# SMART FARMING – IoT ENABLED SMART FARMING APPLICATION

**A PROJECT REPORT**

***Submitted by***

| | |
|---|---|
| **BHARATHI A** | **(711119104015)** |
| **BHARATH NARAYANAN S** | **(711119104016)** |
| **KAVIPRIYA R** | **(711119104034)** |
| **KEERTHANA K** | **(711119104036)** |

***in partial fulfilment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

***In***

**COMPUTER SCIENCE AND ENGINEERING**

**JANSONS INSTITUTE OF TECHNOLOGY, COIMBATORE**

**ANNA UNIVERSITY: CHENNAI-600 025**

# INDEX

# 1. INTRODUCTION

## Introduction to Smart Farming Application

Smart Farming has enabled farmers to reduce waste and enhance productivity with the help of sensors (light, humidity, temperature, soil moisture, etc.) and automation of irrigation systems. Further with the help of these sensors, farmers can monitor the field conditions from anywhere. Internet of Things based Advanced Farming is highly efficient when compared with the conventional approach. The applications of intelligent Agriculture solutions not only targets conventional, large farming. With operations, but could also be new levers to uplift other growing or common trends in agricultural like organic farming, family farming (complex or small spaces, particular cattle and/or cultures, preservation of specific or high-quality varieties, etc.), and enhance highly transparent Farming. Automatic adjustment of farming equipment made possible by linking information like crops/weather and equipment to auto-adjust temperature, humidity, etc. In large farmland, Internet of Things equipped drone helps to receive the current state of crops and send the live pictures of farmland. Analysing farmland from the land using its Solutions you will know the current situation of fields and crops in.

## 1.1Project Overview:

In Internet of Things based smart agriculture, a system is formed to monitor the farmland with the help of sensors, which senses components like temperature, light, humidity, soil moisture, etc. Then, automate the irrigation system and allow farmers to monitor their field conditions from anywhere through IoT Analytics Platform. To make the agricultural process even smarter and accurate, precision agriculture is used. This makes agricultural practice more controlled and precise in terms of raising livestock and farming. Internet of Things based Advanced Farming plays a vital role when it comes to the use of IT and other elements like sensors, agricultural drones, autonomous vehicles, control systems, automated hardware, robotics, variable speed technology, and others. he most common challenge for the Internet of Things in agriculture is connectivity. Every area doesn’t have proper internet connectivity. The second most common challenge for Internet of Things based Advanced Farming is the lack of awareness among consumers. Due to various service providers, it becomes really difficult to maintain interoperability between different IoT systems. A scalable solution that can be integrated with thousands of IoT devices for large farms is seen as a challenge.

## 1.2Purpose:

Internet of Things Smart technology enables new digital agriculture. Today technology has become a necessity to meet current challenges and several sectors are using the latest technologies to automate their tasks. Advanced agriculture, based on Internet of Things technologies, is envisioned to enable producers and farmers to reduce waste and improve productivity by optimizing the usage of fertilizers to boost the efficiency of plants. It gives better control to the farmers for their livestock, growing crops, cutting costs, and resources. The world's total population touched 6.60 billion in 2000 but is projected to grow to 9.32 billion by 2050. Hence, it is necessary to increase the yield on the limited farmland. It is a high-tech system to grow crop cleanly and sustainably for the masses. It is the application of modern Information and Communication Technologies in agriculture. The parameters like temperature, humidity, and soil moisture are updated to the Watson IoT platform. The device will subscribe to the commands from the mobile application and control the motors accordingly. APIs are developed using Node-RED service for communicating with Mobile Application. A mobile application is developed using the MIT App inventor to monitor the sensor parameters and control the motors.

# 2. LITERATURE SURVEY

## Review – 1:

**Title of the paper:** DIGITAL TECHNOLOGIES IN AGRICULTURE AND RURAL AREAS

**Author:** Nikola M. Trendov, Samuel Varas, and Meng Zeng

**Outcome:** The agriculture and food sector is facing multiple challenges. With the global population projected to grow from 7.6 billion in 2018 (UN DESA, 2019) to over 9.6 billion in 2050 there will be a significant increase in the demand for food (UN DESA, 2017). At the same time, the availability of natural resources such as fresh water and productive arable land is becoming increasingly constrained. Production is not the only concern; although agricultural output is currently enough to feed the world, 821 million people still suffer from hunger (FAO, 2018). Processes such as the rapid rate of urbanization are also having important implications for patterns of food production and consumption. The agri-food sector remains critical for livelihoods and employment. There are more than 570 million smallholder farms worldwide (Louder et al., 2016) and agriculture and food production accounts for 28% of the entire global workforce (ILOSTAT, 2019). Achieving the UN Sustainable Development Goal of a 'world with zero hunger' by 2030 will require more productive, efficient, sustainable, inclusive, transparent and resilient food systems (FAO, 2017b p. 140). This will require an urgent transformation of the current agri-food system

## Review – 2:

**Title of the Paper:** Agriculture 4.0: Broadening Responsible Innovation in an Era of Smart Farming

**Author:** Rose, David Christian; Chilvers, Jason

**Outcome:** Agriculture is undergoing a new technology revolution supported by policy-makers around the world. While smart technologies, such as Artificial Intelligence, robotics, and the Internet of Things, could play an important role in achieving enhanced productivity and greater eco-efficiency, critics have suggested that the consideration of social implications is being side-lined. Research illustrates that some agricultural practitioners are concerned about using certain smart technologies. Indeed, some studies argue that agricultural societies may be changed, or "re-scripted," in undesirable ways, and there is precedent to suggest that wider society may be concerned about radical new agricultural technologies. We therefore encourage policy-makers, funders, technology companies, and researchers to consider the views of both farming communities and wider society. In agriculture, the concept of responsible innovation has not been widely considered, although two recent papers have made useful suggestions. We build on these interventions by arguing that key dimensions of responsible innovation—anticipation, inclusion, reflexivity, and responsiveness—should be applied to this fourth agricultural revolution. We argue, however, that ideas of responsible innovation should be further developed in order to make them relevant and robust for emergent Agri-tech, and that frameworks should be tested in practice to see if they can actively shape innovation trajectories. In making suggestions on how to construct a more comprehensive framework for responsible innovation in sustainable agriculture, we call for a more systemic approach that maps and attends to the wider ecology of innovations associated with this fourth agricultural revolution; a broadening of notions of "inclusion" in responsible innovation to account better for diverse and already existing spaces of participation in Agri-tech.

## Review – 3:

**Title of the Paper:** Smart Agriculture System using IoT Technology

**Author:** Adithya Vadapalli

**Outcome:** The farming of agriculture has started past 12000 years back, Neolithic age gave birth of civilization, Farming and later being continued as traditional farming practices. India being an agrarian's country, Mostly Indian farming are dependent on rains, soil, dampness and environment challenges. Our farmers upgraded to modern state of art technology in cultivation. Globally the IoT systems has contributed its application in many fields and proven to be successful. It is the time that Indian farmer need to introduce the Smart Agricultural systems for higher crop yield. The productivity with compilation of data from sensors, actuators and modern electronic gadgets the farmer can monitor agricultural fields. Smart Agriculture can forecast weather data, switching ON the pump motor acknowledging the dampness of soil terms of moisture levels with help of sensors which are interfaced to process module Arduino-UNO. The Smart agriculture system can be operated from anywhere with help of networking technology. On joining process in research and development in Smart Agriculture& Artificial Intelligence can be cutting edge technology in data compiling and resource optimization. The pest & insect's controls that protects damaging the crop and also optimisation resources utilisation can be breakthrough

## Review – 4:

**Title of the Paper:** Smart Farming: IoT Based Smart Sensors Agriculture Stick for Live Temperature and Moisture Monitoring using Arduino, Cloud Computing & Solar Technology

**Author:** Er. Vikram Puri

**Outcome:** Internet of Things (IoT) technology has brought revolution to each and every field of common man's life by making everything smart and intelligent. IoT refers to a network of things which make a self-configuring network. The development of Intelligent Smart Farming IoT based devices 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 paper is to propose a Novel Smart IoT based Agriculture Stick assisting farmers in getting Live Data (Temperature, Soil Moisture) for efficient environment monitoring which will enable them to do smart farming and increase their overall yield and quality of products. The agriculture stick being proposed via this paper is integrated with Arduino Technology, Breadboard mixed with various sensors and live data feed can be obtained online from Thingsspeak.com. The product being proposed is tested on Live Agriculture Fields giving inaccuracy over 98% in data feeds

### Review – 5:

**Title of the Paper:** A RESEARCH PAPER ON SMART AGRICULTURE USING IOT

**Author:** Ritika Srivastava, Vandana Sharma, Vishal Jaiswal, Sumit Raj

**Outcome:** Smart agriculture is an emerging concept, because IOT sensors are capable of providing information about agriculture fields and then act upon based on the user input. The feature of this paper includes development of a system which can monitor temperature, level of water, moisture and even the movement if any happens in the field which may destroy the crops in agricultural field through sensors using Arduino UNO board. Smart agriculture is an emerging concept, because IOT sensors are capable of providing information about agriculture fields and then act upon based on the user input. The project aims at making use of evolving technology i.e., IOT and smart agriculture using automation. Once hardware has been developed depending on the change in requirements and technology the software needs the updating. The updated hardware is called new version of the software. This new version is required to be tested in order to ensure changes that are made in the old version work correctly and it will not bring bugs in other part of the software. This is necessary because updating in one part of the hardware may bring some undesirable effects in other part of the hardware

## 2.1 Existing problem:

Choosing sensor brands and types of sensors are typically the easiest tasks when building a smart agriculture system. Setting up monitoring and automation software packages to implement actions, such as watering triggers, alarms and notifications, will require software expertise that some agricultural experts will find challenging. For example, how often should a crop be watered? What is the minimum level of soil moisture that should be permitted before starting a watering routine? Software integration with sensors and agricultural equipment is a time-consuming activity, but also one where seasonal conditions and crop expertise need to be considered.

**The Challenges of Smart Farming application:**

1.The second most common challenge for Internet of Things based Advanced Farming is the lack of awareness among consumers.

2.The most common challenge for the Internet of Things in agriculture is connectivity. Every area doesn’t have proper internet connectivity.

3.A unified solution which can be integrated with different types of Internets of Things devices.

## 2.2 References:

https://miro.com/app/board/uXjVPIvbQCY=/

## 2.3 Problem Statement Definition:

Create a problem statement to understand user’s needs and requirements. Large farm owners can benefit from IoT applications to collect data regarding the location, well-being, and health of their cattle. To provide efficient decision support system using wireless sensor network which handle different activities of farm and gives useful information related to farm field to farmer

| Problem Statement (PS) | I am (Customer) | I’m trying to | But | Because | Which makes me feel |
|---|---|---|---|---|---|
| PS-1 | Smart Farmer | To make the user handle easily and remote | They must have the knowledge to use the device | Integration of sensors analytics driving automation and response activities | Being Monitored |

# 3. IDEATION & PROPOSED SOLUTION

## 3.1 Empathy Map Canvas:

An empathy map helps to map what a design team knows about the potential audience. This tool helps to understand the reason behind some actions a user takes deeply. This tool helps build Empathy towards users and helps design teams shift focus from the device to the users who are going to use the device. As the team learns more about the users, they place that information on the chart and gain an in-depth view of the user behavior, problems, opportunities.

The benefits include are below:

More understanding of the Target Audience

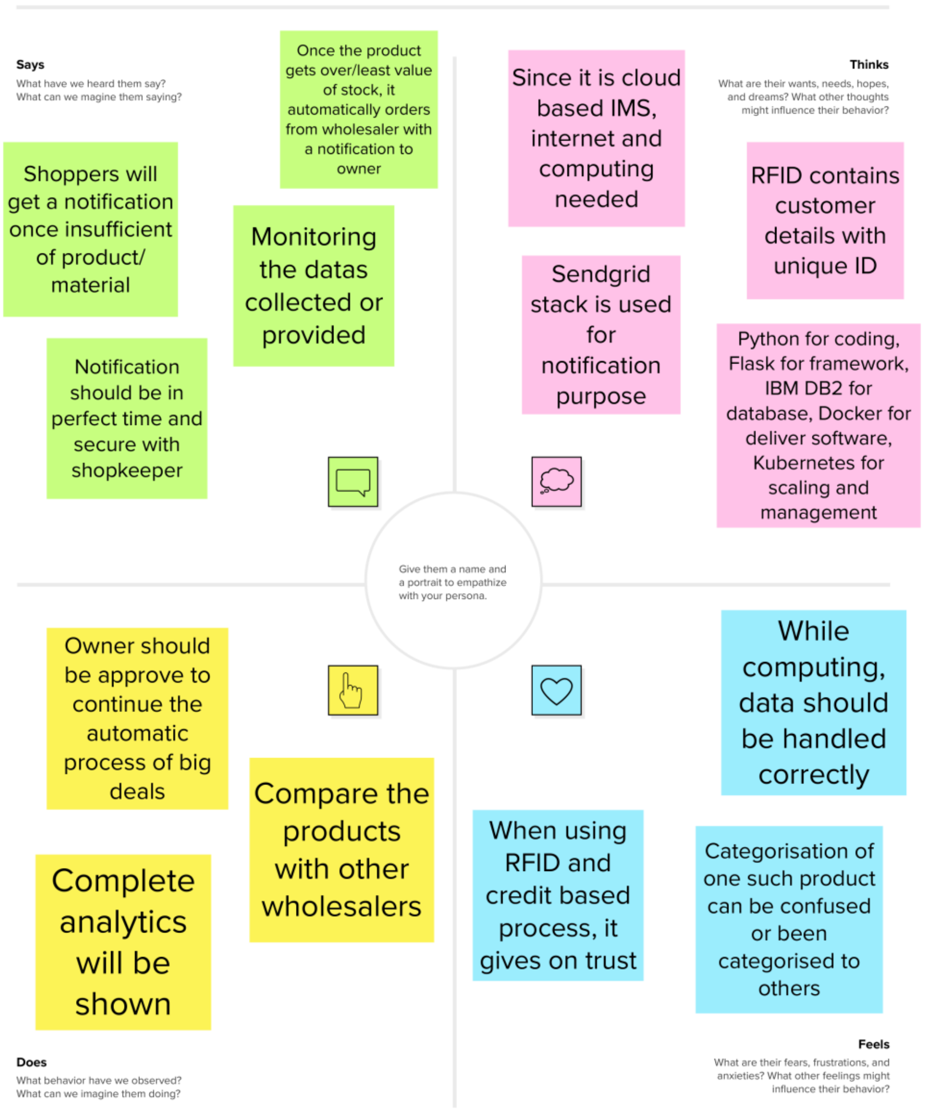
More Organized Information in easy to understand format

Fast and Inexpensive

Easy Customization

Common Understanding and same mindset of whole team members

It describes what users **think, say, feel, do**



**Says**

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

Once the product gets over/least value of stock, it automatically orders from wholesaler with a notification to owner

Shoppers will get a notification once insufficient of product/ material

Monitoring the datas collected or provided

Notification should be in perfect time and secure with shopkeeper

**Thinks**

What are their wants, needs, hopes, and dreams? What other thoughts might influence their behavior?

Since it is cloud based IMS, internet and computing needed

RFID contains customer details with unique ID

Sendgrid stack is used for notification purpose

Python for coding, Flask for framework, IBM DB2 for database, Docker for deliver software, Kubernetes for scaling and management

Give them a name and a portrait to empathize with your persona.

Owner should be approve to continue the automatic process of big deals

Compare the products with other wholesalers

Complete analytics will be shown

**Does**

What behavior have we observed?
What can we imagine them doing?

While computing, data should be handled correctly

When using RFID and credit based process, it gives on trust

Categorisation of one such product can be confused or been categorised to others

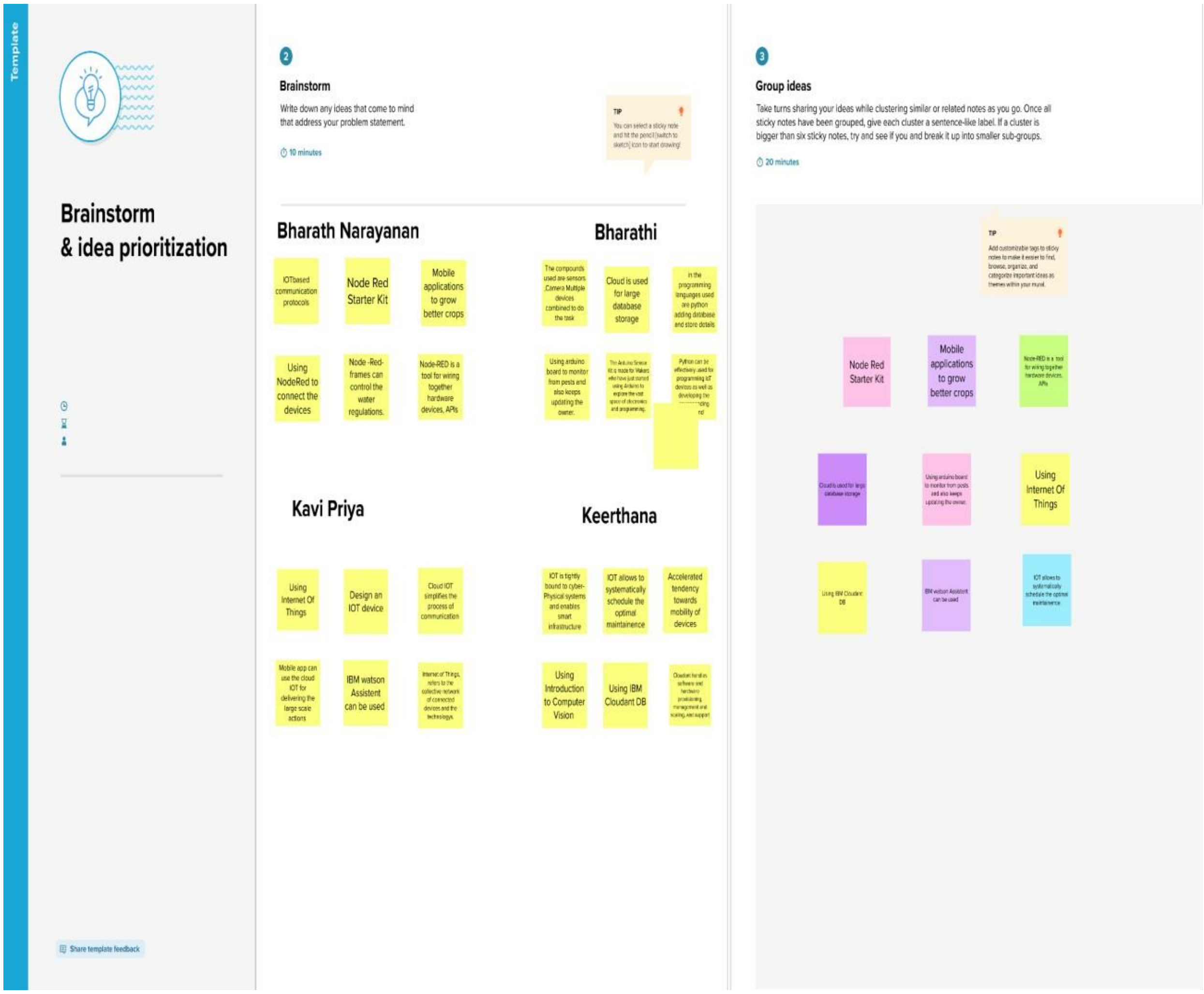
**Feels**

What are their fears, frustrations, and anxieties? What other feelings might influence their behavior?

## 3.2 Ideation & Brainstorming:

Brainstorming is when you deliberately try to think up new ideas or solutions to problems. In writing—whether creative, academic, or business—it's a beneficial preliminary stage that helps writers know precisely what's going into their projects. Ideas are the most valuable resource in any communication, which makes brainstorming for writing a crucial part of the process. But for people who mostly wait around for ideas to find them, brainstorming can be quite difficult or even frustrating.

# Step-1: Brainstorm, Idea Listing and Grouping



Template

Brainstorm & idea prioritization

Share template feedback

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!

Bharath Narayanan

IOTbased communication protocols

Node Red Starter Kit

Mobile applications to grow better crops

Using NodeRed to connect the devices

Node -Red- frames can control the water regulations.

Node-RED is a tool for wiring together hardware devices, APIs

Bharathi

The compounds used are sensors ,Camera Multiple devices combined to do the task

Cloud is used for large database storage

in the programming languges used are python adding database and store details

Using arduino board to monitor from pests and also keeps updating the owner.

The Arduino Sensor Kit is made for Makers who have just started using Arduino to explore the vast space of electronics and programming.

Python can be effectively used for programming IoT devices as well as developing the ...ing ...nd

Kavi Priya

Using Internet Of Things

Design an IOT device

Cloud IOT simplifies the process of communication

Mobile app can use the cloud IOT for delivering the large scale actions

IBM watson Assistent can be used

Internet of Things, refers to the collective network of connected devices and the technology.

Keerthana

IOT is tightly bound to cyber-Physical systems and enables smart infrastructure

IOT allows to systematically schedule the optimal maintainence

Accelerated tendency towards mobility of devices

Using Introduction to Computer Vision

Using IBM Cloudant DB

Cloudant handles software and hardware provisioning, management and scaling, and support

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.

Node Red Starter Kit

Mobile applications to grow better crops

Node-RED is a tool for wiring together hardware devices, APIs

Cloud is used for large database storage

Using arduino board to monitor from pests and also keeps updating the owner.

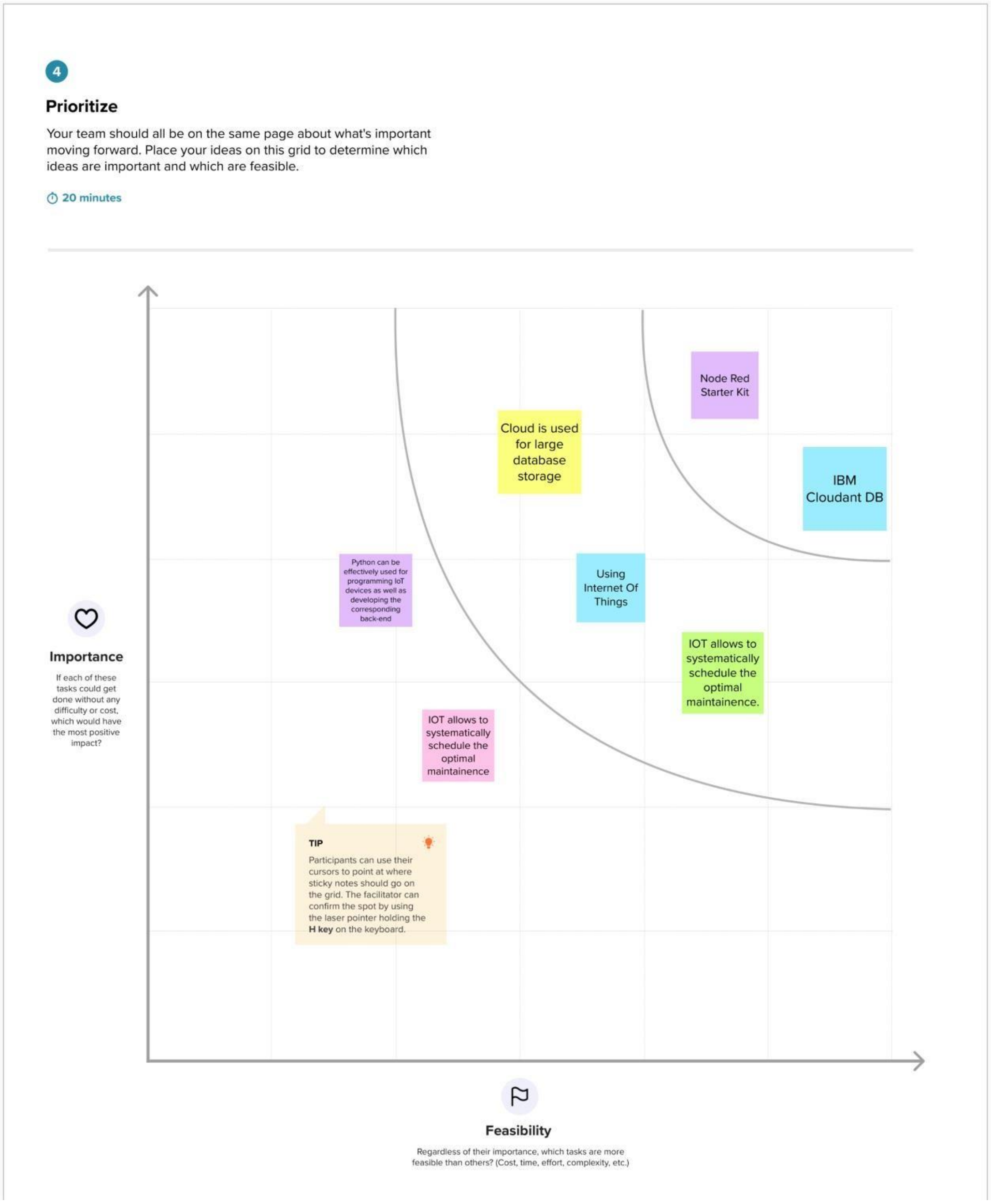
Using Internet Of Things

Using IBM Cloudant DB

IBM watson Assistent can be used

IOT allows to systematically schedule the optimal maintainence

# Step-2: Idea Prioritization



4

Prioritize

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

20 minutes

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Node Red Starter Kit
Cloud is used for large database storage
IBM Cloudant DB
Python can be effectively used for programming IoT devices as well as developing the corresponding back-end
Using Internet Of Things
IOT allows to systematically schedule the optimal maintainence.
IOT allows to systematically schedule the optimal maintainence

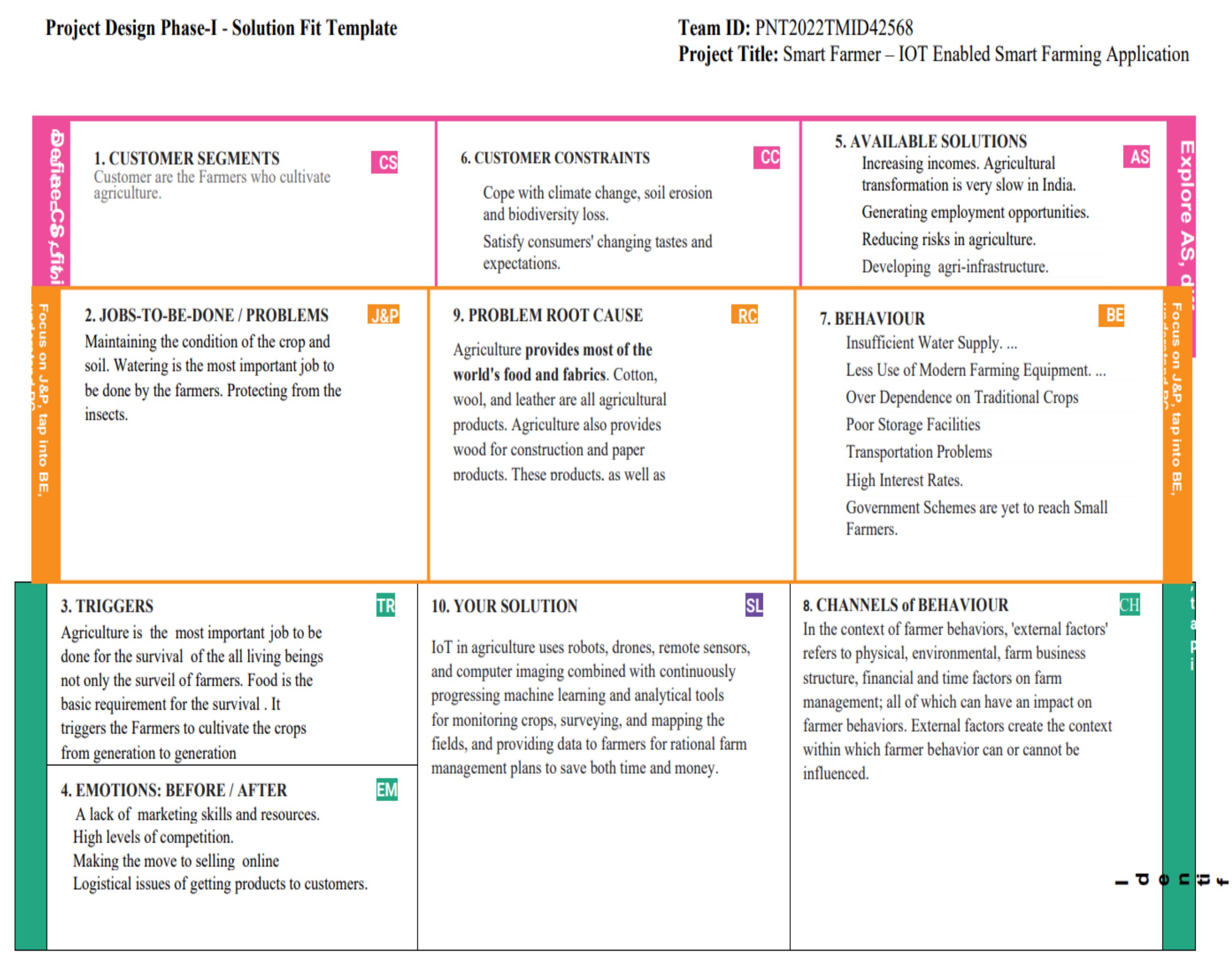
## 3.3 Proposed Solution:

Project team shall fill the following information in proposed solution.

| S.No | Parameter | Description |
|---|---|---|
| • | Problem Statement (Problem to be solved) | To solve farmer issues like<br>• Lack of Modernization andMechanization<br>• Invest in farm productivity andimproving yield production.<br>• Cope with climate change, soilerosion<br>• Watering the field is a difficult process, Farmers have to wait in the field until the water covers the wholefarm field.<br>• Power Supply is also one of the problems. In Village Side, the powersupply may vary.<br>The Biggest Challenges Faced by IoTin the Agricultural Sector are Lack ofInformation, High Adoption, Cost and Security Concerns, etc |
| • | Idea / Solution description | • Smart Farming systems uses moderntechnology to increase the quantity and quality of agricultural products.<br>• Livestock tracking and Geo fencing. Smart logistics and warehousing. Smart pest management. SmartGreenhouses |
| • | Novelty / Uniqueness | □ Smart farming systems reduce waste, improve productivity and enable management of a greater number of resources through remote sensing<br>□ . In traditional farming methods, it was a mainstay for the farmer to be out in the field, constantly monitoring the land and condition of crops.<br>**REMOTE ACCESS –** It helps the farmer to operate the motor from anywhere. |

| | | |
|---|---|---|
| | | **ALERT MESSAGE** – IoT sensor nodes collect information from the farming environment, such as soil moisture, air humidity, temperature,nutrient ingredients of soil, pest images, and water quality, then transmit collected data to IoT backhaul devices |
| 4. | Social Impact / Customer Satisfaction | • Reduces the wages for labors whowork in the agricultural field.<br>• It saves a lot of time.<br>• IoT can help improve customer relationships by enhancing the customer's overall experience.<br>• Easily identify maintenance needs,build better products, send personalized communications, andmore.<br>• IoT can also help e-commerce businesses thrive and increase sales.<br>It makes a wealthy society |
| 5. | Business Model (Revenue Model) | It's a subscription model, where user haveto pay for their internet.<br>• Customer services are supported<br>• It supports third party devices also<br>Reach customers via Referral, Agents,Third party applications |
| 6. | Scalability of the Solution | • Scalability in smart farming refers to the adaptability of a system to increase the capacity, for example, the number of technology devices such as sensors and actuators, while enabling timely analysis.<br>• Our product is scalable with our devices (extra addons) as well as third party devices also. Ability toprovides various features in a application like reports generation etc. |

# 3.4 Problem Solution fit:



Define CS, fit into
Focus on J&P, tap into BE,
Explore AS, d
Focus on J&P, tap into BE,
Identif

**Project Design Phase-I - Solution Fit Template**

**Team ID:** PNT2022TMID42568
**Project Title:** Smart Farmer – IOT Enabled Smart Farming Application

**1. CUSTOMER SEGMENTS** CS

Customer are the Farmers who cultivate agriculture.

**6. CUSTOMER CONSTRAINTS** CC

Cope with climate change, soil erosion and biodiversity loss.

Satisfy consumers' changing tastes and expectations.

**5. AVAILABLE SOLUTIONS** AS

Increasing incomes. Agricultural transformation is very slow in India.

Generating employment opportunities.

Reducing risks in agriculture.

Developing agri-infrastructure.

**2. JOBS-TO-BE-DONE / PROBLEMS** J&P

Maintaining the condition of the crop and soil. Watering is the most important job to be done by the farmers. Protecting from the insects.

**9. PROBLEM ROOT CAUSE** RC

Agriculture **provides most of the world's food and fabrics**. Cotton, wool, and leather are all agricultural products. Agriculture also provides wood for construction and paper products. These products, as well as

**7. BEHAVIOUR** BE

Insufficient Water Supply. ...

Less Use of Modern Farming Equipment. ...

Over Dependence on Traditional Crops

Poor Storage Facilities

Transportation Problems

High Interest Rates.

Government Schemes are yet to reach Small Farmers.

**3. TRIGGERS** TR

Agriculture is the most important job to be done for the survival of the all living beings not only the surveil of farmers. Food is the basic requirement for the survival . It triggers the Farmers to cultivate the crops from generation to generation

**4. EMOTIONS: BEFORE / AFTER** EM

A lack of marketing skills and resources.
High levels of competition.
Making the move to selling online
Logistical issues of getting products to customers.

**10. YOUR SOLUTION** SL

IoT in agriculture uses robots, drones, remote sensors, and computer imaging combined with continuously progressing machine learning and analytical tools for monitoring crops, surveying, and mapping the fields, and providing data to farmers for rational farm management plans to save both time and money.

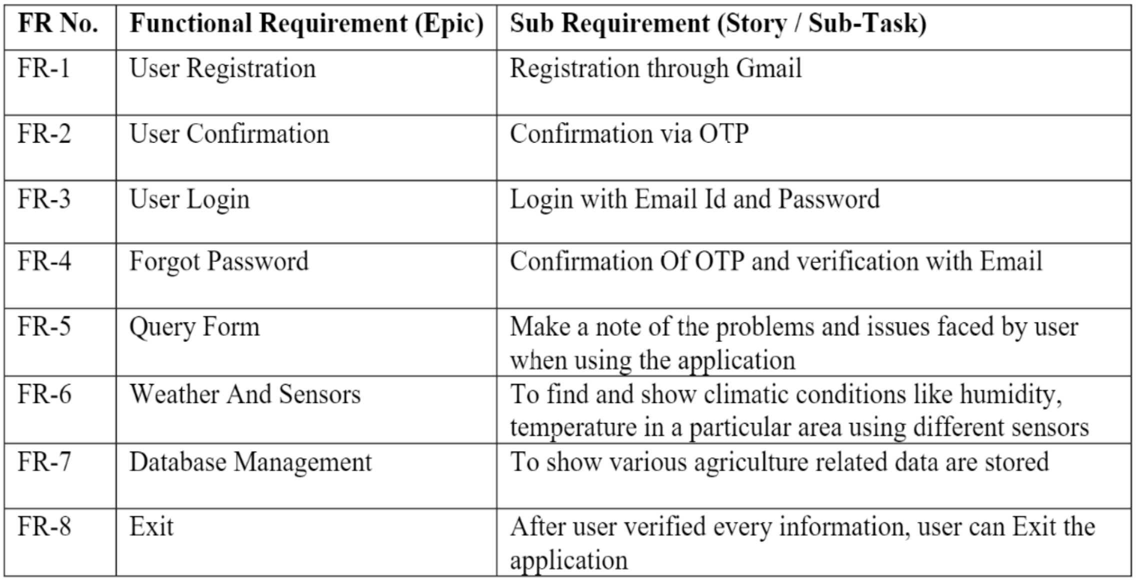
**8. CHANNELS of BEHAVIOUR** CH

In the context of farmer behaviors, 'external factors' refers to physical, environmental, farm business structure, financial and time factors on farm management; all of which can have an impact on farmer behaviors. External factors create the context within which farmer behavior can or cannot be influenced.

-*

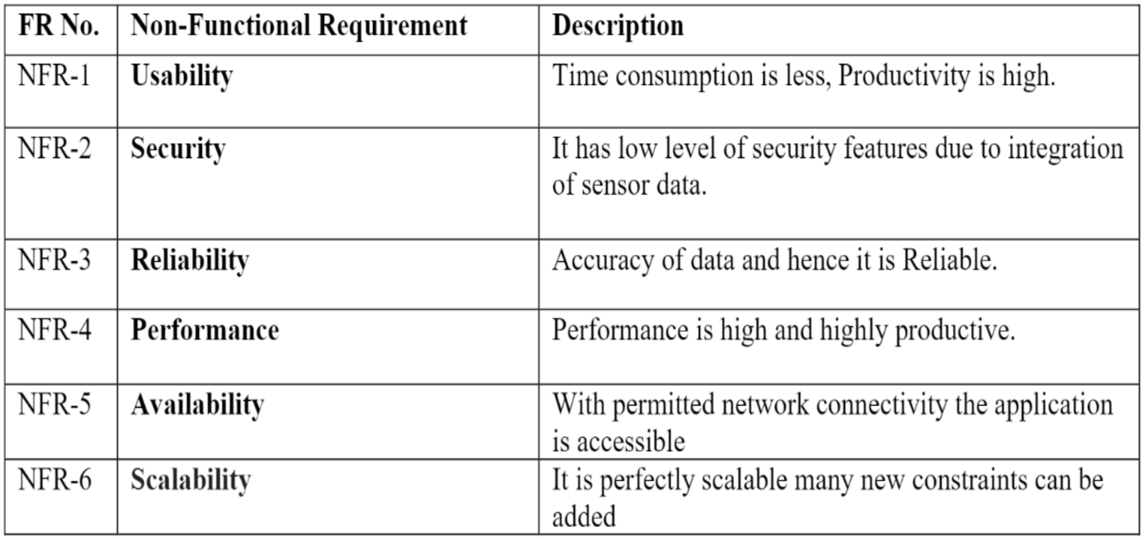
# 4. REQUIREMENT ANALYSIS

## 4.1 Functional requirement:



| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User Registration | Registration through Gmail |
| FR-2 | User Confirmation | Confirmation via OTP |
| FR-3 | User Login | Login with Email Id and Password |
| FR-4 | Forgot Password | Confirmation Of OTP and verification with Email |
| FR-5 | Query Form | Make a note of the problems and issues faced by user when using the application |
| FR-6 | Weather And Sensors | To find and show climatic conditions like humidity, temperature in a particular area using different sensors |
| FR-7 | Database Management | To show various agriculture related data are stored |
| FR-8 | Exit | After user verified every information, user can Exit the application |

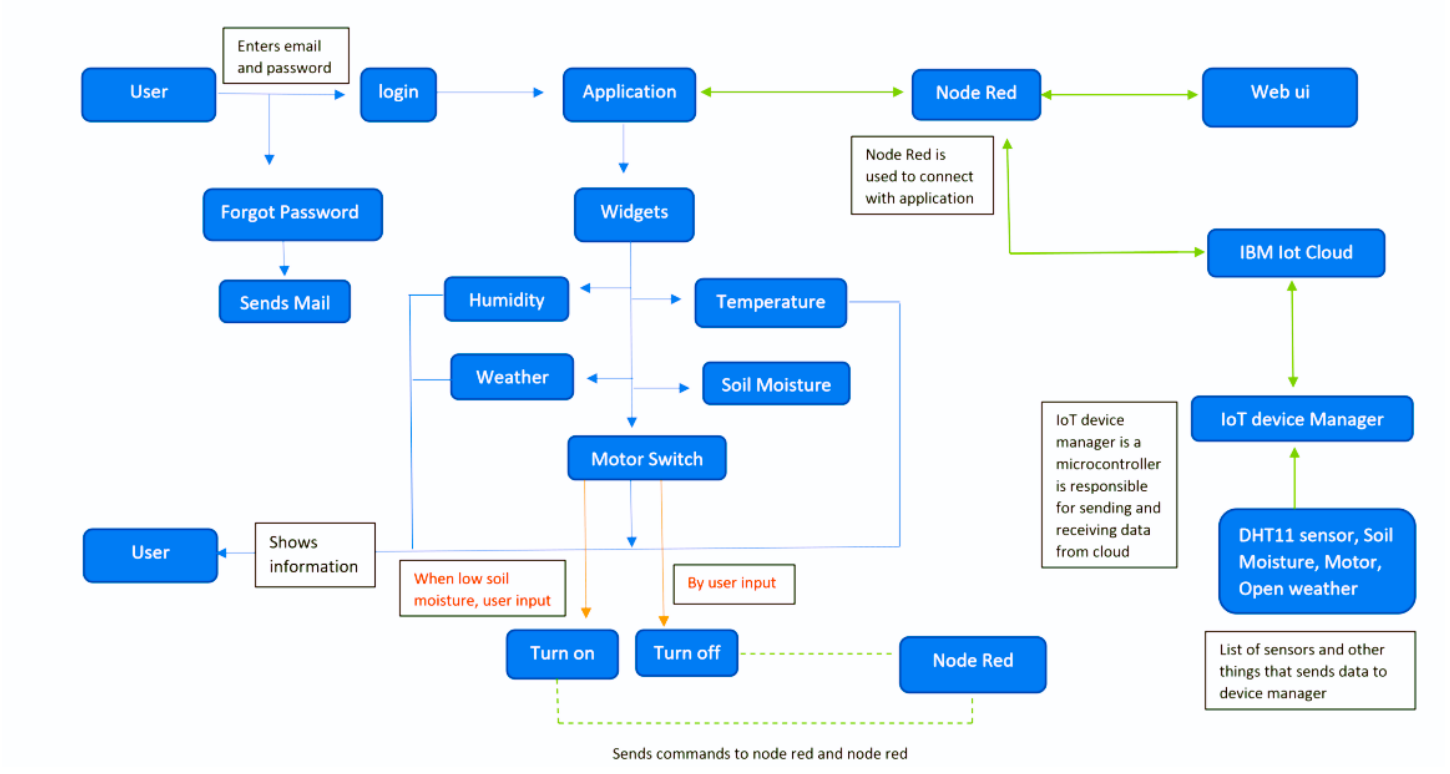
## 4.2 Non-Functional requirements:



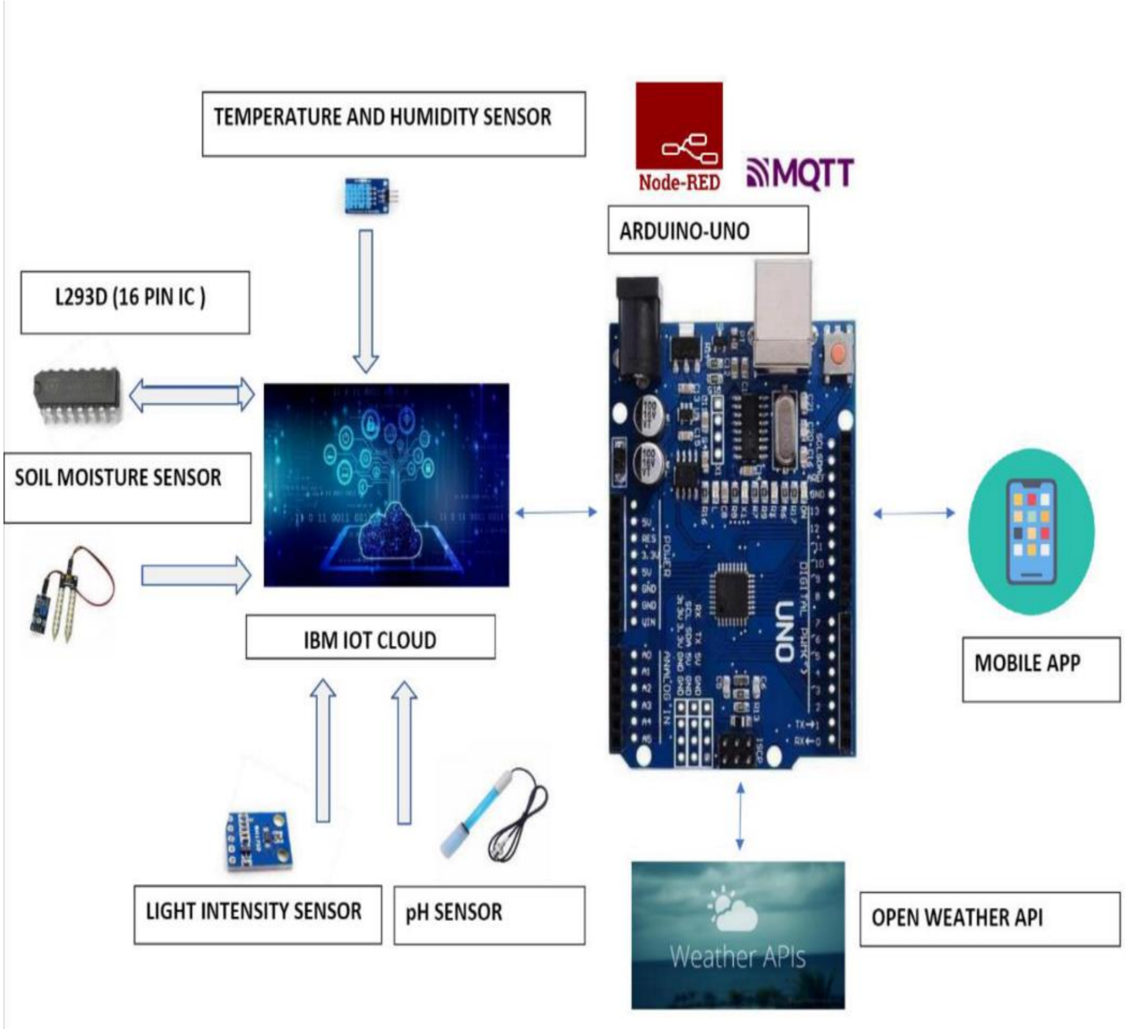
| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | Time consumption is less, Productivity is high. |
| NFR-2 | **Security** | It has low level of security features due to integration of sensor data. |
| NFR-3 | **Reliability** | Accuracy of data and hence it is Reliable. |
| NFR-4 | **Performance** | Performance is high and highly productive. |
| NFR-5 | **Availability** | With permitted network connectivity the application is accessible |
| NFR-6 | **Scalability** | It is perfectly scalable many new constraints can be added |

# 5. PROJECT DESIGN

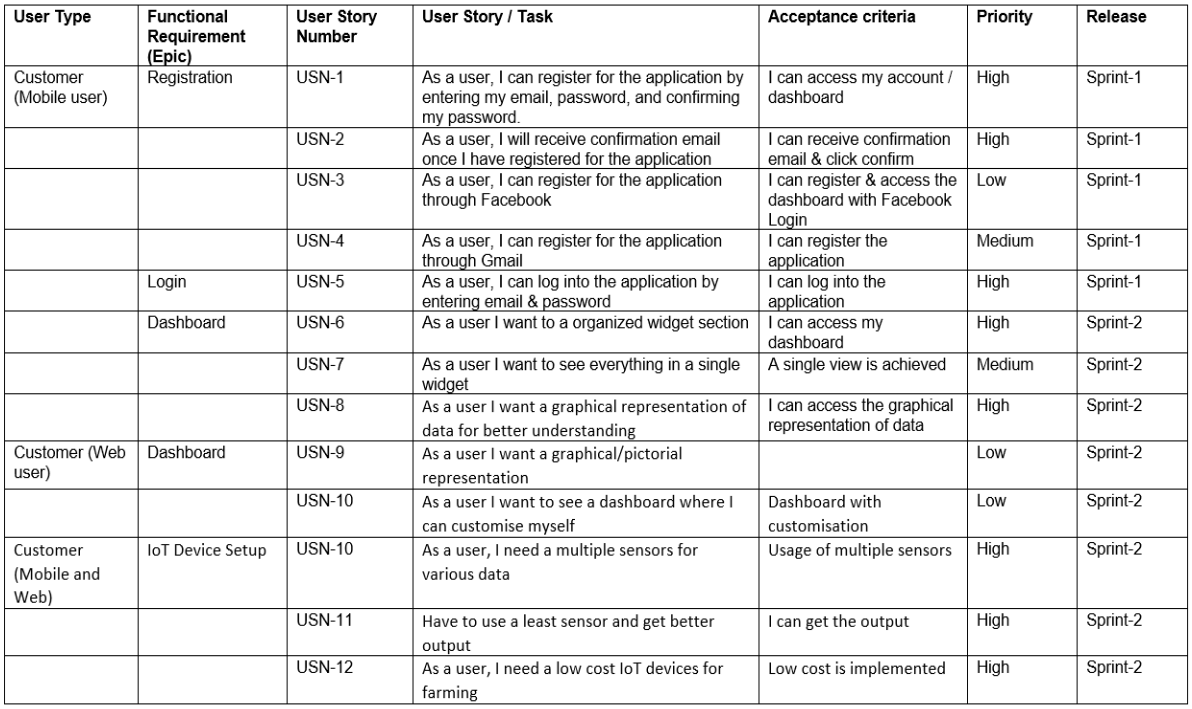
## 5.1 Data Flow Diagrams:


User
Enters email and password
login
Application
Node Red
Web ui
Forgot Password
Sends Mail
Widgets
Node Red is used to connect with application
IBM Iot Cloud
Humidity
Temperature
Weather
Soil Moisture
IoT device Manager
Motor Switch
IoT device manager is a microcontroller is responsible for sending and receiving data from cloud
User
Shows information
When low soil moisture, user input
By user input
DHT11 sensor, Soil Moisture, Motor, Open weather
Turn on
Turn off
Node Red
List of sensors and other things that sends data to device manager
Sends commands to node red and node red

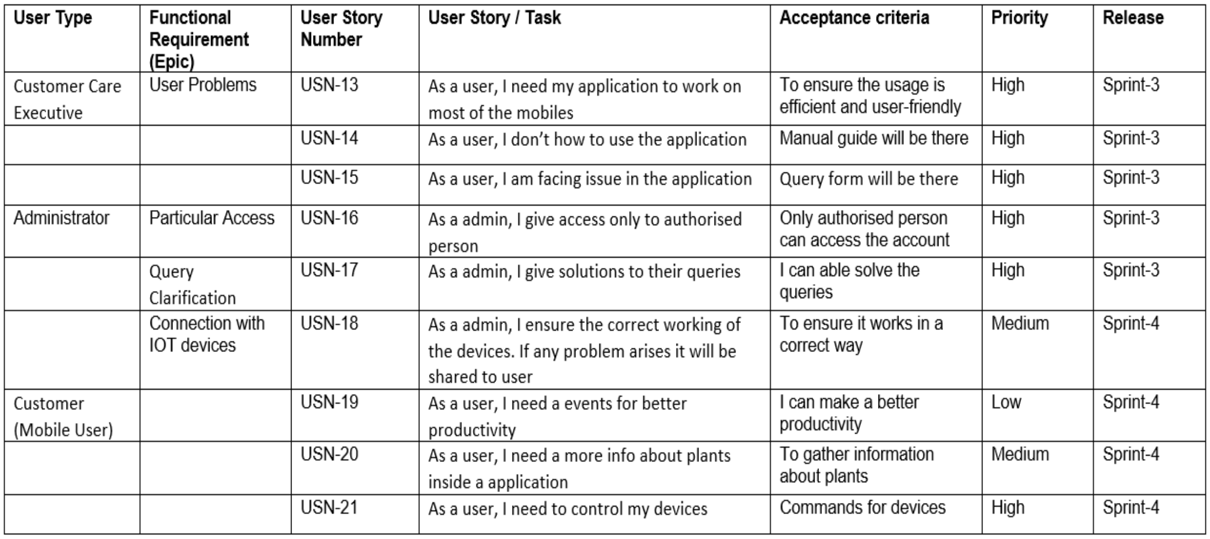
## 5.2 Solution & Technical Architecture:



## 5.3 User Stories:



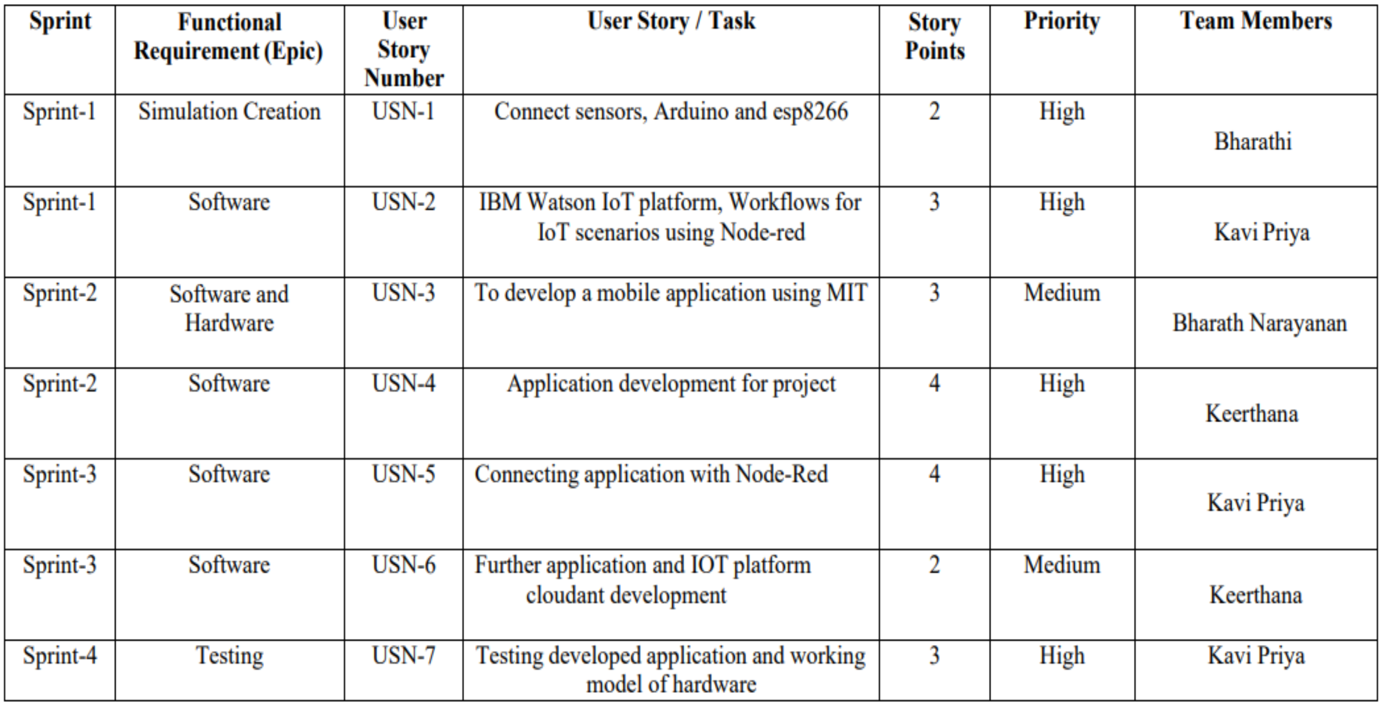
| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
| | | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
| | | USN-3 | As a user, I can register for the application through Facebook | I can register & access the dashboard with Facebook Login | Low | Sprint-1 |
| | | USN-4 | As a user, I can register for the application through Gmail | I can register the application | Medium | Sprint-1 |
| | Login | USN-5 | As a user, I can log into the application by entering email & password | I can log into the application | High | Sprint-1 |
| | Dashboard | USN-6 | As a user I want to a organized widget section | I can access my dashboard | High | Sprint-2 |
| | | USN-7 | As a user I want to see everything in a single widget | A single view is achieved | Medium | Sprint-2 |
| | | USN-8 | As a user I want a graphical representation of data for better understanding | I can access the graphical representation of data | High | Sprint-2 |
| Customer (Web user) | Dashboard | USN-9 | As a user I want a graphical/pictorial representation | | Low | Sprint-2 |
| | | USN-10 | As a user I want to see a dashboard where I can customise myself | Dashboard with customisation | Low | Sprint-2 |
| Customer (Mobile and Web) | IoT Device Setup | USN-10 | As a user, I need a multiple sensors for various data | Usage of multiple sensors | High | Sprint-2 |
| | | USN-11 | Have to use a least sensor and get better output | I can get the output | High | Sprint-2 |
| | | USN-12 | As a user, I need a low cost IoT devices for farming | Low cost is implemented | High | Sprint-2 |



| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer Care Executive | User Problems | USN-13 | As a user, I need my application to work on most of the mobiles | To ensure the usage is efficient and user-friendly | High | Sprint-3 |
| | | USN-14 | As a user, I don't how to use the application | Manual guide will be there | High | Sprint-3 |
| | | USN-15 | As a user, I am facing issue in the application | Query form will be there | High | Sprint-3 |
| Administrator | Particular Access | USN-16 | As a admin, I give access only to authorised person | Only authorised person can access the account | High | Sprint-3 |
| | Query Clarification | USN-17 | As a admin, I give solutions to their queries | I can able solve the queries | High | Sprint-3 |
| | Connection with IOT devices | USN-18 | As a admin, I ensure the correct working of the devices. If any problem arises it will be shared to user | To ensure it works in a correct way | Medium | Sprint-4 |
| Customer (Mobile User) | | USN-19 | As a user, I need a events for better productivity | I can make a better productivity | Low | Sprint-4 |
| | | USN-20 | As a user, I need a more info about plants inside a application | To gather information about plants | Medium | Sprint-4 |
| | | USN-21 | As a user, I need to control my devices | Commands for devices | High | Sprint-4 |

# 6. PROJECT PLANNING & SCHEDULING

## 6.1 Sprint Planning & Estimation:

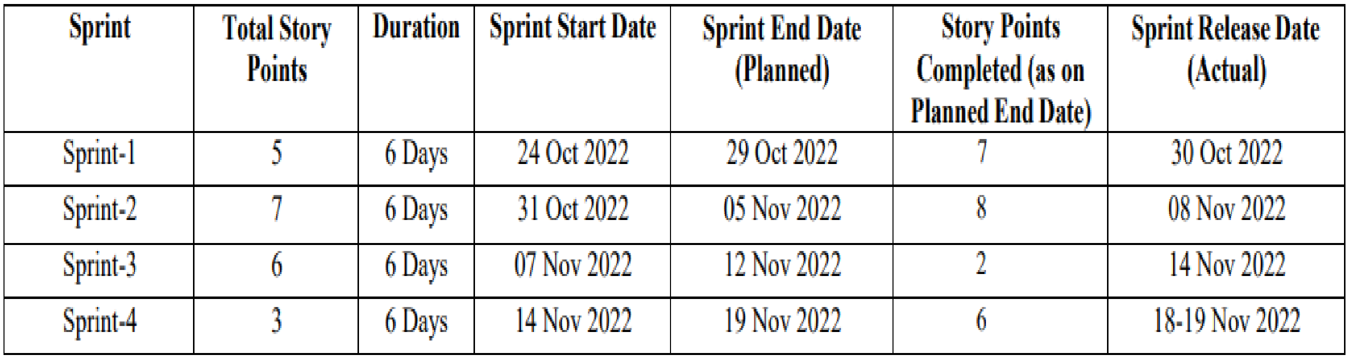


| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Simulation Creation | USN-1 | Connect sensors, Arduino and esp8266 | 2 | High | Bharathi |
| Sprint-1 | Software | USN-2 | IBM Watson IoT platform, Workflows for IoT scenarios using Node-red | 3 | High | Kavi Priya |
| Sprint-2 | Software and Hardware | USN-3 | To develop a mobile application using MIT | 3 | Medium | Bharath Narayanan |
| Sprint-2 | Software | USN-4 | Application development for project | 4 | High | Keerthana |
| Sprint-3 | Software | USN-5 | Connecting application with Node-Red | 4 | High | Kavi Priya |
| Sprint-3 | Software | USN-6 | Further application and IOT platform cloudant development | 2 | Medium | Keerthana |
| Sprint-4 | Testing | USN-7 | Testing developed application and working model of hardware | 3 | High | Kavi Priya |

## 6.2. Sprint Delivery Schedule

| TITLE | DESCRIPTION | DATE |
|---|---|---|
| **Literature Survey & Information Gathering** | Gathering information by referring technical papers, research publications describe literature survey. | 16 SEPTEMBER 2022 |
| **Prepare Empathy Map** | To establish users’ pros and cons,prepare the empathy map canvas on problem statement. | 19 SEPTEMBER 2022 |
| **Ideation** | Establishing brainstorm sessionsand emphasize the top ideas based on the importance of | 20 SEPTEMBER 2022 |

| | | |
|---|---|---|
| | scalability and feasibility. | |
| **Proposed Solution** | Prepare the proposed solution which describes idea, uniqueness, customer satisfaction, business model and scalability of solution. | 22 SEPTEMBER 2022 |
| **Problem Solution Fit** | Prepare problem - solution fit which describes the existence of a problem. | 24 SEPTEMBER 2022 |
| **Solution Architecture** | Defining process of developing solution based on pre-defined processes. | 25 SEPTEMBER 2022 |
| **Customer Journey** | Prepare a customer journey mapwhich understand the customers on user interaction and experiences from scratch tofinding solution. | 03 OCTOBER 2022 |
| **Functional Requirement** | Prepare the functional requirement document. | 08 OCTOBER 2022 |
| **Data Flow Diagrams** | Draw the data flow diagramsbased on problem statement. | 11 OCTOBER 2022 |
| **Technology Architecture** | Prepare a technology architecturediagram. | 14 OCTOBER 2022 |
| **Prepare Milestone & Activity List** | Prepare the milestones & activitylist for the project. | 16 OCTOBER 2022 |
| **Project Development - Delivery of Sprint-1, 2, 3 & 4** | Develop & submit the developedcode by testing it. | 19 NOVEMBER 2022 |



| 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 | 5 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 7 | 30 Oct 2022 |
| Sprint-2 | 7 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 8 | 08 Nov 2022 |
| Sprint-3 | 6 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 2 | 14 Nov 2022 |
| Sprint-4 | 3 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 6 | 18-19 Nov 2022 |

## Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 21 (points per sprint). Let's calculate the team's average velocity (AV) periteration unit (story points per day)

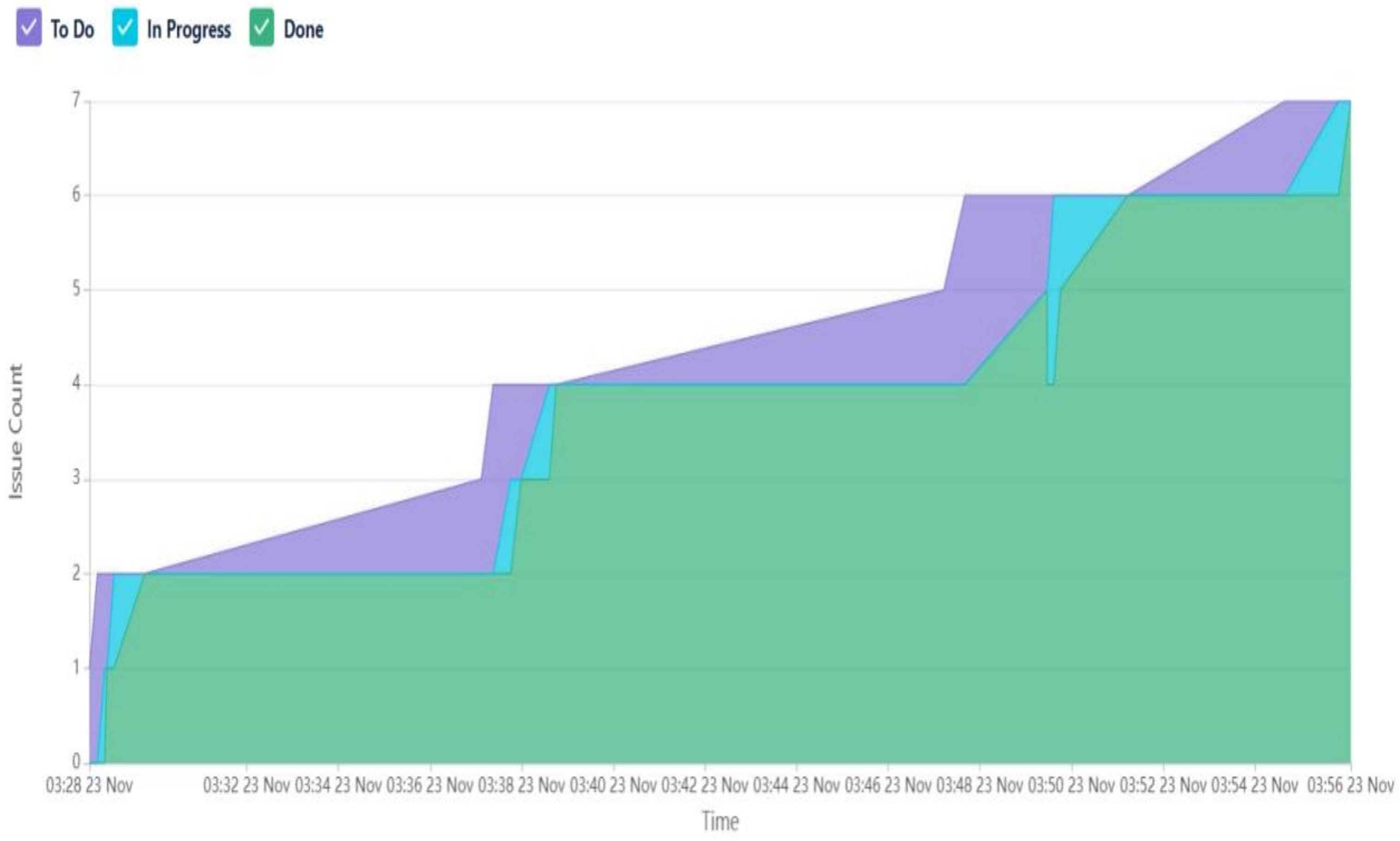
Total Sprint Points = 21

Total Sprint = 7

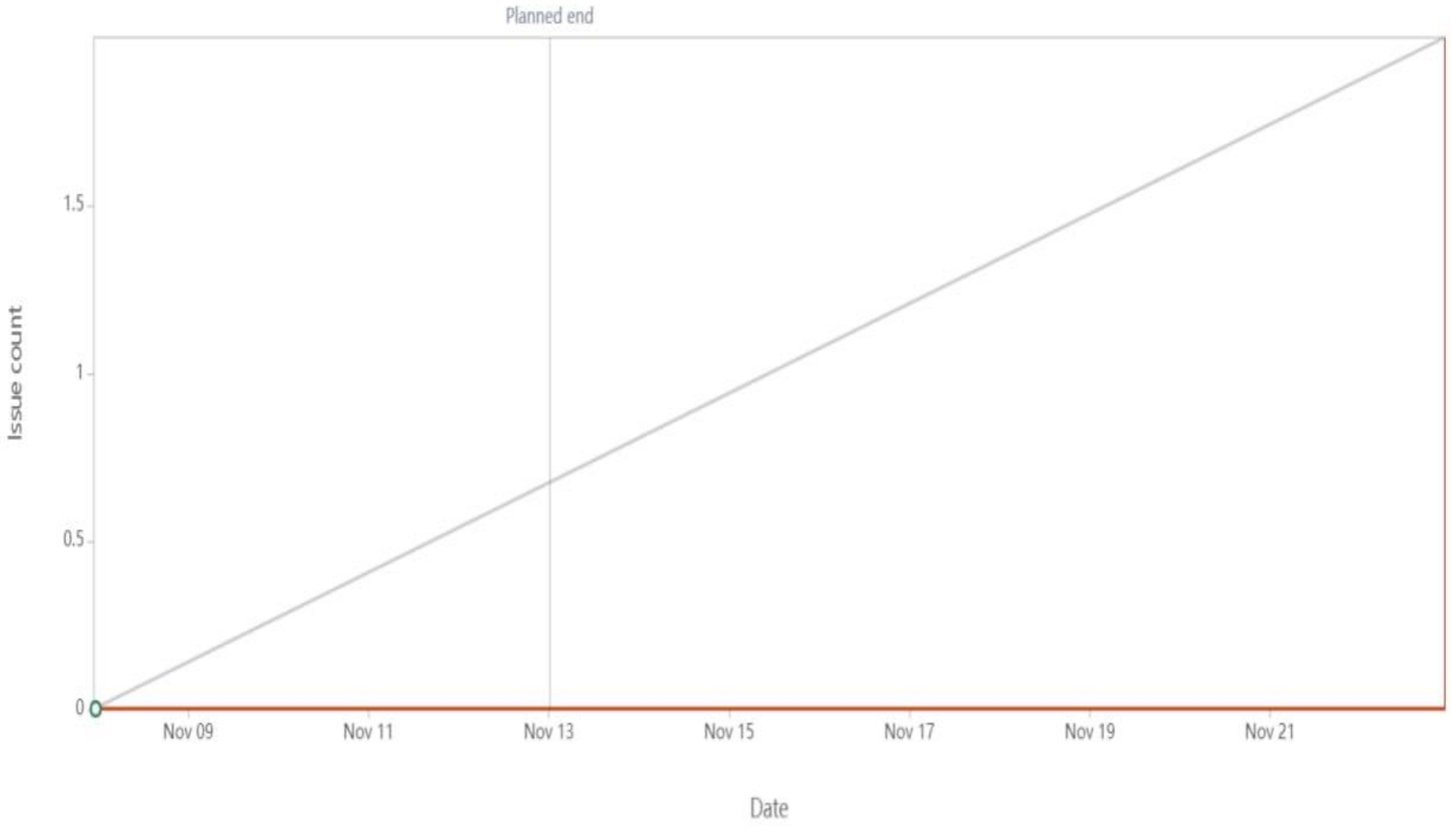
Average Velocity = 21/7 = 3

## 6.3 Reports from JIRA:

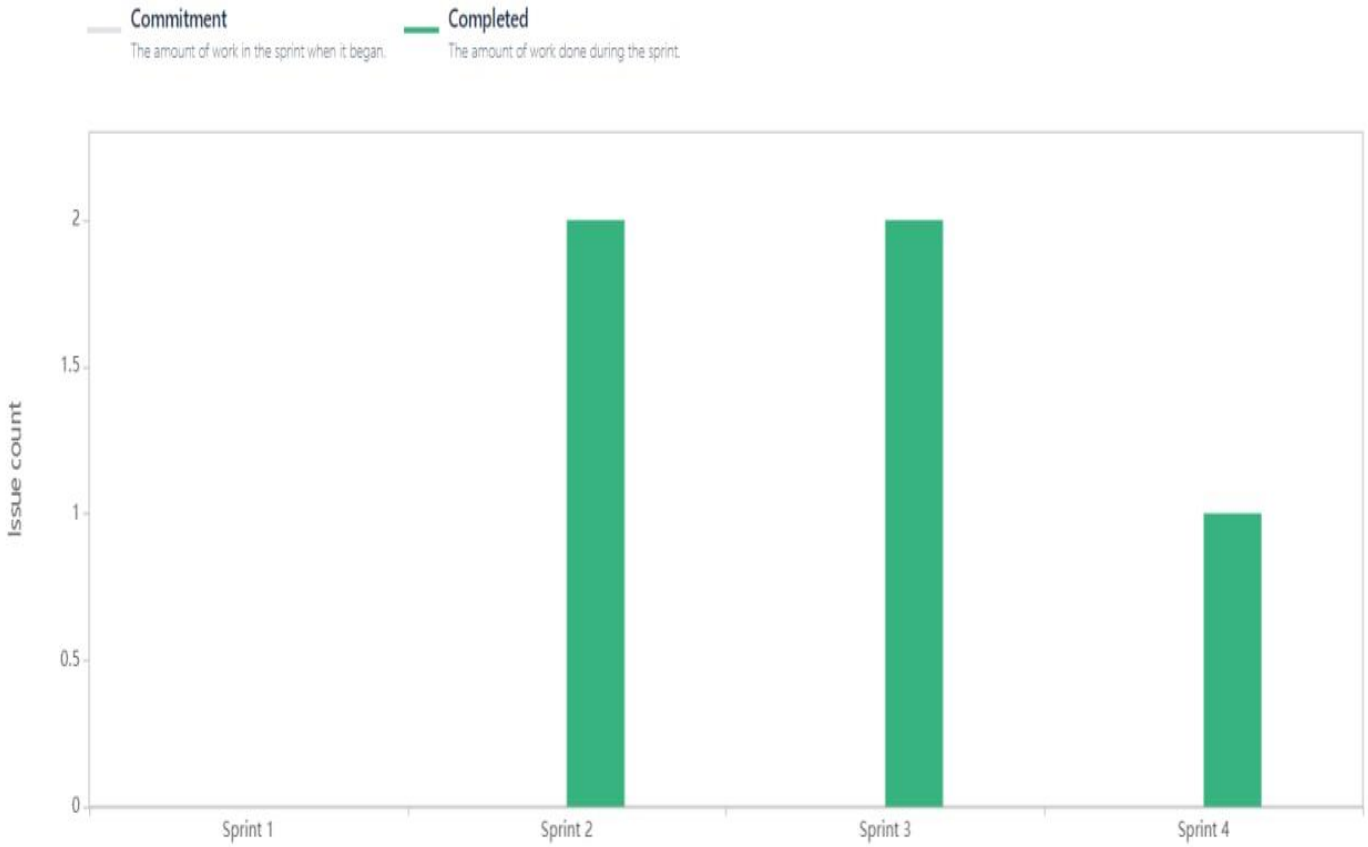
### Cumulative Flow Report



# Sprint Burndown Report



## Velocity Report



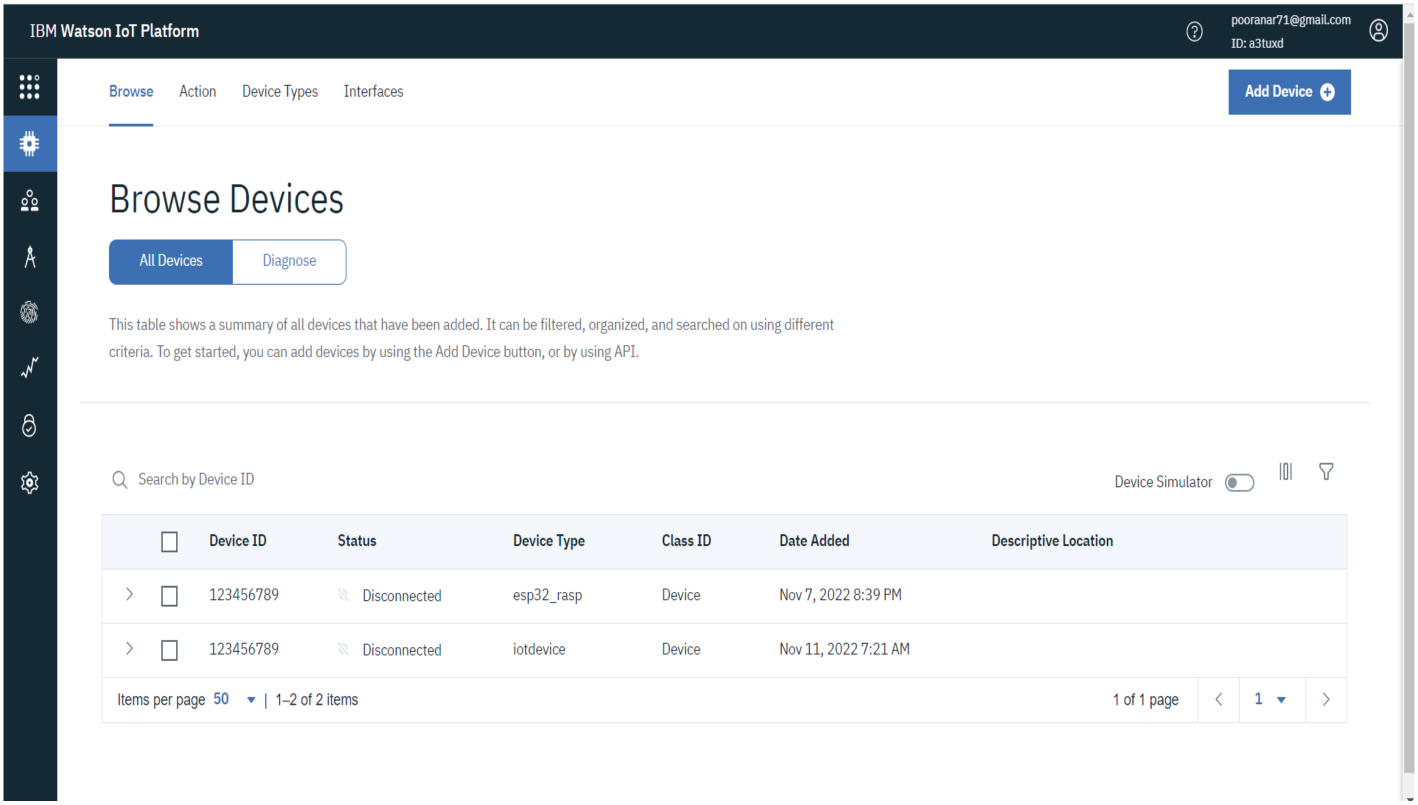
# 7. CODING & SOLUTIONING (Explain the features added in the project along with code)

## 7.1 Feature 1:

## IBM Watson IOT Platform interfaced with NODE-RED

IBM Watson™ IoT Platform is a fully managed, cloud-hosted service that makes it simple to derive value from Internet of Things (IoT) devices.

Simply register and connect your device, be it a sensor, a gateway, or something else, to Watson IoT Platform and start sending data securely up to the cloud using the open, lightweight MQTT messaging protocol. You can set up and manage your devices using your online dashboard or our secure APIs, so that your apps can access and use your live and historical data.



Browse Devices

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

| Device ID | Status | Device Type | Class ID | Date Added | Descriptive Location |
|---|---|---|---|---|---|
| 123456789 | Disconnected | esp32_rasp | Device | Nov 7, 2022 8:39 PM | |
| 123456789 | Disconnected | iotdevice | Device | Nov 11, 2022 7:21 AM | |

## Basic concepts

Familiarize yourself with some basic Platform Service concepts.

### Organizations

When you register with the Platform Service, you are provided with an organization ID, a unique six-character identifier for your account. Organizations ensure that your data is only accessible by your devices and applications.

After registration, devices and API keys are bound to a single organization. When an application connects to the service by using an API key, it registers to the organization that is associated with that API key.

**Important**: For security reasons, direct communication between different organizations is not possible. To transmit data between two organizations, you must register one application with each organization and have the two applications communicate with each other to transfer data.

### Devices

| Device type | Description |
|---|---|
| Managed devices | Devices that contain a device management agent. A device management agent is a set of logic that allows the device to interact with the Platform Service Device Management service by using the Device Management Protocol. Managed devices can perform device management operations, including location updates, firmware downloads and updates, restarts, and factory resets. |
| Unmanaged devices | Devices without a device management agent. Unmanaged devices can connect to the Platform Service and send and receive events and commands, but they cannot send device management requests or perform device management operations. |

A device can be anything that has a connection to the internet and that can push data into the cloud. However, devices cannot communicate directly with other devices, instead devices accept commands from applications, and send events to applications.

Devices in Platform Service are identified by a unique authentication token. Devices must be registered before they can connect to Platform Service.

### Gateways

Gateways are specialized devices that have the combined capabilities of an application and a device. This lets them serve as access points for other devices. Devices that cannot connect directly to the internet can access the Platform Service service by first connecting to the gateway device.

Gateways must be registered before they can connect to the service, and like standard devices can be managed or unmanaged.

### Applications

An application is anything that has a connection to the internet and interacts with data from devices and control the behaviour of those devices. Applications identify themselves with the Platform Service by using an API key and a unique application ID. Unlike devices, individual applications do not need to register before they can connect to the Platform Service. However, they must use a valid registered API key.

### Events

Events are the mechanism by which devices publish data to the Platform Service. Devices control the content of their messages, and assign a name for each event that is sent. The Platform Service uses the credentials that are attached to each event received to determine which device sent the event. This architecture prevents devices from impersonating one another.

Applications can process events in real time, and see the source of the event and the data that is contained if. Applications must be configured to define which devices and events they subscribe to.

## Automated Alerting using Fast2sms

**Fast2SMS**is a popular **bulk SMS service provider in India**. It was started in 21st July 2011. Due to its simplicity and ease of use it has become one of the mostly used SMS portals and has 2 million users

## Features of Fast2SMS

**Bulk SMS** – Bulk SMS refers to business sending SMS to one or more recipients and can scale up to millions of persons at the same time. It refers to sending large number of messages to a predefined set of customers.

**Quick SMS feature** – Fast2SMS provides a very unique and useful feature which is not available in any other bulk SMS service provider. You can send SMS to DND and Non DND numbers even if you are not registered in the DLT portal.

**API SMS** – API refers to Application Programming Interface. **SMS API integration**is the fastest and simplest way to send automated messages directly from your platform. Fast2SMS provide **API for bulk SMS,** which ensures security and it is a very reliable source of sending data.

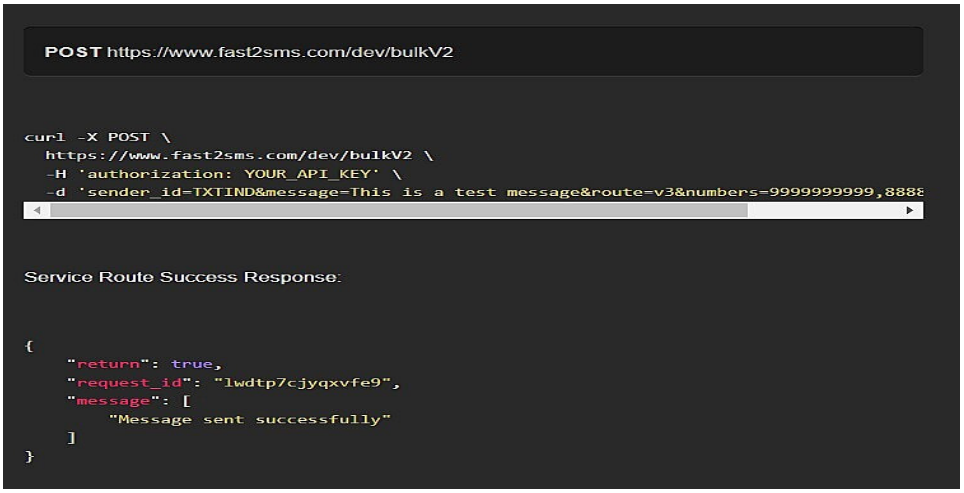
**Add contacts with QR** – Fast2SMS offers the facility of adding contacts with QR. QR stands for quick response. The main benefit of QR code is it takes very less space for storing information. It is a form of barcode and has become quite popular these days. In this feature you can add contacts by creating a QR code. Simply you need to add your group name and group URL name and then click on create QR. The QR code gets created

There is a lot of debate on the topic whether to go for bulk SMS or email marketing. Let's look into some facts related to email and SMS:

- The average person receives 178 text messages in a month.
- And they receive approx. 1216 emails in a month.
- 98% of text messages get read in a month
- 22% of emails are read in a month.
- 91% of adults have their smartphone within arm's reach.
- There are 6 billion active mobile phones in the world.
- 6 billion Email accounts are there in the world.
- According to the findings, 19% of people click the URL in the SMS.
- According to the findings, 2% of people click the URL sent in the email.

The picture is clear. SMS is a more powerful way to connect to the target audience and establish rapport with them. The open rate of an SMS is more than that of an email. As everybody of us might have noticed that as soon as a notification sound comes in our mobile, we immediately grab our phone and check who has sent SMS and the details. This is why SMS has a high readability and is more likely to get opened.

In case of an email, we postpone reading it as it is usually very long. The positive points of an SMS are that it is to the point, conveys essential information in a clear-cut manner and does not require an internet connection to view it.



**POST** https://www.fast2sms.com/dev/bulkV2

```
curl -X POST \
  https://www.fast2sms.com/dev/bulkV2 \
  -H 'authorization: YOUR_API_KEY' \
  -d 'sender_id=TXTIND&message=This is a test message&route=v3&numbers=9999999999,8888
```

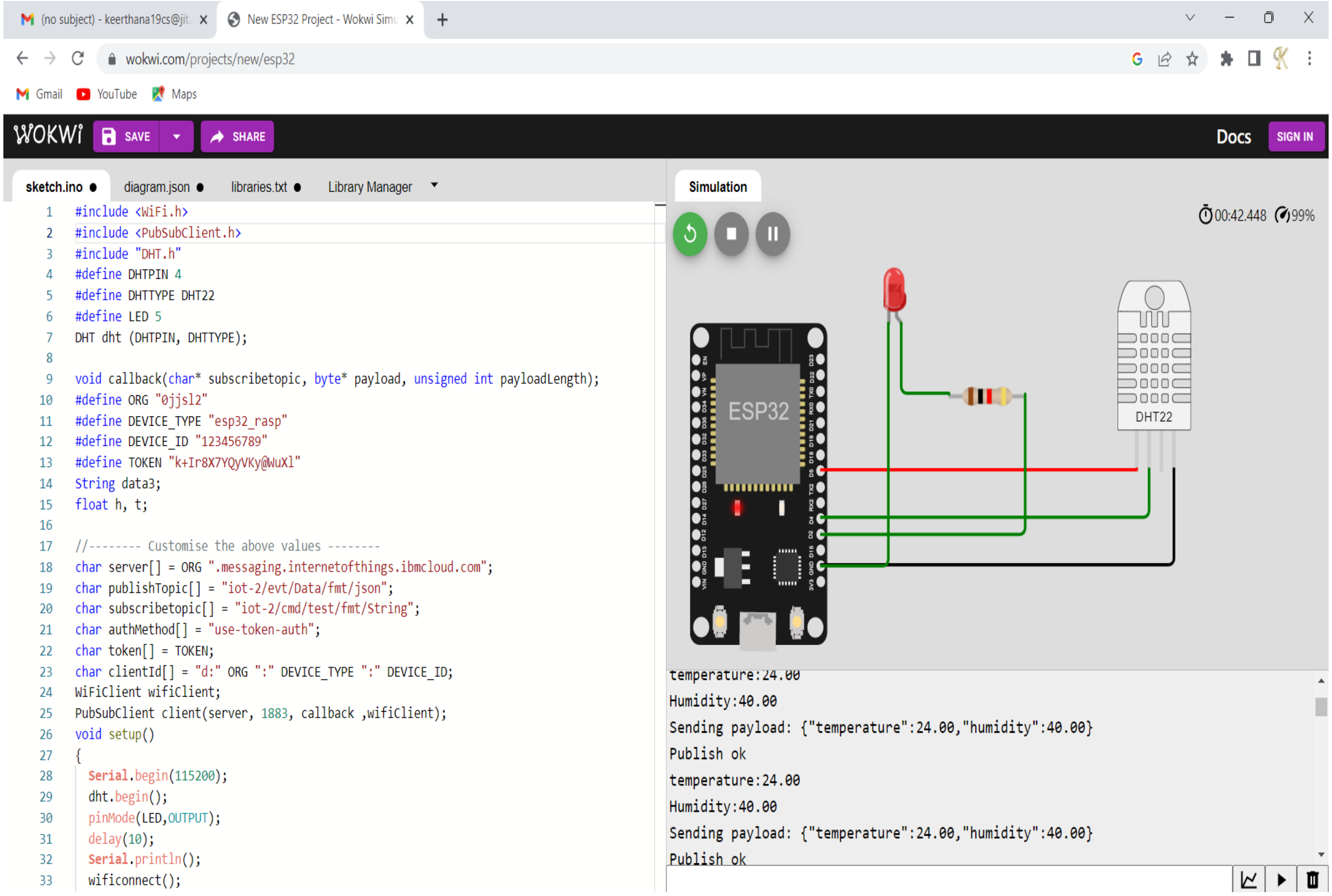
Service Route Success Response:

```
{
    "return": true,
    "request_id": "lwdtp7cjyqxvfe9",
    "message": [
        "Message sent successfully"
    ]
}
```

## PYTHON CODE

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

organization = "Org ID"

deviceType = "Device Type"

deviceId = "Device ID"

authMethod = "token"

authToken = "authtoken"

def myCommandCallback(cmd):

print("Command received: %s" % cmd.data)

if cmd.data['command']=='motoron':

 print("MOTOR ON IS RECEIVED")
```

```
elif cmd.data['command']=='motoroff':

print("MOTOR OFF IS RECEIVED")

  if cmd.command == "setInterval":

if 'interval' not in cmd.data:

  print("Error - command is missing required information: 'interval'")

else:

interval = cmd.data['interval']

elif cmd.command == "print":

if 'message' not in cmd.data:

print("Error - command is missing required information: 'message'")

else:

output=cmd.data['message']

  print(output)

  try:

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}

deviceCli = ibmiotf.device.Client(deviceOptions)

except Exception as e:

print("Caught exception connecting device: %s" % str(e))

sys.exit()

deviceCli.connect()

  while True:

deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
```

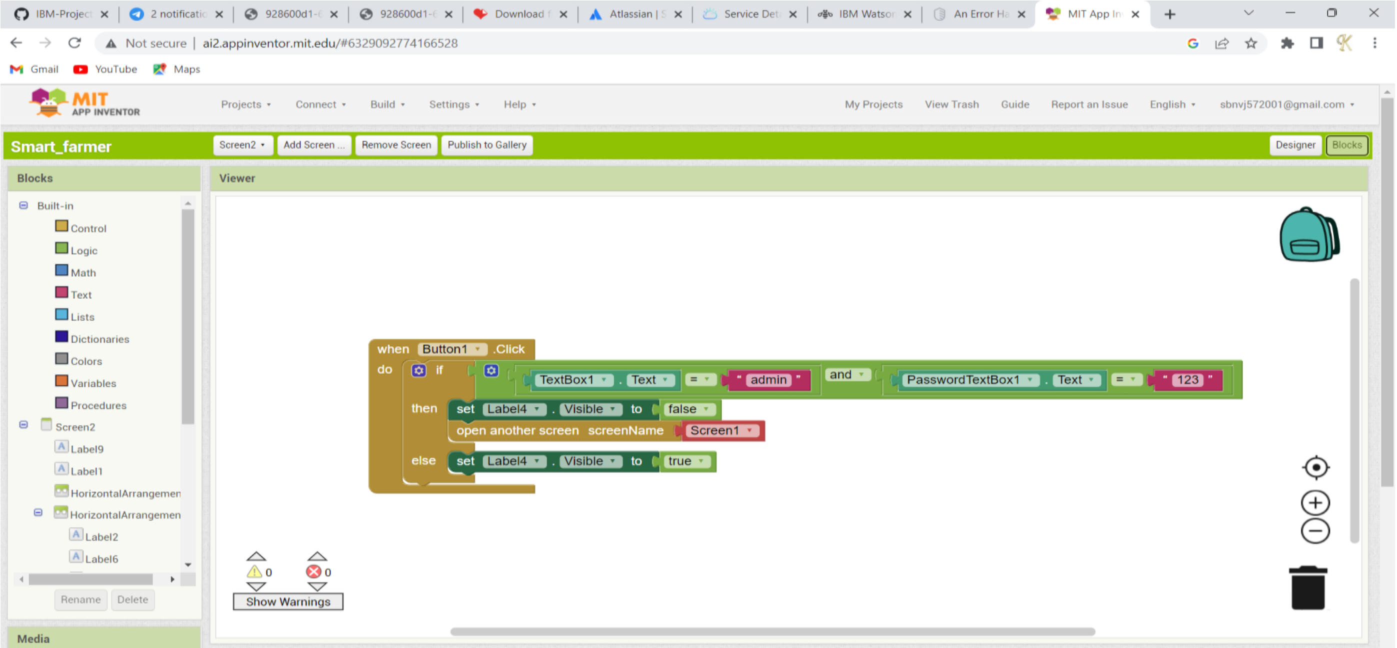


```
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"
#define DHTPIN 4
#define DHTTYPE DHT22
#define LED 5
DHT dht (DHTPIN, DHTTYPE);

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "0jjsl2"
#define DEVICE_TYPE "esp32_rasp"
#define DEVICE_ID "123456789"
#define TOKEN "k+Ir8X7YQyVKy@WuXl"
String data3;
float h, t;

//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
void setup()
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
```

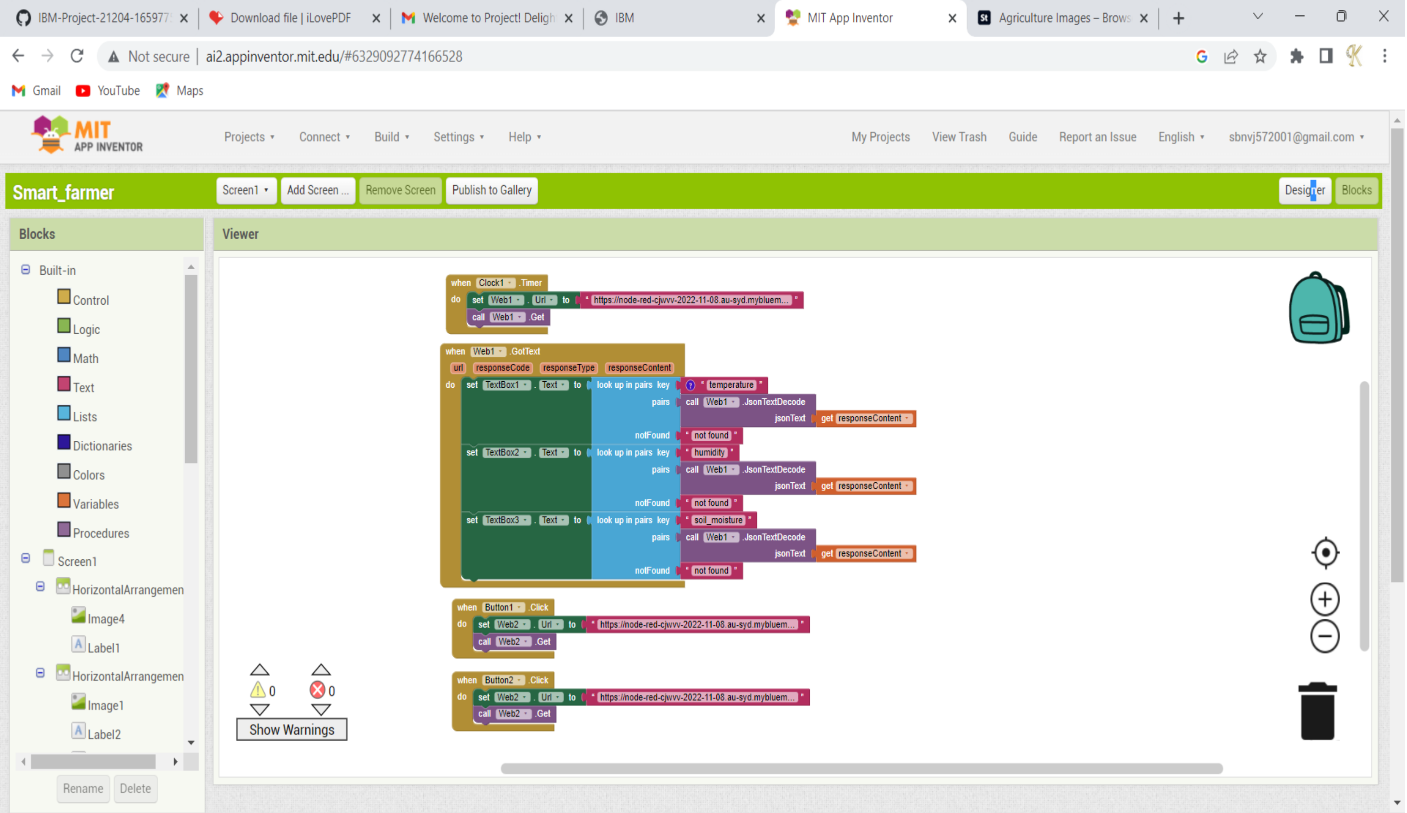
```
temperature:24.00
Humidity:40.00
Sending payload: {"temperature":24.00,"humidity":40.00}
Publish ok
temperature:24.00
Humidity:40.00
Sending payload: {"temperature":24.00,"humidity":40.00}
Publish ok
```

## 7.2 Feature 2:

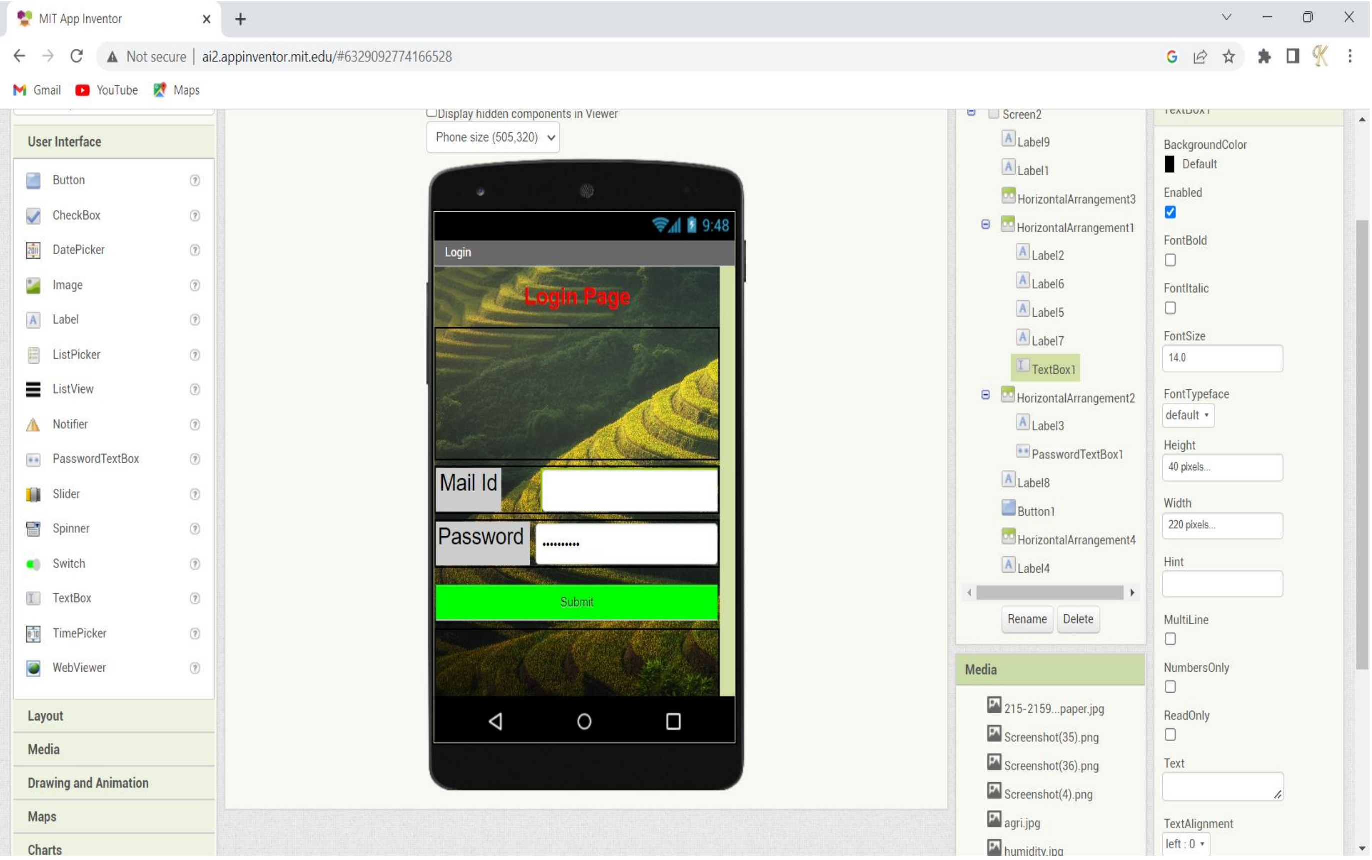
## CODE BLOCKS AND DESIGN UI



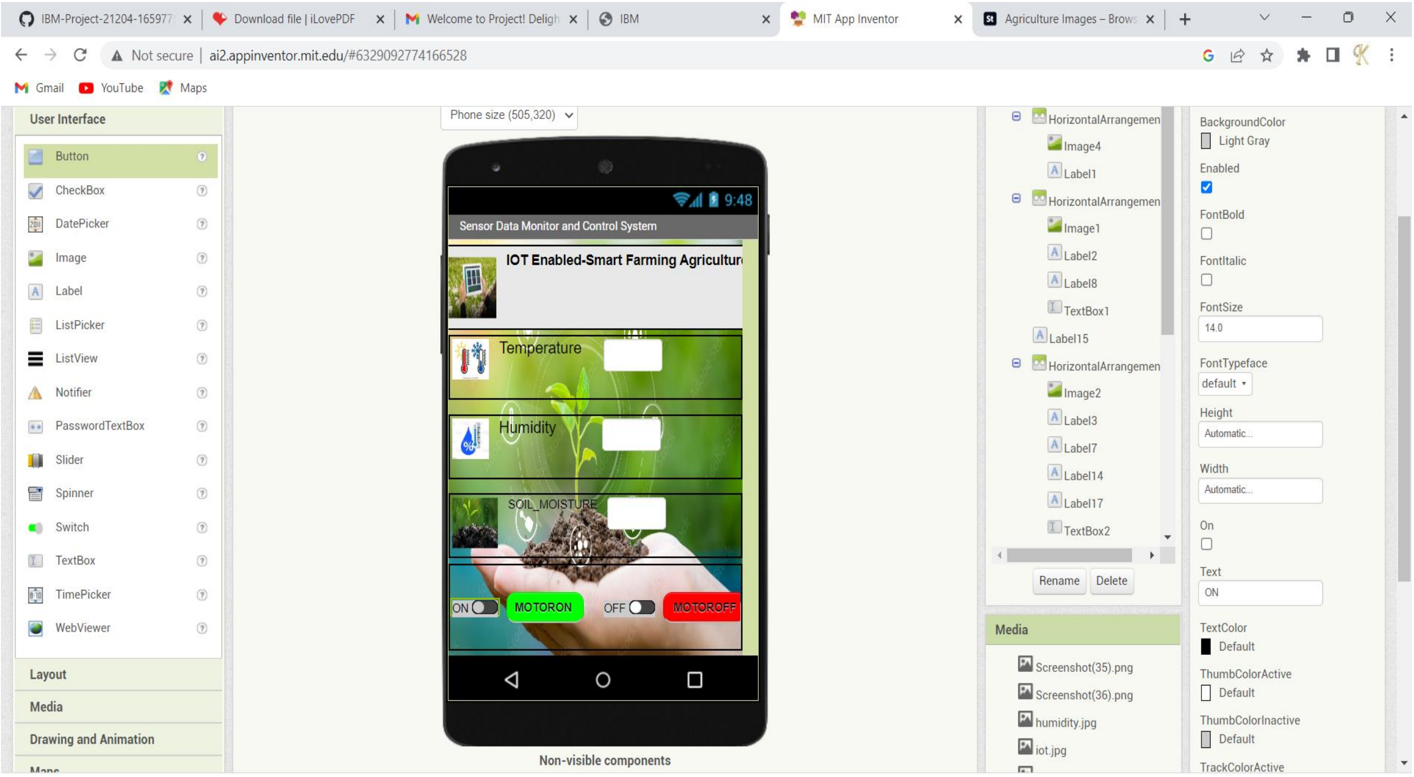
```
when Button1 .Click
do  if  TextBox1 . Text = " admin " and PasswordTextBox1 . Text = " 123 "
    then  set Label4 . Visible to false
          open another screen screenName Screen1
    else  set Label4 . Visible to true
```



```
when Clock1 .Timer
do  set Web1 . Url to " https://node-red-cjwvv-2022-11-08.au-syd.mybluem... "
    call Web1 .Get

when Web1 .GotText
 url  responseCode  responseType  responseContent
do  set TextBox1 . Text to  look up in pairs  key " temperature "
                                              pairs call Web1 .JsonTextDecode
                                                         jsonText get responseContent
                                              notFound " not found "
    set TextBox2 . Text to  look up in pairs  key " humidity "
                                              pairs call Web1 .JsonTextDecode
                                                         jsonText get responseContent
                                              notFound " not found "
    set TextBox3 . Text to  look up in pairs  key " soil_moisture "
                                              pairs call Web1 .JsonTextDecode
                                                         jsonText get responseContent
                                              notFound " not found "

when Button1 .Click
do  set Web2 . Url to " https://node-red-cjwvv-2022-11-08.au-syd.mybluem... "
    call Web2 .Get

when Button2 .Click
do  set Web2 . Url to " https://node-red-cjwvv-2022-11-08.au-syd.mybluem... "
    call Web2 .Get
```

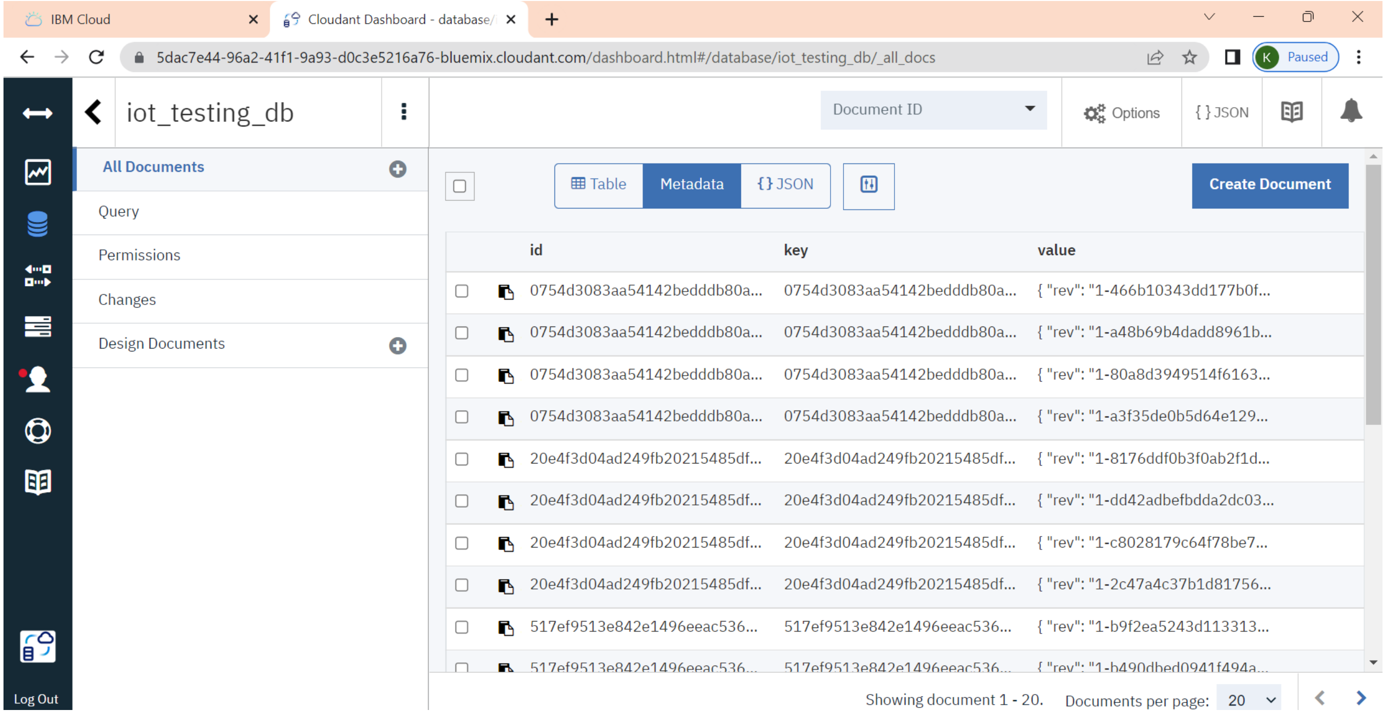


BackgroundColor: Default
Enabled: ☑
FontBold: □
FontItalic: □
FontSize: 14.0
FontTypeface: default
Height: 40 pixels...
Width: 220 pixels...
Hint:
MultiLine: □
NumbersOnly: □
ReadOnly: □
Text:
TextAlignment: left : 0



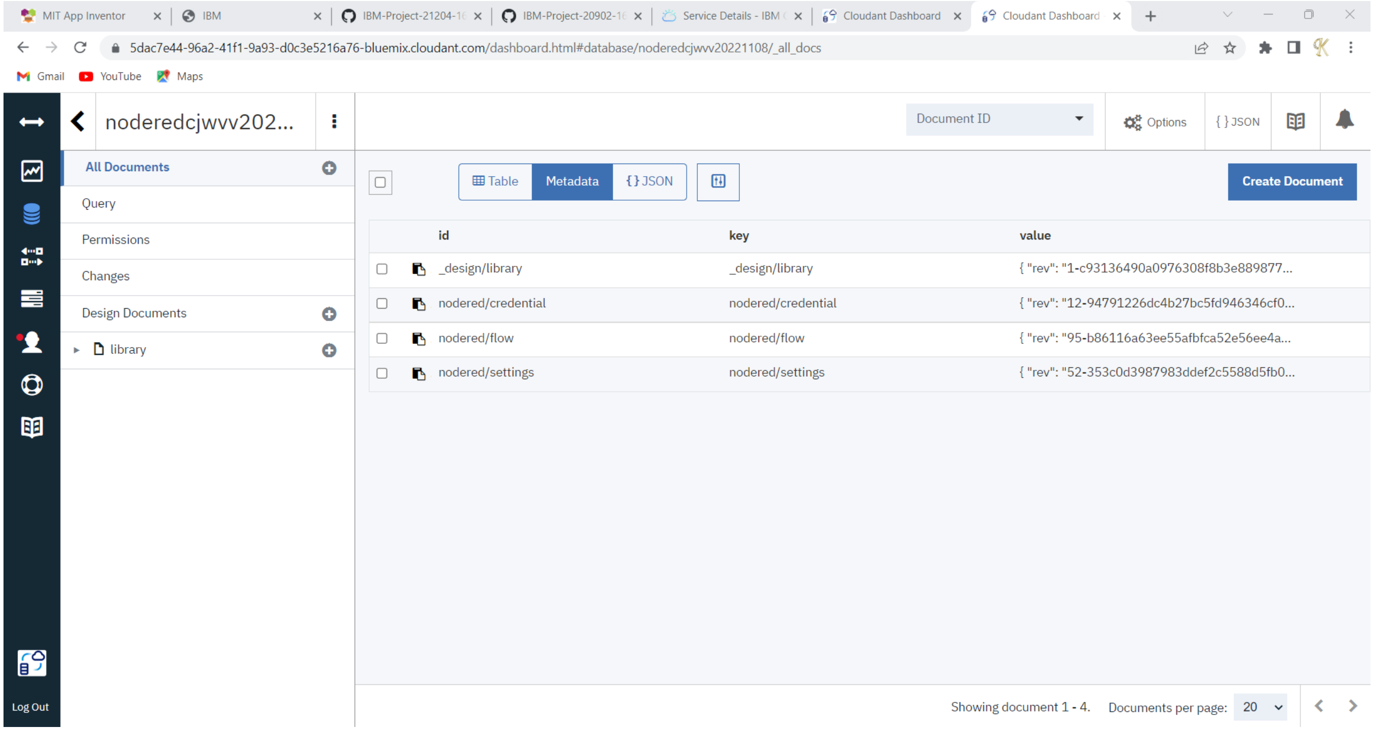
BackgroundColor: Light Gray
Enabled: ☑
FontBold: □
FontItalic: □
FontSize: 14.0
FontTypeface: default
Height: Automatic...
Width: Automatic...
On: □
Text: ON
TextColor: Default
ThumbColorActive: Default
ThumbColorInactive: Default
TrackColorActive

## 7.3 Database Schema (if Applicable) :



| | id | key | value |
|---|---|---|---|
| □ | 0754d3083aa54142bedddb80a... | 0754d3083aa54142bedddb80a... | { "rev": "1-466b10343dd177b0f... |
| □ | 0754d3083aa54142bedddb80a... | 0754d3083aa54142bedddb80a... | { "rev": "1-a48b69b4dadd8961b... |
| □ | 0754d3083aa54142bedddb80a... | 0754d3083aa54142bedddb80a... | { "rev": "1-80a8d3949514f6163... |
| □ | 0754d3083aa54142bedddb80a... | 0754d3083aa54142bedddb80a... | { "rev": "1-a3f35de0b5d64e129... |
| □ | 20e4f3d04ad249fb20215485df... | 20e4f3d04ad249fb20215485df... | { "rev": "1-8176ddf0b3f0ab2f1d... |
| □ | 20e4f3d04ad249fb20215485df... | 20e4f3d04ad249fb20215485df... | { "rev": "1-dd42adbefbdda2dc03... |
| □ | 20e4f3d04ad249fb20215485df... | 20e4f3d04ad249fb20215485df... | { "rev": "1-c8028179c64f78be7... |
| □ | 20e4f3d04ad249fb20215485df... | 20e4f3d04ad249fb20215485df... | { "rev": "1-2c47a4c37b1d81756... |
| □ | 517ef9513e842e1496eeac536... | 517ef9513e842e1496eeac536... | { "rev": "1-b9f2ea5243d113313... |
| □ | 517ef9513e842e1496eeac536... | 517ef9513e842e1496eeac536... | { "rev": "1-b490dbed0941f494a... |

Showing document 1 - 20. Documents per page: 20



| | id | key | value |
|---|---|---|---|
| □ | _design/library | _design/library | { "rev": "1-c93136490a0976308f8b3e889877... |
| □ | nodered/credential | nodered/credential | { "rev": "12-94791226dc4b27bc5fd946346cf0... |
| □ | nodered/flow | nodered/flow | { "rev": "95-b86116a63ee55afbfca52e56ee4a... |
| □ | nodered/settings | nodered/settings | { "rev": "52-353c0d3987983ddef2c5588d5fb0... |

Showing document 1 - 4. Documents per page: 20

# 8. TESTING

## 8.1 Test Cases:

A test case is a document which has a set of conditions or actions that are performed on the software application in order to verify the expected functionality of the feature.

After test scripts, test cases are the second most detailed way of documenting testing work. They describe a specific idea that is to be tested, without detailing the exact steps to be taken or data to be used. This doesn't mention how to apply the coupons or whether there are multiple ways to apply. It also doesn't mention if the tester uses a link to apply a discount, or enter a code, or have a customer service apply it. They give flexibility to the tester to decide how they want to execute the test.

**Benefits of Writing Test Cases**

The key purpose of a test case is to ensure if different features within an application are working as expected. It helps tester, validate if the software is free of defects and if it is working as per the expectations of the end users. Other benefits of test cases include:

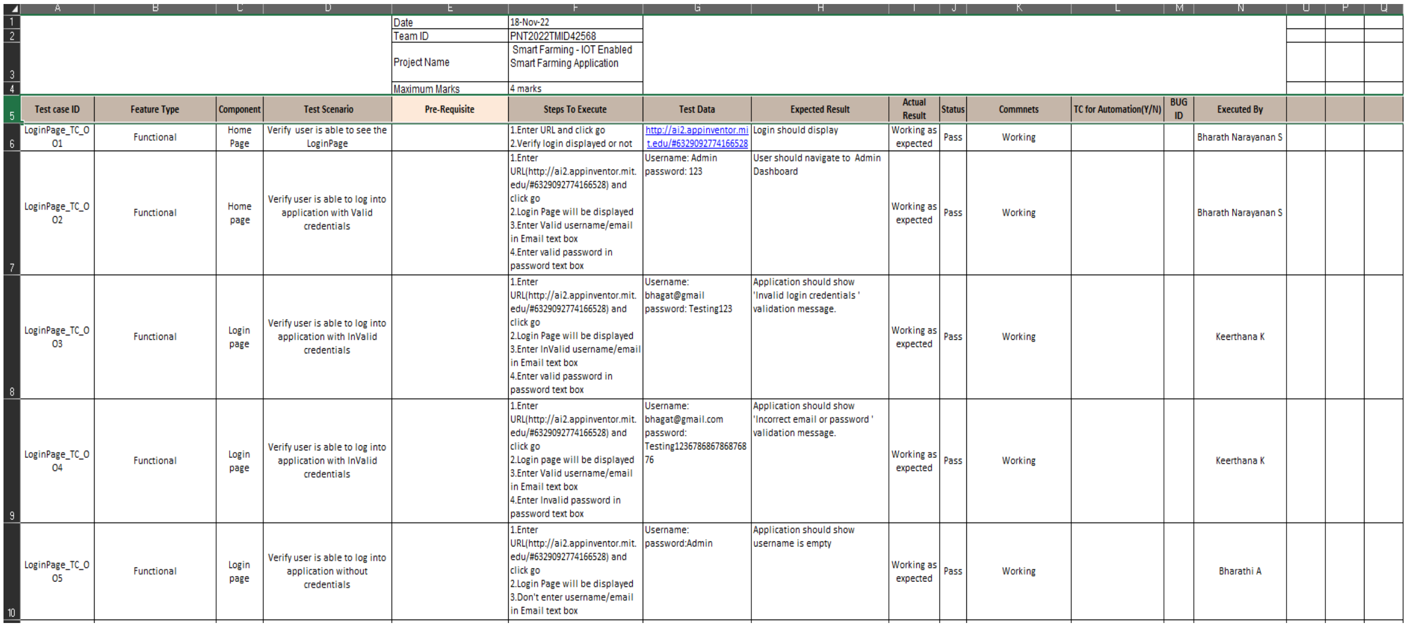
- Test cases ensure good test coverage
- Help improve the quality of software,
- Decreases the maintenance and software support costs
- Help verify that the software meets the end user requirements
- Allows the tester to think thoroughly and approach the tests from as many angles as possible
- Test cases are reusable for the future – anyone can reference them and execute the test.

So, these are a few reasons why test cases are extremely useful in software testing. Test cases are powerful artifacts that work as a good source of truth for how a system and a particular feature of software works. However, before we deep dive into the lessons for writing top-notch test cases, let us have a basic idea on the terminologies associated with them.

**Test Case Format**

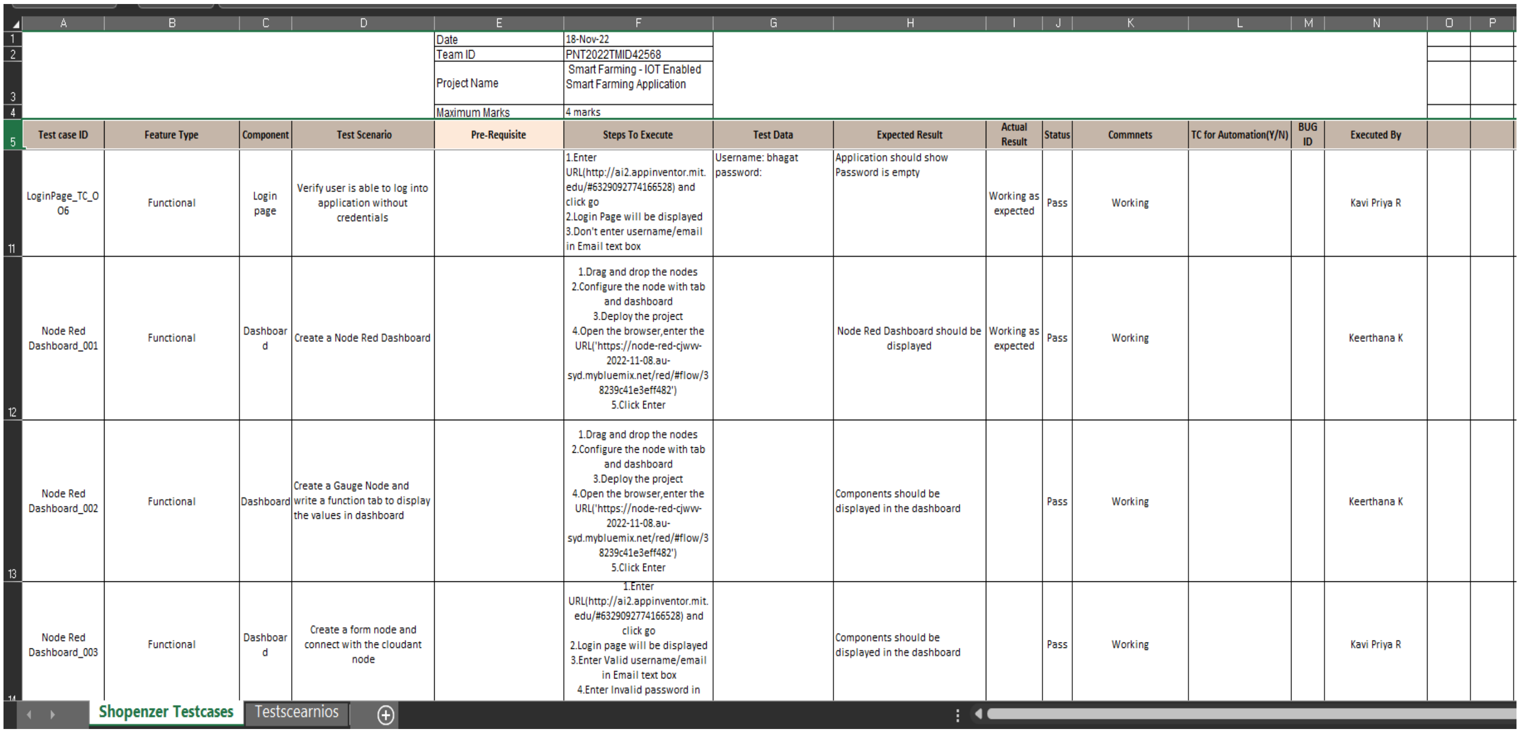
The primary ingredients of a test case are an ID, description, bunch of inputs, few actionable steps, as well as expected and actual results. Let's learn what each of them is:

- **Test Case Name:** A test case should have a name or title that is self-explanatory.
- **Test Case Description:** The description should tell the tester what they’re going to test in brief.
- **Pre-Conditions:** Any assumptions that apply to the test and any preconditions that must be met prior to the test being executed should be listed here.
- **Test Case Steps:** The test steps should include the necessary data and information on how to execute the test. The steps should be clear and brief, without leaving out essential facts.
- **Test Data:** It’s important to select a data set that gives sufficient coverage. Select a data set that specifies not only the positive scenarios but negative ones as well.
- **Expected Result:** The expected results tell the tester what they should experience as a result of the test steps.
- **Actual Result:** They specifies how the application actually behaved while test cases were being executed.
- **Comments:** Any other useful information such as screenshots that tester want’s to specify can be included here.



| | |
|---|---|
| Date | 18-Nov-22 |
| Team ID | PNT2022TMID42568 |
| Project Name | Smart Farming - IOT Enabled Smart Farming Application |
| Maximum Marks | 4 marks |

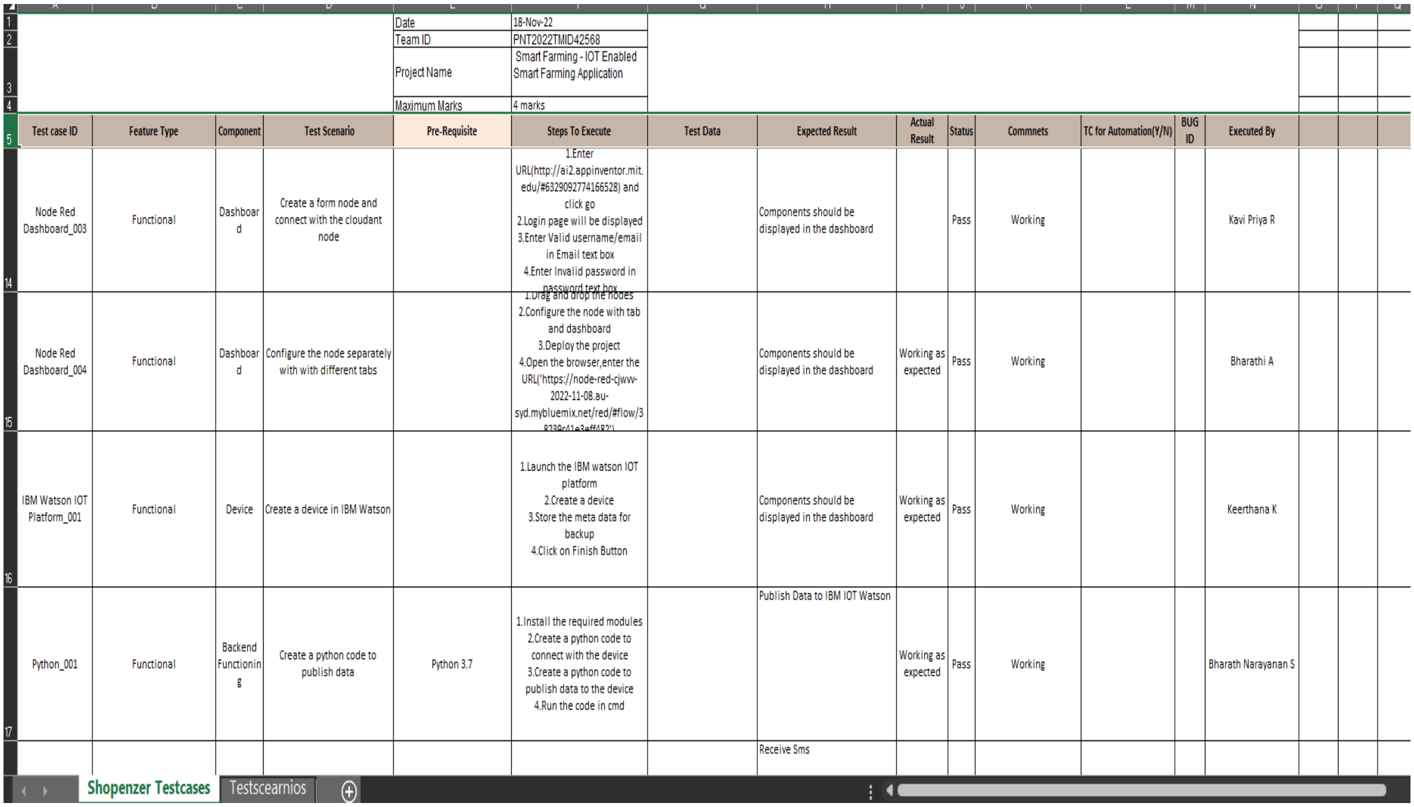
| Test case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commnets | TC for Automation(Y/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LoginPage_TC_O01 | Functional | Home Page | Verify user is able to see the LoginPage | | 1.Enter URL and click go<br>2.Verify login displayed or not | http://ai2.appinventor.mit.edu/#6329092774166528 | Login should display | Working as expected | Pass | Working | | | Bharath Narayanan S |
| LoginPage_TC_O02 | Functional | Home page | Verify user is able to log into application with Valid credentials | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login Page will be displayed<br>3.Enter Valid username/email in Email text box<br>4.Enter valid password in password text box | Username: Admin<br>password: 123 | User should navigate to Admin Dashboard | Working as expected | Pass | Working | | | Bharath Narayanan S |
| LoginPage_TC_O03 | Functional | Login page | Verify user is able to log into application with InValid credentials | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login Page will be displayed<br>3.Enter InValid username/email in Email text box<br>4.Enter valid password in password text box | Username: bhagat@gmail<br>password: Testing123 | Application should show 'Invalid login credentials ' validation message. | Working as expected | Pass | Working | | | Keerthana K |
| LoginPage_TC_O04 | Functional | Login page | Verify user is able to log into application with InValid credentials | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login page will be displayed<br>3.Enter Valid username/email in Email text box<br>4.Enter Invalid password in password text box | Username: bhagat@gmail.com<br>password: Testing123678686786876876 | Application should show 'Incorrect email or password ' validation message. | Working as expected | Pass | Working | | | Keerthana K |
| LoginPage_TC_O05 | Functional | Login page | Verify user is able to log into application without credentials | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login Page will be displayed<br>3.Don't enter username/email in Email text box | Username:<br>password:Admin | Application should show username is empty | Working as expected | Pass | Working | | | Bharathi A |



| | |
|---|---|
| Date | 18-Nov-22 |
| Team ID | PNT2022TMID42568 |
| Project Name | Smart Farming - IOT Enabled Smart Farming Application |
| 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_O O6 | Functional | Login page | Verify user is able to log into application without credentials | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login Page will be displayed<br>3.Don't enter username/email in Email text box | Username: bhagat<br>password: | Application should show Password is empty | Working as expected | Pass | Working | | | Kavi Priya R |
| Node Red Dashboard_001 | Functional | Dashboard | Create a Node Red Dashboard | | 1.Drag and drop the nodes<br>2.Configure the node with tab and dashboard<br>3.Deploy the project<br>4.Open the browser,enter the URL('https://node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/red/#flow/38239c41e3eff482')<br>5.Click Enter | | Node Red Dashboard should be displayed | Working as expected | Pass | Working | | | Keerthana K |
| Node Red Dashboard_002 | Functional | Dashboard | Create a Gauge Node and write a function tab to display the values in dashboard | | 1.Drag and drop the nodes<br>2.Configure the node with tab and dashboard<br>3.Deploy the project<br>4.Open the browser,enter the URL('https://node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/red/#flow/38239c41e3eff482')<br>5.Click Enter | | Components should be displayed in the dashboard | | Pass | Working | | | Keerthana K |
| Node Red Dashboard_003 | Functional | Dashboard | Create a form node and connect with the cloudant node | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login page will be displayed<br>3.Enter Valid username/email in Email text box<br>4.Enter Invalid password in | | Components should be displayed in the dashboard | | Pass | Working | | | Kavi Priya R |

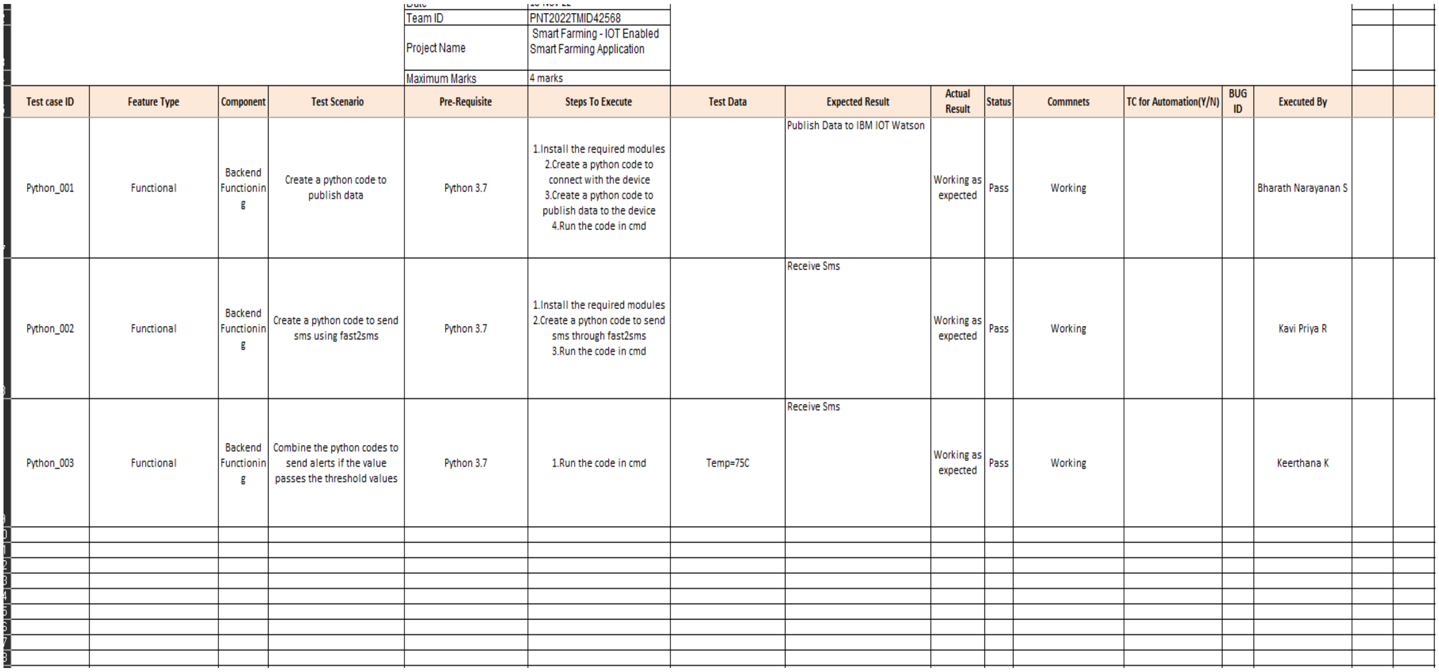
Shopenzer Testcases Testscearnios



| | |
|---|---|
| Date | 18-Nov-22 |
| Team ID | PNT2022TMID42568 |
| Project Name | Smart Farming - IOT Enabled Smart Farming Application |
| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Node Red Dashboard_003 | Functional | Dashboard | Create a form node and connect with the cloudant node | | 1.Enter URL(http://ai2.appinventor.mit.edu/#6329092774166528) and click go<br>2.Login page will be displayed<br>3.Enter Valid username/email in Email text box<br>4.Enter Invalid password in password text box | | Components should be displayed in the dashboard | | Pass | Working | | | Kavi Priya R |
| Node Red Dashboard_004 | Functional | Dashboard | Configure the node separately with with different tabs | | 1.Drag and drop the nodes<br>2.Configure the node with tab and dashboard<br>3.Deploy the project<br>4.Open the browser,enter the URL('https://node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/red/#flow/38239c41e3eff482') | | Components should be displayed in the dashboard | Working as expected | Pass | Working | | | Bharathi A |
| IBM Watson IOT Platform_001 | Functional | Device | Create a device in IBM Watson | | 1.Launch the IBM watson IOT platform<br>2.Create a device<br>3.Store the meta data for backup<br>4.Click on Finish Button | | Components should be displayed in the dashboard | Working as expected | Pass | Working | | | Keerthana K |
| Python_001 | Functional | Backend Functioning | Create a python code to publish data | Python 3.7 | 1.Install the required modules<br>2.Create a python code to connect with the device<br>3.Create a python code to publish data to the device<br>4.Run the code in cmd | | Publish Data to IBM IOT Watson | Working as expected | Pass | Working | | | Bharath Narayanan S |
| | | | | | | | Receive Sms | | | | | | |

Shopenzer Testcases Testscearnios

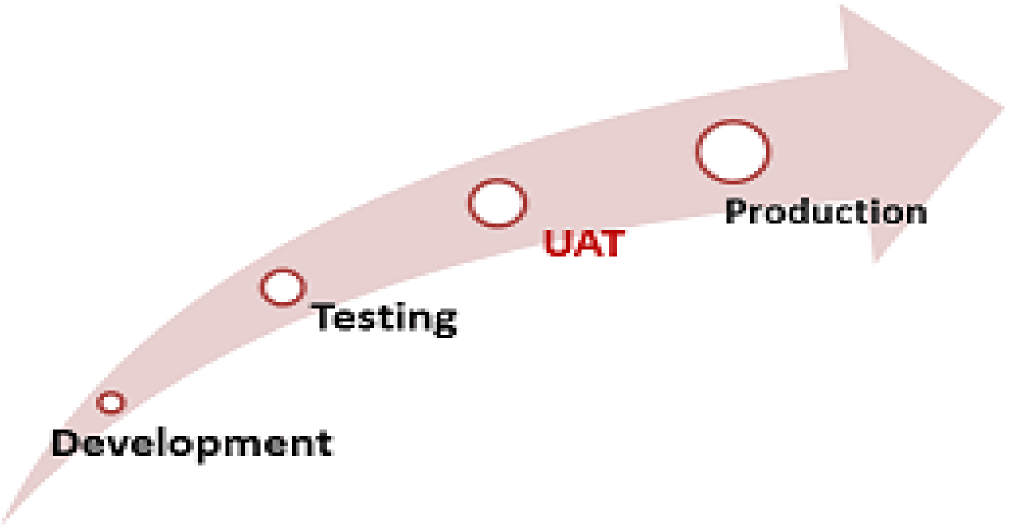


| | |
|---|---|
| Team ID | PNT2022TMID42568 |
| Project Name | Smart Farming - IOT Enabled Smart Farming Application |
| Maximum Marks | 4 marks |

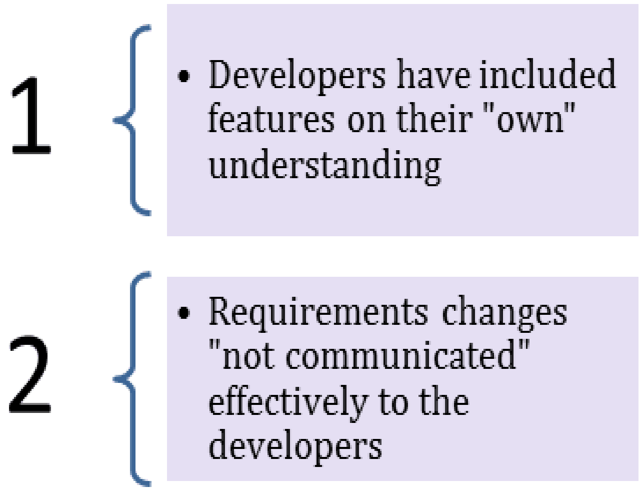
| Test case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commnets | TC for Automation(Y/N) | BUG ID | Executed By | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Python_001 | Functional | Backend Functioning | Create a python code to publish data | Python 3.7 | 1.Install the required modules<br>2.Create a python code to connect with the device<br>3.Create a python code to publish data to the device<br>4.Run the code in cmd | | Publish Data to IBM IOT Watson | Working as expected | Pass | Working | | | Bharath Narayanan S | | |
| Python_002 | Functional | Backend Functioning | Create a python code to send sms using fast2sms | Python 3.7 | 1.Install the required modules<br>2.Create a python code to send sms through fast2sms<br>3.Run the code in cmd | | Receive Sms | Working as expected | Pass | Working | | | Kavi Priya R | | |
| Python_003 | Functional | Backend Functioning | Combine the python codes to send alerts if the value passes the threshold values | Python 3.7 | 1.Run the code in cmd | Temp=75C | Receive Sms | Working as expected | Pass | Working | | | Keerthana K | | |

## 8.2 User Acceptance Testing:

**User Acceptance Testing (UAT)**is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.



## Purpose of UAT



1 • Developers have included features on their "own" understanding

2 • Requirements changes "not communicated" effectively to the developers

The purpose of this document is to briefly explain the test coverage and open issues of the [Iot Enabled Smart Farming Application] project at the time of the release to User Acceptance Testing (UAT). The UAT test plan outlines the strategy that will be used to verify and ensure an application meets its business in smart farming requirements. It documents entry and exit criteria for UAT, Test scenarios and test cases approach and timelines of testing. UAT is performed by, • Consumer • Administrator UAT is performed by –

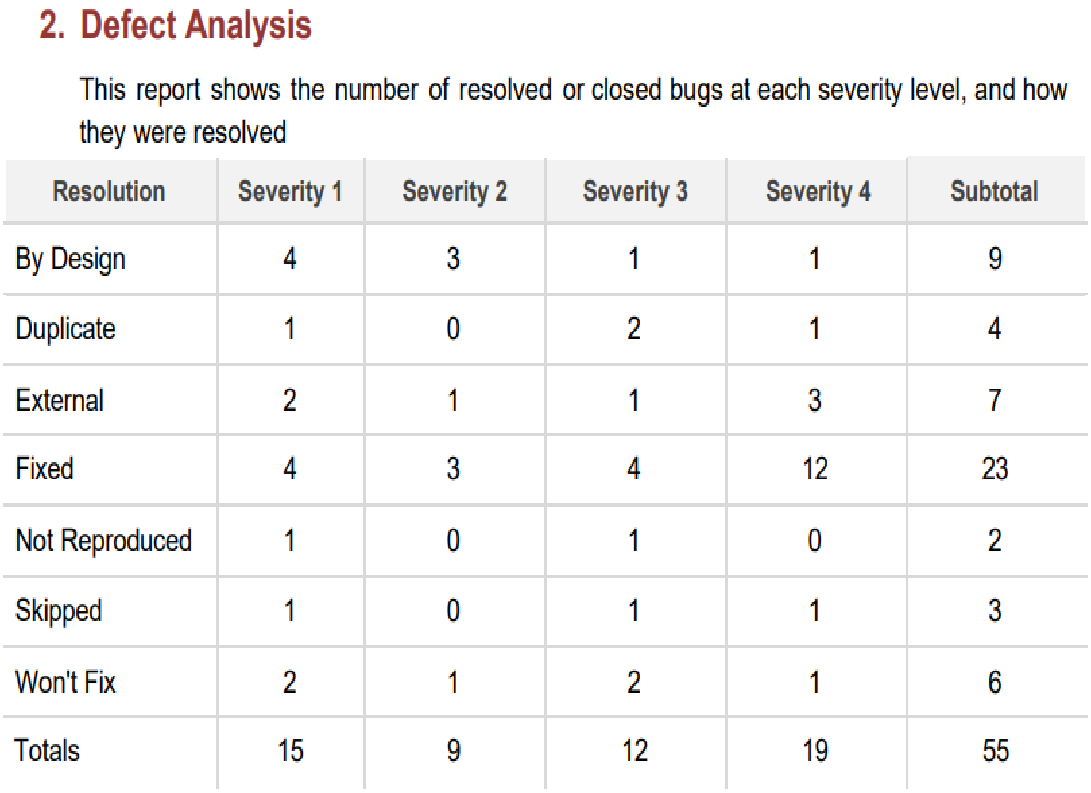
□ Consumers

□ Administrator

## Need of User Acceptance Testing

**Need of User Acceptance Testing** arises once software has undergone Unit, Integration and System testing because developers might have built software based on requirements document by their own understanding and further required changes during development may not be effectively communicated to them, so for testing whether the final product is accepted by client/end-user, user acceptance testing is needed.

- Developers code software based on requirements document which is their “own” understanding of the requirements and **may not actually be what the client needs from the software**.
- Requirements changes during the course of the project may not be communicated effectively to the developers.



## 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 | 4 | 3 | 1 | 1 | 9 |
| Duplicate | 1 | 0 | 2 | 1 | 4 |
| External | 2 | 1 | 1 | 3 | 7 |
| Fixed | 4 | 3 | 4 | 12 | 23 |
| Not Reproduced | 1 | 0 | 1 | 0 | 2 |
| Skipped | 1 | 0 | 1 | 1 | 3 |
| Won't Fix | 2 | 1 | 2 | 1 | 6 |
| Totals | 15 | 9 | 12 | 19 | 55 |

## 3. Test Case Analysis

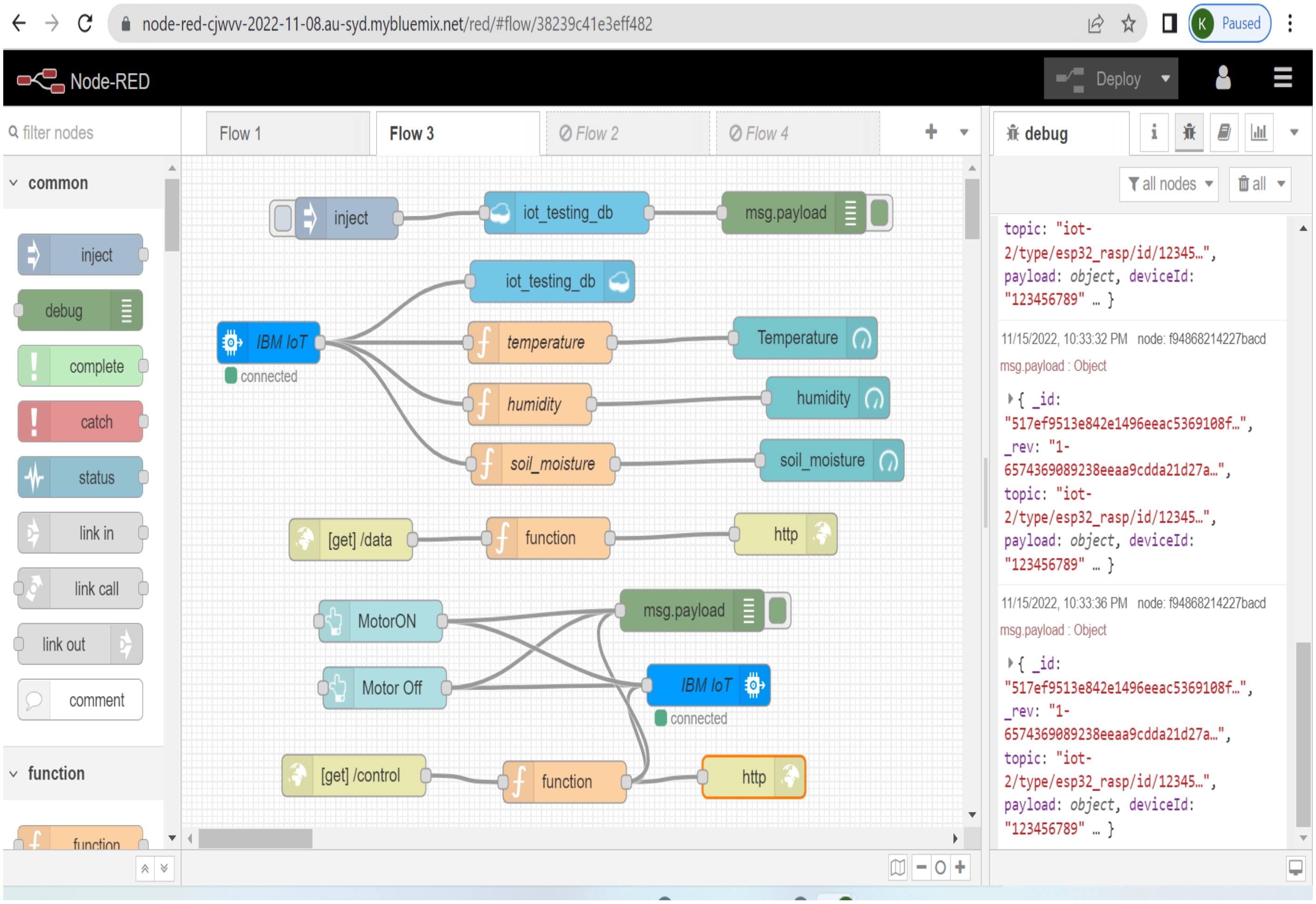
This report shows the number of test cases that have passed, failed, and untested

| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| Login | 6 | 0 | 0 | 6 |
| Node-red Dashboard | 19 | 0 | 0 | 19 |
| IBM Watson IOT Platform | 5 | 0 | 0 | 5 |
| MIT App Inventor | 2 | 0 | 0 | 2 |

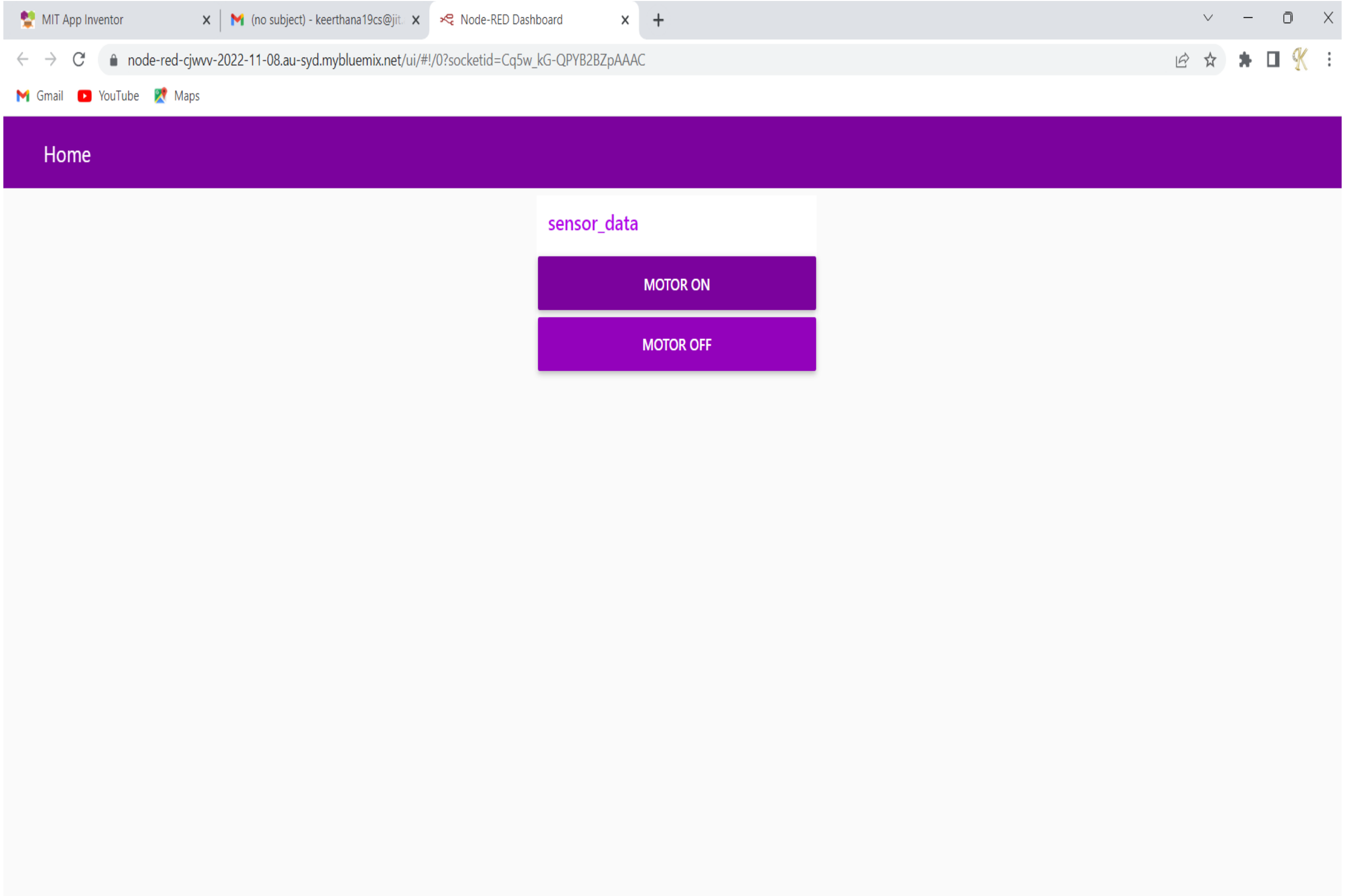
# 9. RESULTS

## 9.1 Performance Metrics:

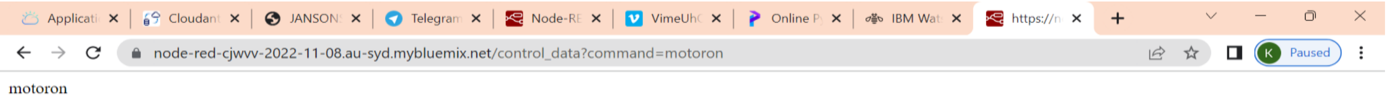
- ✓ These metrics are used to track and measure the effectiveness and profitability of various projects.
- ✓ Each stage of the project is tracked and measured against the goals that the project set out to achieve.
- ✓ The data compiled from the metrics can be used to plan future projects and gives insight on how to make projects more efficient.



```
topic: "iot-
2/type/esp32_rasp/id/12345…",
payload: object, deviceId:
"123456789" … }

11/15/2022, 10:33:32 PM   node: f94868214227bacd
msg.payload : Object
▸{ _id:
"517ef9513e842e1496eeac5369108f…",
_rev: "1-
6574369089238eeaa9cdda21d27a…",
topic: "iot-
2/type/esp32_rasp/id/12345…",
payload: object, deviceId:
"123456789" … }

11/15/2022, 10:33:36 PM   node: f94868214227bacd
msg.payload : Object
▸{ _id:
"517ef9513e842e1496eeac5369108f…",
_rev: "1-
6574369089238eeaa9cdda21d27a…",
topic: "iot-
2/type/esp32_rasp/id/12345…",
payload: object, deviceId:
"123456789" … }
```

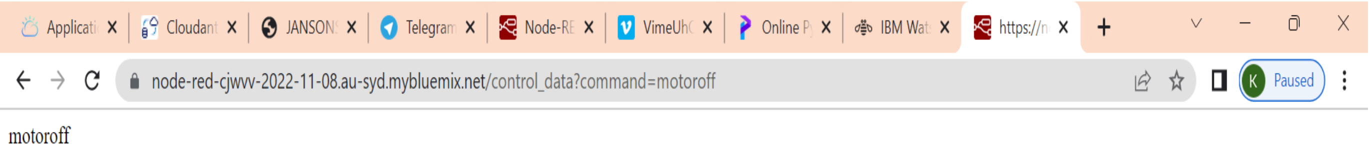


node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/ui/#!/0?socketid=Cq5w_kG-QPYB2BZpAAAC



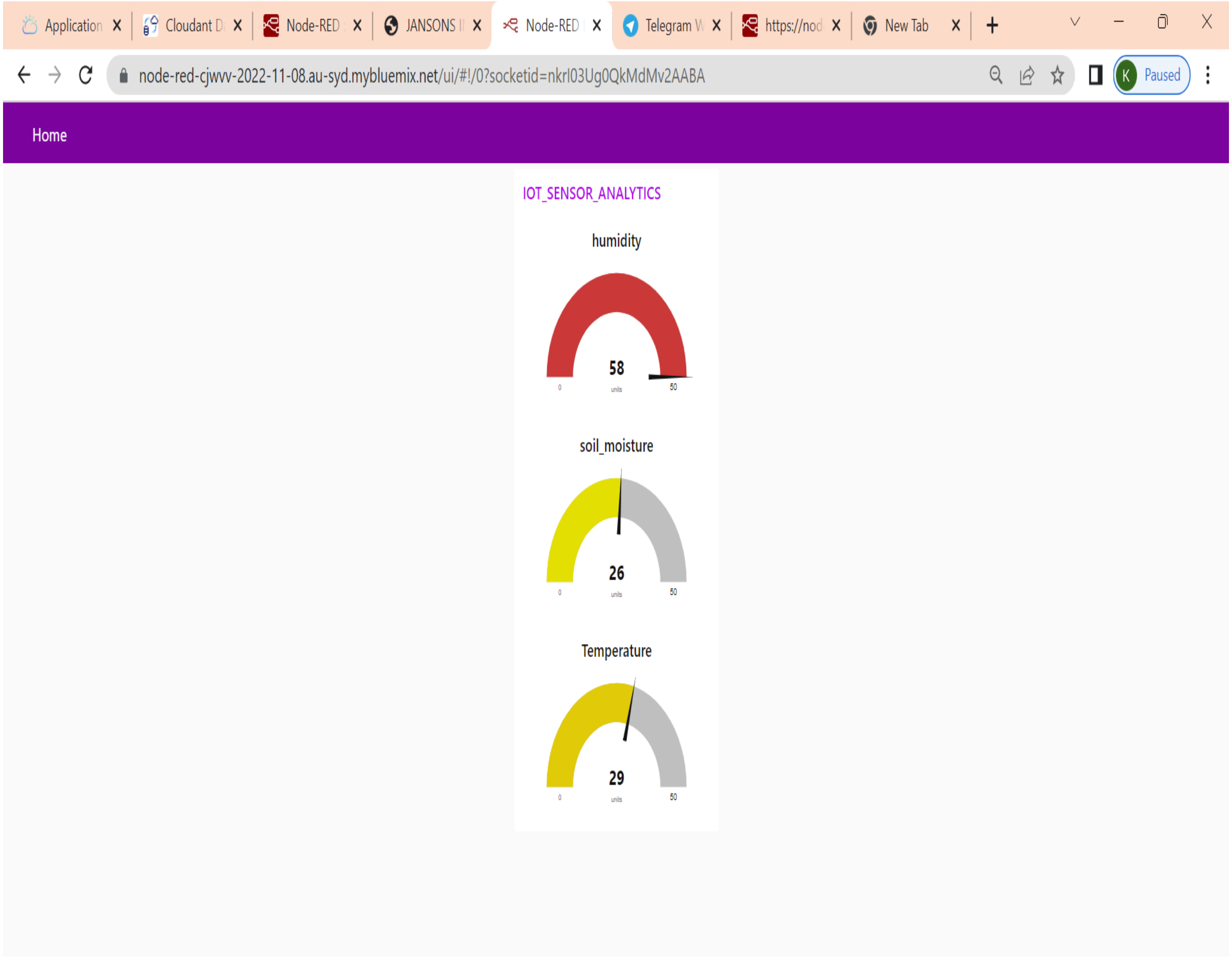
node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/control_data?command=motoron

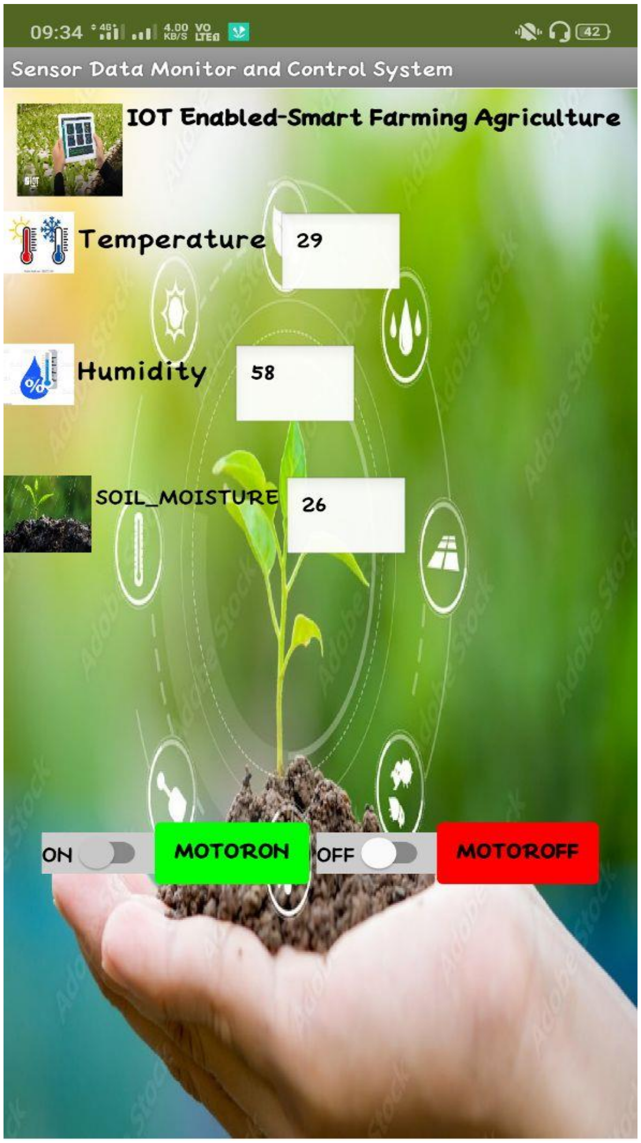
```
motoron
```



node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/control_data?command=motoroff

```
motoroff
```


Application
Cloudant D
Node-RED
JANSONS I
Node-RED
Telegram W
https://nod
New Tab
node-red-cjwvv-2022-11-08.au-syd.mybluemix.net/ui/#!/0?socketid=nkrl03Ug0QkMdMv2AABA
K
Paused
Home
IOT_SENSOR_ANALYTICS
humidity
58
0
units
50
soil_moisture
26
0
units
50
Temperature
29
0
units
50



Sensor Data Monitor and Control System

IOT Enabled-Smart Farming Agriculture

Temperature 29

Humidity 58

SOIL_MOISTURE 26

# 10. ADVANTAGES & DISADVANTAGES

## ADVANTAGES

- It allows farmers to maximize yields using minimum resources such as water, fertilizers, seeds etc.

- Smart agriculture uses drones and robots which helps in many ways. These improves data collection process and helps in wireless monitoring and control.

- It is cost effective method
  .
- It delivers high quality crop production.

## DISADVANTAGES

- The smart agriculture needs availability of internet continuously. Rural part of most of the developing countries do not fulfil this requirement. Moreover, internet connection is slower.

- The smart farming based equipment require farmers to understand and learn the use of technology. This is major challenge in adopting smart agriculture farming at large scale across the countries.

# 11. CONCLUSION

Smart Agriculture using IoT can have a booming effect on today’s agricultural growth. From this article, the main objective of this project is to improve the crop yield and thereby meet the demand. This project remotely measures and monitor water moisture levels in the soil to ensure that crops are getting optimal water resources and automatically trigger sprinkler systems to address low moisture levels in the soil to prevent crop damage or loss. This idea will improve the crop yield and manage them. The environmental conditions and the soil conditions can be easily and accurately monitored with the help of IoT based sensors. This methodology has a capability of sensing the moisture, humidity, temperature and obstacle. These values can be used in an efficient and effective way to take all necessary steps and automate the process to agriculture.

# 12. FUTURE SCOPE

**IoT solutions are focused on helping farmers close the supply demand gap**, by ensuring high yields, profitability, and protection of the environment. The approach of using IoT technology to ensure optimum application of resources to achieve high crop yields and reduce operational costs is called precision agriculture. IoT in agriculture technologies comprise specialized equipment, wireless connectivity, software and IT services.

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 (compared to being slightly over $5 billion back in 2016).

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. IoT smart farming solutions is a system that is built for monitoring the crop field with the help of sensors (light, humidity, temperature, soil moisture, crop health, etc.) and automating the irrigation system. **The farmers can monitor the field conditions from anywhere.** They can also select between manual and automated options for taking necessary actions based on this data. For example, if the soil moisture level decreases, the farmer can deploy sensors to start the irrigation. Smart farming is highly efficient when compared with the conventional approach.

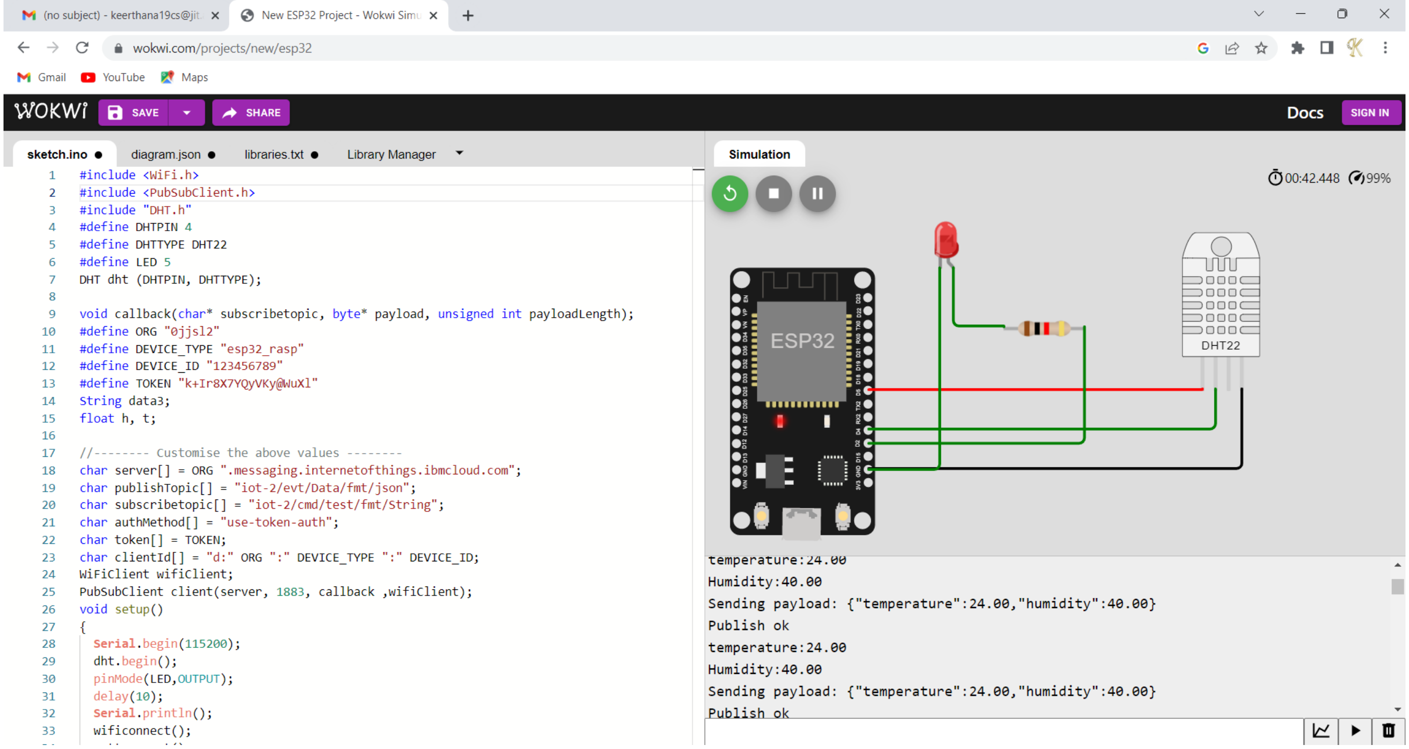
# 13. APPENDIX

## Source Code:

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

organization = "Org ID" #replace the ORG ID

deviceType = "Device Type"#replace the Device type wi

deviceId = "Device ID"#replace Device ID

authMethod = "token"

authToken = "authtoken" #Replace the authtoken

def myCommandCallback(cmd):

print("Command received: %s" % cmd.data)

if cmd.data['command']=='lighton':

  print("MOTOR ON IS RECEIVED")

elif cmd.data['command']=='lightoff':

print("MOTOR OFF IS RECEIVED")

  if cmd.command == "setInterval":

if 'interval' not in cmd.data:

  print("Error - command is missing required information: 'interval'")

else:

interval = cmd.data['interval']

elif cmd.command == "print":

if 'message' not in cmd.data:

print("Error - command is missing required information: 'message'")
```

```
else:

output=cmd.data['message']

  print(output)

  try:

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}

deviceCli = ibmiotf.device.Client(deviceOptions)

except Exception as e:

print("Caught exception connecting device: %s" % str(e))

sys.exit()

deviceCli.connect()

  while True:

deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
```

# OUTPUT



```
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"
#define DHTPIN 4
#define DHTTYPE DHT22
#define LED 5
DHT dht (DHTPIN, DHTTYPE);

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "0jjsl2"
#define DEVICE_TYPE "esp32_rasp"
#define DEVICE_ID "123456789"
#define TOKEN "k+Ir8X7YQyVKy@WuXl"
String data3;
float h, t;

//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
void setup()
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
```

```
temperature:24.00
Humidity:40.00
Sending payload: {"temperature":24.00,"humidity":40.00}
Publish ok
temperature:24.00
Humidity:40.00
Sending payload: {"temperature":24.00,"humidity":40.00}
Publish ok
```

# GitHub &amp; Project Demo Link :

## GitHub &amp;:

https://github.com/IBM-EPBL/IBM-Project-21204-1659775126

## Project Demo Link:

https://www.youtube.com/watch?v=Z4TlfN5Ag-w