PROJECT REPORT

SMART FARMER – IoT ENABLED SMART FARMING APPLICATION

TEAM ID: PNT2022TMID17614

ROLL NUMBER	NAME
713319EC010	ARUNPRASANTH E
713319EC038	JAIKRISHNAN V
713319EC040	JAYENDRA S
713319EC043	KAMALESH P

Project Report Format

1. **INTRODUCTION**
   - 1.1 Project Overview
   - 1.2 Purpose
2. **LITERATURE SURVEY**
   - 2.1 Existing problem
   - 2.2 References
   - 2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   - 3.1 Empathy Map Canvas
   - 3.2 Ideation & Brainstorming
   - 3.3 Proposed Solution
   - 3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   - 4.1 Functional requirement
   - 4.2 Non-Functional requirements
5. **PROJECT DESIGN**
   - 5.1 Data Flow Diagrams
   - 5.2 Solution & Technical Architecture
   - 5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
   - 6.1 Sprint Planning & Estimation
   - 6.2 Sprint Delivery Schedule
   - 6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   - 7.1 Feature 1
   - 7.2 Feature 2
   - 7.3 Database Schema (if Applicable)
8. **TESTING**
   - 8.1 Test Cases
   - 8.2 User Acceptance Testing
9. **RESULTS**
   - 9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

    Source Code

    GitHub & Project Demo Link

INTRODUCTION

1.1 Project Overview:

People who use the internet of things can live and work more intelligently and have total control over their life. IoT is crucial to business in addition to providing smart home automation devices. With the help of IoT, organizations can see in real time how their systems actually function, gaining insights into anything from equipment performance to supply chain and logistics activities. This paper presents an Internet of Things (IoT) based Smart Farmer - IoT Enabled Smart Farming Application. The system is implemented using an ultrasonic sensor which is connected to Arduino UNO as to monitor the water level. In this system, the depth level will be sent via Arduino Ethernet Shield with an Internet connection to the IBM Cloud.

1.2 Purpose:

Since the dawn of human civilization, agriculture has been considered to be the most significant activity. Traditional irrigation techniques, such as flood irrigation and overhead sprinkler irrigation, are not very effective. They waste a significant amount of water, and the excessive moisture in the soil can also encourage the growth of diseases like fungus. Since water is a valuable resource and indirectly supports the survival of the farm, an automated irrigation system is crucial. Around 85% of all water resources are used exclusively for irrigation purposes worldwide. This need is anticipated to rise in the coming years due to population growth. We must implement new strategies that reduce the amount of water needed for irrigation in order to meet this demand. In an automation system, sensors are used to monitor the crop's access to water, and controlled irrigation is used to water the crop as needed. IoT is a new platform that is extremely beneficial to people all over the world. IoT is at the heart of such revolutionary growth engines. IoT is possible because of adequate power supply and internet connectivity. The term "Internet of Things" is commonly used to describe a framework in which sensors are connected to objects and allow these objects to share their "digital voice" with the outside world via an internet connection. IoT has recently evolved into a collection of purpose-built networks.

LITERATURE SURVEY

2.1 Existing Problem:

S.NO	TITLE	AUTHOR AND YEAR OF PUBLICATION	METHODOLOGY USED
1.	Mobile integrated Smart irrigation management and monitoring system using IoT	S. Vaishali et.al, 08 February 2018	In order to control and monitor the irrigation process, smart and automated irrigation system is developed, Implemented and tested. There is a need for automated irrigation system because it is simple and easy to install. This system uses values ON and OFF to control water motor. Python programming language is been used for automation purpose.
2.	IoT Based Smart Irrigation Monitoring And Controlling System	Shweta B. Saraf et.al ,15 January 2018	In this paper proposed system is based on IoT that uses real time input data. Smart farm irrigation system uses android phone for remote monitoring and controlling of drips through wireless sensor network. Zigbee is used for communication between sensor nodes and base station. Real time sensed data handling and demonstration on the server is accomplished using web-based java graphical user interface.

2.2 References:

1.Mobile Integrated Smart Irrigation Management and Monitoring System Using IoT

Date of conference: 06-08

April 2017 Publisher: IEEE

Date added to IEEE Xplore: 08February 2018

DOI: 10.1109/ICCSP.2017.8286792

2.IoT Based Smart Irrigation Monitoring and Controlling System

Date Added to IEEE Xplore: 15January 2018

ISBN Information: Electronic ISBN: 978-1-5090

Date of conference: 19-20 May 2017

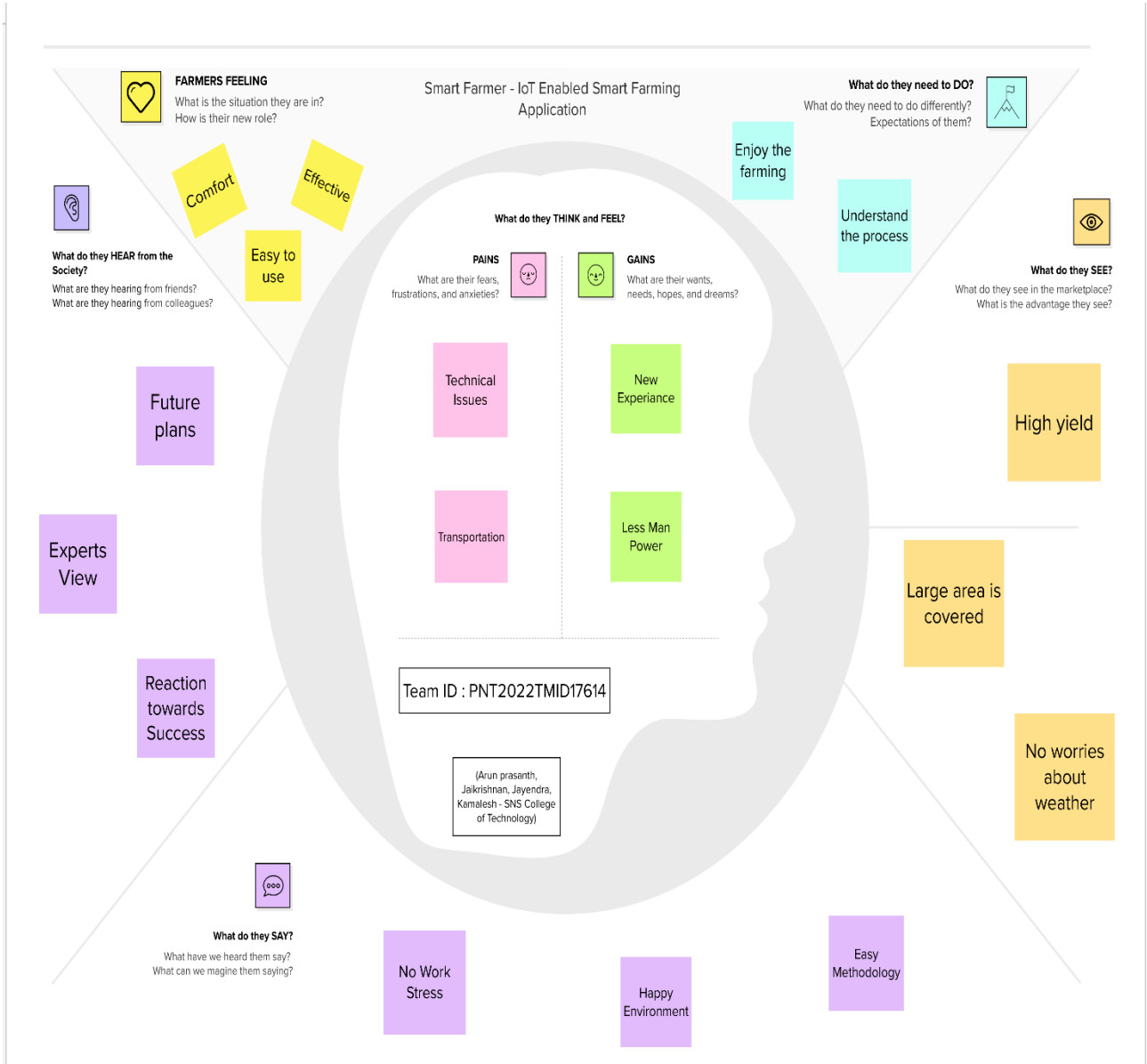
INSPEC Accession Number: 17504411

2.3 Problem statement definition:

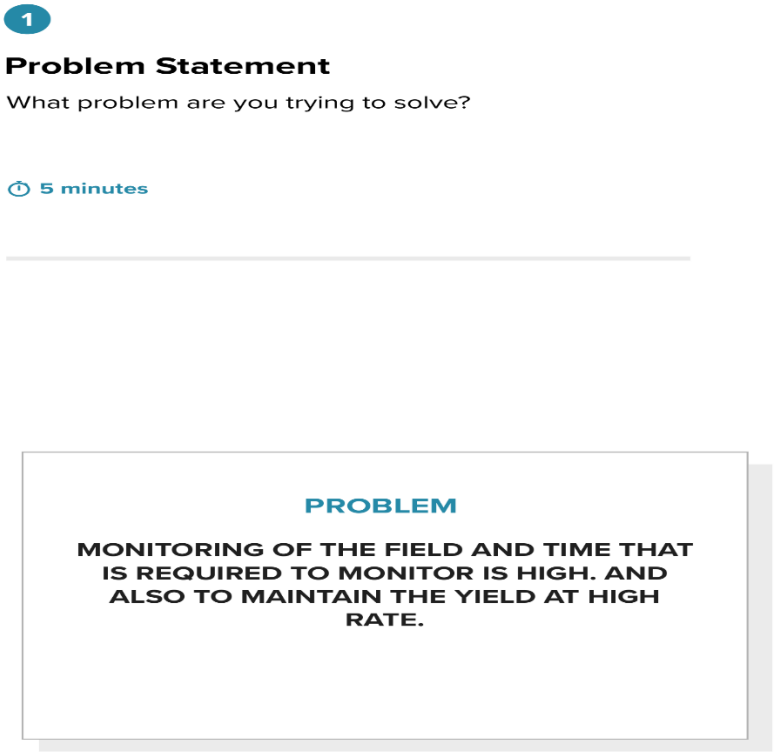
Agriculture is the Backbone of Our Country. Traditional methods that are used for irrigation. They result in a lot of wastage of water. About 85% of total available water resources across the world are solely used for the irrigation purpose. In upcoming years this demand is likely to increase because of increasing population. To meet this demand, we must adopt new techniques which will conserve need of water for irrigation process. In this paper proposed system is based on IoT that uses real time input data. This Water level monitoring irrigation system the excess availability of water in crop is monitored through sensors and reduces the water consumption. This idea is also to focus on parameters such as temperature and soil moisture. The main objective of this project is to control reduce the water supply, save the crops and monitor the plants. The system is implemented using an ultrasonic sensor which is connected to Arduino UNO as to monitor Farm Field level. In this system, Farm Field depth level will be sent via Arduino Ethernet Shield with an Internet connection to the IBM IoT Cloud. The IBM Cloud store the collected Farm field level data into IoT database and display the Farm Field depth level on online dashboard for real-time visualization. The IBM event manager invoke a notification alert to the owner of the farmer mobile phone via a SMS when the farm field is nearly filled and It automatically Switch Off the Water Motor.

IDEATION AND PROPOSED SOLUTION

3.1 Empathy map canvas:



3.2 Ideation and Brainstorming:



2

Brainstorm

Ideas that address the Problems

10 minutes

Jaikrishnan

- Communication is based on MQTT protocol
- Light intensity BH 1750 sensor is used
- ACQUI WEATHER COULD BE USED FOR WEATHER MONITOR
- L293D is Used as the Motor driver

Arun Prasanth

- Decision Tree Algorithm is used
- Optical Sensor is Used place in the drone for moisture monitoring
- Semiconductor based Sensor (range from -70 to 180 degree celcius)
- Open Weather Map is used

Jayendra

- IOT core is used
- RTD sensor is used for temperature controll
- Tensiometer Sensor is used
- SPI serial pheriperal interface is used

Kamalesh

- ATREE soil Sensor for moisture measurement
- Serial USART can be used for Communication
- RS-PH-N01-TR-1 can be used asPH sensor
- Node Red can be used as the Programming tool

3

Group ideas

This is a textbox...

20 minutes

Protocols used

Decision Tree Algorithm is used

Communication is based on MQTT protocol

Sensord used

Semiconductor based Sensor (range from -70 to 180 degree celcius)

RS-PH-N01-TR-1 can be used asPH sensor

Tensiometer Sensor is used

Programming tool

Node Red can be used as the Programming tool

Online Stimulation is done through Tinkercad

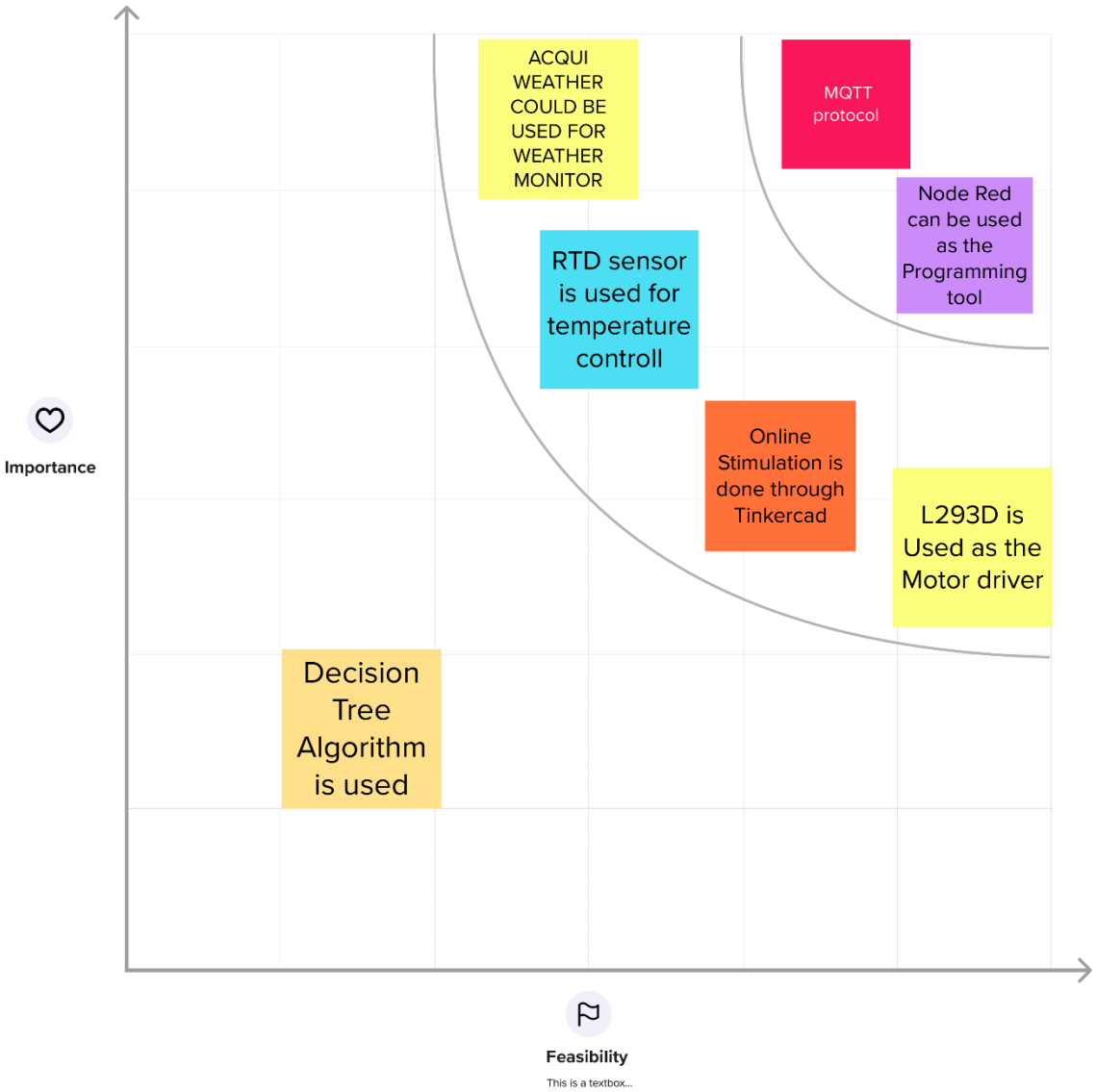
Interfacing Motor

L293D is Used as the Motor driver

4

Prioritize

20 minutes



This is a textbox...

3.3 Proposed solution fit:

S. No	Parameter	Description
1.	Problem Statement (Problem to be solved)	To monitor the water level of crops during irrigation.
2.	Idea / Solution description	To make crop irrigation easy and effective and monitored using IBM Watson IoT Platform and Node-RED Service.
3.	Novelty / Uniqueness	A mobile application is developed for monitoring and controlling the irrigation.
4.	Social Impact / Customer Satisfaction	No need to worry about the condition of the crops and water supply. The present condition of the field can be generated as a report.
5.	Business Model (Revenue Model)	One time subscription
6.	Scalability of the Solution	It can handle the irrigation process and can perform several actions based on the climate condition and seasonal changes.

3.4 Problem solution fit:

Define CS, fit into CC

1. CUSTOMER SEGMENT(S)

Who is your customer?

Farmers are our customers

6. CUSTOMER CONSTRAINTS

What constraints prevent your customers from taking action or limit their choices

of solutions?

Time Consumption, Easy to use, the availability of device, proper network facilities and budget are several constraints, Knowledge about the application.

5. AVAILABLE SOLUTIONS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have?

Most commonly used irrigation type is Drip irrigation the most common disadvantage is when the water is not filtered properly there will be clogs and the tubes will get affected easily. In smart farming we can use solar empowered smart

Explore AS, differentiate

2. JOBS-TO-BE-DONE / PROBLEMS

Which jobs-to-be-done (or problems) do you address for your customers?

Soil analysis, climatic conditions can be monitored, devices can be controlled using mobile.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?

Knowledge on how to water the plants accordingly, how to make use of the software etc.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?

The customers will reach us when they don't have idea on how to analyse the soil and to improve the current

3. TRIGGERS

TR

What triggers customers to act?

To get correct accuracy on what to be done on the farm and to produce more crops and livestock quantitively.

4. EMOTIONS: BEFORE / AFTER

EM

How do customers feel when they face a problem or a job and afterwards?

Crop yield is high, as when the productivity increases farmers will be

satisfied. They will not worry about the loss. Irrigation will be more efficient than before

10. YOUR SOLUTION

SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.

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

Team ID: **(PNT2022TMID17614)**

8. CHANNELS of BEHAVIOUR

CH

8.1 ONLINE

8.2

What kind of actions do customers take online?

we will reach the customer directly ask

about their problems and provide effective solutions if their problems

8.3 OFFLINE

What kind of actions do customers take offline?

In offline mode will do digital marketing using advertisements

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 via Email & OTP
FR-3	User Location	Share location through the mobile phones
FR-4	User Enter the No. of Acres	Mention the number of acres through Smart farming applications
FR-5	User Select the Category	Select the display option like Weather, drone services, Soil testing etc.
FR-6	Confirmation of User Details & Payment Process	Enter the valid data and payment through online transaction

4.2 Non-Functional requirements:

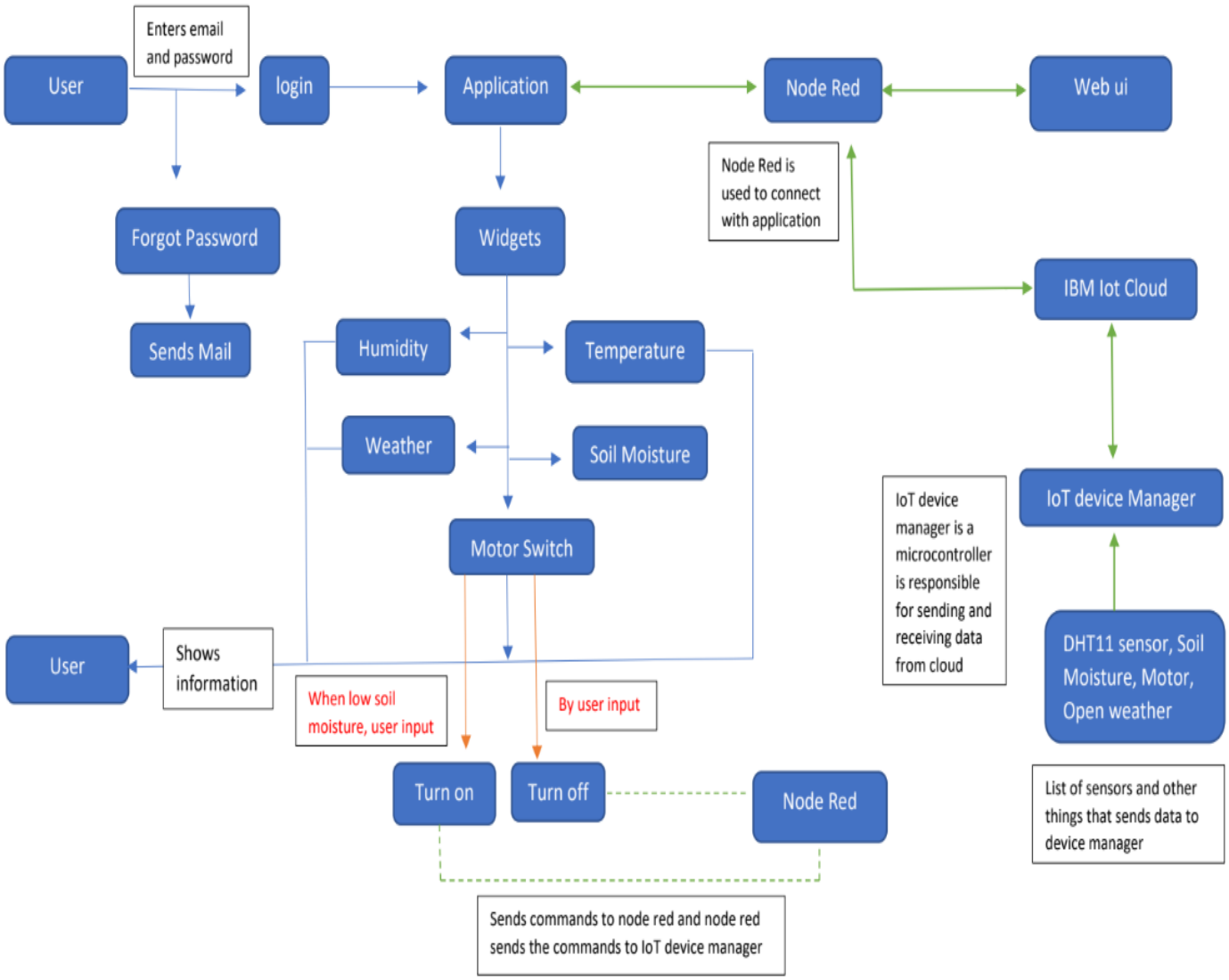
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	User Friendly Guidelines are available for the user in the application
NFR-2	**Security**	The process of protecting data from Unauthorized Access
NFR-3	**Reliability**	Consistency and Accuracy and the shared protection achieves a better trade-off between costs and reliability & Once the Database is Updated the user can access the data at anytime
NFR-4	**Performance**	The Front-Page load time must be no more than 5 seconds for the user based on the network connection
NFR-5	**Availability**	In future, if there is any modification in the application it will not affect the user data, frontend & Back- end of the application
NFR-6	**Scalability**	The Website traffic limit must be scalable

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 where the data is stored.



5.2 Solution & Technical architecture:

Technical Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to;

- Find the best tech solution to solve existing business problems.

- Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.

- Define features, development phases, and solution requirements.

- Provide specifications according to which the solution is defined, managed, and delivered.

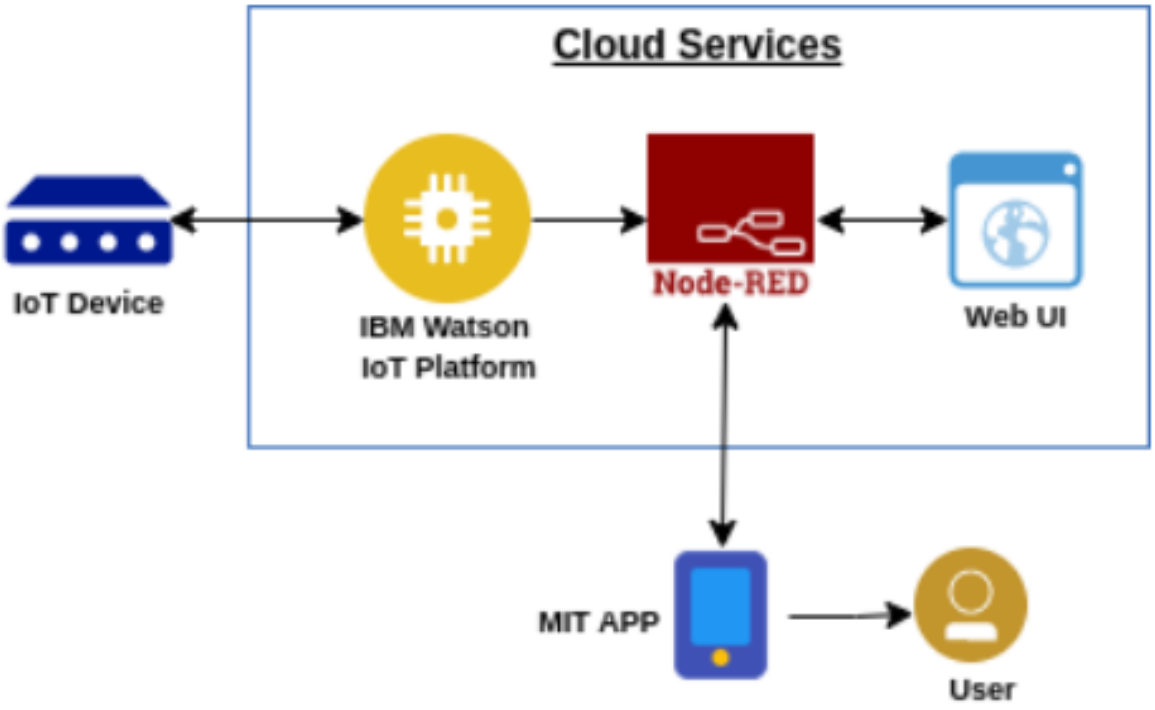


Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	Mobile app. In our application we are data are displayed using widgets like structure. Users interacts with widgets to additional info	MIT App Inventor
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Database Type, Configurations etc.	Firebase is NoSQL database
6.	Cloud Database	Database Service on Cloud	Firebase, IBM Watson IoT Cloud Platform
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Purpose of External API used in the application	Open weather API
9.	External API-2	Purpose of External API used in the application	Firebase API
10.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration, Cloud Server Configuration	Local, IBM Cloud, Firebase
11.	DHT11 sensor, Soil Moisture sensor	It used to monitor the soil, temperature, humidity.	PST's Advanced Ceramic Metal -Oxide Moisture Sensor technology allows for measurements of dew-point, moisture content and trace moisture in both gases and liquids.

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Node Red, MIT App Inventor, Arduino IDE Node Red for connecting with application, MIT App Inventor for building app, Arduino is open-source electronics platform to build hardware and software.	It is a software, which helps in connecting and building application. Node Red, MIT App Inventor, Arduino IDE.
2.	Security Implementations	HTTPS Connections, X-Force Red IoT Testing.	Encryptions, Secured Connection
3.	Scalable Architecture	Architecture is scalable from 10 devices to 300 devices easily and account is also scalable up to thousand connections. For very high scalability we need to upgrade our cloud plan.	Firebase, IBM Cloud
4.	Availability	Availability of our application is 24/7 because which use a cloud technology. Firebase will use commercially reasonable efforts to make Firebase available with a Monthly Uptime Percentage of at least 99.95% and distributed servers.	Firebase, IBM Cloud
5.	Performance	Number of requests is 2 requests per 20 seconds or 4 requests per 30 second and sometimes user request will be added with respective to the requests.	Technology used

5.3 User stories:

User Stories

Use the below template to list all the user stories for the product.

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	Medium	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		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user I want to see everything in single widget		Medium	Sprint-2
		USN-7	As a user I want a organised widgets section		High	Sprint-2
		USN-8	As a user I want a graphical/pictorial representation		Low	Sprint-2
Customer (Web User)	Dashboard	USN-9	As a user I want a graphical representation of data for better understanding		High	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	Have to use a least sensor and get better output		High	Sprint-2
		USN-11	As a user, I need a low cost IoT devices for farming		High	Sprint-2
		USN-12	As a user, I need a multiple sensors for various data		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 don't how to use the application	Manual guide will be there	Medium	Sprint-3
		USN-14	As a user, I need my application to work on most of the mobiles		High	Sprint-3
		USN-15	As a user, I am facing issue in the application	Query form will be there	High	Sprint-3
Administrator	Query Clarification	USN-16	As a admin, I give solutions to their queries		High	Sprint-3
	Particular Access	USN-17	As a admin, I give access only to authorised person		High	Sprint-3
	Connection with IoTdevices	USN-18	As a admin, I ensure the correct working of the devices. If any problem arises it will be shared to user		Medium	Sprint-4
Customer (Mobile user)	Application	USN-19	As a user, I need to control my devices	Commands for devices	High	Sprint-4
		USN-20	As a user, I need a events for better productivity		Low	Sprint-4
		USN-21	As a user, I need a more info about plants inside a application		Medium	Sprint-4

PROJECT PLANNING & SCHEDULING

6.1 Sprint planning & estimation:

S. No	Milestone	Activities	Date	Status
1.	IBM Cloud services	Pre-Requisites	27.08.2022	Completed
		MIT app inventor		
		Creating an account in Fast2SMS dashboard		
		Software installation (Python IDLE)		
2.	Project objectives	Prepare the project objectives	27.08.2022	Completed
3.	Create and Configure IBM cloud services	Create an IBM Watson IOT platform and devices	03.09.2022	Completed
		Create Node Red services	10.09.2022	Completed
4.	Develop a Python script	Develop a Python Code	19.09.2022	Completed
5.	Build a web application using Node-Red services	Building a web application using Node-Red	12.10.2022	Completed
6.	Develop a mobile application	Developing a mobile application	15.10.2022	Completed
7.	Ideation phase	Literature survey	03.09.2022	Completed
		Empathy map	10.09.2022	Completed
		Brainstorming	17.09.2022	Completed
8.	Project design phase - I	Proposed solution	19.09.2022	Completed
		Problem solution fit	01.10.2022	Completed
		Solution Architecture	02.10.2022	Completed

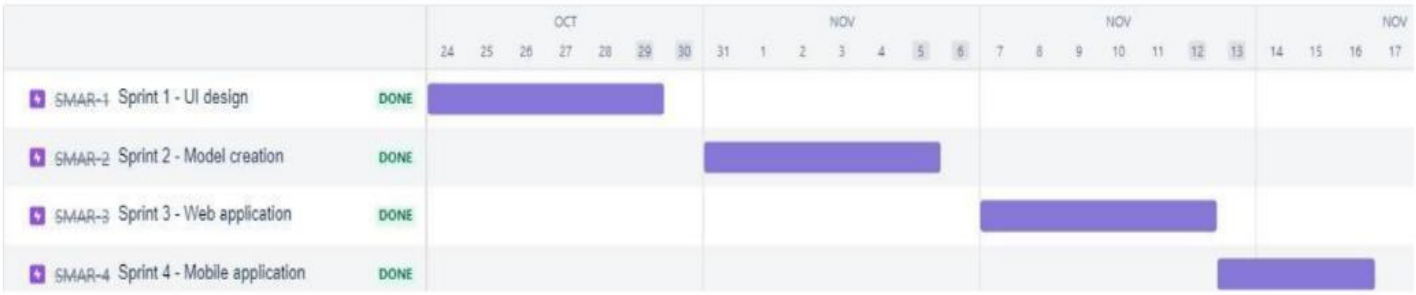
9.	Project Design phase - II	Customer journey	08.10.2022	Completed
		Functional requirement	15.10.2022	Completed
		Data flow diagrams	17.10.2022	Completed
		Technology architecture	19.10.2022	Completed
10.	Project planning phase	Milestone and activity list	26.10.2022	Completed
		Sprint delivery plan	28.10.2022	Completed
11.	Project development phase	Delivery of sprint 1	01.11.2022	Completed
		Delivery of sprint 2	05.11.2022	Completed
		Delivery of sprint 3	09.11.2022	Completed
		Delivery of sprint 4	12.11.2022	Completed

6.2 Sprint delivery schedule:

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Simulation	USN-1	Connect sensors, Arduino andesp8266	10	High	Arunprasanth E Jaikrishnan V
Sprint-1	Software	USN-2	Develop an application with MIT App inventor (Login page)	10	High	Jayendra S Kamalesh P

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Software and Hardware	USN-3	Connect the hardware with IBM Cloud and API Integration	10	Medium	Arunprasanth E Jaikrishnan V
Sprint-2	Software	USN-4	Application development for project	10	High	Jayendra S Kamalesh P
Sprint-3	Software	USN-5	Establishing Node-Red connection	10	Medium	Arunprasanth E Jaikrishnan V
Sprint-3	Software	USN-6	Connecting application with Node Red and further application development	10	High	Jayendra S Kamalesh P
Sprint-4	Hardware	USN-7	Testing developed application and working model of hardware	20	High	Arunprasanth E Jaikrishnan V Jayendra S Kamalesh P

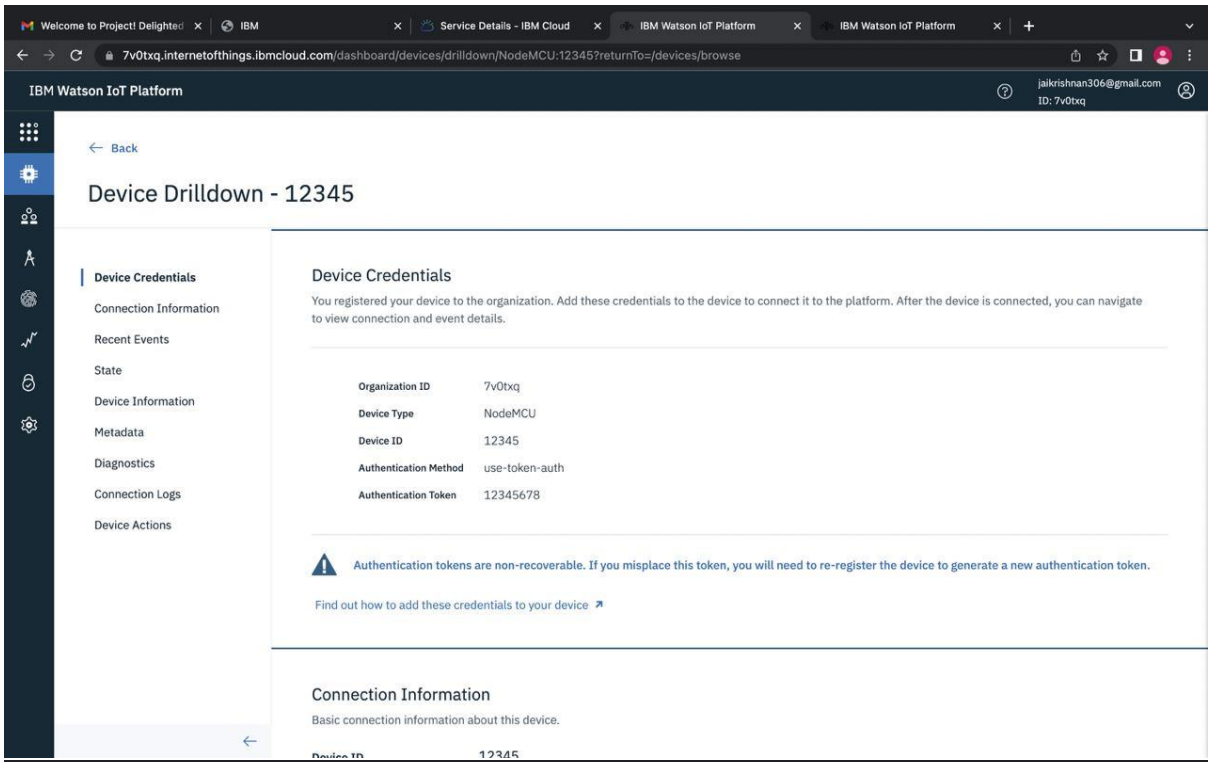
6.3 Reports from JIRA:

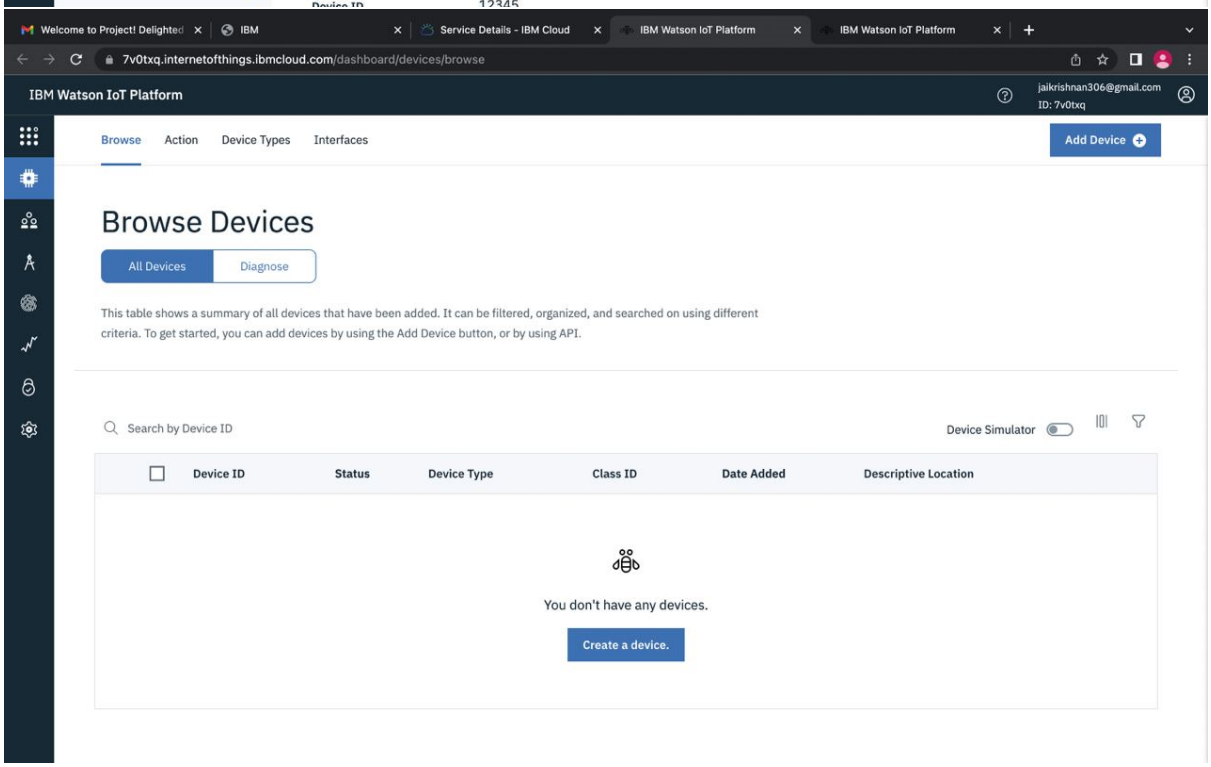


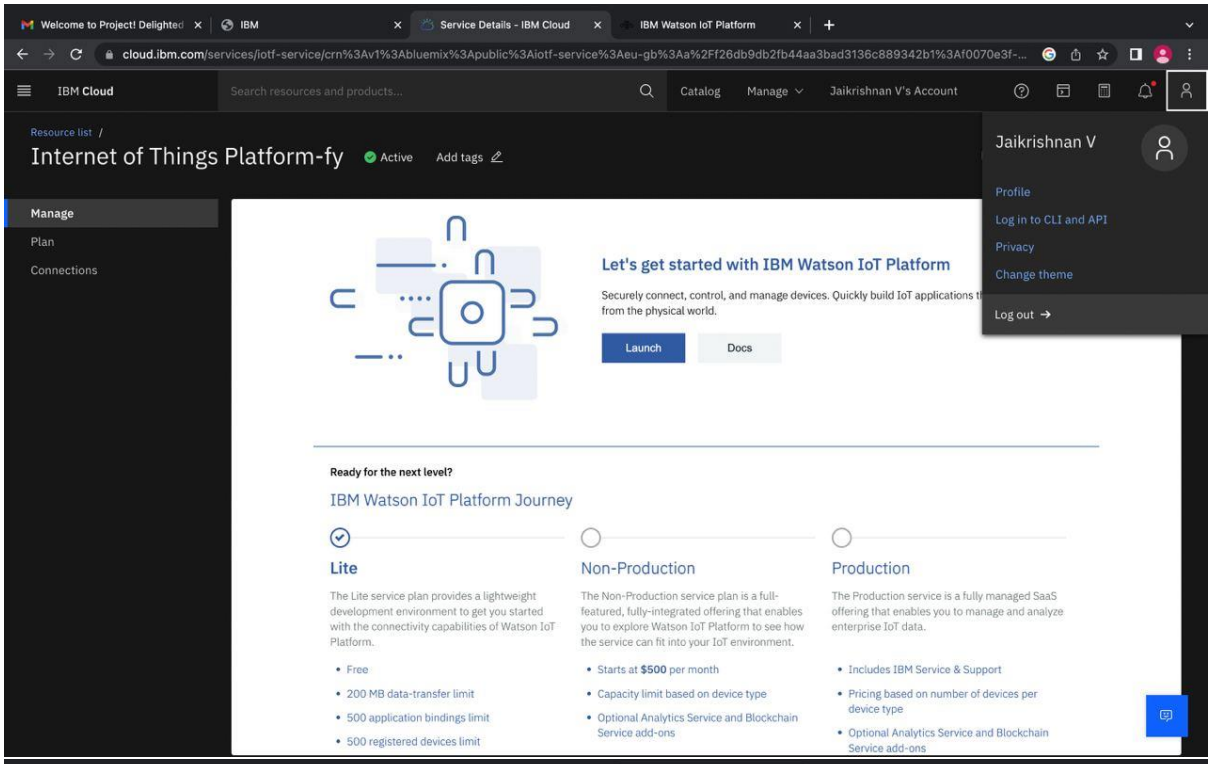
CODING AND SOLUTIONING

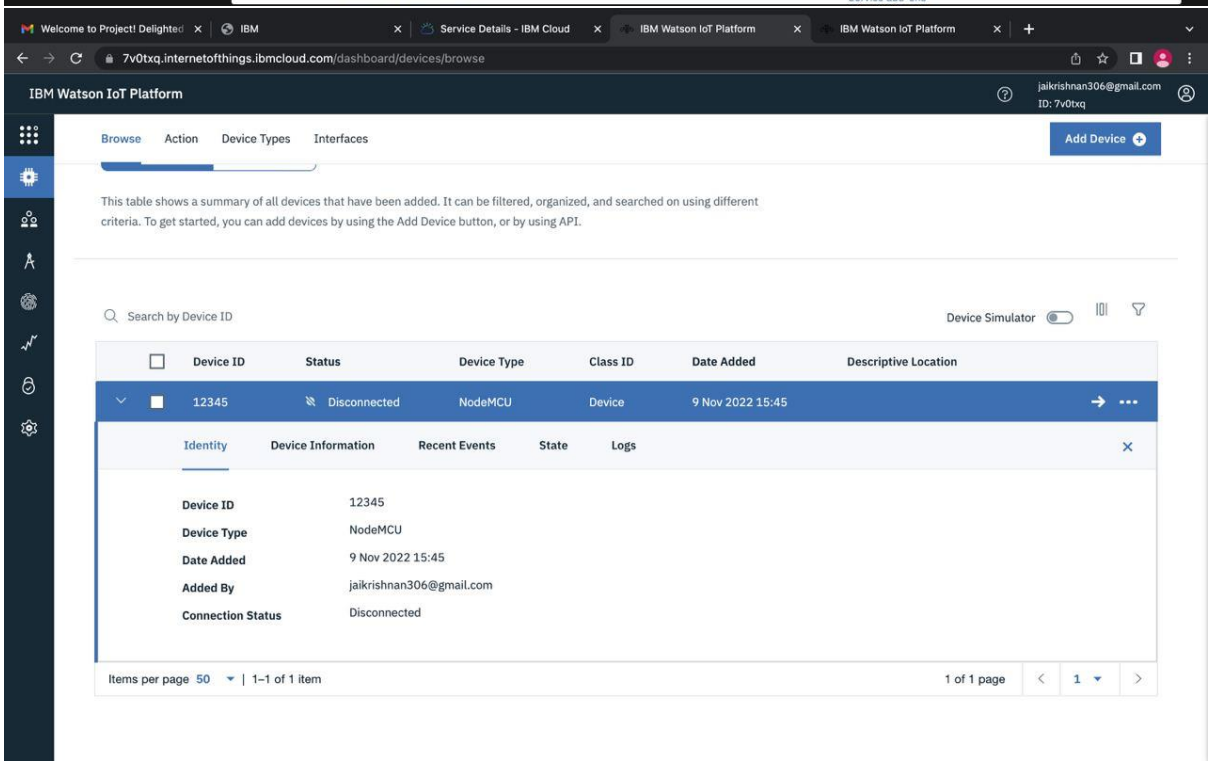
7.1 Feature 1:

Sprint 1 & Sprint 2



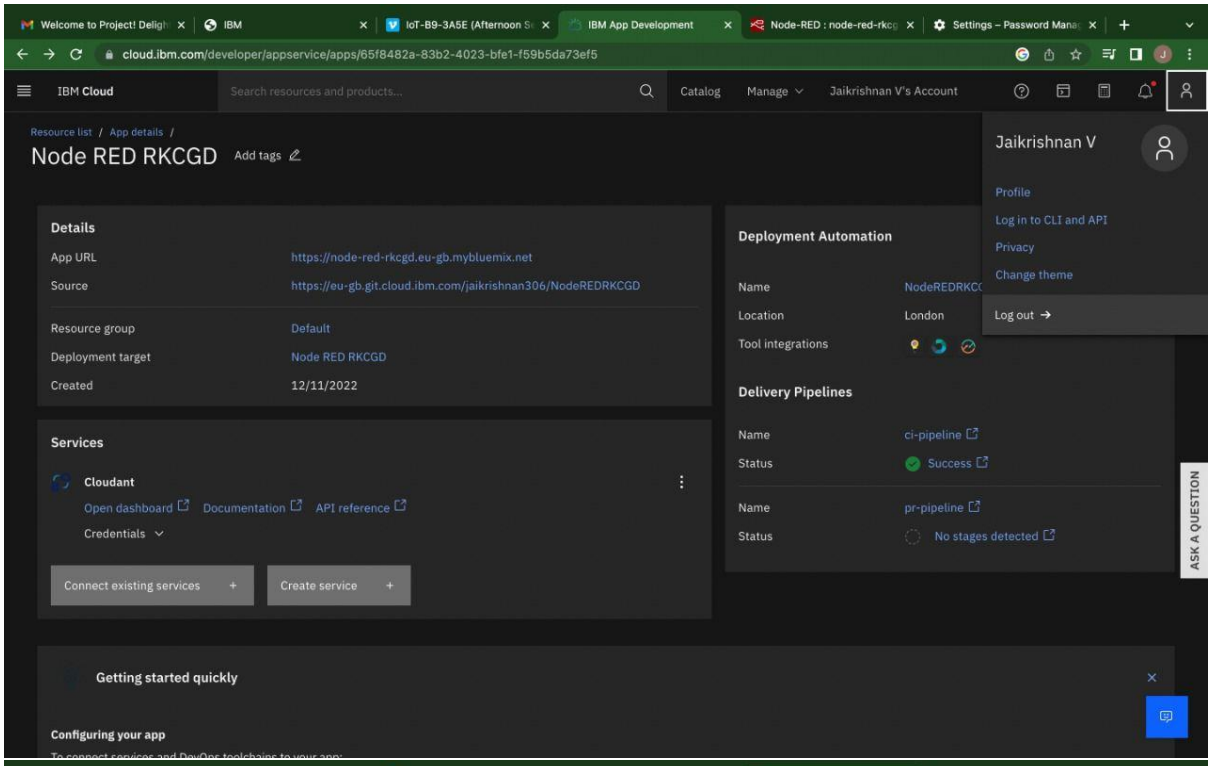


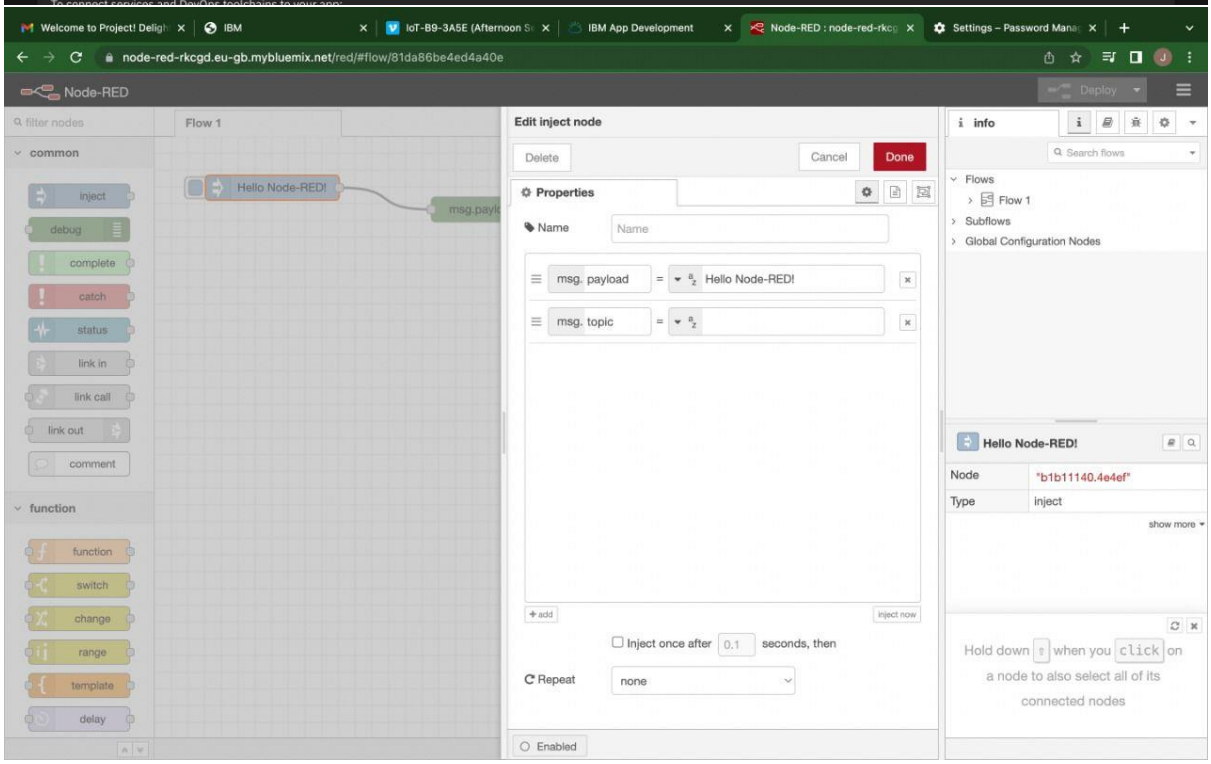

Welcome to Project! Delighted
IBM
Service Details - IBM Cloud
IBM Watson IoT Platform
cloud.ibm.com/services/iotf-service/crn%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-gb%3Aa%2Ff26db9db2fb44aa3bad3136c889342b1%3Af0070e3f-...
IBM Cloud
Search resources and products...
Catalog
Manage
Jaikrishnan V's Account
Resource list /
Internet of Things Platform-fy
Active
Add tags
Manage
Plan
Connections
Jaikrishnan V
Profile
Log in to CLI and API
Privacy
Change theme
Log out
Let's get started with IBM Watson IoT Platform
Launch
Docs
Ready for the next level?
IBM Watson IoT Platform Journey
Lite
The Lite service plan provides a lightweight development environment to get you started with the connectivity capabilities of Watson IoT Platform.
Free
200 MB data-transfer limit
500 application bindings limit
500 registered devices limit
Non-Production
The Non-Production service plan is a full-featured, fully-integrated offering that enables you to explore Watson IoT Platform to see how the service can fit into your IoT environment.
Starts at $500 per month
Capacity limit based on device type
Optional Analytics Service and Blockchain Service add-ons
Production
The Production service is a fully managed SaaS offering that enables you to manage and analyze enterprise IoT data.
Includes IBM Service & Support
Pricing based on number of devices per device type
Optional Analytics Service and Blockchain Service add-ons

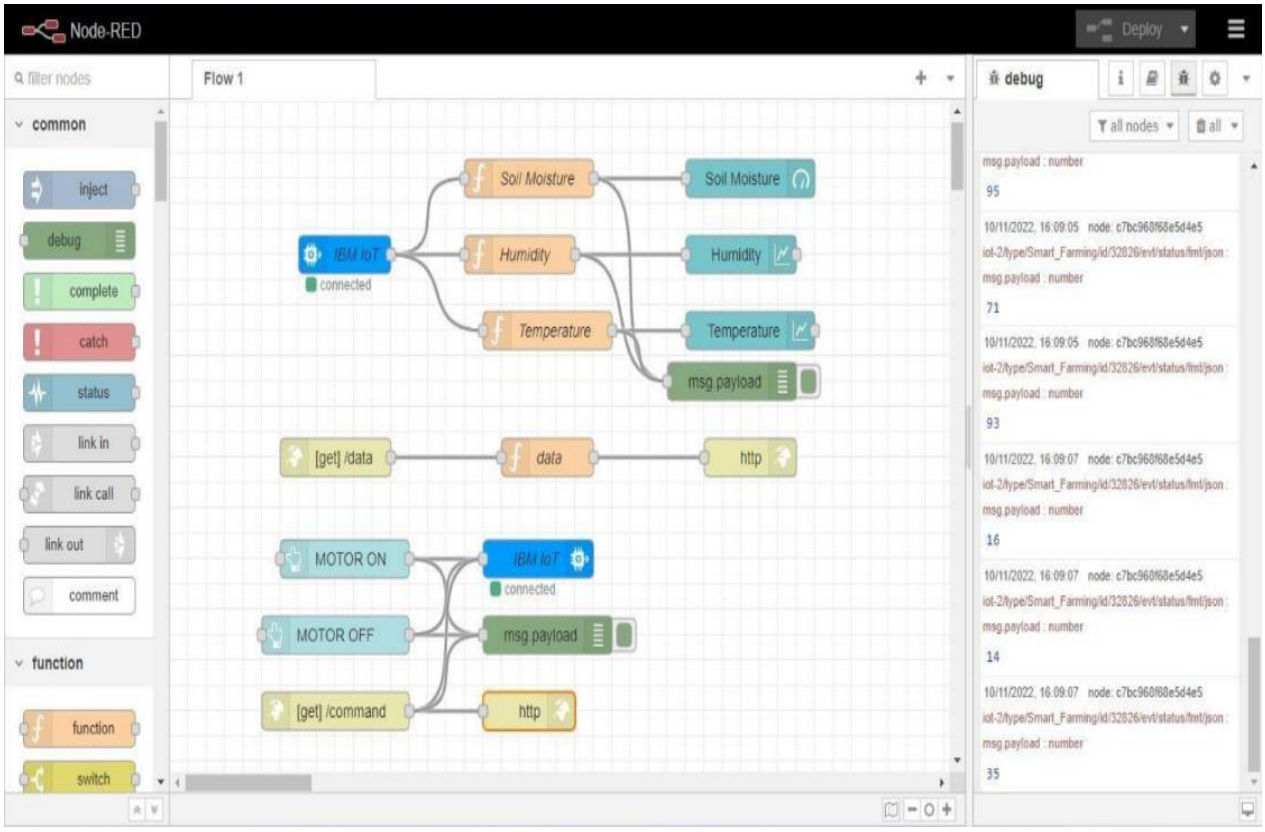

Welcome to Project! Delighted
IBM
Service Details - IBM Cloud
IBM Watson IoT Platform
IBM Watson IoT Platform
7v0txq.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
jaikrishnan306@gmail.com
ID: 7v0txq
Browse
Action
Device Types
Interfaces
Add Device
This table shows a summary 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.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
12345
Disconnected
NodeMCU
Device
9 Nov 2022 15:45
Identity
Device Information
Recent Events
State
Logs
Device ID
12345
Device Type
NodeMCU
Date Added
9 Nov 2022 15:45
Added By
jaikrishnan306@gmail.com
Connection Status
Disconnected
Items per page 50
1–1 of 1 item
1 of 1 page
1

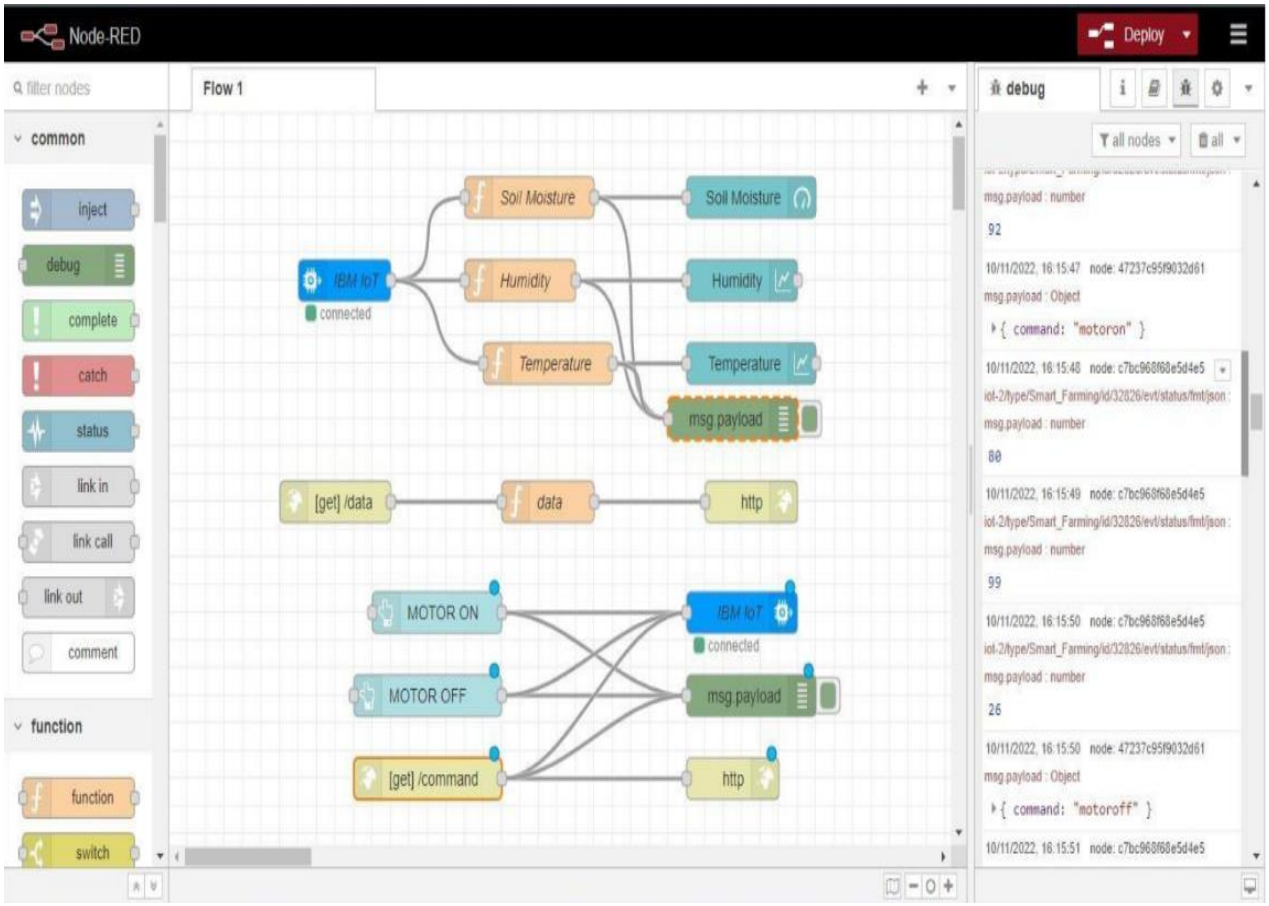
7.2 Feature 2:

NODE-RED



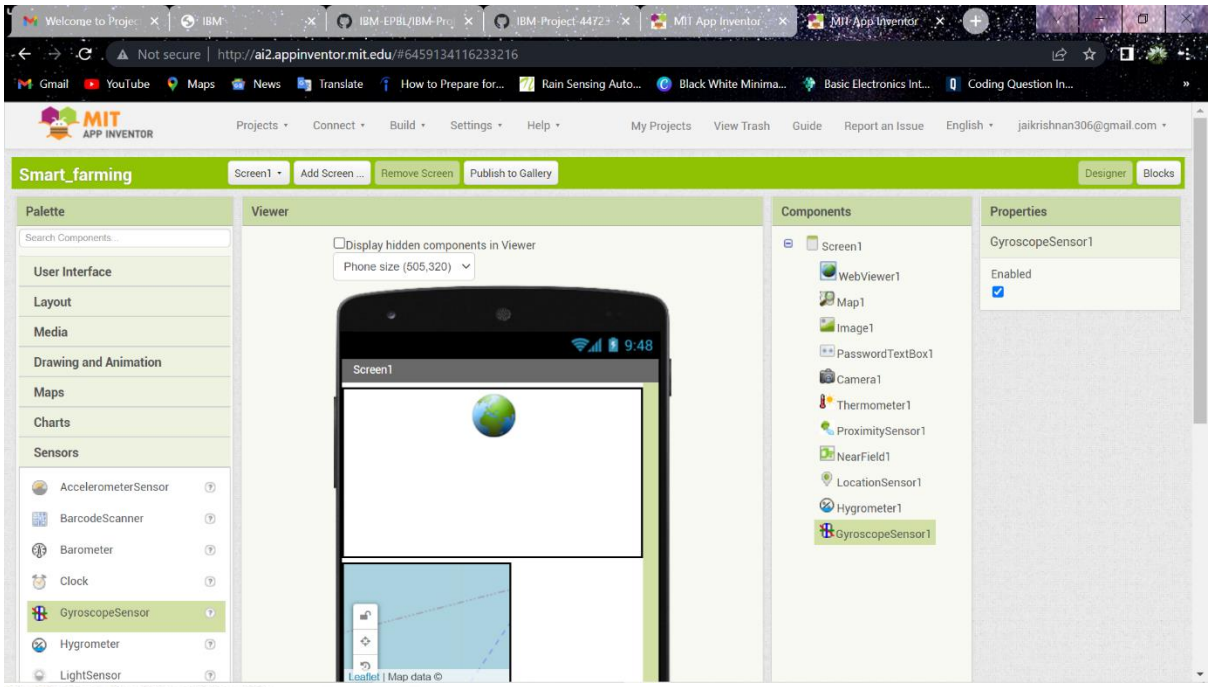



Node-RED
Deploy
filter nodes
Flow 1
debug
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
Soil Moisture
Soil Moisture
IBM IoT
connected
Humidity
Humidity
Temperature
Temperature
msg.payload
[get] /data
data
http
MOTOR ON
IBM IoT
connected
MOTOR OFF
msg.payload
[get] /command
http
all nodes
all
msg.payload : number
95
10/11/2022, 16:09:05 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
71
10/11/2022, 16:09:05 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
93
10/11/2022, 16:09:07 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
16
10/11/2022, 16:09:07 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
14
10/11/2022, 16:09:07 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
35

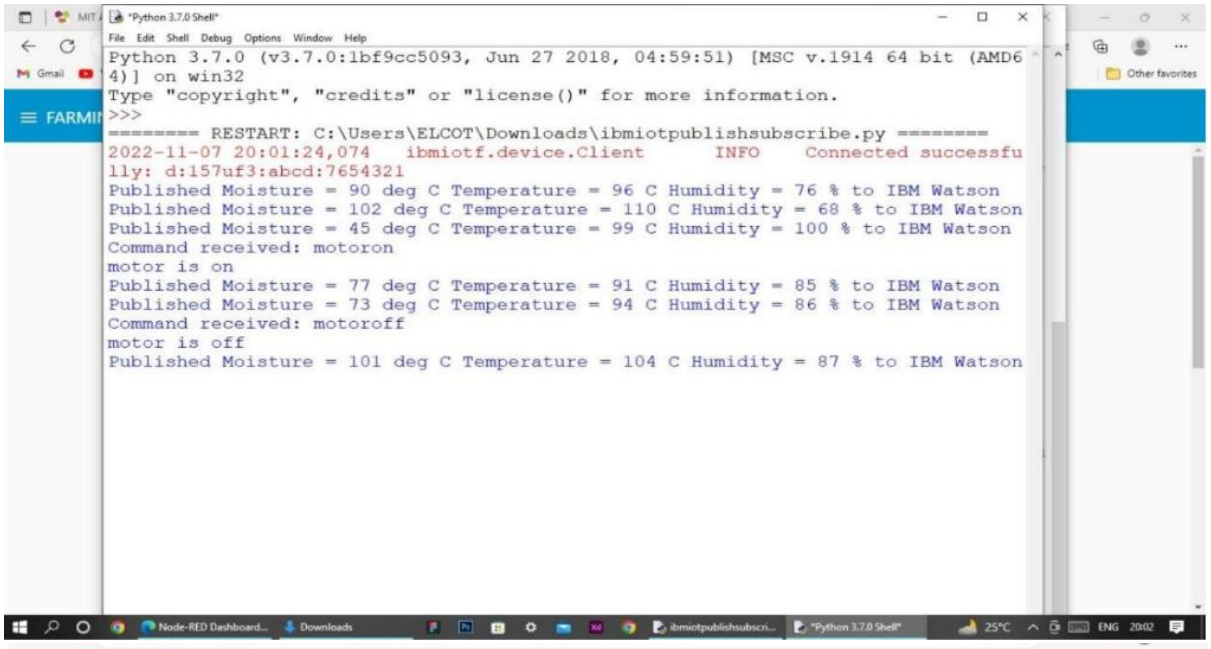

Node-RED
Deploy
filter nodes
Flow 1
debug
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
Soil Moisture
Soil Moisture
IBM IoT
connected
Humidity
Humidity
Temperature
Temperature
msg.payload
[get] /data
data
http
MOTOR ON
IBM IoT
connected
MOTOR OFF
msg.payload
[get] /command
http
all nodes
all
msg.payload : number
92
10/11/2022, 16:15:47 node: 47237c95f9032d61
msg.payload : Object
{ command: "motoron" }
10/11/2022, 16:15:48 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
80
10/11/2022, 16:15:49 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
99
10/11/2022, 16:15:50 node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json : msg.payload : number
26
10/11/2022, 16:15:50 node: 47237c95f9032d61
msg.payload : Object
{ command: "motoroff" }
10/11/2022, 16:15:51 node: c7bc968f68e5d4e5

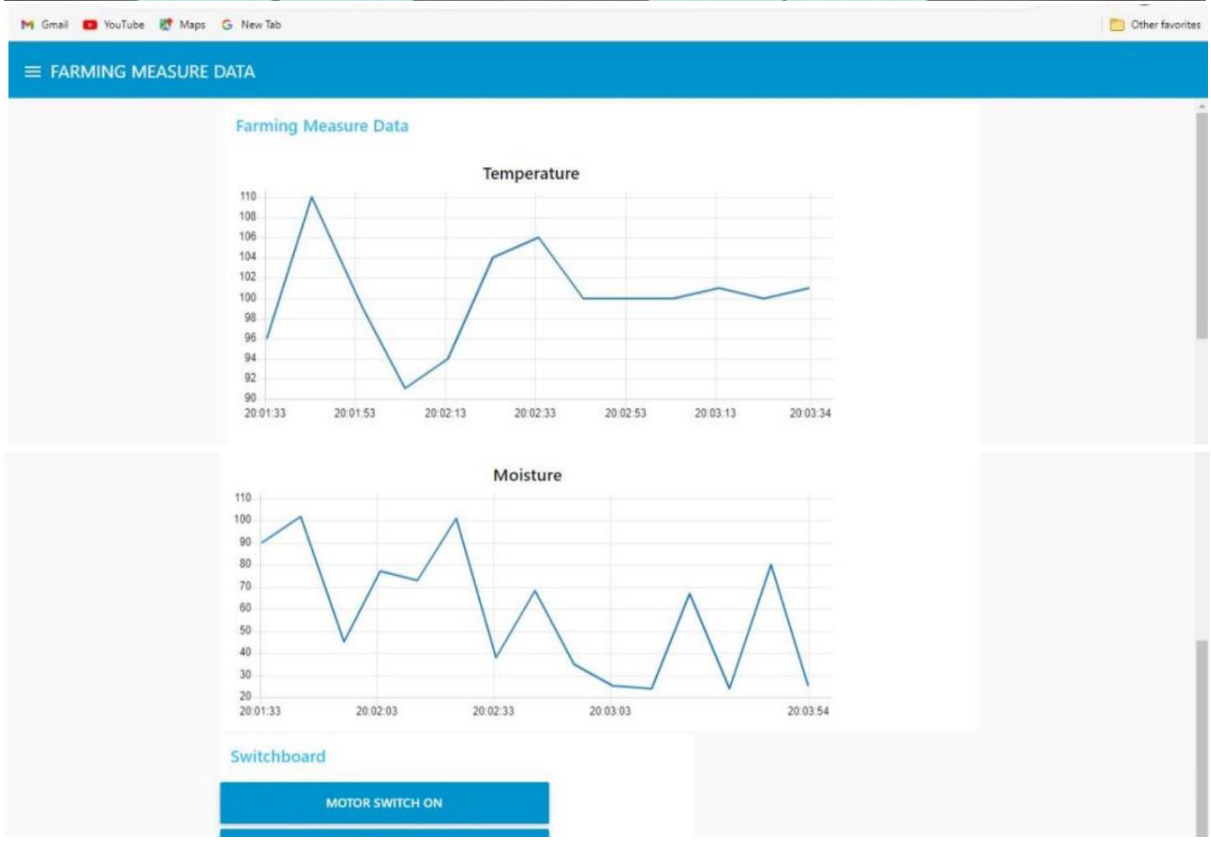
TESTING

8.1 Test cases:



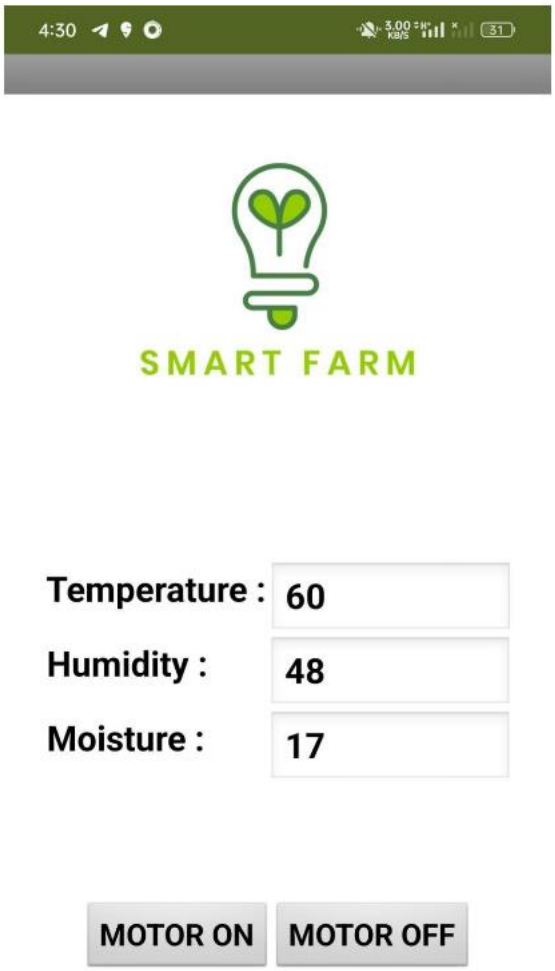

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

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

# Initialize GPIO
def myCommandCallback(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": authMe
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................................
```


"Python 3.7.0 Shell"
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] 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 successfully: 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


Gmail
YouTube
Maps
New Tab
Other favorites
FARMING MEASURE DATA
Farming Measure Data
Temperature
110
108
106
104
102
100
98
96
94
92
90
20:01:33
20:01:53
20:02:13
20:02:33
20:02:53
20:03:13
20:03:34
Moisture
110
100
90
80
70
60
50
40
30
20
20:01:33
20:02:03
20:02:33
20:03:03
20:03:54
Switchboard
MOTOR SWITCH ON

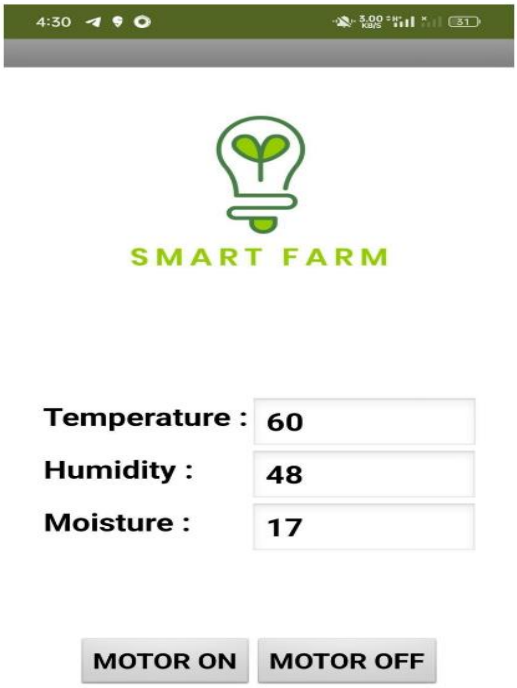
8.2 User acceptance testing:

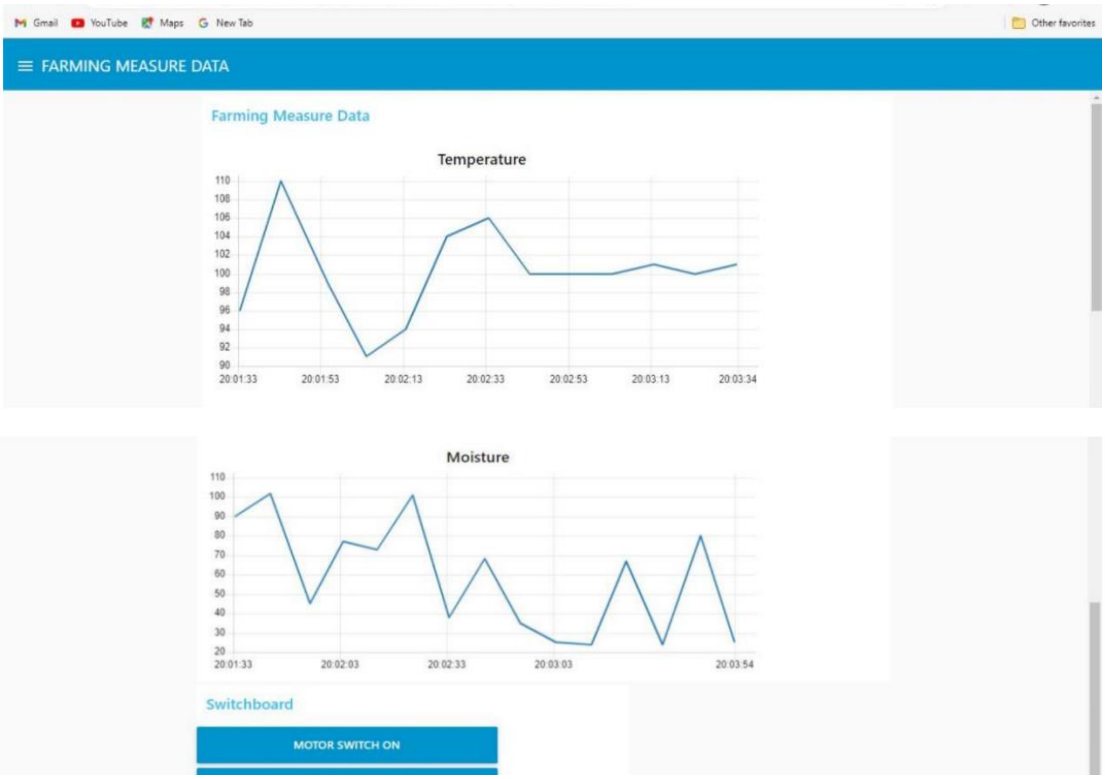
User acceptance testing is a type of software testing in which the different units, modules or components of a software application are tested as a combined entity. However, these modules may be coded by different programmers. Integration Testing is a type of software testing, which is performed on software to determine the flow between two or more modules by combining them. This testing makes sure that the interactions between different components of the software is completed smoothly without any complication. The purpose of the integration testing is to expose faults in the interaction between integrated units. Once all the modules have been unit tested, integration testing is performed**.** It is the final testing done before the marketing plan.



RESULTS

9.1 Performance metrics:





ADVANTAGES AND DISADVANTADES

Advantages:

- As it is a mobile friendly application one can access all the metrics in one touch.

- It has clean User interface so that user have smooth control over the application.

- The consumption of electric power is less as compared to other application.

- The moisture level and the temperature levels are monitored at regular intervals.

- It can run on all android versions.

- The application requires less memory and storage space.

Disadvantages:

- When the network connectivity is poor the performance of the application will be affected.

- As it is platform dependent it cannot run on all devices.

- The application will produce inaccurate values when there is a fault or any change in API.

- The user should be more aware on the results produced.

CONCLUSION

In this work, we successfully developed a system that can help in an automated irrigation system by analyzing the moisture level of the ground. The smart irrigation system proves to be useful system as it automates and regulates the watering without any manual intervention. The primary applications for this project are for farmers and gardeners who do not have enough time to water crops/plants. The cloud computing devices are used at the end of the system that can create a whole computing system from sensors to tools that observe data from the agricultural field. The farmers are facing major problems in watering their agriculture fields. So that the farmers can water their plant with smart techniques. The objective of this project is to implement an IoT system in order to help farmers to control and monitor their farms and that is done successfully.

FUTURE SCOPE

Agriculture domains encounters with many challenges starting from soil parameters, seed sowing, crop growth and its quality, weed handling, disease management till harvesting and storing crop. Artificial intelligence driven techniques along with other available tools and automation can address these challenges and proven the revolution in agriculture. Most popular AI application in agriculture is use of Robot and Drones, they perform almost all task like humans even at a faster rate with accuracy. From literature review it is clear that precision farming is probable by integrating sensors, cameras, data analytics, GPS and remote sensing. Image recognitions software's, IoT sensors can be used for disease recognition at primary stages and hence crop health can be supervised which increases superior quality production with minimum loss. Table 1 demonstrate the various applications in view of Smart Agriculture for improved evolution as well as superiority. Still there are several challenges associated with AI and IoT application in smart agriculture which is the promising future to be explored area for researchers. Some of major challenges are: - Awareness issues, Hardware implementation challenges, Cost of software and hardware, Network management, Energy management, Privacy issues, Security challenges, Interoperability of systems with the induction of Computer vision, Deep learning, big data also agriculture sector has influenced a lot. Researchers can integrate IoT sensors along with smart systems and computational optimization algorithms to overcome the limitations/shortcomings. Smart Agriculture has a budding potential towards productivity, precision, optimization, adaptive resource management and intelligent food traceability. It will contribute to environment also in terms of efficient use of water, prevent disease contamination and precise use of pesticides.

APPENDIX

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "7v0txq",
        "typeId": "NodeMCU",
        "deviceId":"12345"
    },
    "auth": {
        "token": "12345678"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']

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

while True:
    soil=random.randint(0,100)
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
```

```
myData={'soil_moisture': soil, 'temperature':temp, 'humidity':hum}
print("Published data Successfully: %s", myData)
client.commandCallback = myCommandCallback
time.sleep(2)
client.disconnect()
```