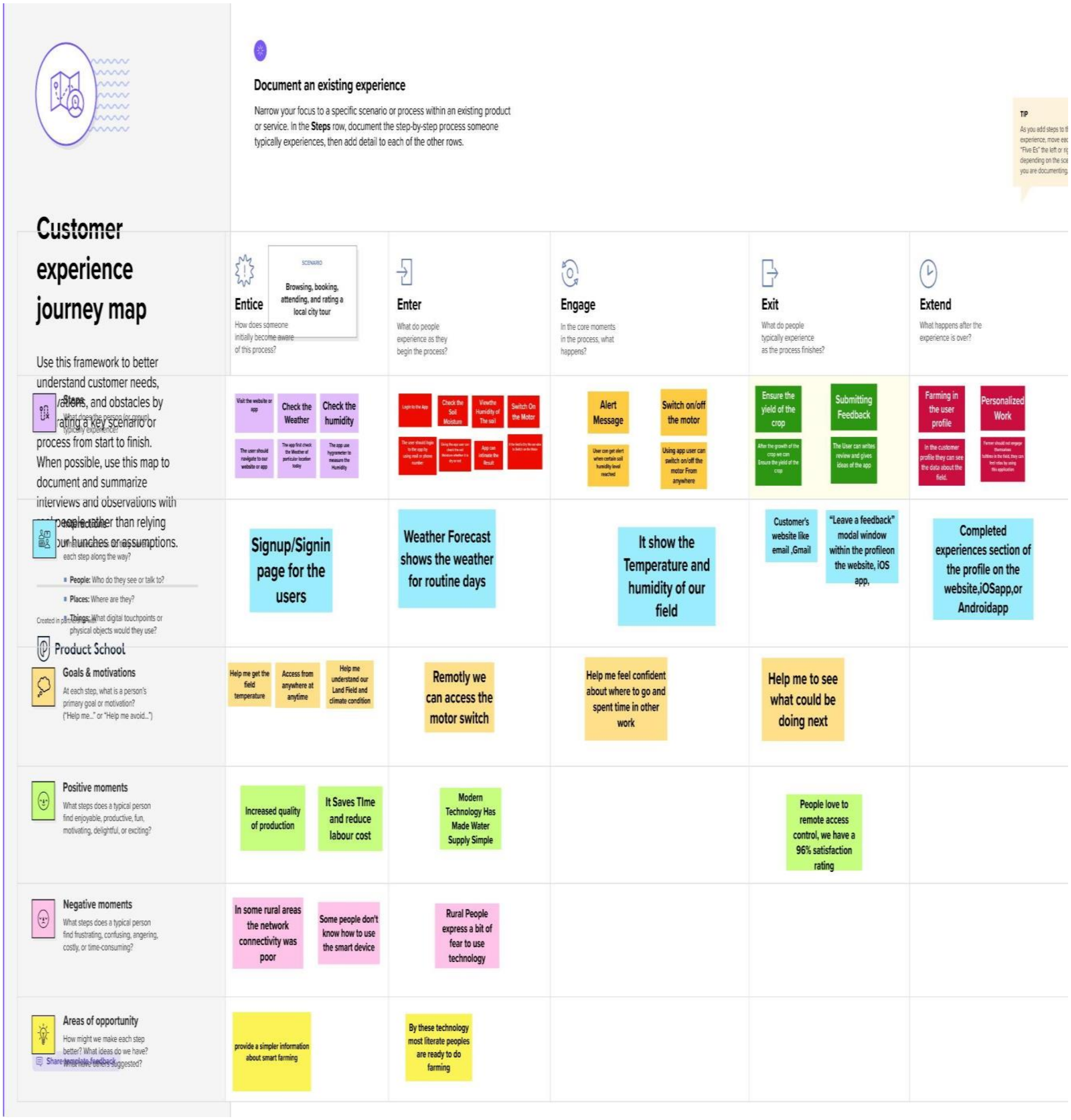
1. The different soil parameters temperature, soil moistures and then humidity are sensed using different sensors and obtained value is stored in the IBM B2 cloud.
2. Arduino UNO is used as a processing Unit that process the data obtained from the sensors and whether data from the weather API.
3. NODE-RED is used as a programming tool to write the hardware, software and APIs. The MQTT protocol is followed

for the communication.

4. All the collected data are provided to the user through a mobile application that was developed using the MIT app inventor. The user could make decision through an app, weather to water the field or not depending upon the sensor values. By using the app, they can remotely operate the motor switch.



5.3.User stories:



Customer experience journey map

Use this framework to better understand customer needs, motivations, and obstacles by illustrating a key scenario or process from start to finish. When possible, use this map to document and summarize interviews and observations with real people rather than relying on your hunches or assumptions.

Created in partnership with Product School

Document an existing experience

Narrow your focus to a specific scenario or process within an existing product or service. In the **Steps** row, document the step-by-step process someone typically experiences, then add detail to each of the other rows.

TIP

As you add steps to t experience, move eac "Five Es" the left or ri depending on the sce you are documenting.

SCENARIO

Browsing, booking, attending, and rating a local city tour

	Entice How does someone initially become aware of this process?	Enter What do people experience as they begin the process?	Engage In the core moments in the process, what happens?	Exit What do people typically experience as the process finishes?	Extend What happens after the experience is over?
Steps What does the person (or group) typically experience?	Visit the website or app Check the Weather Check the humidity The user should navigate to our website or app The app first check the Weather of particular location today The app use hygrometer to measure the Humidity	Login to the App Check the Soil Moisture Viewthe Humidity of The soil Switch On the Motor The user should login to the app by using mail or phone number Using the app user can check the soil Moisture whether it is dry or not App can intimate the Result If the field is Dry the can also to Switch on the Motor	Alert Message Switch on/off the motor User can get alert when certain soil humidity level reached Using app user can switch on/off the motor From anywhere	Ensure the yield of the crop Submitting Feedback After the growth of the crop we can Ensure the yield of the crop The User can writes review and gives ideas of the app	Farming in the user profile Personalized Work In the customer profile they can see the data about the field. Farmer should not engage themselves fulltime in the field, they can feel relax by using this application
Expectations What ... each step along the way? People: Who do they see or talk to? Places: Where are they? Things: What digital touchpoints or physical objects would they use?	Signup/Signin page for the users	Weather Forecast shows the weather for routine days	It show the Temperature and humidity of our field	Customer's website like email ,Gmail "Leave a feedback" modal window within the profileon the website, iOS app,	Completed experiences section of the profile on the website,iOSapp,or Androidapp
Goals & motivations At each step, what is a person's primary goal or motivation? ("Help me..." or "Help me avoid...")	Help me get the field temperature Access from anywhere at anytime Help me understand our Land Field and climate condition	Remotly we can access the motor switch	Help me feel confident about where to go and spent time in other work	Help me to see what could be doing next	
Positive moments What steps does a typical person find enjoyable, productive, fun, motivating, delightful, or exciting?	Increased quality of production It Saves TIme and reduce labour cost	Modern Technology Has Made Water Supply Simple		People love to remote access control, we have a 96% satisfaction rating	
Negative moments What steps does a typical person find frustrating, confusing, angering, costly, or time-consuming?	In some rural areas the network connectivity was poor Some people don't know how to use the smart device	Rural People express a bit of fear to use technology			
Areas of opportunity How might we make each step better? What ideas do we have? What have others suggested?	provide a simpler information about smart farming	By these technology most literate peoples are ready to do farming			

Share template feedback

6.Project Planning & Scheduling:

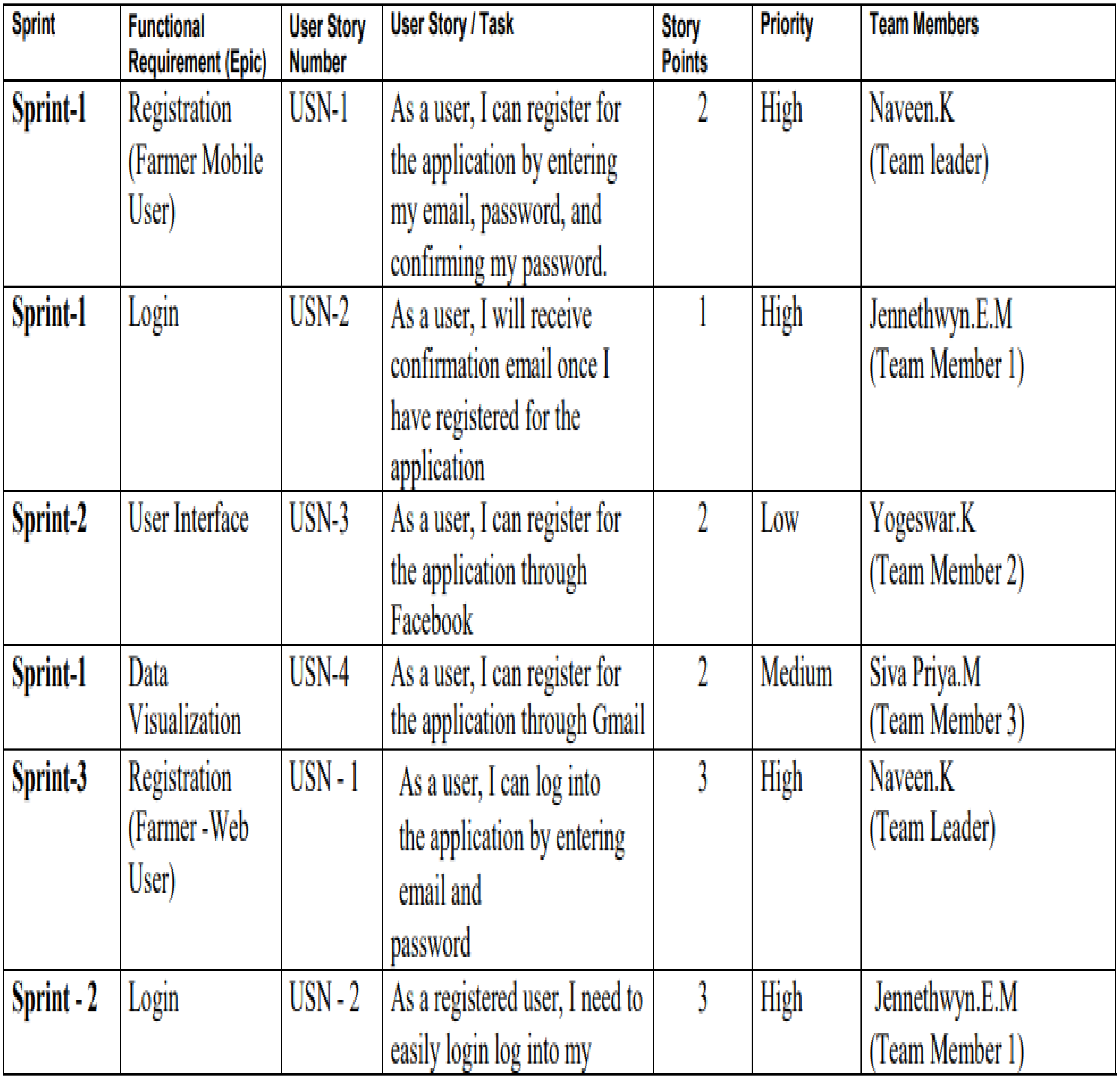
6.1.Sprint Planning & Estimation:

S.NO	ACTIVITY TITLE	ACTIVITY-DESCRIPTION	DURATION
1	**Understanding the Project**	Assign the team members after that create repository in theGitHub and then assigntask to each member and guide them how to access the GitHub while submitting the assignments	1 week
2	**Staring the Project**	Team Members to Assign All the TasksBased on Sprints and Work on It Accordingly.	1 week
3	**Completing Every Task**	Team Leader shouldensure that whether every team member have completed theassigned task or not	1 week
4	**Stand Up Meetings**	Team Lead Must Havea Stand-Up Meeting with The Team and Work on The Updates and Requirement Session	1 week
5	**Deadline**	Ensure that team members are completing every task within the deadline	1 week

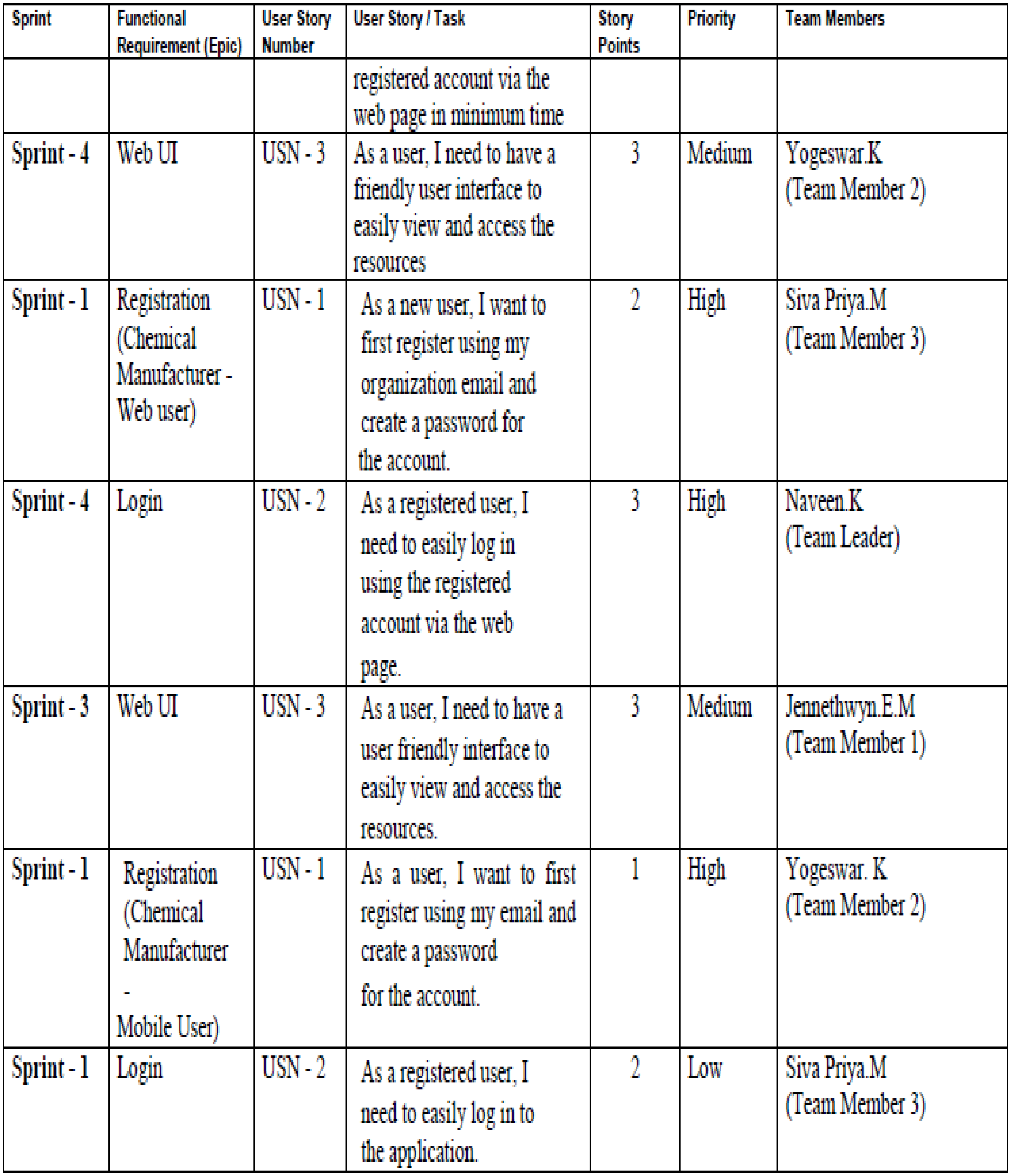
6	**Budget and Scope of project**	Analyze the overall budget which must be within certain limit it should be favorable to every person	1 week

6.2.Sprint Delivery Schedule:

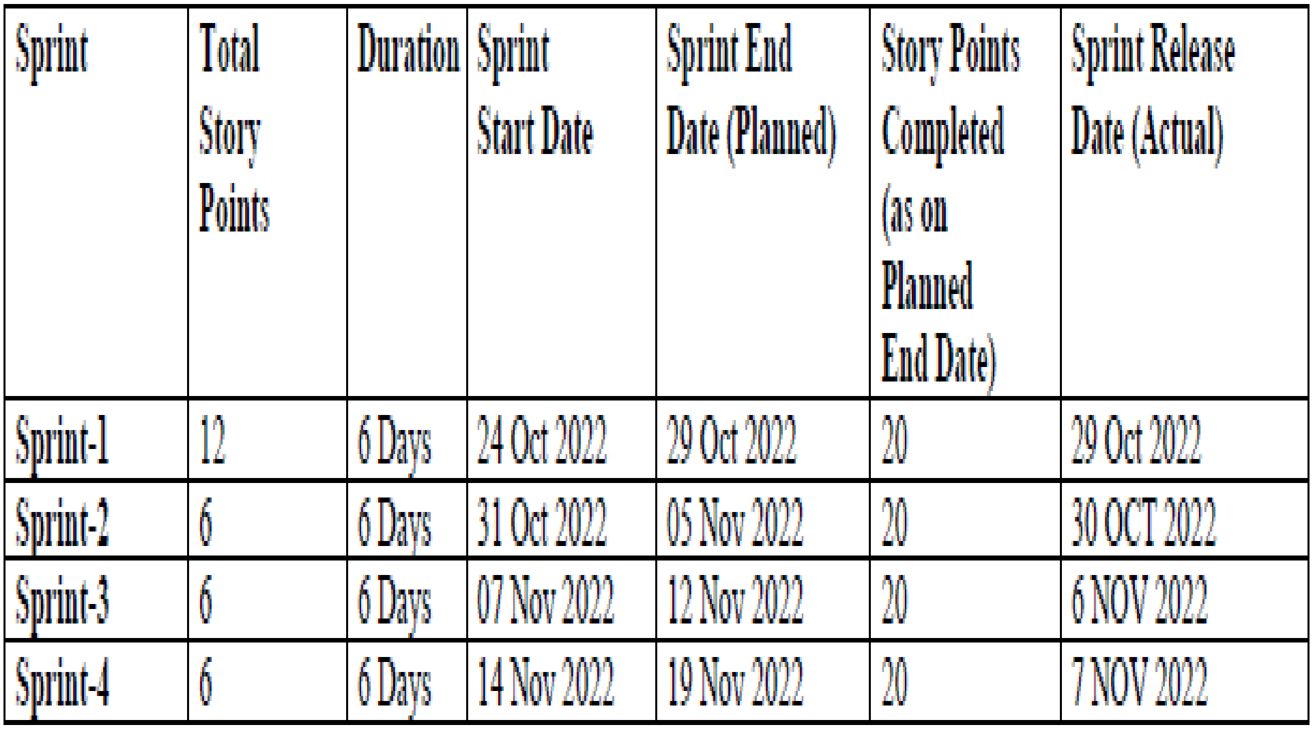
Product Backlog, Sprint Schedule, and Estimation



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration (Farmer Mobile User)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Naveen.K (Team leader)
Sprint-1	Login	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Jennethwyn.E.M (Team Member 1)
Sprint-2	User Interface	USN-3	As a user, I can register for the application through Facebook	2	Low	Yogeswar.K (Team Member 2)
Sprint-1	Data Visualization	USN-4	As a user, I can register for the application through Gmail	2	Medium	Siva Priya.M (Team Member 3)
Sprint-3	Registration (Farmer -Web User)	USN - 1	As a user, I can log into the application by entering email and password	3	High	Naveen.K (Team Leader)
Sprint - 2	Login	USN - 2	As a registered user, I need to easily login log into my	3	High	Jennethwyn.E.M (Team Member 1)



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			registered account via the web page in minimum time			
Sprint - 4	Web UI	USN - 3	As a user, I need to have a friendly user interface to easily view and access the resources	3	Medium	Yogeswar.K (Team Member 2)
Sprint - 1	Registration (Chemical Manufacturer - Web user)	USN - 1	As a new user, I want to first register using my organization email and create a password for the account.	2	High	Siva Priya.M (Team Member 3)
Sprint - 4	Login	USN - 2	As a registered user, I need to easily log in using the registered account via the web page.	3	High	Naveen.K (Team Leader)
Sprint - 3	Web UI	USN - 3	As a user, I need to have a user friendly interface to easily view and access the resources.	3	Medium	Jennethwyn.E.M (Team Member 1)
Sprint - 1	Registration (Chemical Manufacturer - Mobile User)	USN - 1	As a user, I want to first register using my email and create a password for the account.	1	High	Yogeswar. K (Team Member 2)
Sprint - 1	Login	USN - 2	As a registered user, I need to easily log in to the application.	2	Low	Siva Priya.M (Team Member 3)



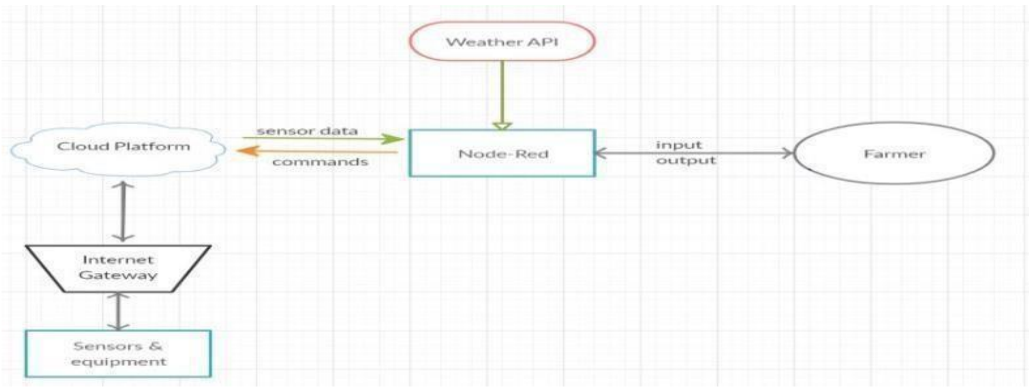
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	12	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	6	6 Days	31 Oct 2022	05 Nov 2022	20	30 OCT 2022
Sprint-3	6	6 Days	07 Nov 2022	12 Nov 2022	20	6 NOV 2022
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022	20	7 NOV 2022

Project planning & Scheduling

1.1 Sprint Delivery planning & Estimation

SPRINT DELIVERY OVERVIEW :

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

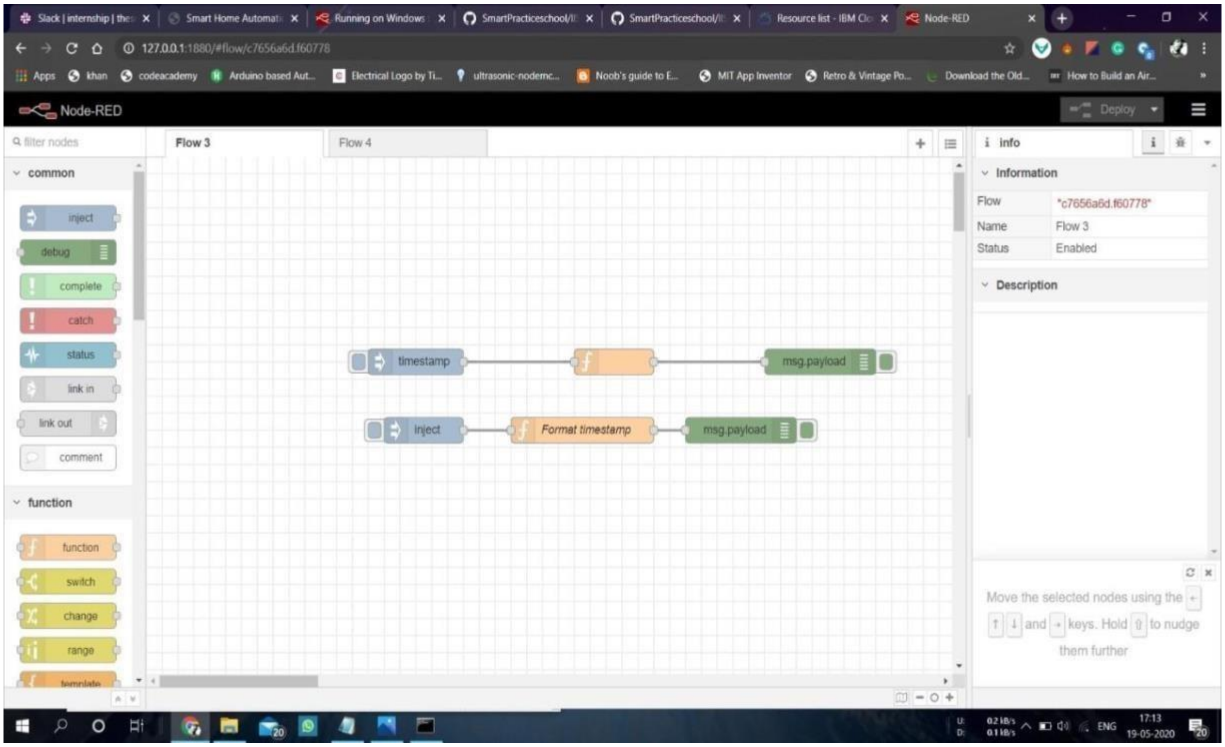


1.**Required Software Installation**

1.1 A Node-Red

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.

NodeRED provides a web browser-based flow editor, which can be used to create JavaScript functions.



Flow "c7656a6d.f60778"
Name Flow 3
Status Enabled

Move the selected nodes using the ← ↑ ↓ and → keys. Hold ⇧ to nudge them further

Installation :

- First install npm/node.js
- Open cmd prompt
- Type => npm install node-red **To run the application :**

- Open cmd prompt
- Type=> node-red
- Then open http://localhost:1880/ in browser

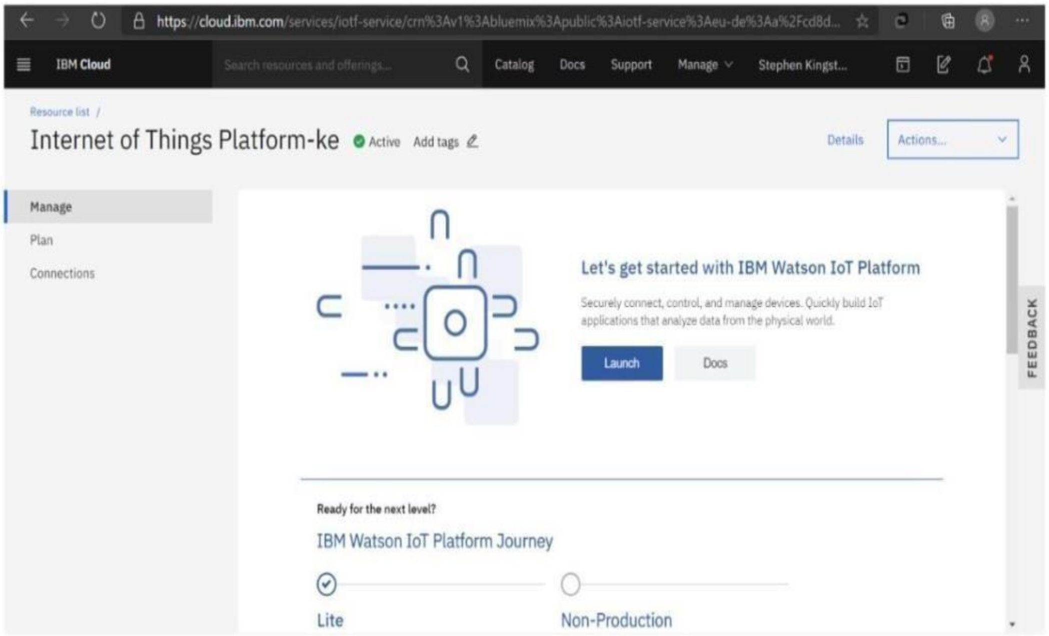
Installation of IBM IoT and Dashboard nodes for Node-Red

In order to connect to IBM Watson IoT platform and create the Web App UI these nodes are required

1. IBM IoT node
2. Dashboard node
3.

1.2. IBM Watson IoT Platform

A fully managed, cloud-hosted service with capabilities for device registration, connectivity, control, rapid visualization and data storage.



https://cloud.ibm.com/services/iotf-service/crn%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-de%3Aa%2Fcd8d...

Internet of Things Platform-ke

Let's get started with IBM Watson IoT Platform

Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.

Ready for the next level?

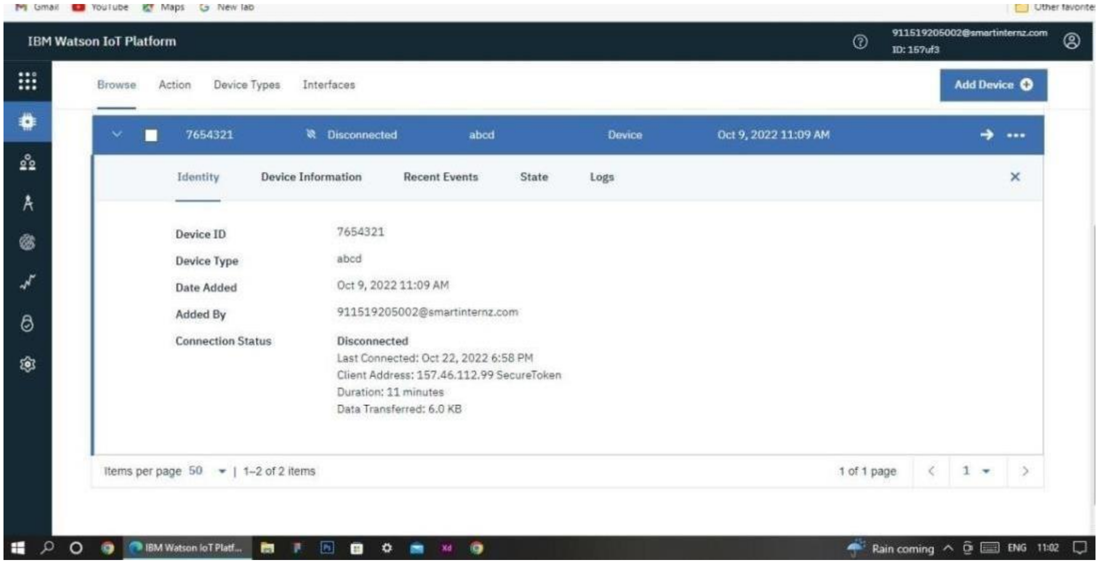
IBM Watson IoT Platform Journey

IBM Watson IoT Platform is a managed, cloud-hosted service designed to make it simple to derive value from your IoT devices.

Steps to configure:

- Create an account in IBM cloud using your email ID
- Create IBM Watson Platform in services in your IBM cloud account

- Launch the IBM Watson IoT Platform
- Create a new device
- Give credentials like device type, device ID, Auth. Token • Create API key and store API key andtoken elsewhere.



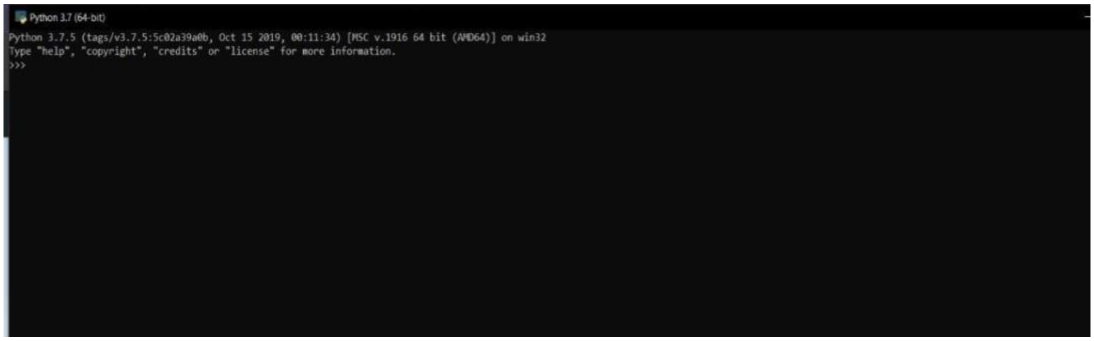
7654321 Disconnected abcd Device Oct 9, 2022 11:09 AM

Device ID 7654321
Device Type abcd
Date Added Oct 9, 2022 11:09 AM
Added By 911519205002@smartinternz.com
Connection Status **Disconnected**
Last Connected: Oct 22, 2022 6:58 PM
Client Address: 157.46.112.99 SecureToken
Duration: 11 minutes
Data Transferred: 6.0 KB

Items per page 50 | 1–2 of 2 items 1 of 1 page 1

1.2.C Python IDE

- Install Python3 compiler
- Install any python IDE to execute python scripts, in my case I used Spyder to execute the code.



```
Python 3.7.5 (tags/v3.7.5:5c02a39a0b, Oct 15 2019, 00:11:34) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

1.3 IoT Simulator

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

The link to simulator: https://watson-iot-sensor-simulator.mybluemix.net/

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

1.4 OpenWeather API

OpenWeatherMap is an online service that provides weather data. It provides current weather data, forecasts and historical data to more than 2 million customer.

Website link: https://openweathermap.org/guide

steps to configure:

oCreate account in OpenWeather oFind the name of your city by searching oCreate API key to your account o Replace "city name" and "your api key" with your city and API key in below red text **api.openweathermap.org/data/2.5/weather?q={city name}&appid={your api key}** Link I used in my project:

http://api.openweathermap.org/data/2.5/weather?q=Gudur,in&appid=62354068e45f41ffa6a5b164714145fe

2.Building Project

2.1 Connecting IoT Simulator to IBM Watson IoT Platform

-
-
- Open link provided in above section 4.3
- Give the credentials of your device

  in IBM Watson IoT Platform

- Click on connect

My credentials given to simulator are:

OrgID : 157uf3 api : a-9wbx5m-1qfklrf7jl

Device type: abcd token: JcU(4(9Z37PdL!Rmz(

Device ID :

7654321 Device

Token :

87654321

- You will receive the simulator data in cloud
- You can see the received data in Recent Events under your device
- Data received in this format(json)

{ o"d": {

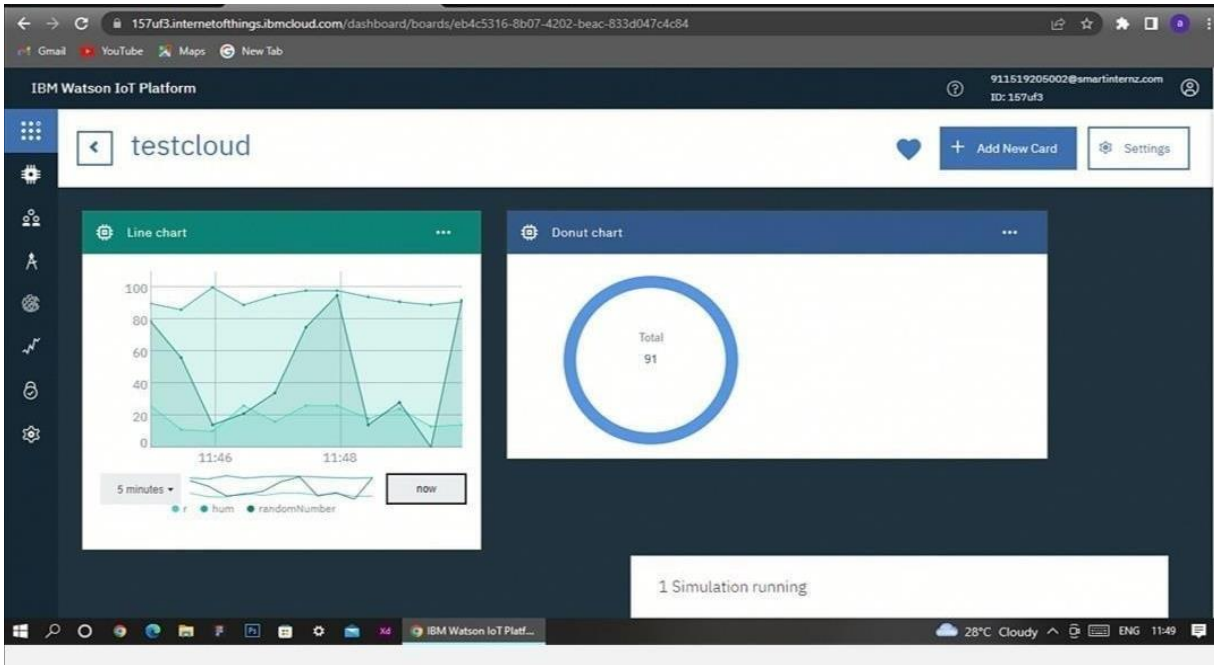
□ "name": "qwerty123",

□ "temperature": 17,

□ "humidity": 76,

□ "objectTemp": 25 o} }

You can see the received data in graphs by creating cards in Boards tab



1 Simulation running

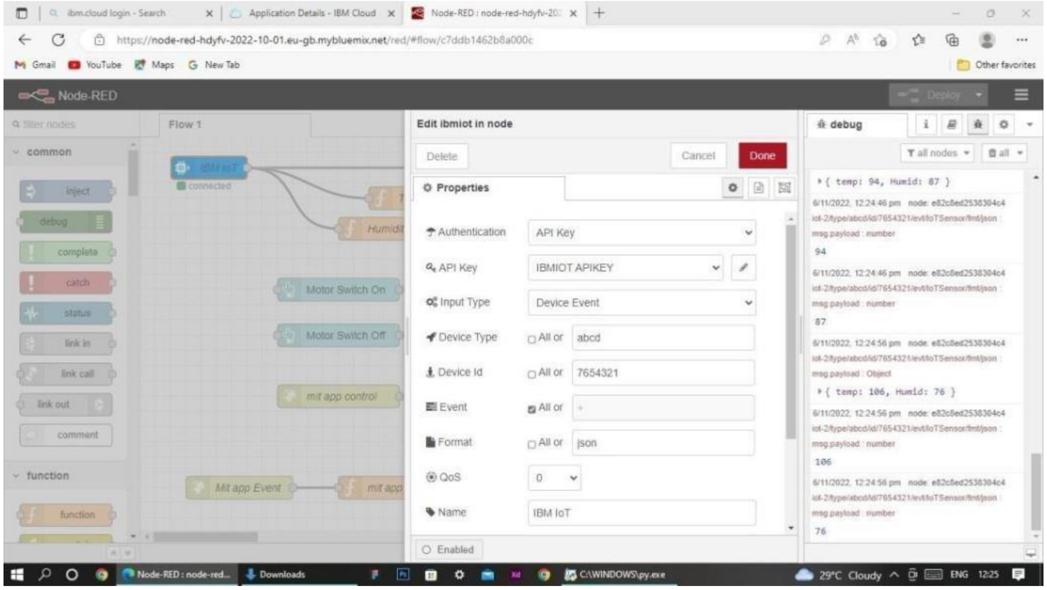
2.2 Configuration of Node-Red to collect IBM CLOUD DATA

The node IBM IoT App In is added to Node-Red workflow. Then the appropriate device credentials obtained earlier are entered into the node to connect and fetch device telemetry to Node-Red

Once it is connected Node-Red receives data from the device Display the data using debug node for verification

Connect function node and write the Java script code to get each reading separately. The Java script code for the function node is:

- msg.payload=msg.payload.d.temperature return msg;

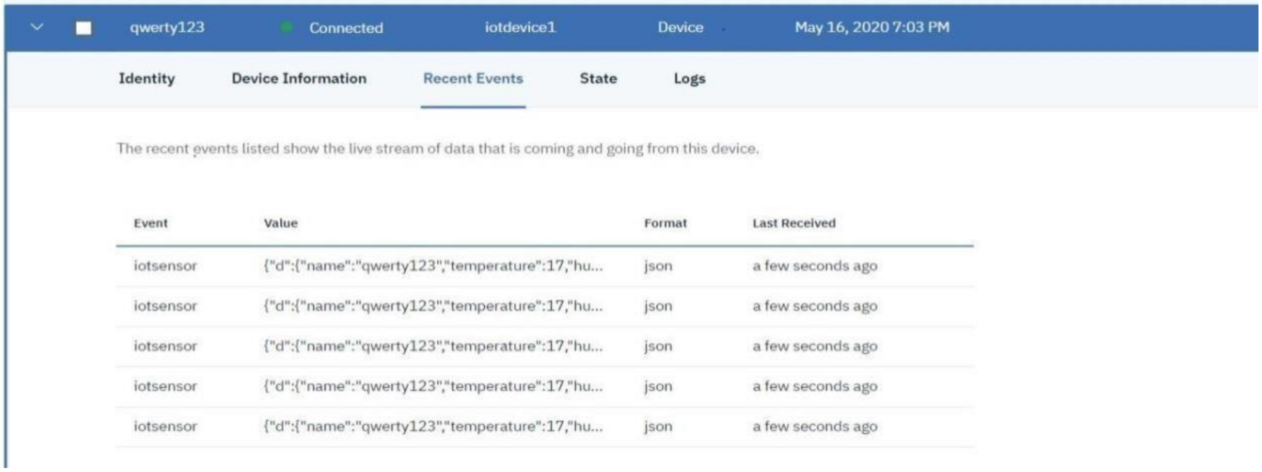


Edit ibmiot in node

- Authentication: API Key
- API Key: IBMIOT APIKEY
- Input Type: Device Event
- Device Type: □ All or abcd
- Device Id: □ All or 7654321
- Event: ☑ All or +
- Format: □ All or json
- QoS: 0
- Name: IBM IoT

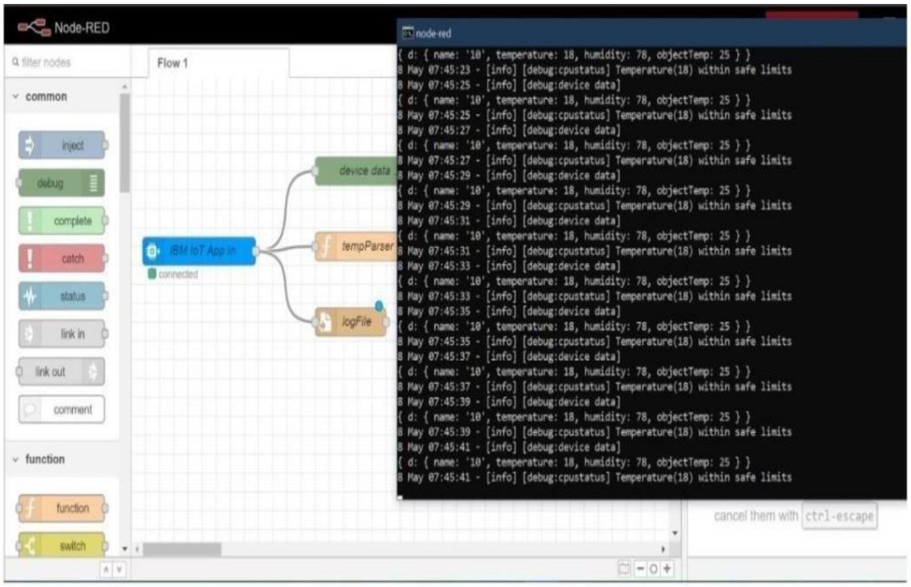
```
▸{ temp: 94, Humid: 87 }
6/11/2022, 12:24:46 pm  node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
94
6/11/2022, 12:24:46 pm  node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
87
6/11/2022, 12:24:56 pm  node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : Object
▸{ temp: 106, Humid: 76 }
6/11/2022, 12:24:56 pm  node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
106
6/11/2022, 12:24:56 pm  node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
76
```

Finally connect Gauge nodes from dashboard to see the data in UI Nodes connecte following manner to get each reading separately



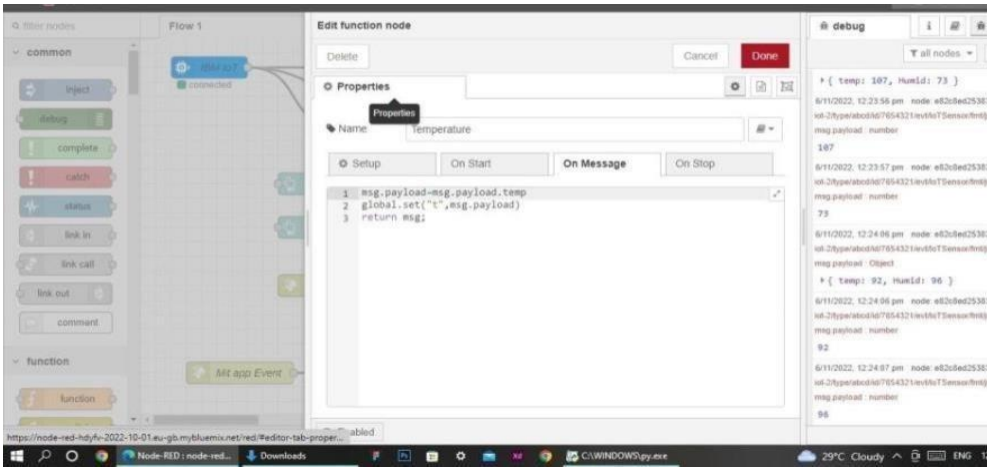
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
iotsensor	{"d":{"name":"qwerty123","temperature":17,"hu...	json	a few seconds ago
iotsensor	{"d":{"name":"qwerty123","temperature":17,"hu...	json	a few seconds ago
iotsensor	{"d":{"name":"qwerty123","temperature":17,"hu...	json	a few seconds ago
iotsensor	{"d":{"name":"qwerty123","temperature":17,"hu...	json	a few seconds ago
iotsensor	{"d":{"name":"qwerty123","temperature":17,"hu...	json	a few seconds ago



```
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:23 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:25 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:25 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:27 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:27 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:29 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:29 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:31 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:31 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:33 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:33 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:35 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:35 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:37 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:37 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:39 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:39 - [info] [debug:cpustatus] Temperature(18) within safe limits
8 May 07:45:41 - [info] [debug:device data]
{ d: { name: '10', temperature: 18, humidity: 78, objectTemp: 25 } }
8 May 07:45:41 - [info] [debug:cpustatus] Temperature(18) within safe limits
```

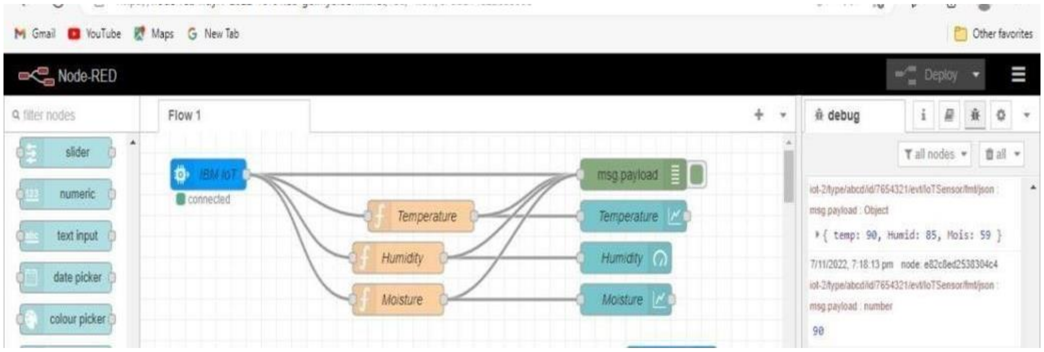
Data received from the cloud in Node-RED console



Name: Temperature

```
msg.payload=msg.payload.temp
global.set("t",msg.payload)
return msg;
```

```
▸{ temp: 107, Humid: 73 }
6/11/2022, 12:23:56 pm   node: e82c8ed2538
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/j
msg.payload : number
107
6/11/2022, 12:23:57 pm   node: e82c8ed2538
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/j
msg.payload : number
73
6/11/2022, 12:24:06 pm   node: e82c8ed2538
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/j
msg.payload : Object
▸{ temp: 92, Humid: 96 }
6/11/2022, 12:24:06 pm   node: e82c8ed2538
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/j
msg.payload : number
92
6/11/2022, 12:24:07 pm   node: e82c8ed2538
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/j
msg.payload : number
96
```



```
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : Object
▸{ temp: 90, Humid: 85, Mois: 59 }
7/11/2022, 7:18:13 pm   node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
90
```

This is the Java script code I written for the function node to get Temperature separately.

2.3.Configuration of Node-Red to collect data from OpenWeather

The Node-Red also receive data from the OpenWeather API by HTTP GET request. An inject trigger is added to perform HTTP request for every certain interval.

HTTP request node is configured with URL we saved before in section 4.4

The data we receive from OpenWeather after request is in below JSON format:

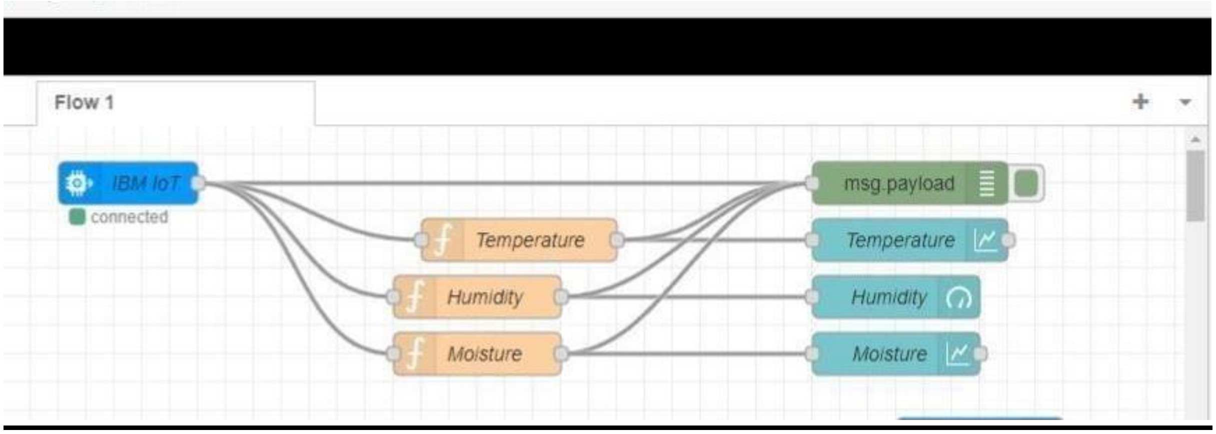
{"coord":{"lon":79.85,"lat":14.13},"weather":[{"id":803,"main":"Clouds","description":"broken
clouds","icon":"04n"}],"base":"stations","main":{"temp":307.59,
"feels_like":305.5,"temp_min":30 7.
59,"temp_max":307.59,"pressure":1002,"humidity":35,"sea_level":1002,"
grnd_level":1000},"wind":
{
"speed":6.23,"deg":170},"clouds":{"all":68},"dt":1589991979,"s
ys":{"country":"IN","sunrise":1589 93
3553,"sunset":1589979720},"timezone":19800,"id":1270791,"name":"
Gūdūr","cod":200} In order to parse the JSON string we use Java script functions and get each parameters

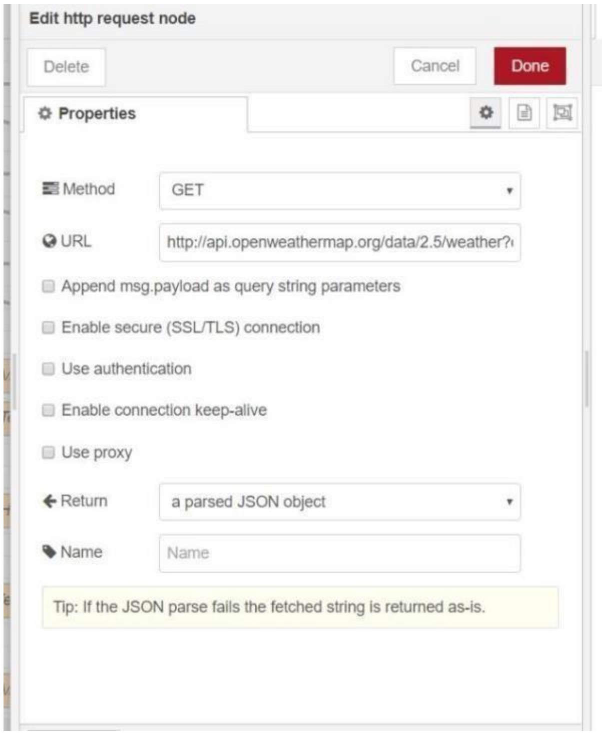
```
var temperature =
msg.payload.main.temp;
temperature = temperature-
273.15; return
{payload : temperature.toFixed(2)};
```

In the above Java script code we take temperature parameter into a new variable and convert it from kelvin to Celsius

Then we add Gauge and text nodes to represent data visually in UI

The above image has the program flow for receiving data from OpenWeather.





Method: GET

URL: http://api.openweathermap.org/data/2.5/weather?

□ Append msg.payload as query string parameters

□ Enable secure (SSL/TLS) connection

□ Use authentication

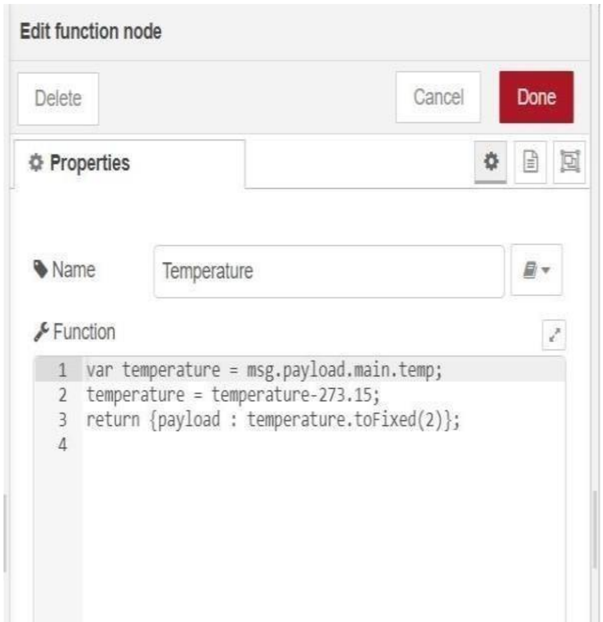
□ Enable connection keep-alive

□ Use proxy

Return: a parsed JSON object

Name: Name

Tip: If the JSON parse fails the fetched string is returned as-is.



Name: Temperature

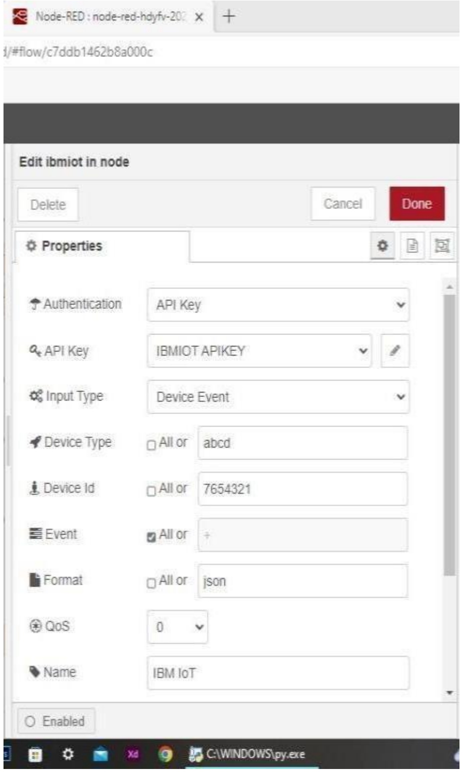
Function

```
var temperature = msg.payload.main.temp;
temperature = temperature-273.15;
return {payload : temperature.toFixed(2)};

```

The above two images contain http request and function node data that needs to be filled.

*2.*4.Configuration of Node-Red to send commands to IBM cloud

ibmiot out node I used to send data from Node-Red to IBM Watson device. So, after adding it to the flow we need to configure it with credentials of our Watson device.



Node-RED : node-red-hdyfv-202

/#flow/c7ddb1462b8a000c

Edit ibmiot in node

Authentication		API Key
API Key		IBMIOT APIKEY
Input Type		Device Event
Device Type	□ All or	abcd
Device Id	□ All or	7654321
Event	☑ All or	+
Format	□ All or	json
QoS		0
Name		IBM IoT

○ Enabled

C:\WINDOWS\py.exe

Here we add three buttons in UI which each
sends a number 0,1 and
2.0 -> for motor off
for motor on

1 -> for running motor continuously 30 minutes

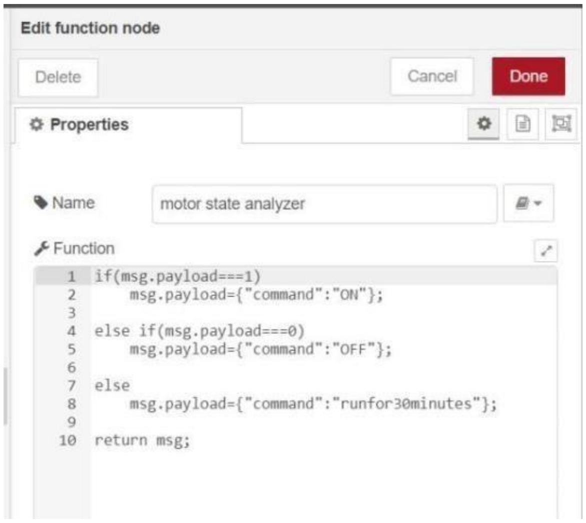
We used a function node to analyse the data received and assign command to each number. The Java script code for the analyser is:

```
if(msg.payload===1)
  msg.payload={"command":"ON"};
  else
  if(msg.payload===0)
  msg.payload={"comm
  and":"OFF"}; else
  msg.payload={"command":"run
for30minutes"}; return msg;
```

Then we use another function node to parse the data and get the command and represent it visually with text node.

The Java script code for that function node is:

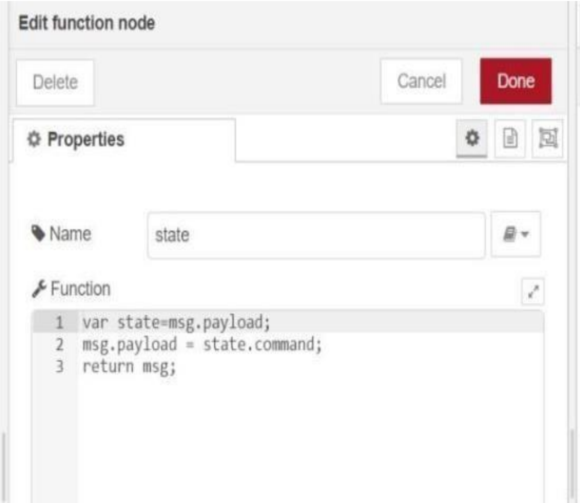
```
var state=msg.payload;
  msg.payload =
state.command; return msg;
```



Name: motor state analyzer

Function

```
if(msg.payload===1)
    msg.payload={"command":"ON"};

else if(msg.payload===0)
    msg.payload={"command":"OFF"};

else
    msg.payload={"command":"runfor30minutes"};

return msg;
```



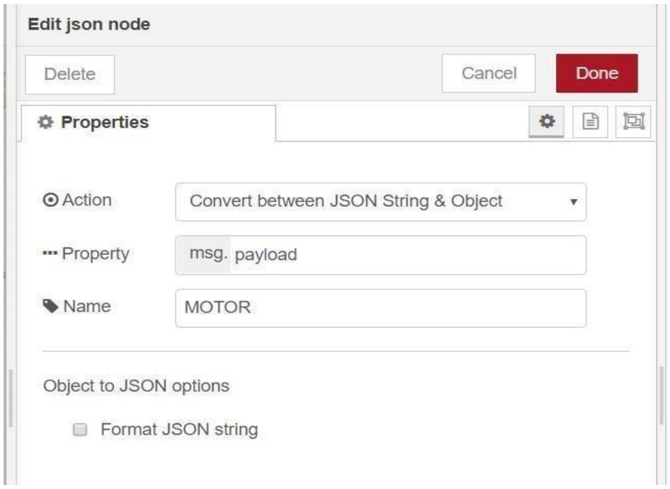
Name: state

Function

```
var state=msg.payload;
msg.payload = state.command;
return msg;
```

The above images show the java script codes of analyzer and state function nodes.

Then we add edit Json node to the conversion between JSON string & object and finally connect it to IBM IoT Out.
Edit JSON node needs to be configured like this


Edit json node

Delete | Cancel | Done

Properties

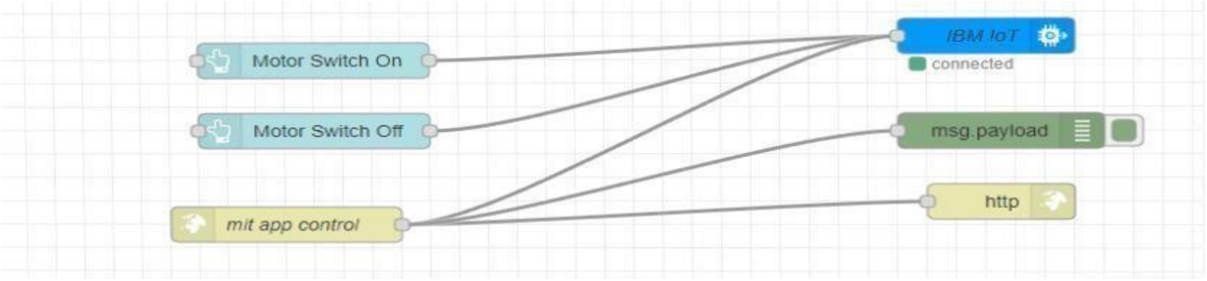
Action: Convert between JSON String & Object

Property: msg. payload

Name: MOTOR

Object to JSON options

□ Format JSON string



This is the program flow for sending commands to IBM cloud.

2.5Adjusting User Interface

In order to display the parsed JSON data a Node-Red dashboard is created

Here we are using Gauges, text and button nodes to display in the UI and helps to monitor the parameters and control the farm equipment.

Below images are the Gauge, text and button node configurations

Adding Background image to the UI

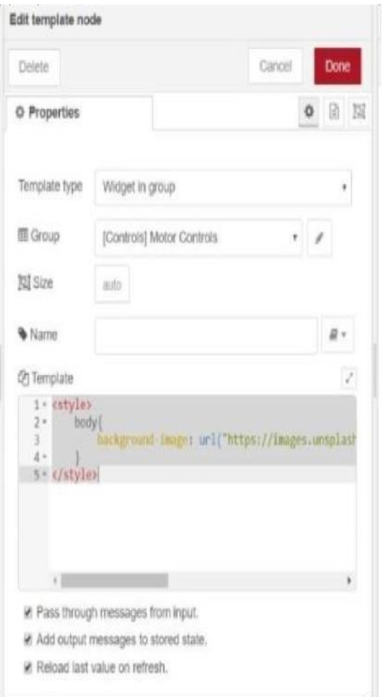
To add the background image we are going to add template node and configure it with below HTML code

- <style> body{ background-image: url("https://images.unsplash.com/photo-1563514227147-6d2ff665 a6a0?ixlib=rb-1.2.1&auto=format&fit=crop&w=1951&q=80");

  }
- </style>

We add URL of image that need to be displayed

Configuration is shown in the beside image



Template type: Widget in group

Group: [Controls] Motor Controls

Size: auto

Name:

Template

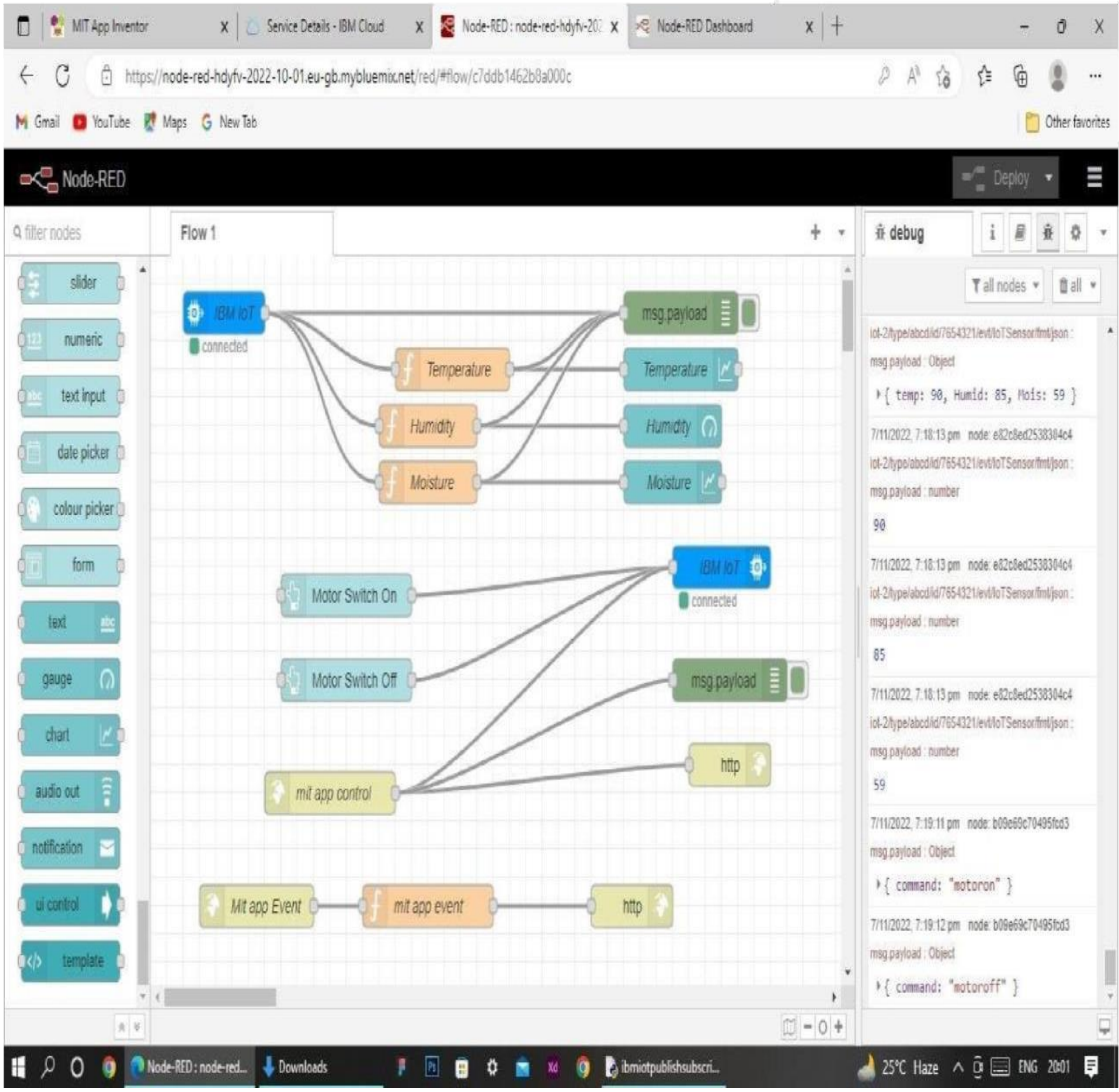
```
<style>
    body{
        background-image: url("https://images.unsplash
    }
</style>
```

☑ Pass through messages from input.

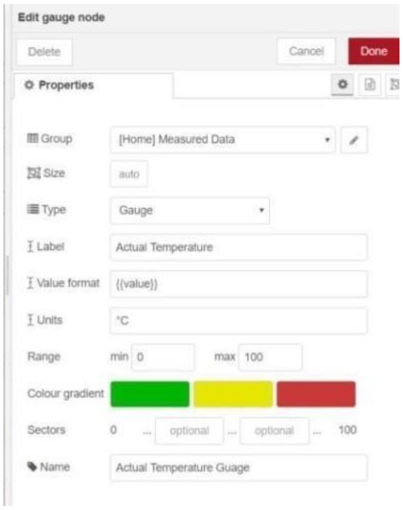
☑ Add output messages to stored state.

☑ Reload last value on refresh.

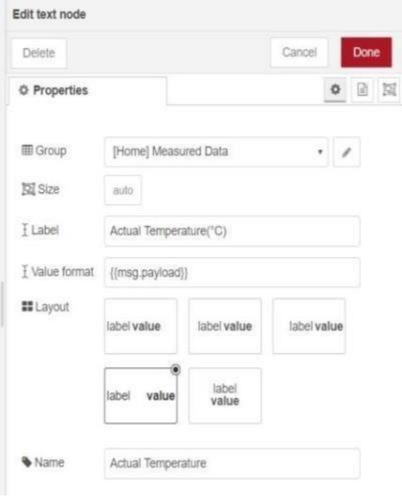
Complete Program Flow



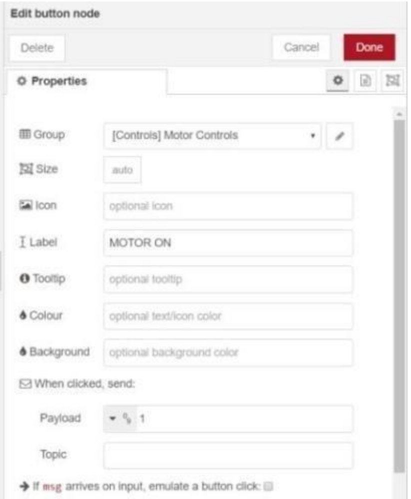
```
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : Object
▸{ temp: 90, Humid: 85, Mois: 59 }
7/11/2022, 7:18:13 pm   node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
90
7/11/2022, 7:18:13 pm   node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
85
7/11/2022, 7:18:13 pm   node: e82c8ed2538304c4
iot-2/type/abcd/id/7654321/evt/IoTSensor/fmt/json :
msg.payload : number
59
7/11/2022, 7:19:11 pm   node: b09e69c70495fcd3
msg.payload : Object
▸{ command: "motoron" }
7/11/2022, 7:19:12 pm   node: b09e69c70495fcd3
msg.payload : Object
▸{ command: "motoroff" }
```



Group: [Home] Measured Data
Size: auto
Type: Gauge
Label: Actual Temperature
Value format: {{value}}
Units: °C
Range: min 0 max 100
Colour gradient: 🟩 🟨 🟥
Sectors: 0 ... optional ... optional ... 100
Name: Actual Temperature Guage



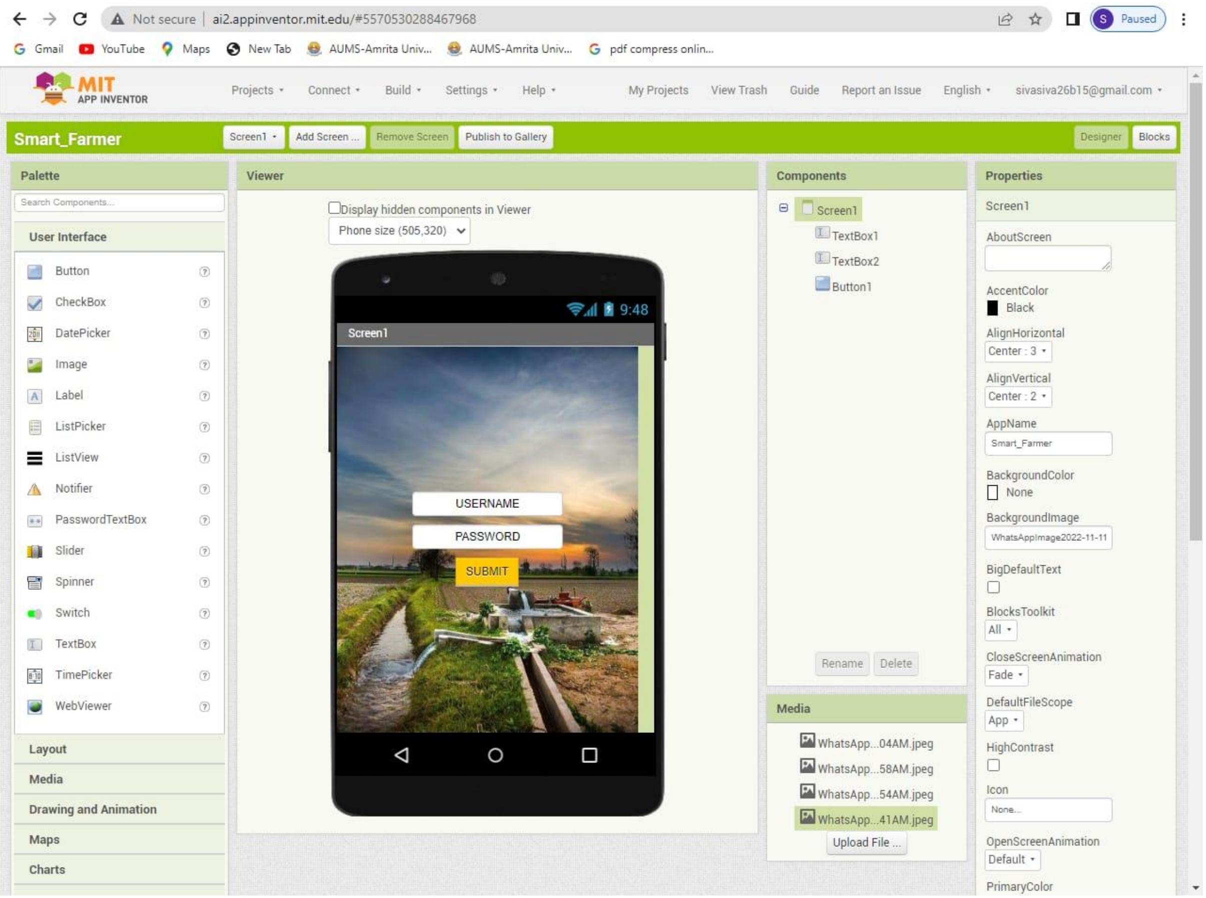
Group: [Home] Measured Data
Size: auto
Label: Actual Temperature(°C)
Value format: {{msg.payload}}
Layout: label value | label value | label value | label value | label value
Name: Actual Temperature



Group: [Controls] Motor Controls
Size: auto
Icon: optional icon
Label: MOTOR ON
Tooltip: optional tooltip
Colour: optional text/icon color
Background: optional background color
When clicked, send:
Payload: 1
Topic:
If msg arrives on input, emulate a button click: ☐

Mobile app:

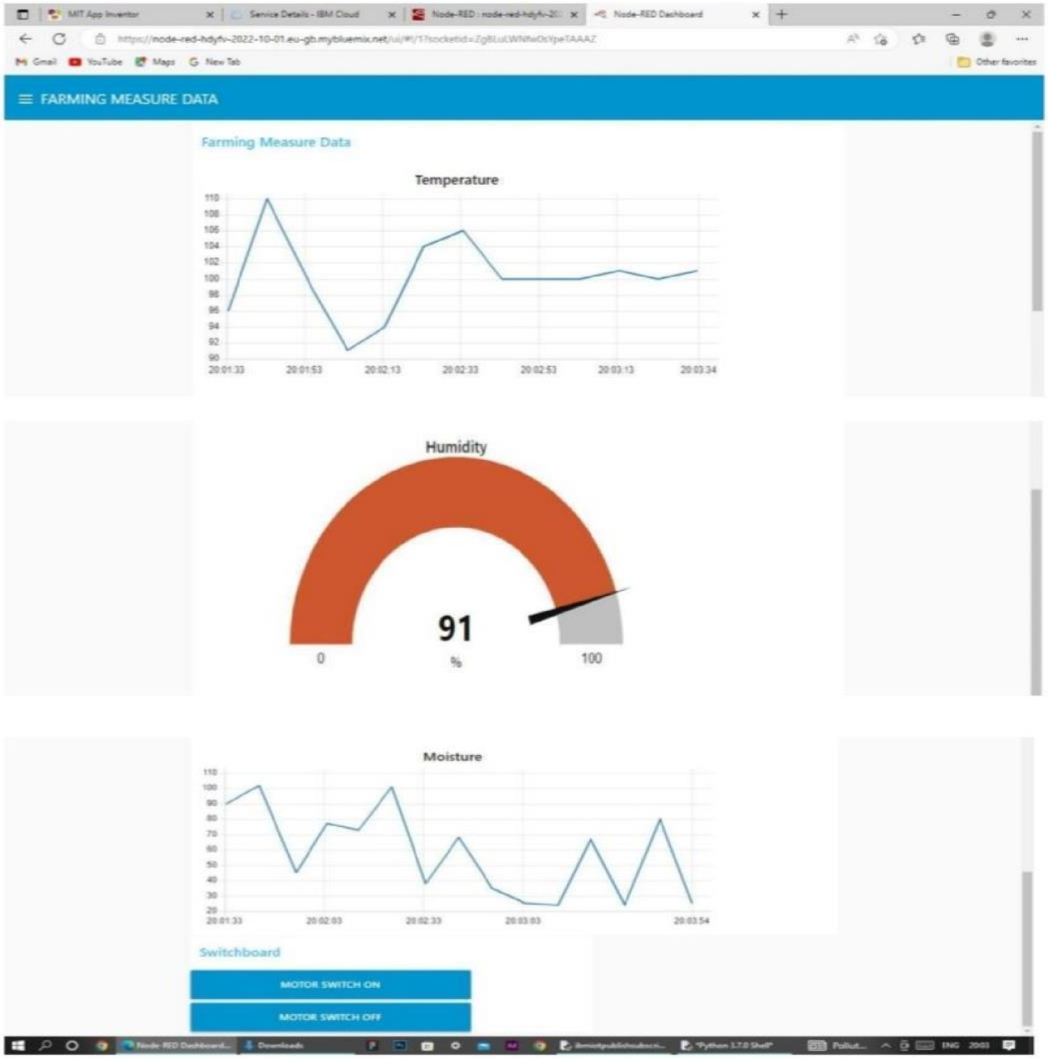
Screen 1:

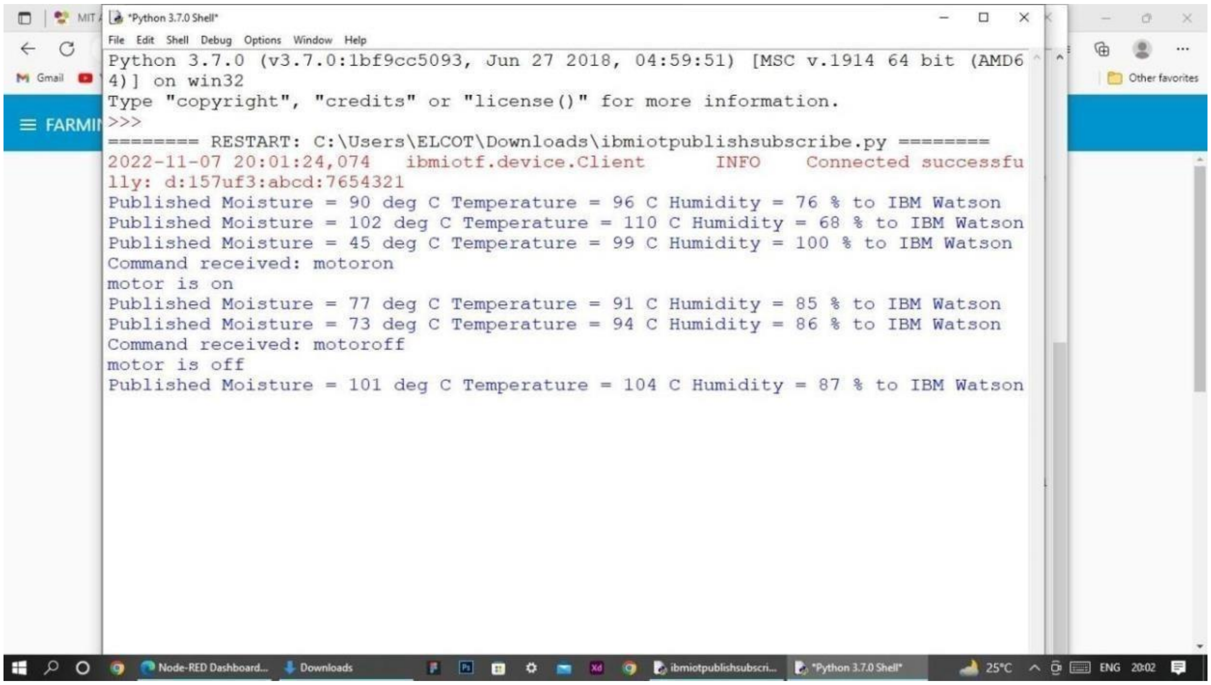


Screen1
AboutScreen:
AccentColor: Black
AlignHorizontal: Center : 3
AlignVertical: Center : 2
AppName: Smart_Farmer
BackgroundColor: None
BackgroundImage: WhatsAppImage2022-11-11
BigDefaultText: ☐
BlocksToolkit: All
CloseScreenAnimation: Fade
DefaultFileScope: App
HighContrast: ☐
Icon: None...
OpenScreenAnimation: Default
PrimaryColor

Web App UI:

Web APP UI Home Tab





```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
======== RESTART: C:\Users\ELCOT\Downloads\ibmiotpublishsubscribe.py ========
2022-11-07 20:01:24,074   ibmiotf.device.Client      INFO    Connected successfu
lly: d:157uf3:abcd:7654321
Published Moisture = 90 deg C Temperature = 96 C Humidity = 76 % to IBM Watson
Published Moisture = 102 deg C Temperature = 110 C Humidity = 68 % to IBM Watson
Published Moisture = 45 deg C Temperature = 99 C Humidity = 100 % to IBM Watson
Command received: motoron
motor is on
Published Moisture = 77 deg C Temperature = 91 C Humidity = 85 % to IBM Watson
Published Moisture = 73 deg C Temperature = 94 C Humidity = 86 % to IBM Watson
Command received: motoroff
motor is off
Published Moisture = 101 deg C Temperature = 104 C Humidity = 87 % to IBM Watson
```

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

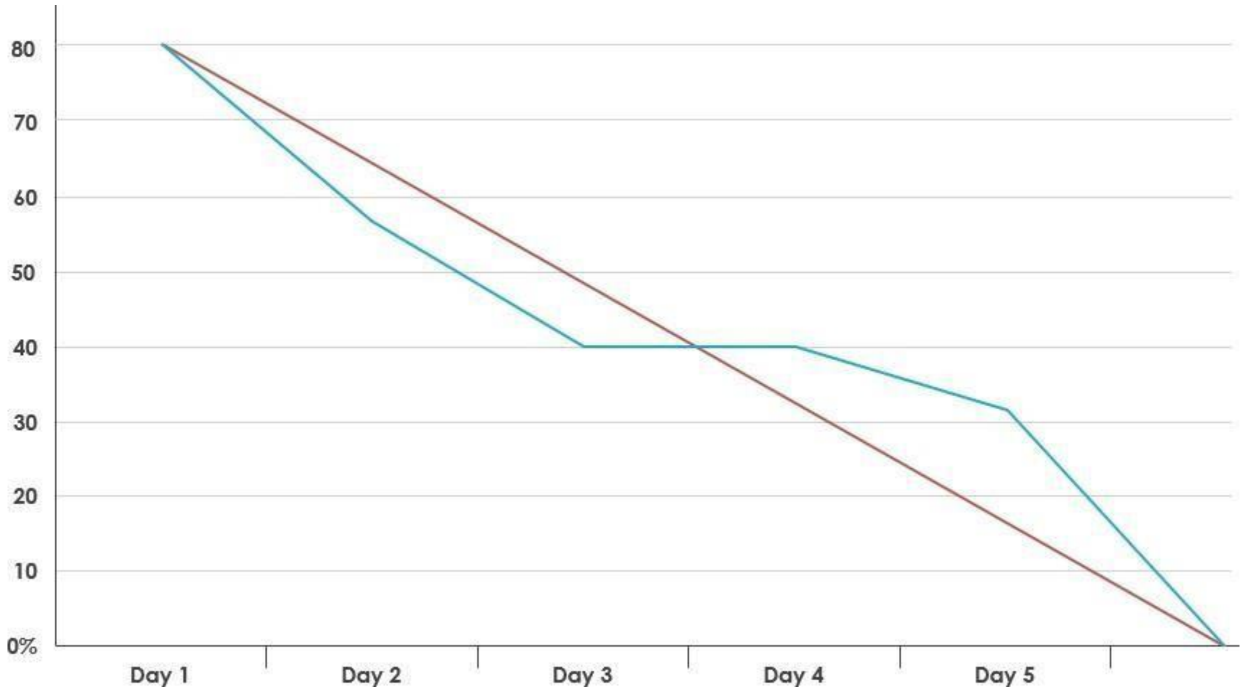
AV for sprint 1= Sprint Duration/Velocity=12/6=2

AV for sprint 2= Sprint Duration/Velocity=6/6=1

AV for Sprint 3=Sprint Duration/Velocity=6/6=1

AV for Sprint 4=Sprint Duration/Velocity=6/6=1

Burndown Chart:



7.1 Feature 1

Receiving commands from IBM cloud using Python program

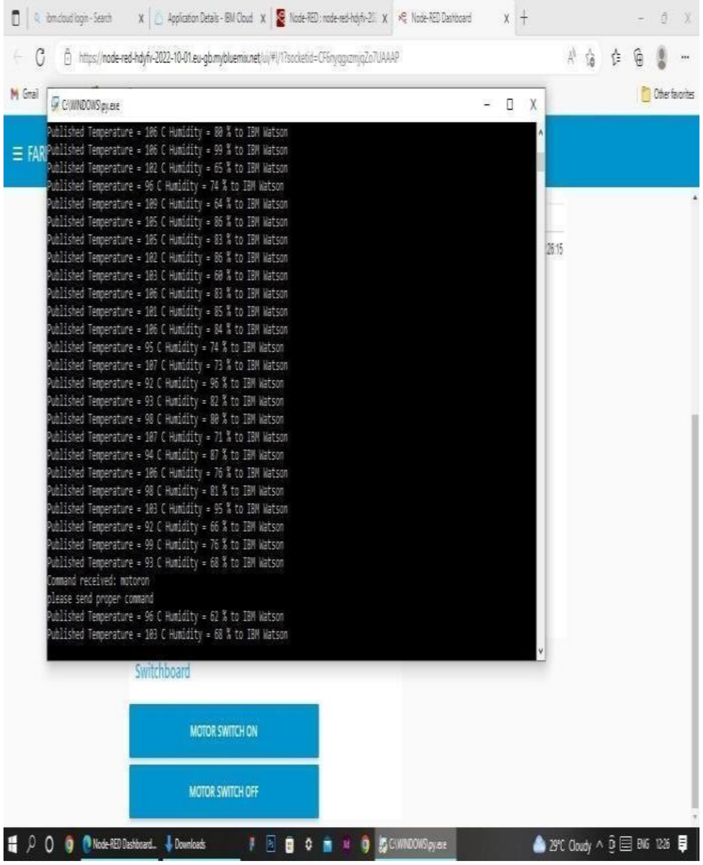
This is the Python code to receive commands from cloud to any device like Raspberry Pi in the farm

```
import time

import sys import

ibmiotf.applicatio

n import

        ibmiotf.d

  evice import
```

```
        random

      #Provide your IBM Watson Device Credentials organization = "157uf3"
      deviceType = "abcd" deviceId
      = "7654321" authMethod = "token" authToken = "87654321"


      # Initialize GPIO
      def
        myCommandCallbac
        k(cmd):
        print("Command
        received: %s" %
cmd.data['command'])


      status=cmd.data['command']
      if
      status=="motoron":
                  print
      ("motor is on")
                  elif status
      == "motoroff":
                  print
      ("motor is
        off")
        else :
            print ("please send proper command")


        try:


            deviceOptions = {"org": organization, "type":

  deviceType, "id": deviceId, "auth-

  method": authMethod, "authtoken":

  authToken}

            deviceCli =

            ibmiotf.device.Client(deviceOp

            tions)

            #..............................................


        except Exception as

print("Caught exception connecting device:

    %s" %

  str(e)) sys.exit()


        # Connect and send a datapoint

  "hello" with value "world" into the cloud

  as an event of type "greeting" 10 times
```

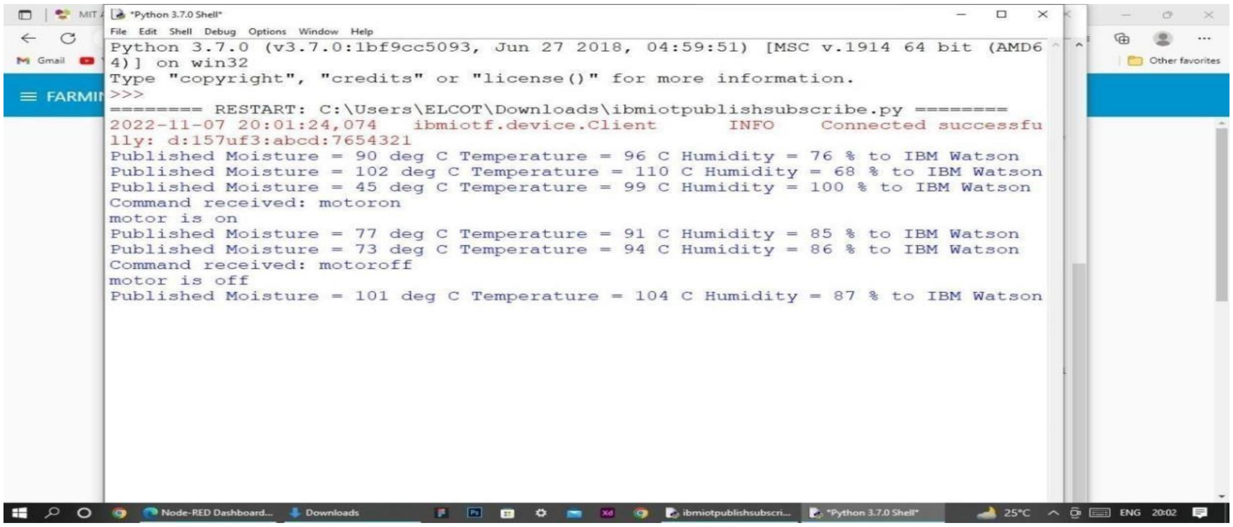
```
          deviceCli.connect()

          while True:
              #Get Sensor Data from DHT11



              temp=random.randint(90,110)

              Humid=random.randint(60,100)

        Mois=random.randint(20,120)

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


              success =
deviceCli.publishEvent("IoTSensor", "json",
data, qos=0, on_publish=myOnPublishCallback)
              if not success:
```

```
    print("Not connected
to IoTF")
                    time.slee
p(10)


    deviceCli.commandCallback = myCommandCallback


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



```
Published Temperature = 106 C Humidity = 80 % to IBM Watson
Published Temperature = 106 C Humidity = 99 % to IBM Watson
Published Temperature = 102 C Humidity = 65 % to IBM Watson
Published Temperature = 96 C Humidity = 74 % to IBM Watson
Published Temperature = 109 C Humidity = 64 % to IBM Watson
Published Temperature = 105 C Humidity = 86 % to IBM Watson
Published Temperature = 105 C Humidity = 83 % to IBM Watson
Published Temperature = 102 C Humidity = 86 % to IBM Watson
Published Temperature = 103 C Humidity = 60 % to IBM Watson
Published Temperature = 106 C Humidity = 83 % to IBM Watson
Published Temperature = 101 C Humidity = 85 % to IBM Watson
Published Temperature = 106 C Humidity = 84 % to IBM Watson
Published Temperature = 95 C Humidity = 74 % to IBM Watson
Published Temperature = 107 C Humidity = 73 % to IBM Watson
Published Temperature = 92 C Humidity = 96 % to IBM Watson
Published Temperature = 93 C Humidity = 82 % to IBM Watson
Published Temperature = 98 C Humidity = 80 % to IBM Watson
Published Temperature = 107 C Humidity = 71 % to IBM Watson
Published Temperature = 94 C Humidity = 87 % to IBM Watson
Published Temperature = 106 C Humidity = 76 % to IBM Watson
Published Temperature = 98 C Humidity = 81 % to IBM Watson
Published Temperature = 103 C Humidity = 95 % to IBM Watson
Published Temperature = 92 C Humidity = 66 % to IBM Watson
Published Temperature = 99 C Humidity = 76 % to IBM Watson
Published Temperature = 93 C Humidity = 68 % to IBM Watson
Command received: motoron
please send proper command
Published Temperature = 96 C Humidity = 62 % to IBM Watson
Published Temperature = 103 C Humidity = 68 % to IBM Watson
```

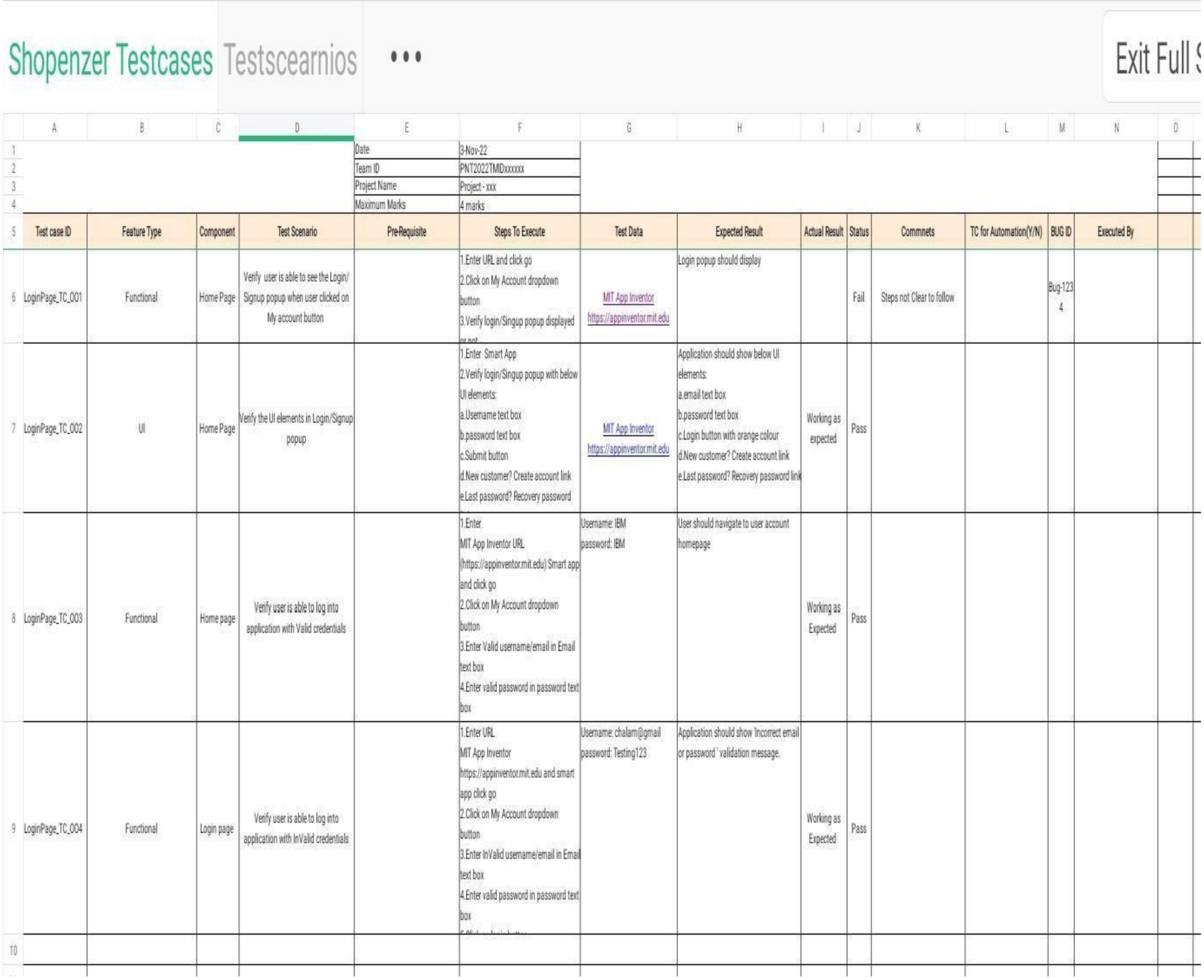
Feature 2:



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
======== RESTART: C:\Users\ELCOT\Downloads\ibmiotpublishsubscribe.py ========
2022-11-07 20:01:24,074   ibmiotf.device.Client      INFO    Connected successfu
lly: d:157uf3:abcd:7654321
Published Moisture = 90 deg C Temperature = 96 C Humidity = 76 % to IBM Watson
Published Moisture = 102 deg C Temperature = 110 C Humidity = 68 % to IBM Watson
Published Moisture = 45 deg C Temperature = 99 C Humidity = 100 % to IBM Watson
Command received: motoron
motor is on
Published Moisture = 77 deg C Temperature = 91 C Humidity = 85 % to IBM Watson
Published Moisture = 73 deg C Temperature = 94 C Humidity = 86 % to IBM Watson
Command received: motoroff
motor is off
Published Moisture = 101 deg C Temperature = 104 C Humidity = 87 % to IBM Watson
```

8.Testing:

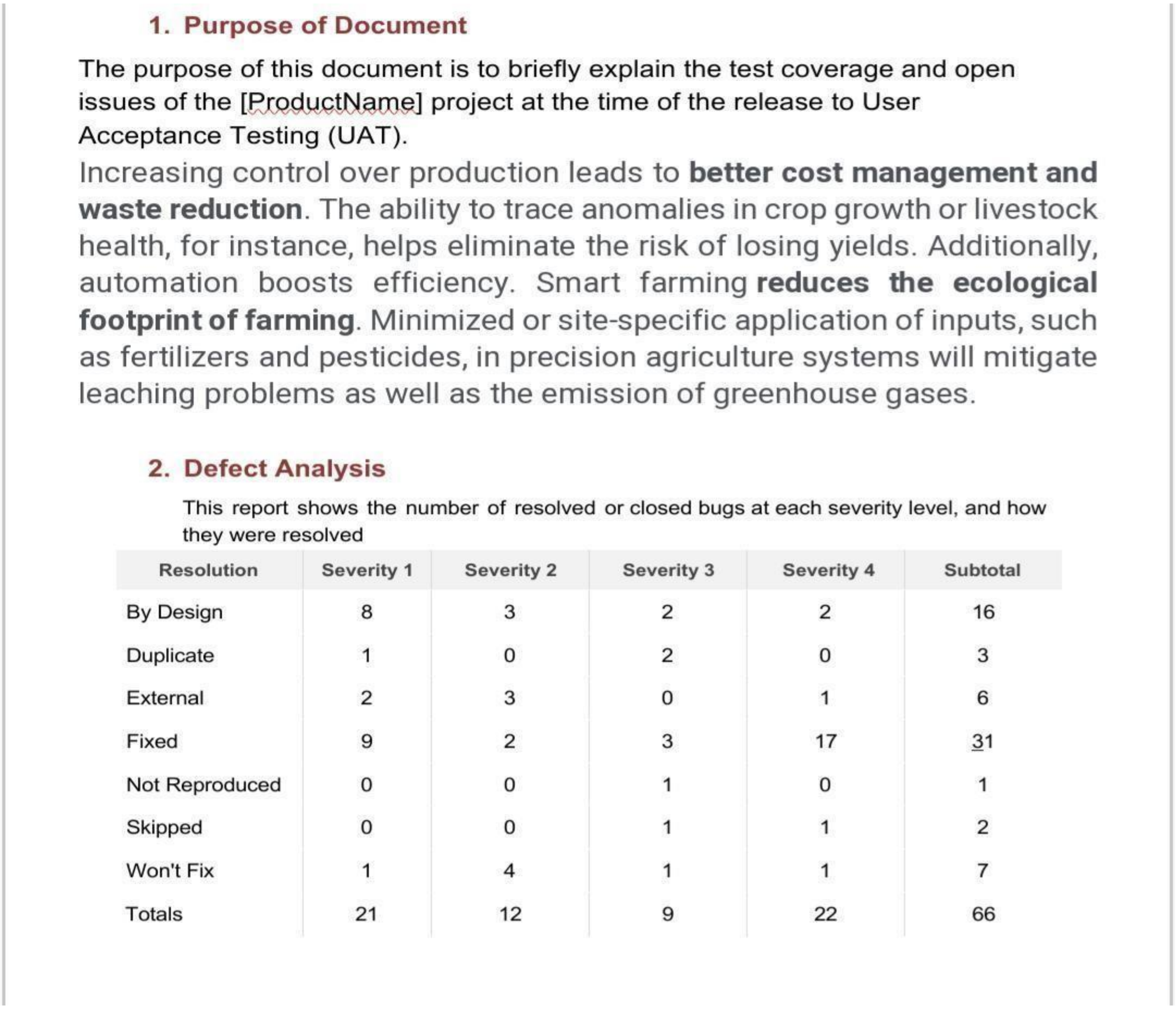
8.1.Test Cases



Date	3-Nov-22
Team ID	PNT2022TMIDxxxxxx
Project Name	Project - xxx
Maximum Marks	4 marks

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/ Signup popup when user clicked on My account button		1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup displayed or not	MIT App Inventor https://appinventor.mit.edu	Login popup should display		Fail	Steps not Clear to follow		Bug-1234	
LoginPage_TC_002	UI	Home Page	Verify the UI elements in Login/Signup popup		1.Enter Smart App 2.Verify login/Singup popup with below UI elements: a.Username text box b.password text box c.Submit button d.New customer? Create account link e.Last password? Recovery password	MIT App Inventor https://appinventor.mit.edu	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Working as expected	Pass				
LoginPage_TC_003	Functional	Home page	Verify user is able to log into application with Valid credentials		1.Enter MIT App Inventor URL (https://appinventor.mit.edu) Smart app and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box	Username: IBM password: IBM	User should navigate to user account homepage	Working as Expected	Pass				
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL MIT App Inventor https://appinventor.mit.edu and smart app click go 2.Click on My Account dropdown button 3.Enter InValid username/email in Email text box 4.Enter valid password in password text box	Username: chalam@gmail password: Testing123	Application should show 'Incorrect email or password ' validation message.	Working as Expected	Pass				

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 [ProductName] project at the time of the release to User Acceptance Testing (UAT).

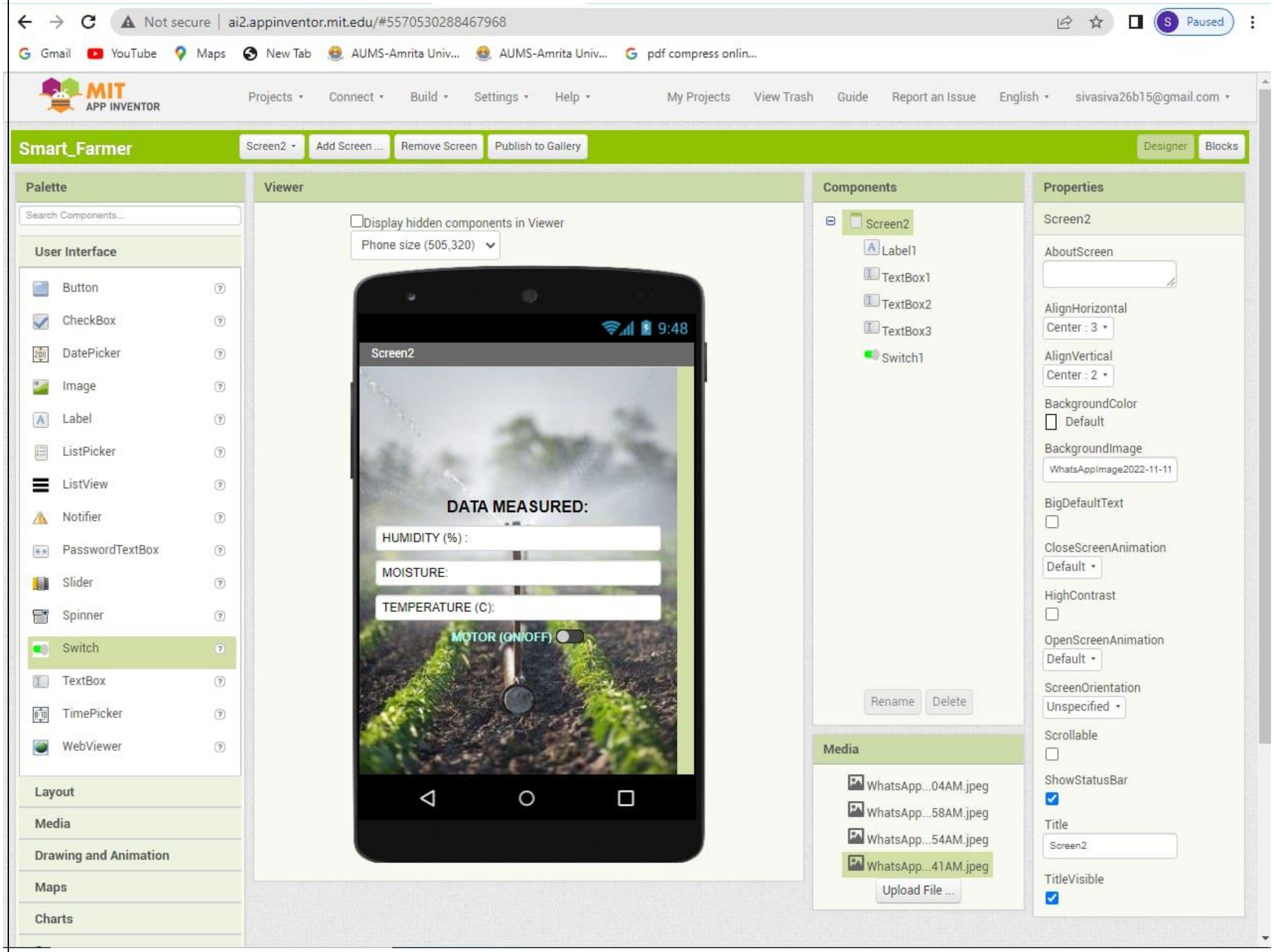
Increasing control over production leads to **better cost management and waste reduction**. The ability to trace anomalies in crop growth or livestock health, for instance, helps eliminate the risk of losing yields. Additionally, automation boosts efficiency. Smart farming **reduces the ecological footprint of farming**. Minimized or site-specific application of inputs, such as fertilizers and pesticides, in precision agriculture systems will mitigate leaching problems as well as the emission of greenhouse gases.

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	8	3	2	2	16
Duplicate	1	0	2	0	3
External	2	3	0	1	6
Fixed	9	2	3	17	31
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	1	4	1	1	7
Totals	21	12	9	22	66

9. Results :



DATA MEASURED:

HUMIDITY (%) :

MOISTURE:

TEMPERATURE (C):

MOTOR (ON/OFF)

AboutScreen

AlignHorizontal: Center : 3

AlignVertical: Center : 2

BackgroundColor: □ Default

BackgroundImage: WhatsAppImage2022-11-11

BigDefaultText: □

CloseScreenAnimation: Default

HighContrast: □

OpenScreenAnimation: Default

ScreenOrientation: Unspecified

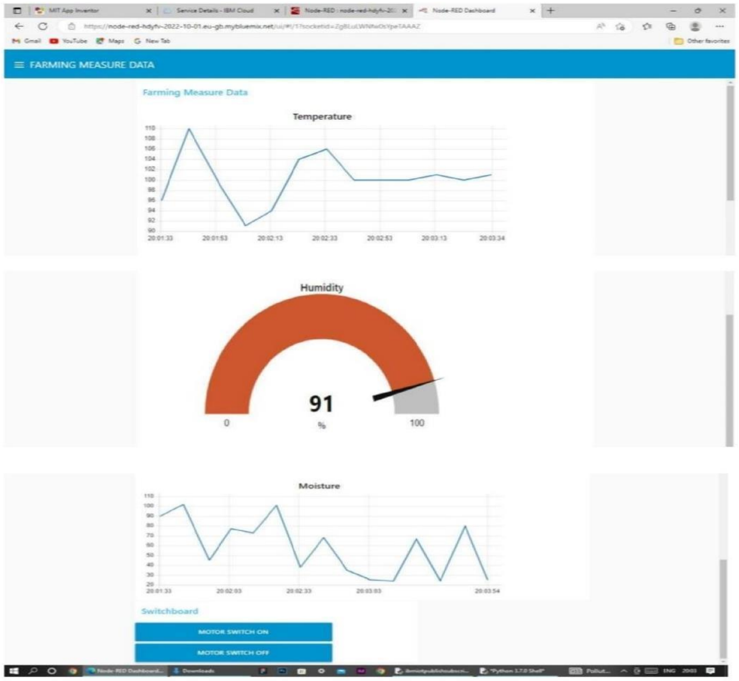
Scrollable: □

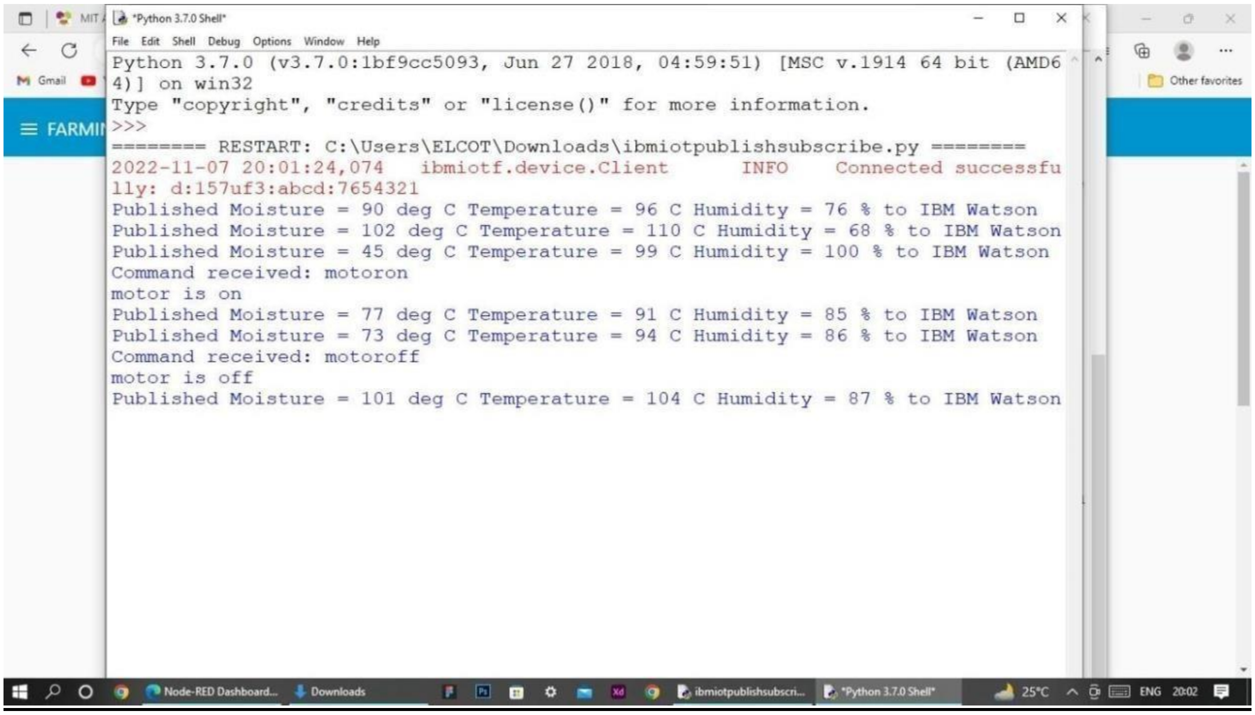
ShowStatusBar: ☑

Title: Screen2

TitleVisible: ☑

Web APP UI Home Tab





```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
======== RESTART: C:\Users\ELCOT\Downloads\ibmiotpublishsubscribe.py ========
2022-11-07 20:01:24,074   ibmiotf.device.Client      INFO    Connected successfu
lly: d:157uf3:abcd:7654321
Published Moisture = 90 deg C Temperature = 96 C Humidity = 76 % to IBM Watson
Published Moisture = 102 deg C Temperature = 110 C Humidity = 68 % to IBM Watson
Published Moisture = 45 deg C Temperature = 99 C Humidity = 100 % to IBM Watson
Command received: motoron
motor is on
Published Moisture = 77 deg C Temperature = 91 C Humidity = 85 % to IBM Watson
Published Moisture = 73 deg C Temperature = 94 C Humidity = 86 % to IBM Watson
Command received: motoroff
motor is off
Published Moisture = 101 deg C Temperature = 104 C Humidity = 87 % to IBM Watson
```

10. Advantages &Disadvantages:

Advantages:

- Smart farming increases the production, efficiency and protection of crops. This enables the farmers to manage their fields remotely via smart gadgets, which will save lot of resources and time. With the help of automation and sensor technology, benefits the society by conservation of water, resource management and better crop yield.
- Real time monitoring allows the user to know the present situation with just an application. The data collected can help the farmers to predict the weather in the upcoming years. The application is very user friendly and so anyone can learn how to use it in very short duration.

Disadvantages:

- In some rural areas the network connectivity is poor.
- Rural people express a bit of fear in using technology.
- Some people does not know how to use the smart device.

11. CONCLUSION:

Our project system has customized function for different types of crops. Each mode comes with its own set of features depending on the crop. This system monitors some parameters of the crop using the sensors and determine the soil moisture, humidity and depending on the need, it automates water supply. The sensors used here is less suspected to environmental conditions therefore giving long lasting performance. Moisture sensor probes are coated with an immersion gold that protects the Nickel probes from oxidation. Several sensors can be used to identify the presence of excess water in the field. This feature is not present in many of the recent Farming applications.

12.Future Scope:

Through collecting data from sensors using IoT devices, you will learn about the real-time state of your crops.The future of IoT in agriculture allows predictive analytics to help you make better harvesting decisions. Smart farming refers to managing farms using modern Information and communication technologies to inrcrease the quantity and quality of products while optimizing the human labour required

10.Appendix :

Source code :

Link : **https://drive.google.com/drive/folders/1FsO4p7lg_V2FPOv8Ca-Hj2P0cMkzOttH**

Links:

IBM cloud reference: **https://cloud.ibm.com/**

Git hub link :

https://github.com/IBM-EPBL/IBM-Project-43393-660716563

IOT Watson simulator :

https://g4wj2o.internetofthings.ibmcloud.com/dashboa

Node-Red : **https://127.0.1:1880**

MIT App Inventor :

http://ai2.appinventor.mit.edu/#5570530288467968