IoT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

TEAM ID : PNT2022TMID46939

ROLL NUMBER	NAME
821919104023	Y.SAFREEN BANU
821919104018	D.PRIYA DHARSHINI
821919104028	M.SUGANYA
821919104020	L.RABIYA

PROJECT REPORT

1. INTRODUCTION

1.1 Project Overview :

Agriculture is the primary source of livelihood for about 58% of India's population. Agriculture is an important sector of Indian economy and it contributes about 17% to the GDP. Agriculture also provides employment to 60% of population. But due to various climatic changes and animal intervention the farmers are facing major losses. There are many traditional methods that are being used by the farmers like scare crows, electric fences, etc, In some areas farmers uses smoke to prevent their farmland, the burn elephant dung or other

materials that create heavy smoke. In some areas people also uses fish or garlic natural emulsion, castor oil to repels the animals. But these are not very effective to save the farms from animals. Hence, we have designed this affordable system to surveillance and to protect the farm effectively. Animals like wild boars, buffaloes, cows, elephant, monkeys, birds, etc. damages the crop a lot which results in loss of production and so of farmer. It is very difficult for a farmer to keep an eye on the field every time. This system is designed to surveillance the field 24*7 which is not possible for a human being and diverts the animals without harming them. The system uses raspberry pi, PIR sensor to detect animal, camera module to look on animal, GSM module to send alert message to farmer, and a buzzer to divert the animals. This system ensures the safety of farm and decreases the loss of farmers.

1.2 PURPOSE :

The purpose of Smart Crop Protection System is to Secure or Protect the farm from the theft in the farm or main purpose of this project is to alert the farmer as well as fear the animals with getting harm to animals. An intelligent crop protection system helps the farmers in protecting the crop from the animals and birds which destroy the crop. This system also helps farmers to monitor the soil moisture levels in the field and also the temperature and humidity values near the field. The motors and sprinklers in the field can be controlled using the mobile application.

2. LITERATURE SURVEY

2.1 Existing Problem :

Agriculture is a field which forms the basis of our economy. Yet it faces a lot of problems in terms of availability of resources, Irrigation, increasing rate of Pesticides, Climatic disasters, insects which ruin the crops and makes huge loss in this sector. In agriculture water is needed for the crops for their growth. If the soil gets dry it is necessary to supply water. But sometime if the farmer doesn't visit the field it is not possible to know the condition of soil. Sometimes over supply of water or less supply of water affects the growth of crops. Sometimes if the weather or temperature changes suddenly it is necessary to take certain actions. The crop protection is majorly dependent on the moisture content of the soil , temperature and humidity of the surrounding environment.

2.2 References :

- Damini Kalra, Praveen Kumar, K. Singh, Apurva Soni “Sensor Based Crop Protection System with IoT monitored Automatic Irrigation” 2nd International conference on Advances in Computing, Communication Control and Networking, 2020.
- S. Giordano, Ilias Nektarios Seitanidis, Mike Oluwatayo Ojo, Davide Adami “IoT solutions for crop protection against wild animal attacks” 2018 IEEE International Conference on Environmental Engineering (EE), March 2018
- Mr. P. Venkatesh Rao, Mr.Ch Siva Rama Krishna, Mr M Samba Siva Reddy “A Smart Crop Protection against Animal Attack”. International Journal of Scientific Research and Review ISSN: 2279 Vol. 8 Issue 05, 2019

2.3 Problem Statement Definition :

User Story Number	User Story / Task	Acceptance Criteria	Priority
USN-1	As an user, I can monitor crop production.	I can monitor the system.	High
USN-2	As an user, I will inform the farmer to ptotect the crops.	I can inform the farmer.	Medium
USN-3	As an user, I can notice the levels of crop in the field.	I can notice the crop level.	Low
USN-4	As an co-user, I can send the alert message to the farmers.	I can alert farmers.	High
USN-5	As a farmer, I will follow the route to the crop which can avoid are detect animal intrusion.	I can reach the crops.	High
USN-6	As an crop protector, I Can Protect the crops.	I can protect the crops.	Medium
USN-7	As a farmer, I can supervise the process and ensure the health of farm	I can manage all these process going good.	High

USN-8	As a crop yielder , I can yield more crops .	I can yield more crops.	Medium
USN-9	As a crop monitor, I check the quality of IoT device's quality	I can check the IoT device.	Medium

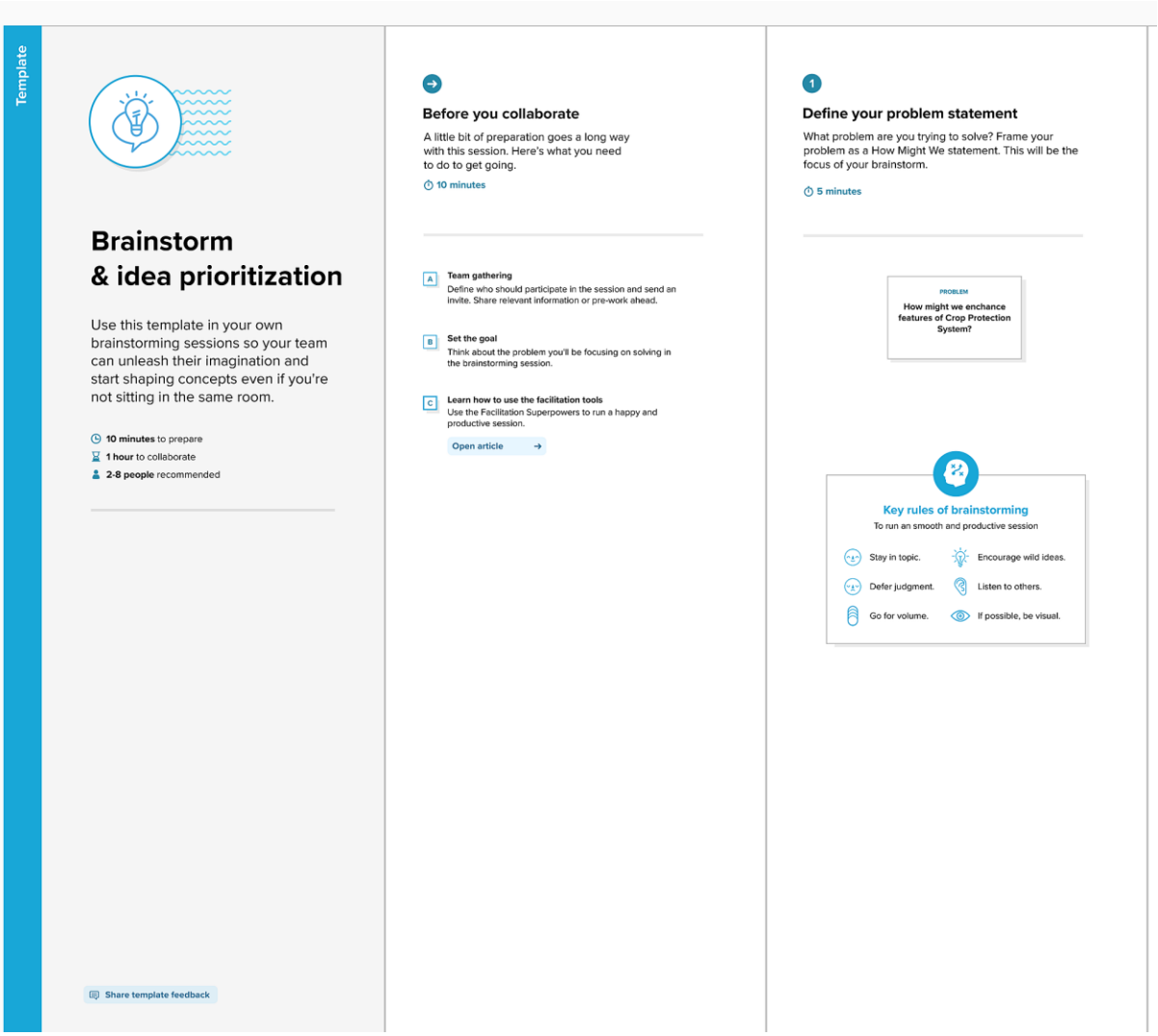
3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas :

An Empathy map is a simple, easy to digest visual that captures knowledge about a user's Behaviour's and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solutions requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consists things from the user's perspective along with his or her goals and challenges.

3.2 Ideation & Brainstorming :

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



Step-2: Brainstorm, Idea Listing and Grouping:

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!

Convesing biodiversity

Crop protection

loweing the food cost

Nutrients in the earth

Agricultural Marketing

Increase the quality

Increase Profitability

Irrigation

optimizing and resorces

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

Mechanical Crops Protection

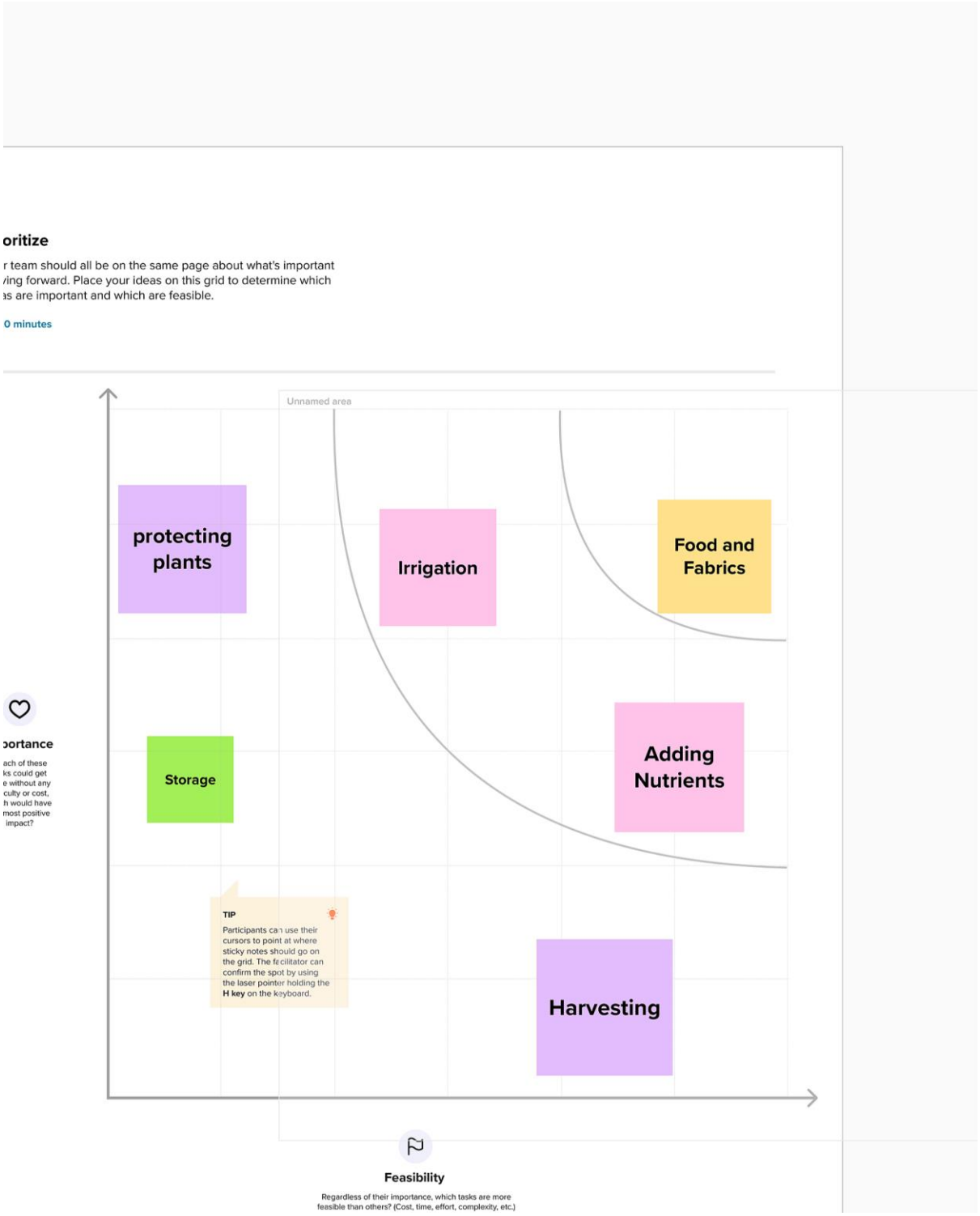
Reducing yields

Maintaining Crop diversity

Organic Pesticides

InterCropping

Step-4 : Idea Prioritization:



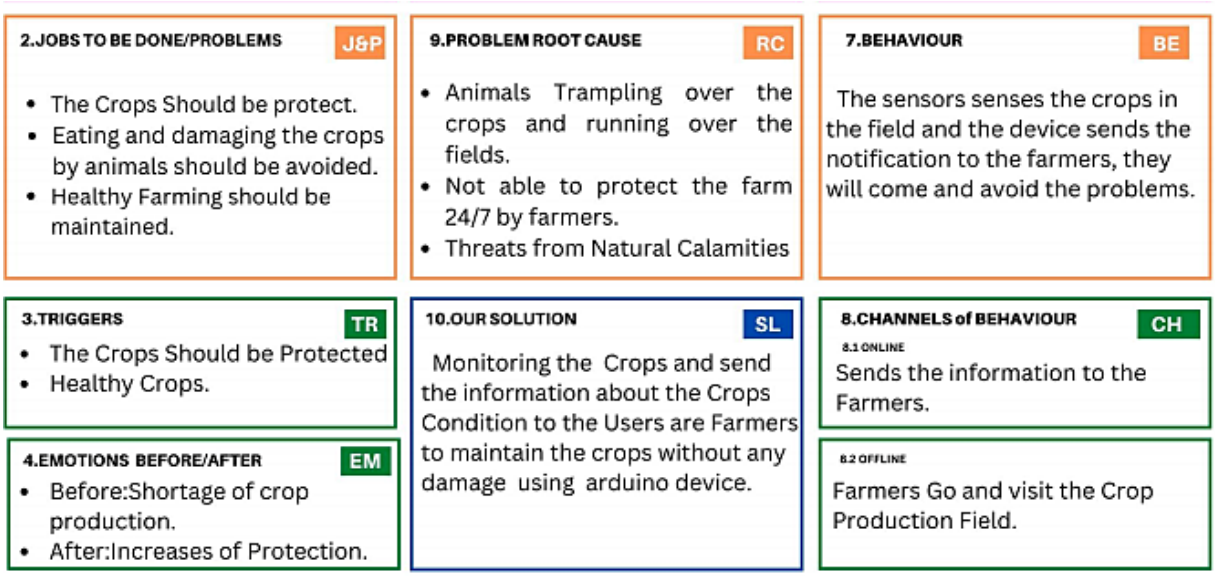
3.3 Proposed Solution :

S No:	Parameter	Description
1.	Problem Statement (Problem to be solved)	Crops in farms are many times damaged by animals like buffaloes, cows, goats, birds and wild elephants. This causes major losses for the farmers. Farmers cannot stay on the field for 24 hours and protect it.
2.	Idea / Solution description	An animal detection system has been designed to detect the presence of animals and it offers a warning and divert the animal without any harm. The designed system will continuously check for any animal to enter the field. IR sensors and ultrasonic sensor are used in this project to detect animal movement and to give a signal to the controller. Further the animals are being diverted by generating sound and signals, and this signal is being transmitted to GSM and instantly give farmers warning, so the farmers will be aware of the difficulty and available to the spot just in case the animals do not show off by the alarm. The complete safety of crops was ensured by this system from animals thus protecting the farmer's loss.
3.	Novelty / Uniqueness	The Problem of Crop Protection System is solved by an animal detection system has been designed to detect the presence of animals and it offers a warning and divert the animal without any harm. The designed system will continuously check for any animal to enter the field. IR sensors and ultrasonic sensor are used

		in this project to detect animal movement and to give a signal to the controller. Further the animals are being diverted by generating sound and signals, and this signal is being transmitted to GSM and instantly give farmers warning, so the farmers will be aware of the difficulty.
4.	Social Impact / Customer Satisfaction	The project is very effective in protecting crops in the field. Rather than using Manual Methods, a monitoring system is used to ensure the crop without any damaged. It has been tested and verified properly to ensure all the different parts work together for a smooth function of the whole system. In most of the cities globally poses a challenge to protect and maintenance of the crops.
5.	Buisness Model (Revenue Model)	• The cost to develop the project is about the sensors used here. • The Arduino device and Cloud platform used here play a vital role in cost. • If any damage occurs to the device during monitoring we need to fix it. • The contribution of the farmers is necessary to make the project succeed in the market.
6.	Scalability of the Solution	The project design is a part of the implication that can be used to improve the Crop Protection. All the technical aspects have been thoroughly designed keeping all the constraints in mind. The project resolves around whether the project will be able to meet the future needs of the users. This project-based on IoT gives users the freedom of changing Hardware

		as well as software specifications as per the arising need. IoT based projects are already designed while keeping future demands in mind and in a rising economy like India where the concept of smart crop protection is new the demand for our project will keep on increasing.

3.4 Problem Solution Fit :



4.REQUIREMENT ANALYSIS

4.1 Functional Requirement :

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Fitting IoT Device in the farm.	The IoT device needs to be fixed in the farm with water proof safety. The IoT device consists of PIR Sensor, Flame Sensor. To

		send data to the cloud GSM is used.
FR-2	Connecting to the cloud.	The device should configure to connect to the cloud. The data of sensors need to be received and processed.
FR-3	Predictions for Crops Destroy.	In this 24x7 Monitoring System is designed for Monitoring the Crops, PIR Sensors is used to sense movement of People, Animals Node Red is used to access the location of the Agriculture farm. LCD display Animal Information when animal is detected, Flame Sensor detects the Fire and via blink application send given Alert Message to farmer. Whenever there is an attack by animals to Crops in Agriculture then Alert Message is sent farm the device to farmers and the cloud. In term farmers can protect the Crop.
FR-4	Real time Monitoring.	This System works in real time to detect the animals in the fields. The System enables the farmer to have a real time view of his fields from any place via internet and even provides manual buzzer controls if the need arises to use sound the buzzer if needed. The System also provides a history of the events taking place in the fields, in the form of images and textual log records.
FR-5	Requires no human supervision.	This System requires almost no human supervision, except for the task of switching the system on and off. The System is capable of turning the buzzers on automatically and warding off the animals thus protecting the fields from any damage.
FR-6	Routes to Crop Protection.	The Crops are protected by insects, animals,

		etc through the use of deliberate sensors connected in the farm field; sensors estimate the motion of inspects and animals nearer to the crop and sent the signal to the Arduino Uno microcontroller for calculation of distance and all.

4.2 Non Functional Requirement :

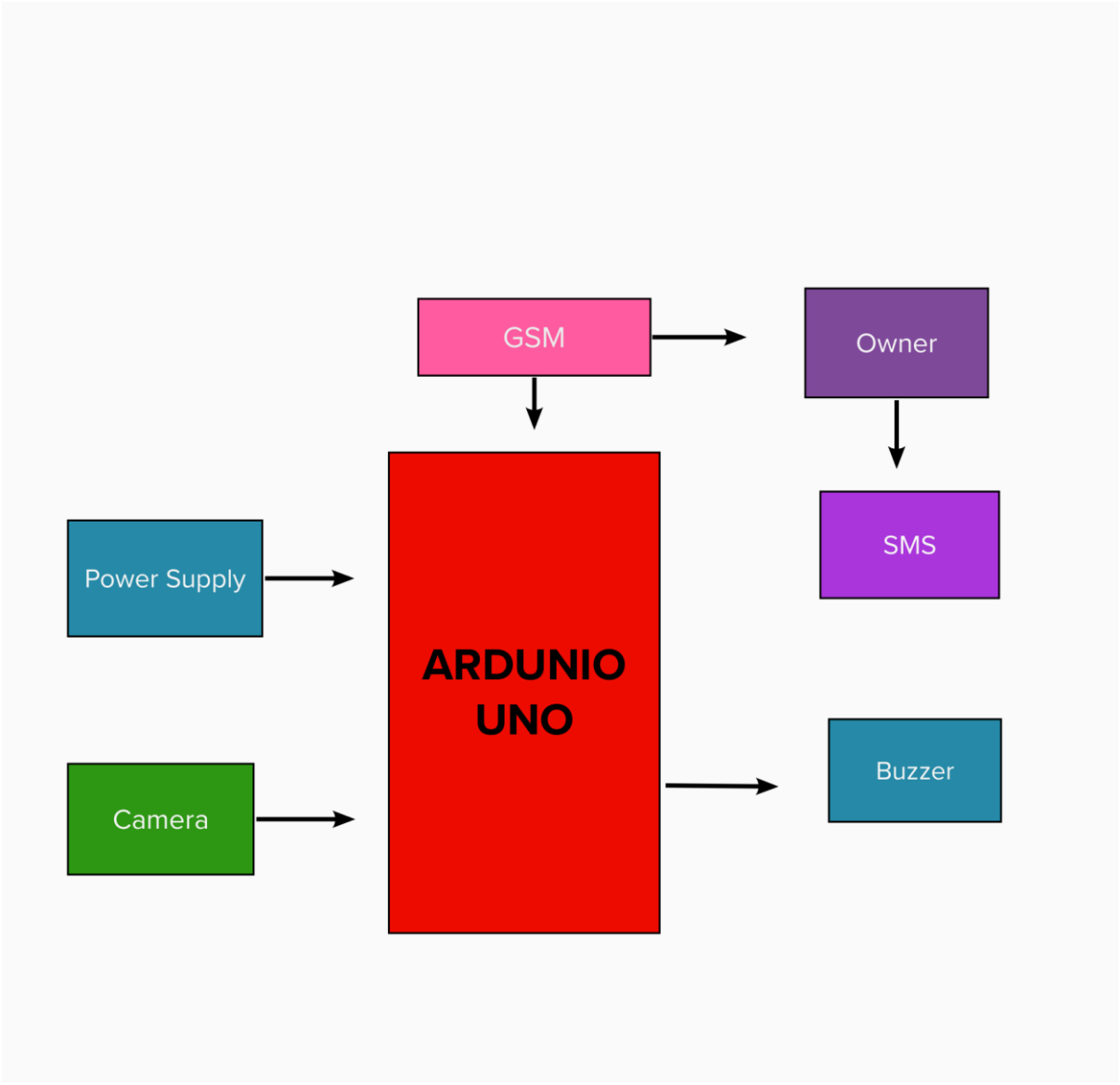
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	IoT solution for Smart crop protection offer advanced machine learning techniques in the system. Due to this the system can be trained to detect different types of animals. This feature of the system makes it highly adaptable to the local sites of deployment. Thus the system is not limited to the detection of only particular type of animals. This make it suitable for different areas of our country.
NFR-2	**Security**	Building and deploying IoT-based smart crop protection in rural areas can be complex time consuming and resource intensive process. Many departments not have resources to support such a project internally.
NFR-3	**Reliability**	One of the difficult operational problems of farmers are facing is the Intrusion or Ravaged of Animals in forms in recent years, Due to Environmental concerns and no of cost most of the farmers have been forced for accessing this crops, and examining then Cost Effectiveness.

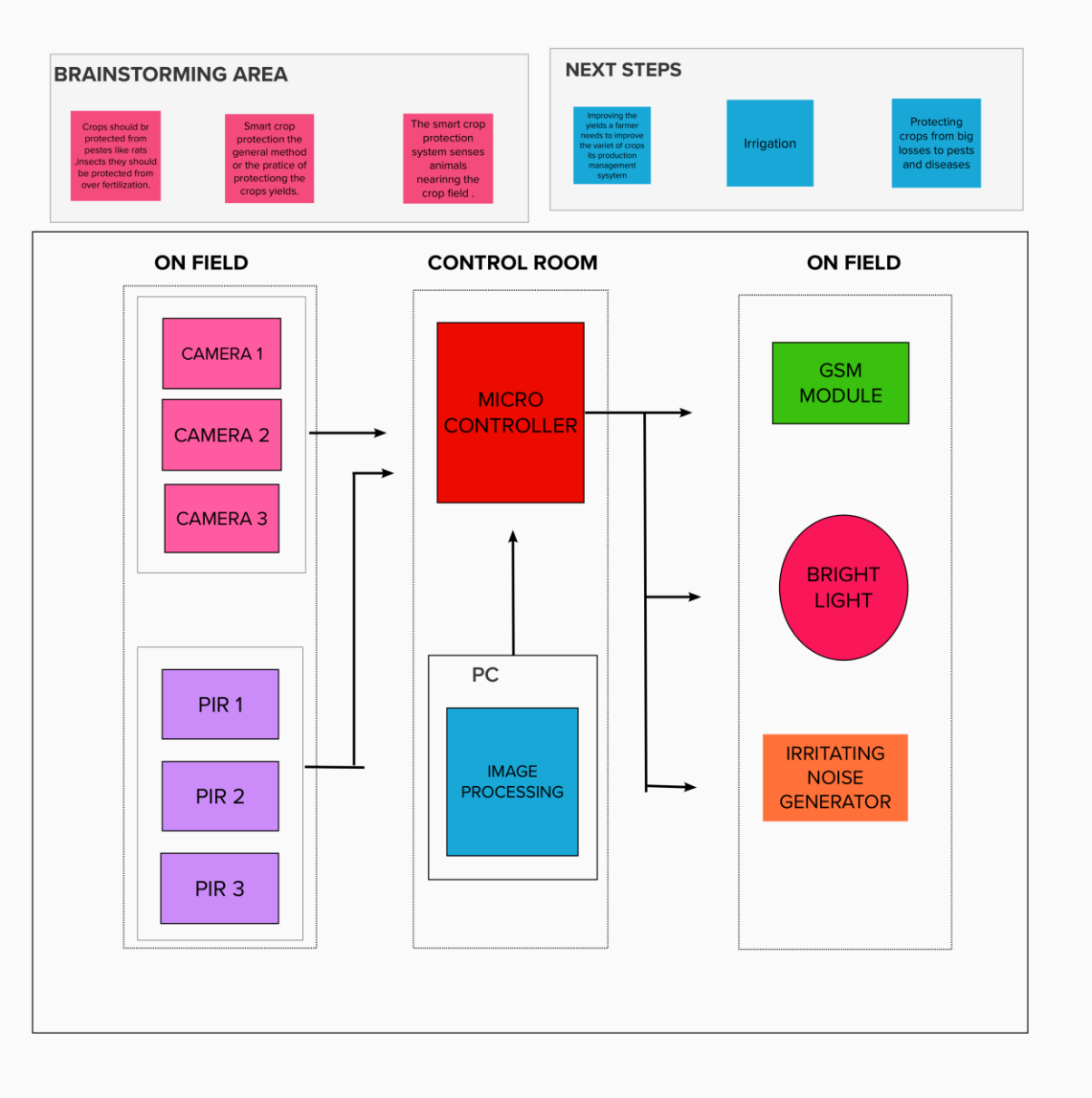
NFR-4	**Performance**	An integrated Aurdino program is developed to microcontroller, display system and communication system. Whenever there is any detection of intrusion in the field the users will get to know about it in the farm of assigned values.
NFR-5	**Availability**	Another purpose of this project is to make the crop protection system as cheap as possible .Ensures complete safety of crops from animals thus protecting the farmer loss.
NFR-6	**Scalability**	The Farm diversity about 80% of its Intrusion, or Ravaged and hopes to go "Better Crop Yields" by the end of 2021. Thus leads to their Economy well being

5.PROJECT DESIGN

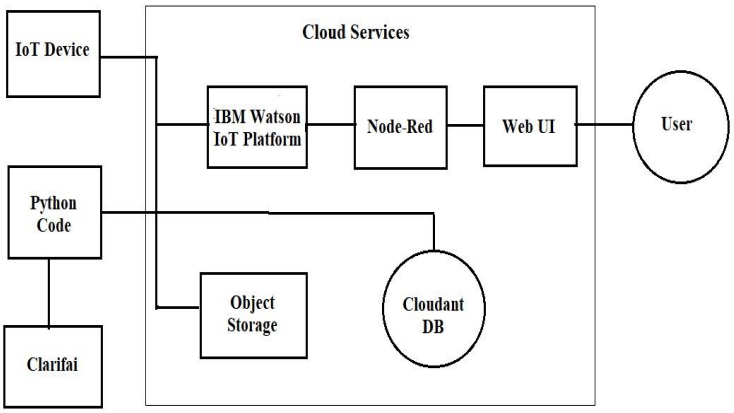
5.1 Data Flow Diagram

- This is an Arduino Uno based framework utilizing microcontroller.
- This framework utilizes a PIR sensor to identify intruders close to the field and additional to it a smoke sensor to identify.
- When animal approach close to the PIR Sensor, it detects the movement.
- After getting the initial input signal from the warm body, it is passed for further processing, then it will be passed one to the microcontroller.
  Then the system will be activated, immediately the buzzer goes on and simultaneouly it ends on SMS to the owner.
- Microcontroller (Arduino Uno) is used for reading the inputs from PIR, Soil Moisture Sensor and Flame sensor.
- The GSM module is used for sending SMS to farmer.


GSM
Owner
SMS
Power Supply
ARDUNIO
UNO
Camera
Buzzer



5.2 Solution & Technical Architecture :



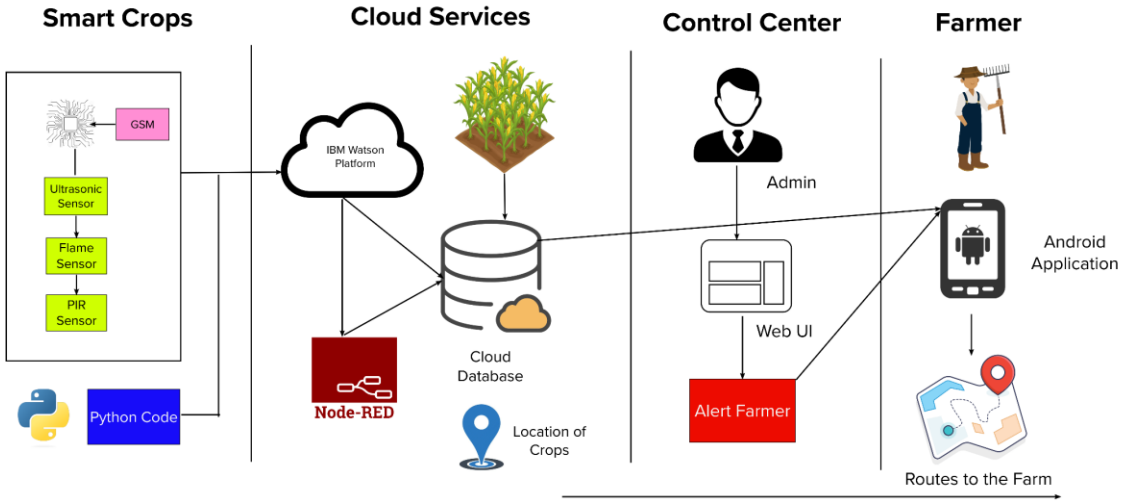


Table-1 : Components & Technologies: Technology

S.No	Component	Description	Technology
1.	Arduino Uno	The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller.	Arduino programming itself is done in **C++.**
2.	Application Logic-1	Logic for Ultrasonic sensor data.	C++/Python
3.	Application	Logic for Flame sensor data.	C++/Python

	Logic-2		
4.	Application Logic-3	Logic for a PIR sensor data	C++/Python
5.	GSM	The Arduino GSM shield allows an Arduino board to connect to the internet, send and receive SMS, and make voice calls using the GSM library.	C++/Python
6.	Cloud Sever	Application deployment on Local System / Cloud	IBM Watson IoT Platform, Node Red
7.	Cloud Database	Database Service on Cloud	IBM Watson IoT platform, Cloudant DB
8.	User Interface	How user interacts with application to alert the Farmer.	HTML, CSS, JavaScript , Python etc.
9.	External API-1	Purpose of External API used in the application to locate the crops.	Google Maps Geolocation API

Table-2: Application Characteristics:

S.No	Component	Description	Technology
1.	Open-Source Microcontroller	Arduino Uno is used to make the IoT device	C++/Python
2.	Security	Encryption/Decryption used for security purpose	GSM,Python
3.	Scalable Architecture	New features can be added	Node Red
4.	Availability	Web application can be accessed from anywhere	IBM Watson IoT Platform, HTML, CSS, JavaScript
5.	Performance	All Farmers can access the application at	Cloudant DB, IBM

		same time	Watson IoT Platform

5.3 User Stories :

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
User	Login	USN-1	As an user, I can monitor Crop production	I can monitor the system.	High	Sprint-4
		USN-2	As an user, I will inform the farmer to ptotect the crops.	I can inform the farmer.	Medium	Sprint-2
		USN-3	As an user, I can notice the levels of crop in the field.	I can notice the crop level.	Low	Sprint-2
User 2	Login	USN-4	As an co-user, I can send the alert message to the farmers	I can alert farmers.	**High**	Spirit-1
Farmer	Login	USN-5	As a farmer, I will follow the route to the crop which can avoid are detect animal intrusion	I can reach the crops.	High	Sprint-2

Crop Protector	Login	USN-6	As an crop protector, I can protect the crops.	I can protect the crop.	Medium	Sprint-2
Farmer	Login	USN-7	As a farmer, I can supervise the process and ensure the health of farm	I can manage all these process going good.	High	Spirit-1
Crop yielder	**Register**	USN-8	As a crop yielder , I can yield more crops .	I can register smart crop	Medium	Spirit-3
Crop Monitor		USN-9	As a crop monitor, I check the quality of IoT device's quality.	I can check the IoT device.	Medium	Spirit-3

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation :

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

S.No	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Monitoring	USN-1	The IoT device will monitor the Crops in the Field.	20	High	Safreen Banu.Y Priya Dharshini.D Suganya.M Rabiya.L
Sprint 1	Surveying	USN-2	As a Farmer I can survey the Crops level in the Field	20	Low	Safreen Banu.Y Priya Dharshini.D Suganya.M Rabiya.L
Sprint-2	Mapping	USN-3	As an admin, I can map the fields and providing data to farmers.	20	High	Safreen Banu.Y Priya Dharshini.D Suganya.M Rabiya.L

Sprint-3	Alert	USN-4	As an admin, I can map the fields and providing data to farmers.	20	High	Safreen Banu.Y Priya Dharshini.D Suganya.M Rabiya.L
Sprint-4	Location View	USN-5	As a Farmer, I will follow the route to the Crops in the field which can avoid animal intrusion.	20	Medium	Safreen Banu.Y Priya Dharshini.D Suganya.M Rabiya.L

6.2. Sprint Delivery Schedule:

Project Tracker, Velocity & Burndown Chart:

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

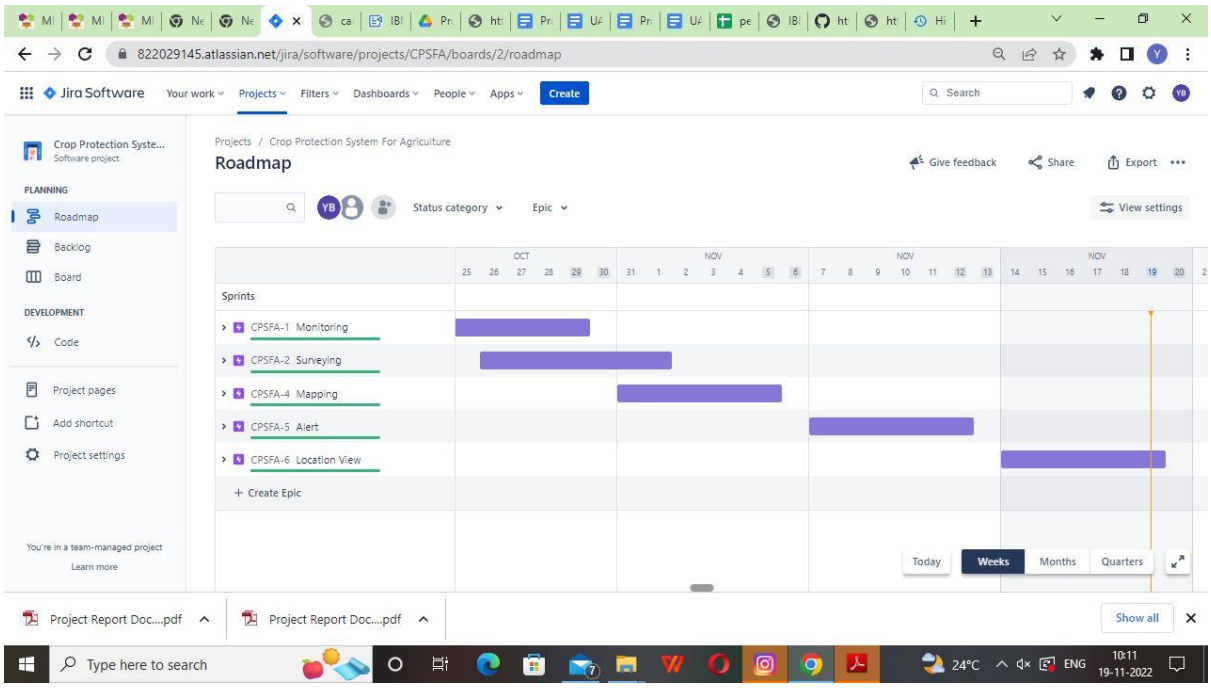
Velocity:

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

(AV) per iteration unit (story points per day)

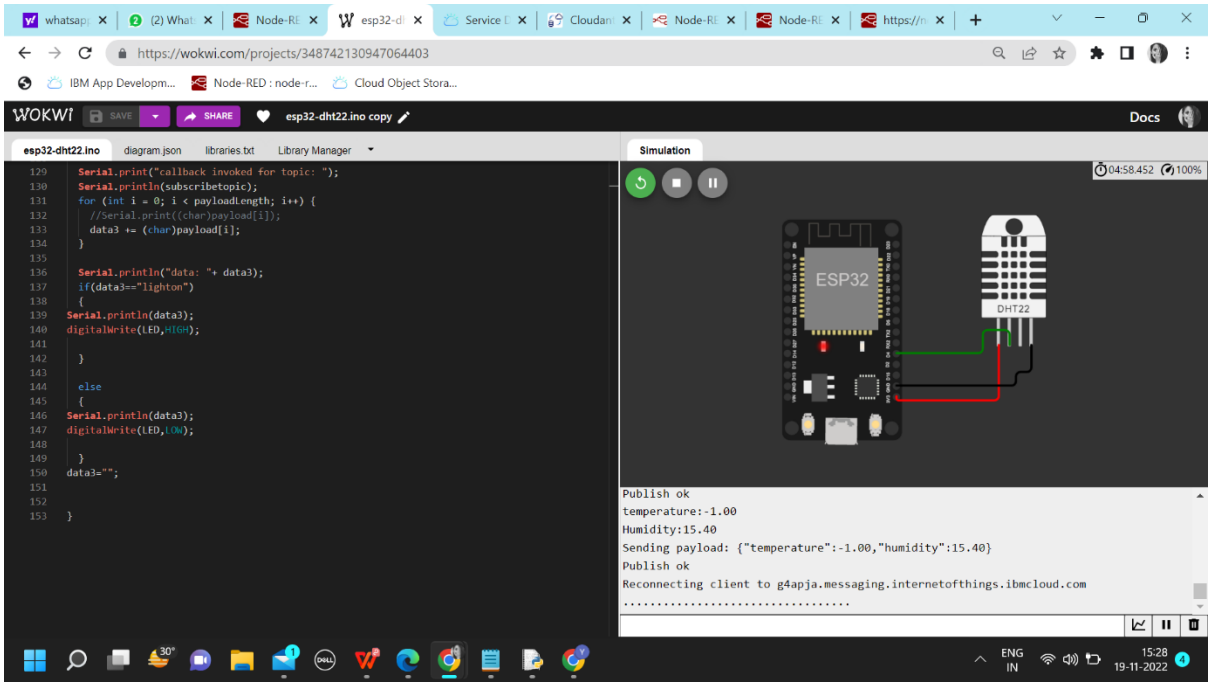
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.3 Reports from JIRA:

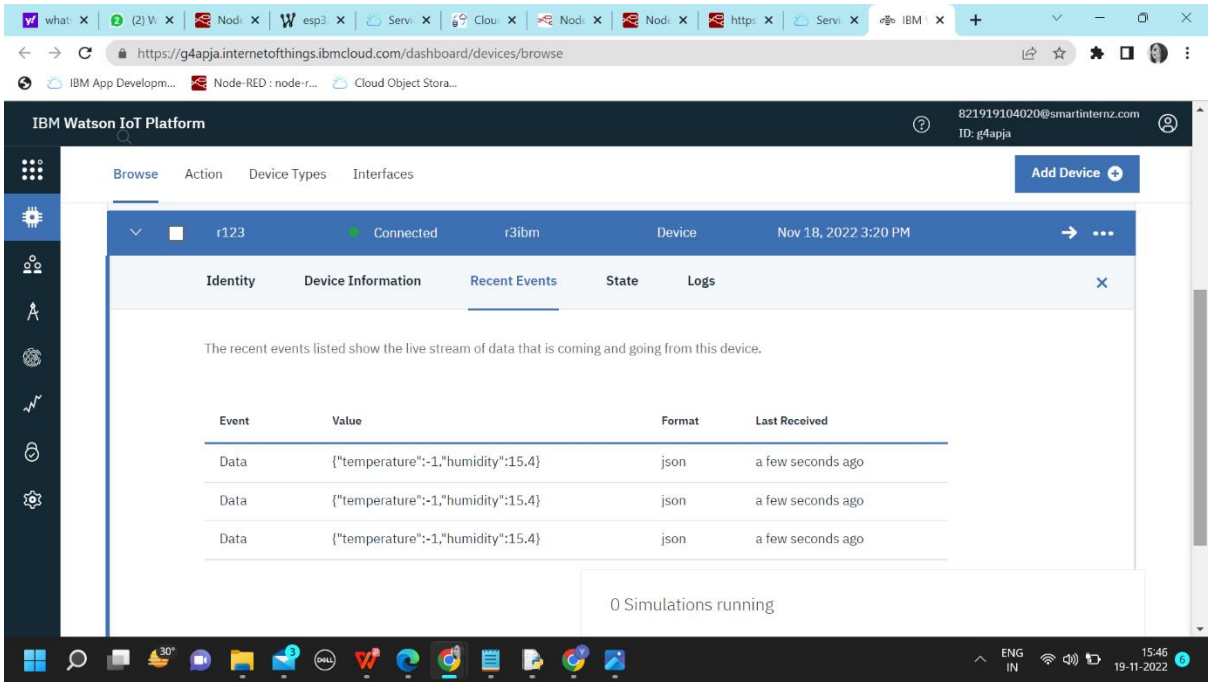


7. CODING & SOLUTIONING (Explain the features added in the project along with code)

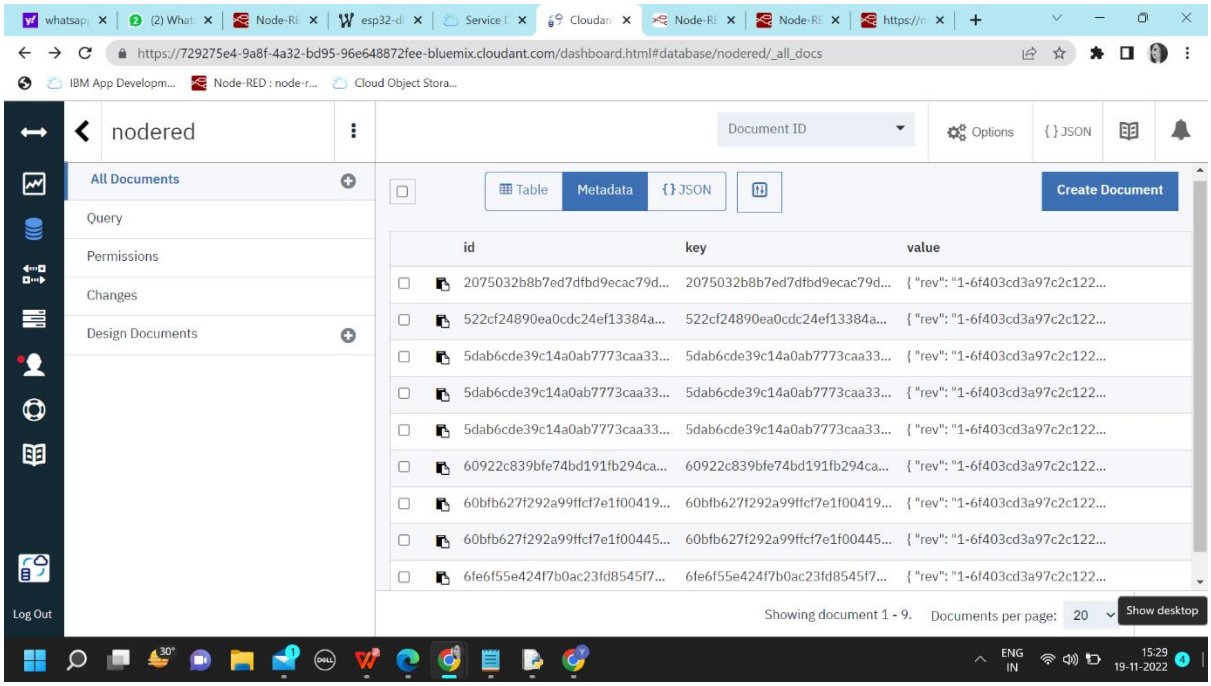
7.1.Wokwi:



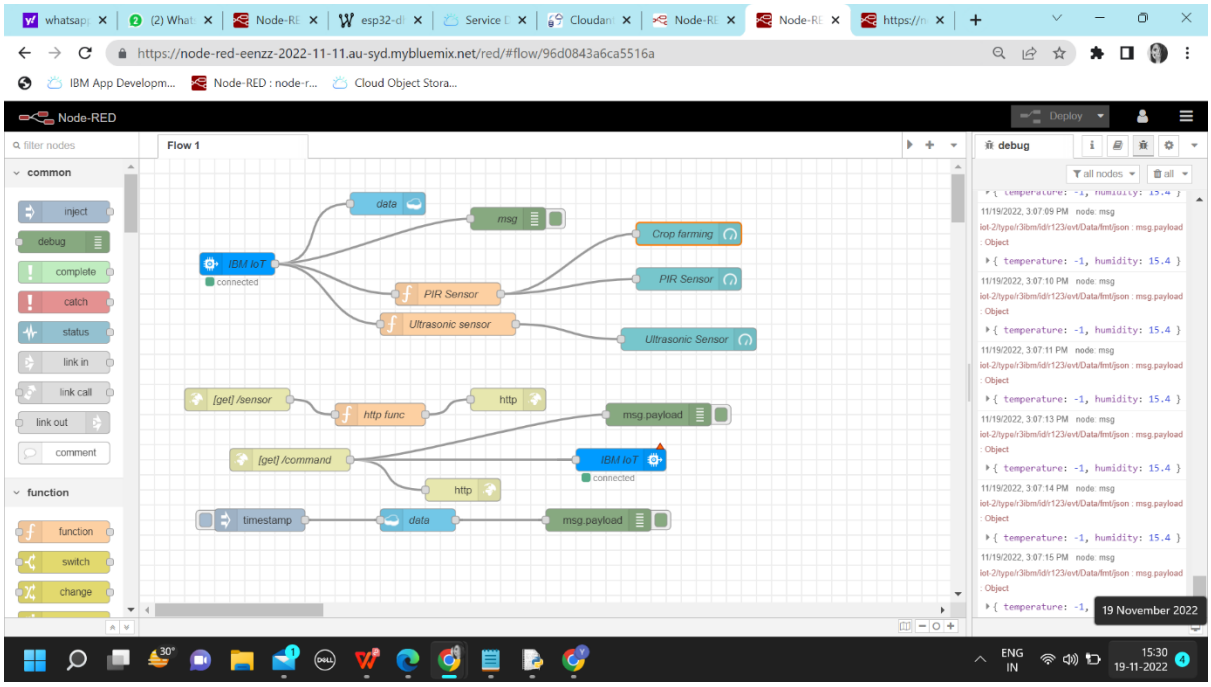
7.2.Watson:

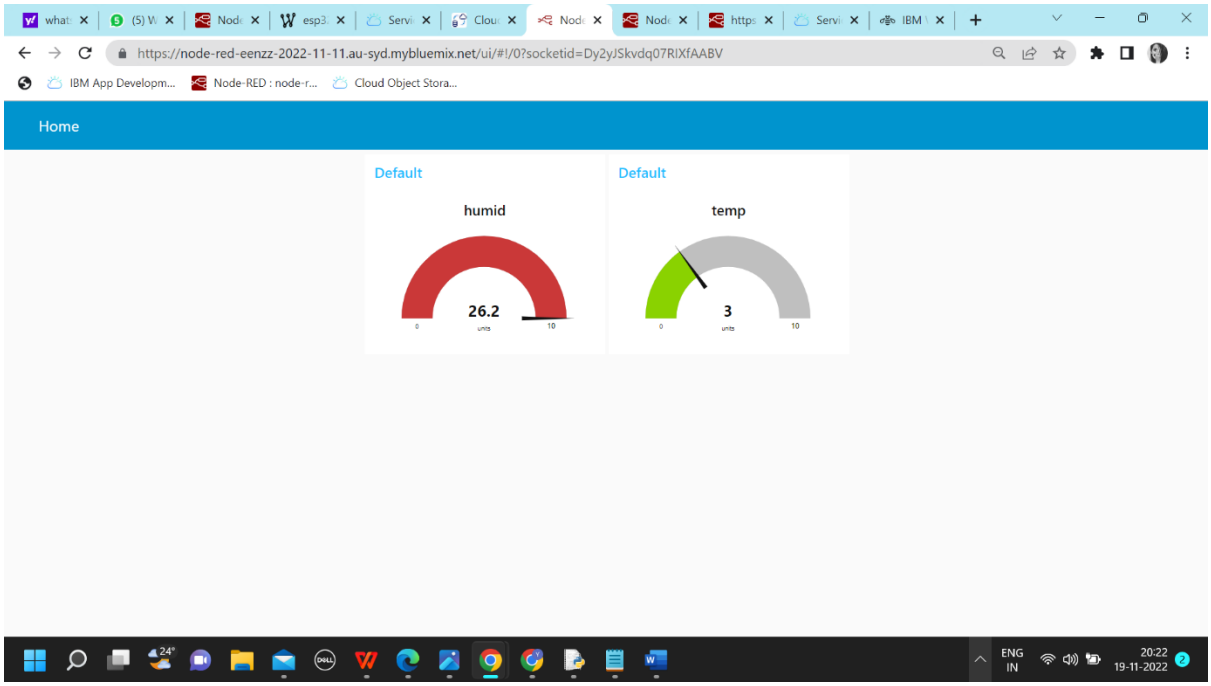


7.3.Cloudant:

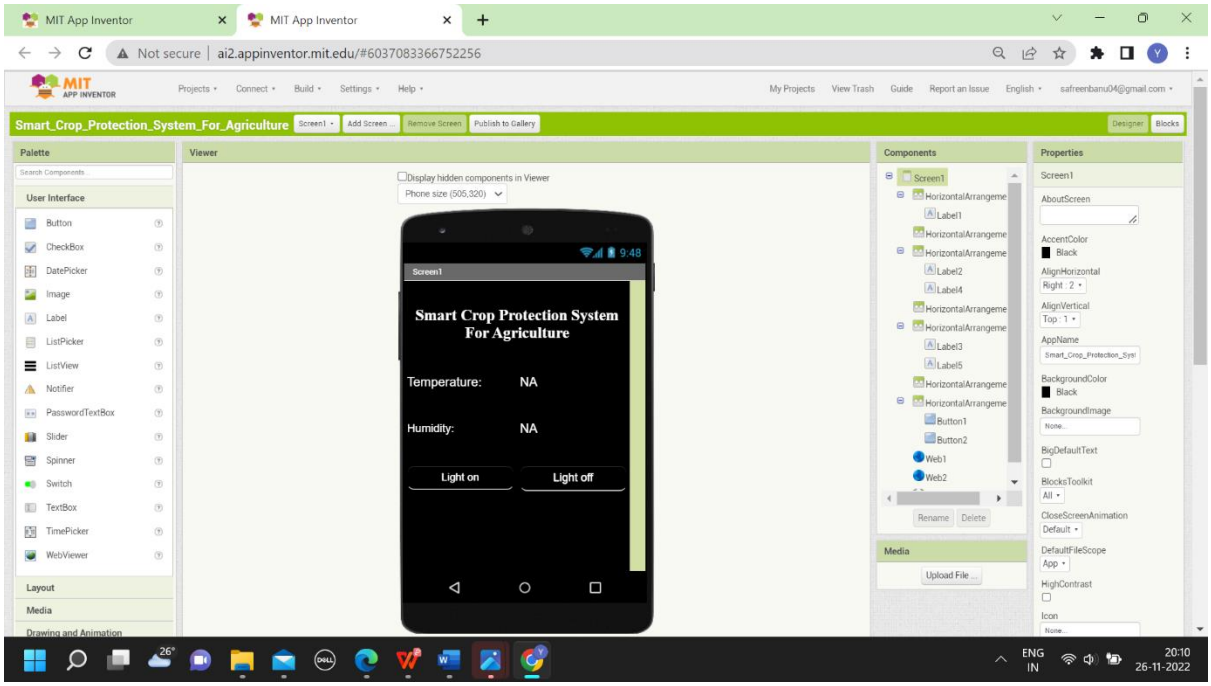

nodered
All Documents
Query
Permissions
Changes
Design Documents
Log Out
Document ID
Options
{ } JSON
Table
Metadata
{ } JSON
Create Document
id
key
value
Showing document 1 - 9.
Documents per page:
20
Show desktop

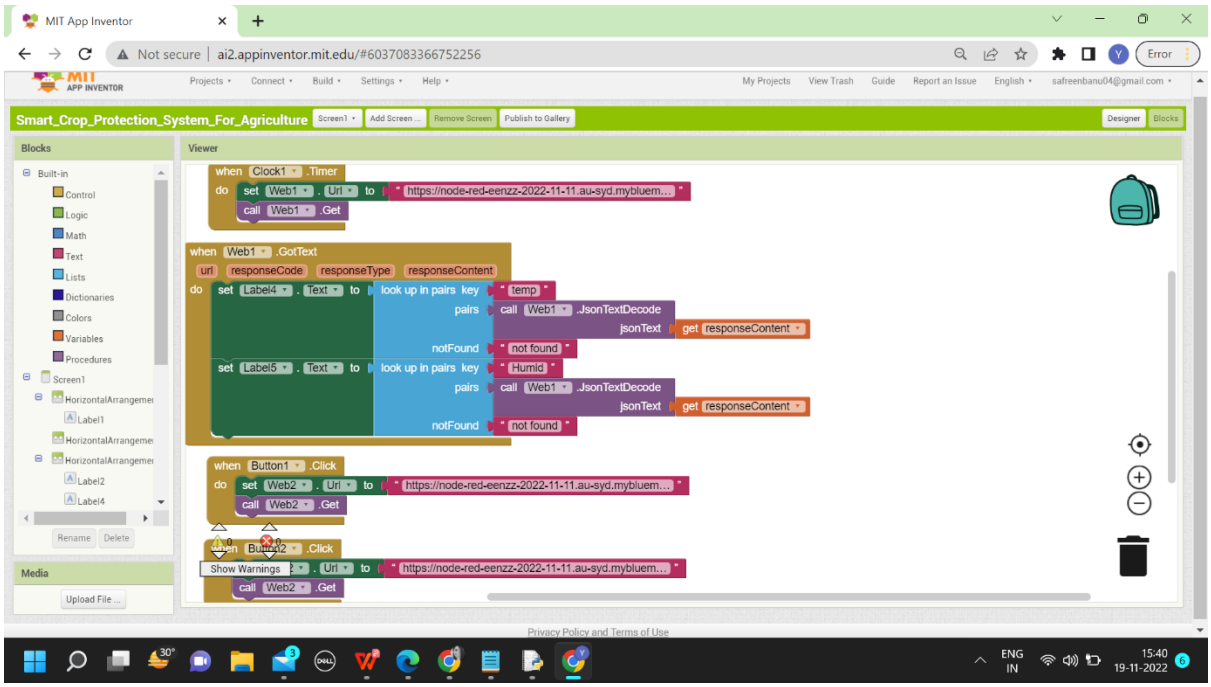
7.4.Nodered:

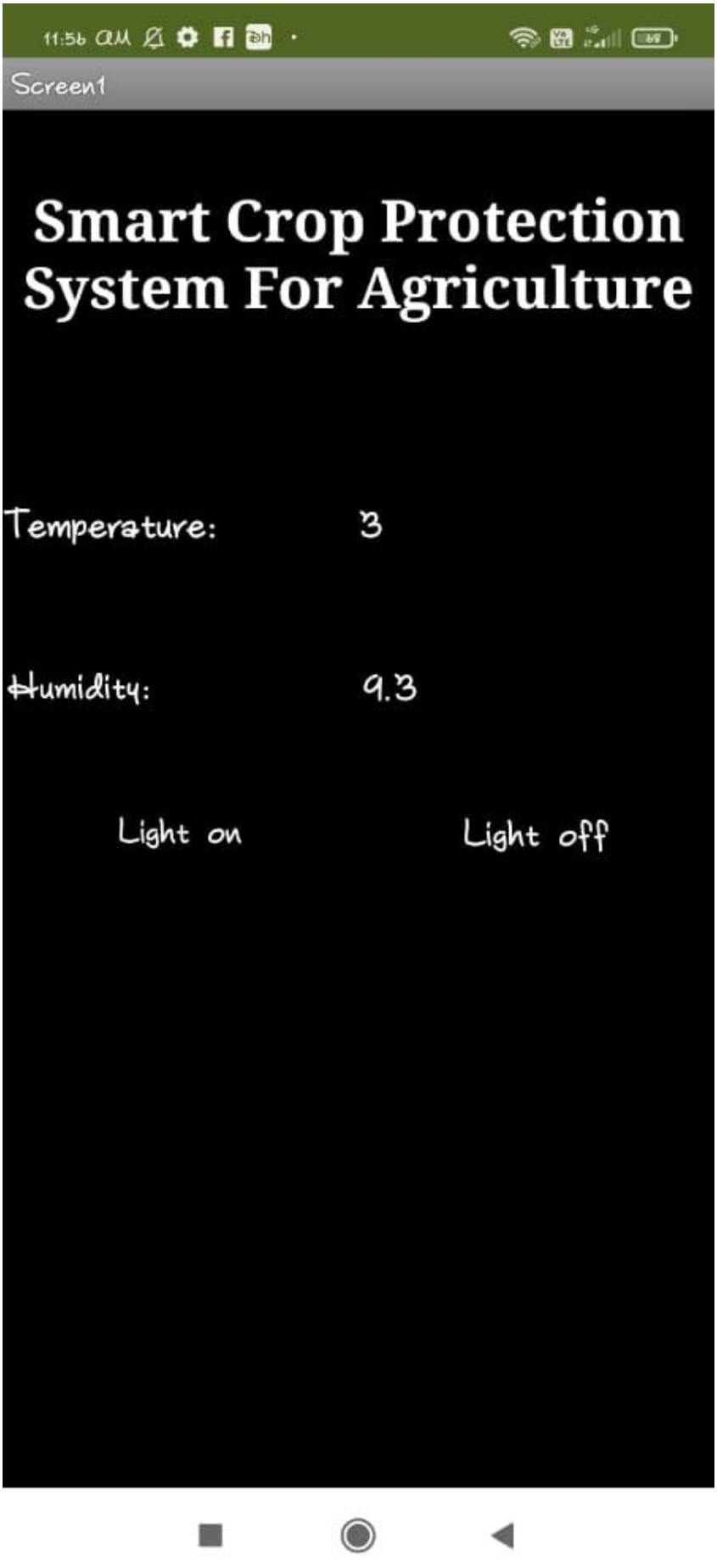

Node-RED
Flow 1
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
change
data
msg
IBM IoT
connected
Crop farming
PIR Sensor
Ultrasonic sensor
Ultrasonic Sensor
[get] /sensor
http func
http
msg.payload
[get] /command
timestamp
debug
19 November 2022


https://node-red-eenzz-2022-11-11.au-syd.mybluemix.net/ui/#!/0?socketid=Dy2yJSkvdq07RlXfAABV
Home
Default
humid
26.2
Default
temp
3

7.5.MIT App:


MIT App Inventor
Not secure | ai2.appinventor.mit.edu/#6037083366752256
Smart_Crop_Protection_System_For_Agriculture
Palette
Viewer
Components
Properties
Screen1
Smart Crop Protection System For Agriculture
Temperature: NA
Humidity: NA
Light on
Light off


MIT App Inventor
Not secure | ai2.appinventor.mit.edu/#6037083366752256
Smart_Crop_Protection_System_For_Agriculture
Blocks
Viewer
when Clock1 .Timer
do set Web1 . Url to https://node-red-eenzz-2022-11-11.au-syd.mybluem...
call Web1 .Get
when Web1 .GotText
url responseCode responseType responseContent
set Label4 . Text to look up in pairs key temp
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound not found
set Label5 . Text to look up in pairs key Humid
when Button1 .Click
set Web2 . Url to https://node-red-eenzz-2022-11-11.au-syd.mybluem...
call Web2 .Get
Show Warnings
Media
Upload File ...
Privacy Policy and Terms of Use


11:56 AM
Screen1
Smart Crop Protection System For Agriculture
Temperature: 3
Humidity: 9.3
Light on
Light off

8.TESTING

8.1 Test Cases :

				Date	25-Nov-22							
				Team ID	PNT2022TMID46939							
				Project Name	IoT based Smart Crop Protection System for Agriculture							
				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
ESP32 Arduino	Circuit	Simulation	Verify the circuit is working correctly and sends the data to watson		1.Run the simulation 2.Verify the output is correct 3.Verify the json data is send to watson platform	https://wokwi.com/projects/348742133947064403	The data in watson displayed properly	Working as expected	Pass				Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini
Python	Random Functions	Random data	Verify the random data is sent to the watson device	Python	1.Run the python code 2.Check the modules installed properly 3.The random data is perfectly generated 4.The data is sent as json to watson		The data in watson displayed properly	Working as expected	Pass	Steps are not clear to follow			Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini
Watson Device	Receive data	Visual Data	Verify the recieved data is sent to the node red		1.Verify the data is recieved 2.Sending the json data to node red 3.The data is processed and Visualized	https://g4cpja.internetofthings.ibmcloud.com/dashboard/devices/browse	The data is visualized	Working as expected	Pass	Sometimes the the json is not sent to node red			Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini
Node red	Dashboard	Web app	Verify the data is Visualized	Node red	1.Verify the data is sent to MIT 2.Verify the data is fetched from watson 3.Verify the data is sent to Cloudant 4.The control logic is working or not	https://node-red-cenqz.2022.11-13.au-syd.mybluemix.net/red/#flow/96d3643a6e55354a	The data is sent to the cloudant	Working as expected	Pass				Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini
Cloudant	Database	JSON data	The data is stroed	Cloudant	1.Verify the data is stored to cloudant database 2.Verify the database is created	https://72917fc4-9a8f-4a32-bd91-9bc648872fdc-bluemix.cloudant.com/dashboard.html#database/hackproof_all_docs	The data perfectly stored as a document	Data Stored	Pass				Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini
MIT app	Alert	Mobile app	Verify the data is showed in app		1.The app is working in mobile 2.The data is recived from node red 3.Verify the command is working or not		The application showed the sensor data	Working as expected	Pass				Y.Safreen Banu , L.Rabiya , M.Suganya , D.Priyadharshini

8.2 User Acceptance Testing :

1.Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	8	3	2	2	15
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	0	1	1
Won't Fix	0	4	2	1	7
Totals	22	12	12	25	71

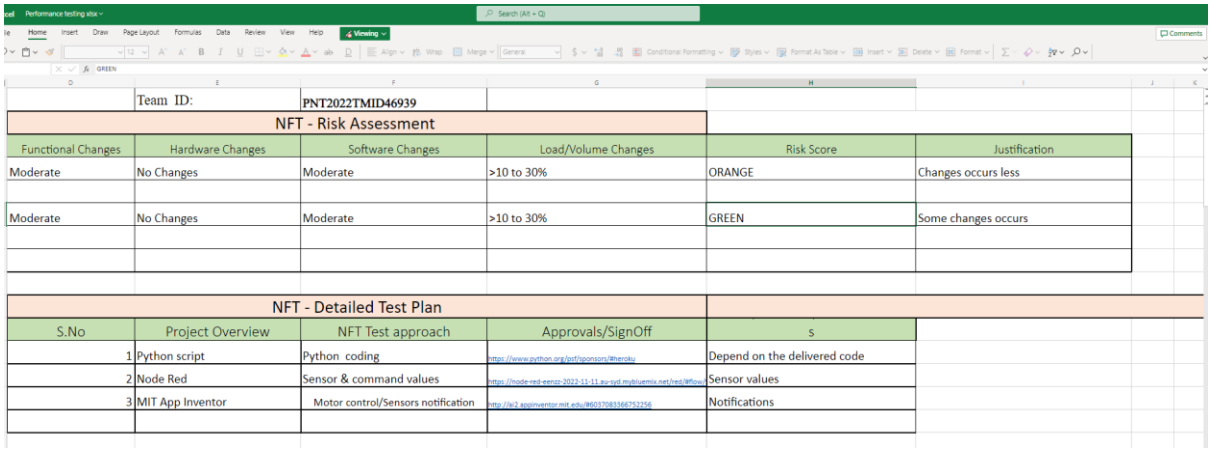
3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	49	0	0	49
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

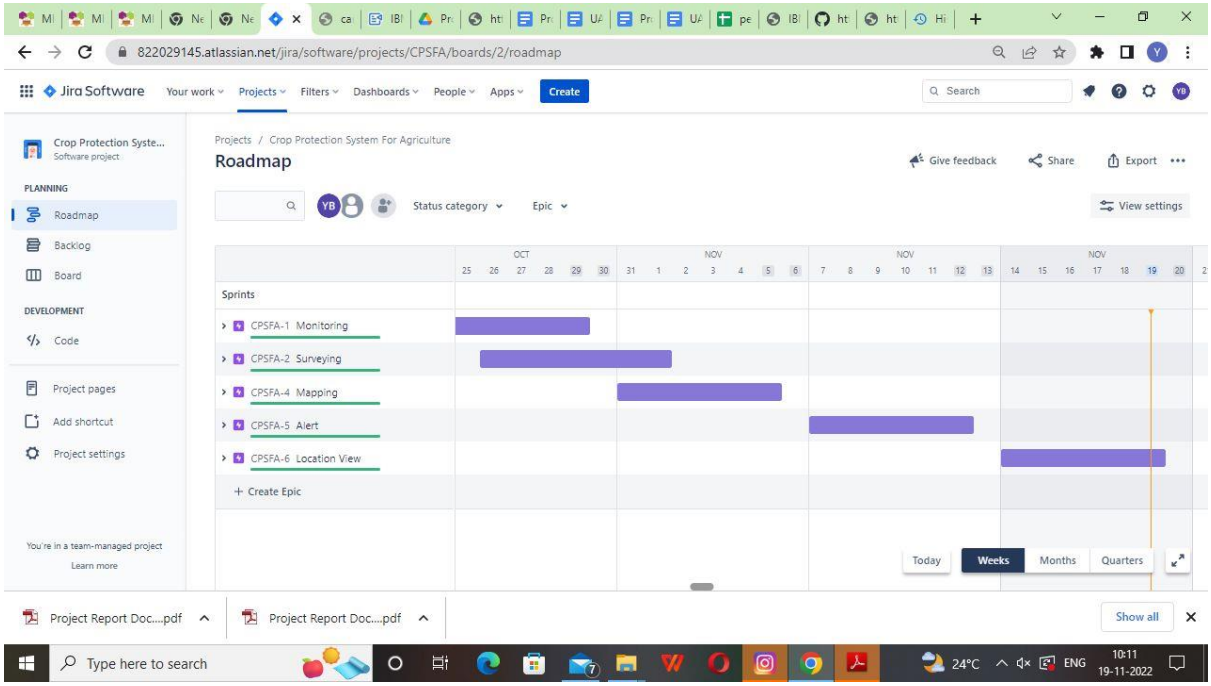
9.1 Performance Metrics :



	Team ID:	PNT2022TMID46939			
NFT - Risk Assessment					
Functional Changes	Hardware Changes	Software Changes	Load/Volume Changes	Risk Score	Justification
Moderate	No Changes	Moderate	>10 to 30%	ORANGE	Changes occurs less
Moderate	No Changes	Moderate	>10 to 30%	GREEN	Some changes occurs

NFT - Detailed Test Plan				
S.No	Project Overview	NFT Test approach	Approvals/SignOff	s
1	Python script	Python coding	https://www.python.org/psf/sponsors/#heroku	Depend on the delivered code
2	Node Red	Sensor & command values	https://node-red-eenzc-2022-11-11.au-syd.mybluemix.net/red/#flow/	Sensor values
3	MIT App Inventor	Motor control/Sensors notification	http://ai2.appinventor.mit.edu/#6037083366752256	Notifications

	Team ID:	PNT2022TMID46939			
NFT - Risk Assessment					
Functional Changes	Hardware Changes	Software Changes	Load/Volume Changes	Risk Score	Justification
Moderate	No Changes	Moderate	>10 to 30%	ORANGE	Changes occurs less
Moderate	No Changes	Moderate	>10 to 30%	GREEN	Some changes occurs
NFT - Detailed Test Plan					
S.No	Project Overview	NFT Test approach	Approvals/SignOff	s	
1	Python script	Python coding	https://www.python.org/psf/sponsors/#heroku	Depend on the delivered code	
2	Node Red	Sensor & command values	https://node-red-eensz-2022-11-11.au-syd.mybluemix.net/red/#flow/	Sensor values	
3	MIT App Inventor	Motor control/Sensors notification	http://ai2.appinventor.mit.edu/#6037088366752256	Notifications	



10. ADVANTAGES & DISADVANTAGE

10.1 Advantages

- All the data like climatic conditions and changes in them, soil or Crop conditions everything can be easily monitored.
- Risk of crop damage can be lowered to a greater extent.
- The process included in farming can be controlled using the web Applications from anywhere, anytime.
- Farmers can monitor the health of farm animals closely, even if they are physically distant.

- Smart farming systems reduce waste, improve productivity and enable management of a greater number of resources through remote sensing

10.2 Disadvantages

- Smart Crop Protection requires internet connectivity continuously, but rural part scan not fulfill this requirement.
- Any faults in the sensors can cause great loss in the agriculture, due to wrong records and the actions of automated processes.
- IoT devices need much money to implement.

11. CONCLUSION

Iot based smart crop protection system for agriculture reduces the ecological footprint of farming and man power. Minimized or site-specific application of inputs, such as fertilizers and pesticides, in precision agriculture systems will mitigate leaching problems as well as the emission of greenhouse gases.The dependency on manual labour has reduced significantly. The processes like pest control, fertilizing, and irrigation are increasingly becoming automated, and farmers can control them remotely. The use of smart IoT sensors can maintain these processes, increasing crop production.

12. FUTURE SCOPE

IoT smart agriculture products are designed to help monitor crop fields using sensors and by automating irrigation systems. As a result of this smart crop protection, farmers and associated brands can easily monitor the field conditions from anywhere without any hassle the future scope makes the farmers to monitor the crops and produce a good yield of crop production.

13. APPENDIX

13.1 SOURCE CODE :

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random
```

```
#Provide your IBM Watson Device Credentials

authMethod = "token"

organization = "yet4pm"

authToken = "12345678910"


deviceType1 = "Sensor"

deviceId1 = "DHT"



deviceType3 = "Actuator"

deviceId3 = "Water_pump"


deviceType2 = "Sensor1"

deviceId2 = "soil_moisture"


# Initialize GPIO



def myCommandCallback(cmd):

print("Command received: %s" % cmd.data['command'])

status=cmd.data['command']

if status=="Waterpump_on":
```

```
print ("Water Pump is Turned ON \n")

else :

print ("Water pump is off")


#print(cmd)




try:

deviceOptions1 = {"org": organization, "type": deviceType1, "id": deviceId1, "auth-method": authMethod, "auth-token": authToken}

deviceCli1 = ibmiotf.device.Client(deviceOptions1)


deviceOptions2 = {"org": organization, "type": deviceType2, "id": deviceId2, "auth-method": authMethod, "auth-token": authToken}

deviceCli2 = ibmiotf.device.Client(deviceOptions2)


deviceOptions3 = {"org": organization, "type": deviceType3, "id": deviceId3, "auth-method": authMethod, "auth-token": authToken}

deviceCli3 = ibmiotf.device.Client(deviceOptions3)


#.....


except Exception as e:
```

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

sys.exit()


# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times

deviceCli1.connect()

deviceCli2.connect()

deviceCli3.connect()


while True:

#Get Sensor Data from esp32


temp=random.randint(0,45)

Humid=random.randint(0,100)


data1 = { 'Temperature' : temp , 'Humidity': Humid}

#print data

def myOnPublishCallback():

print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid,"to IBM Watson \n")


success1 = deviceCli1.publishEvent("DHT Sensor", "json", data1, qos=0, on_publish=myOnPublishCallback)

if not success1:

print("Not connected to IoTF\n")
```

```
time.sleep(1)


Soil_moisture=random.randint(0,100)
data2 = { 'Soil_moisture' : Soil_moisture}


def myOnPublishCallback2():
print ("Published Soil_moisture = %s %%" % temp, "to IBM Watson")
success2 = deviceCli2.publishEvent("Soil Moisture Sensor", "json", data2, qos=0, on_publish=myOnPublishCallback2)


if not success2:
print("Not connected to IoTF")
time.sleep(1)
deviceCli3.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud
deviceCli1.disconnect()
deviceCli2.disconnect()
```

Wokwi Code

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 4     // what pin we're connected to
#define DHTTYPE DHT11   // define type of sensor DHT 11
#define LED 5
```

```
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

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

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

#define ORG "g4apja"//IBM ORGANITION ID
#define DEVICE_TYPE "r3ibm"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "r123"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678910"     //Token
String data3;
float h, t;


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


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

```
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

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

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



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

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

```
  payload += temp;
  payload += "," "\"humidity\":";
  payload += humid;
  payload += "}";


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


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

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

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function definition for wifi connect
{
  Serial.println();
```

```
  Serial.print("Connecting to ");

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

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

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

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

  Serial.println("data: "+ data3);
```

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

  }

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

  }
data3="";


}
```

13.2 GITHUB & PROJECT DEMO LINK:

GITHUB:

https://github.com/IBM-EPBL/IBM-Project-45720-1660731821

Wokwi Link :

https://wokwi.com/projects/348742130947064403

MIT App Link:

http://ai2.appinventor.mit.edu/#6037083366752256

Node Link:

https://node-red-eenzz-2022-11-11.au-syd.mybluemix.net/red/#flow/8f5b618a88505785

Video Demo Link:

https://youtu.be/tcF1vpkxDVA