IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

PROJECT REPORT

TEAM ID	PNT2022TMID35844
TEAM MEMBERS	ALBEN RICHARDS M J - 2019504507 HARINANDINI N - 2019504526 MAHESWARI G - 2019504545 ABISHEK R - 2019504001
COLLEGE	MADRAS INSTITUTE OF TECHNOLOGY , ANNA UNIVERSITY

1 . INTRODUCTION

1.1 PROJECT OVERVIEW

With increasing population across the world, food production and farming needs to get increasingly productive and capable of high yields in limited time. The scope for manual experimentation, viability assessment through trial and error etc are no longer feasible. According to the UN Food and Agriculture Organization, “the world will need to produce 70% more food in 2050 than it did in 2006”. Low productivity of crops is one of the main problems faced by the farmers in our country. This can be because of two main reasons. Crops destroyed by wild animals and because of bad weather condition increase in temperature , humidity , soil moisture values. This paper provides a solution to the destruction of crops by animals. This system will provide a complete technical solution using the Internet of things (IOT) to the farmers to prevent their crops from wild animals and provide information to the farmers to maximize their production.It also helps the users to supervise the soil moisture content, temperature and humidity values near the cultivation field. And also provides control over periodic watering using application. Thus an IoT based crop protection system has been introduced to improve the standard of farming by preventing the cultivation field from varying climatic changes and haunting animals.

1.2 PURPOSE

- The main purpose of this project is to increase food production and reduce the financial losses by implementing IoT technology into farming to make cultivation in a better and organized way.
- Requirement of low man power and less time consumption.
- Reduced working hours and work load.
- Financial losses may be due to the destruction of crops due to animal

intrusion which is detected prevented using generation of an alarm.

- And also smart farming is implemented by turning on and off the motors based on the field status.
- Field status include parameters such as temperature , humidity , soil moisture etc.

2 . LITERATURE SURVEY

2.1 EXISTING PROBLEM

"Food” is the important thing, which is necessary for survival. For the growing food consumption along with population,farmers are doing their part in an effective manner, while which they face few problems such as:

There are increasing pressures from climatic changes, soil erosion and biodiversity loss and from consumer's varying opinion over food varieties and concerns about the steps to be taken to meet the increasing demand.

And the natural world that farming works with plants, pests and diseases continues to pose their own challenges.

The effects of climate change affect farmer's ability to increase the production and achieve profit.

Increasingly volatile weather and more extreme events like floods, droughts limit the availability of water, allow weeds, pests and fungi to thrive, and can reduce crop productivity.

2.2 REFERENCES

1) https://www.agrivi.com/blog/top-five-strategies-to-protect-crops-from-wild-animals/

2) https://article.murata.com/en-eu/article/measures-against-wildlife-damage-through-iot

3) Tanmay Baranwal” Development of IOT based Smart Security and

Monitoring Devices for Agriculture”, Department of Computer Science Lovely Professional University Phagwara, Punjab, IEEE-2016.

4) P. Deotale and P. Lokulwar, "Smart Crop Protection System from Wild Animals Using IoT," 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA), 2021, pp. 1-4, doi: 10.1109/ICCICA52458.2021.9697315.

5) S. Pandey and S. B. Bajracharya, “Crop protection and its effectiveness against wildlife: A case study of two villages of Shivpuri national park, Nepal,” Nepal Journal of Science and Technology, vol. 16, no. 1, pp. 1– 10, 2015.

6) Hanshi Wang; Jingli Lu; Lizhen Liu; Wei Song; Zhaoxia Wang; “Community Alarm System Design BasedOnMCU And GSM” Year:2015

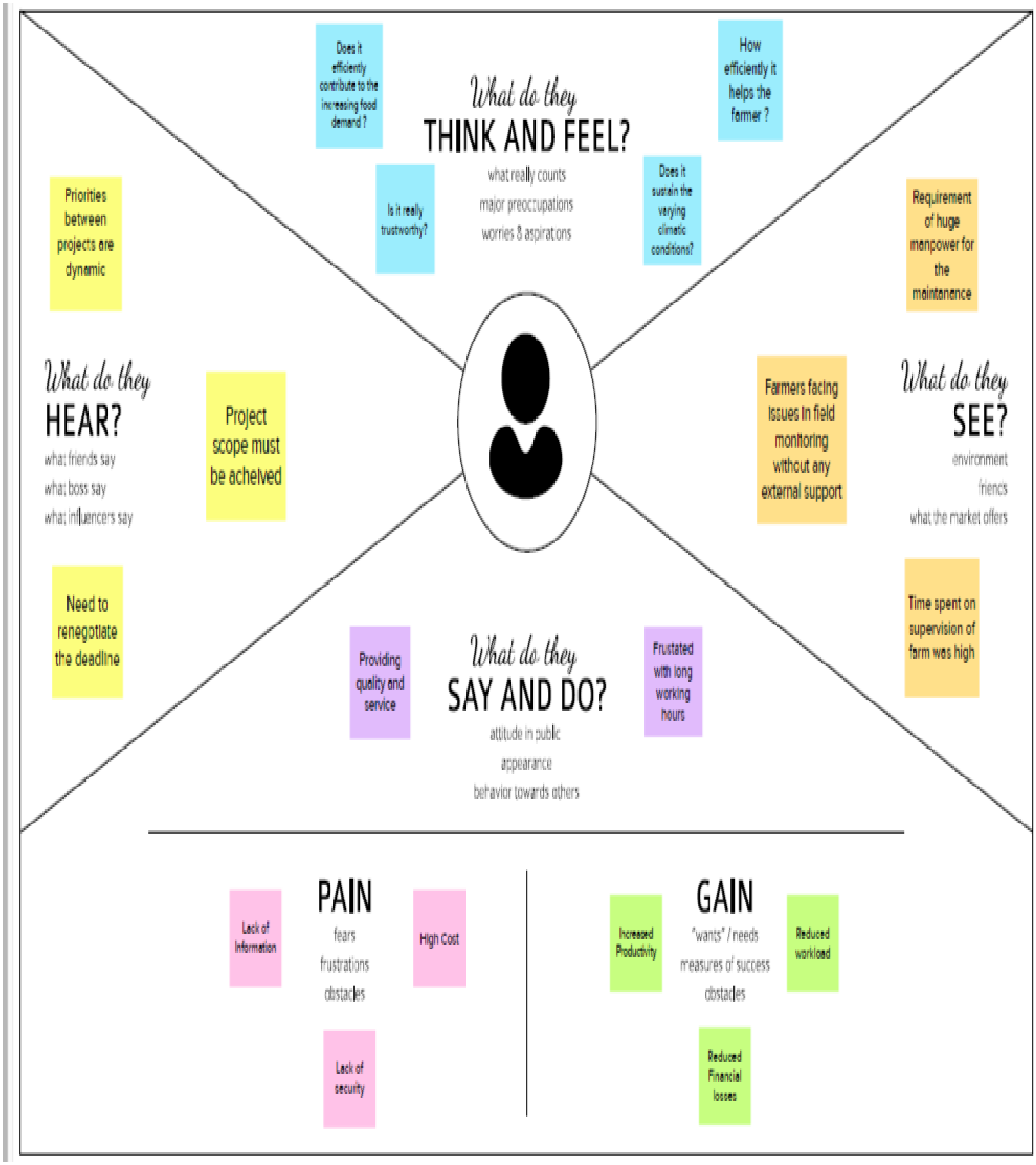
2.3 PROBLEM STATEMENT

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.

3. IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

Empathy map discusses the feelings, thoughts and fears of the users on using the product in their farm. It also discusses the questions arising in the minds of the farmers which will help to improve customer satisfaction. It also discusses the lack of awareness regarding the availability of the product.



What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

Does it efficiently contribute to the increasing food demand ?

Is it really trustworthy?

How efficiently it helps the farmer ?

Does it sustain the varying climatic conditions?

What do they
HEAR?
what friends say
what boss say
what influencers say

Priorities between projects are dynamic

Project scope must be acheived

Need to renegotiate the deadline

What do they
SEE?
environment
friends
what the market offers

Requirement of huge manpower for the maintenance

Farmers facing Issues in field monitoring without any external support

Time spent on supervision of farm was high

What do they
SAY AND DO?
attitude in public
appearance
behavior towards others

Providing quality and service

Frustated with long working hours

PAIN
fears
frustrations
obstacles

Lack of Information

High Cost

Lack of security

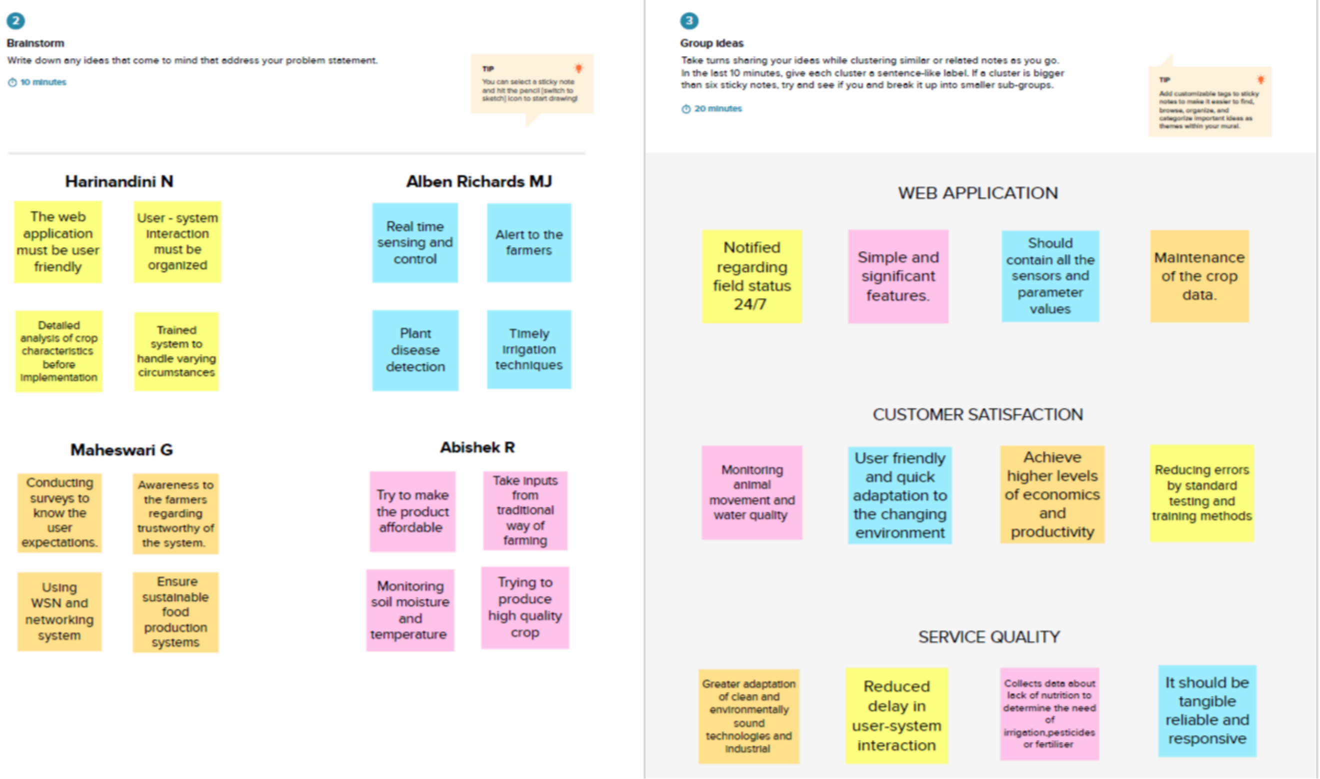
GAIN
"wants" / needs
measures of success
obstacles

Increased Productivity

Reduced workload

Reduced Financial losses

3.2 IDEATION AND BRAIN STORMING



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!

Harinandini N

- The web application must be user friendly
- User - system interaction must be organized
- Detailed analysis of crop characteristics before implementation
- Trained system to handle varying circumstances

Alben Richards MJ

- Real time sensing and control
- Alert to the farmers
- Plant disease detection
- Timely irrigation techniques

Maheswari G

- Conducting surveys to know the user expectations.
- Awareness to the farmers regarding trustworthy of the system.
- Using WSN and networking system
- Ensure sustainable food production systems

Abishek R

- Try to make the product affordable
- Take inputs from traditional way of farming
- Monitoring soil moisture and temperature
- Trying to produce high quality crop

3

Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, 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.

WEB APPLICATION

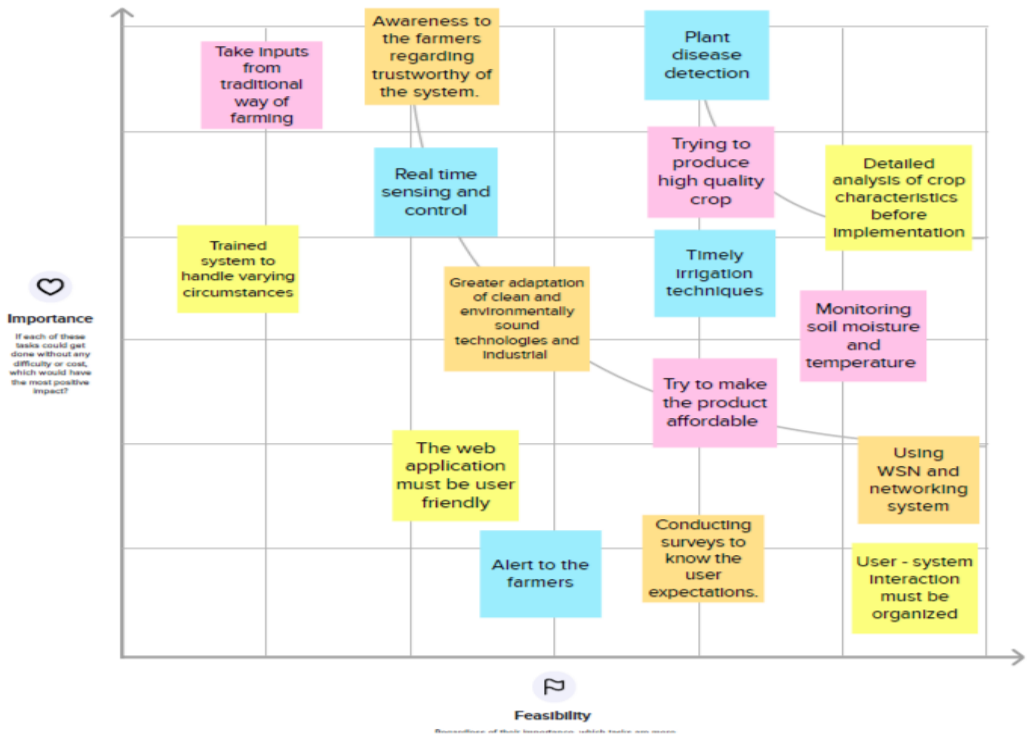
- Notified regarding field status 24/7
- Simple and significant features.
- Should contain all the sensors and parameter values
- Maintenance of the crop data.

CUSTOMER SATISFACTION

- Monitoring animal movement and water quality
- User friendly and quick adaptation to the changing environment
- Achieve higher levels of economics and productivity
- Reducing errors by standard testing and training methods

SERVICE QUALITY

- Greater adaptation of clean and environmentally sound technologies and industrial
- Reduced delay in user-system interaction
- Collects data about lack of nutrition to determine the need of irrigation,pesticides or fertiliser
- It should be tangible reliable and responsive

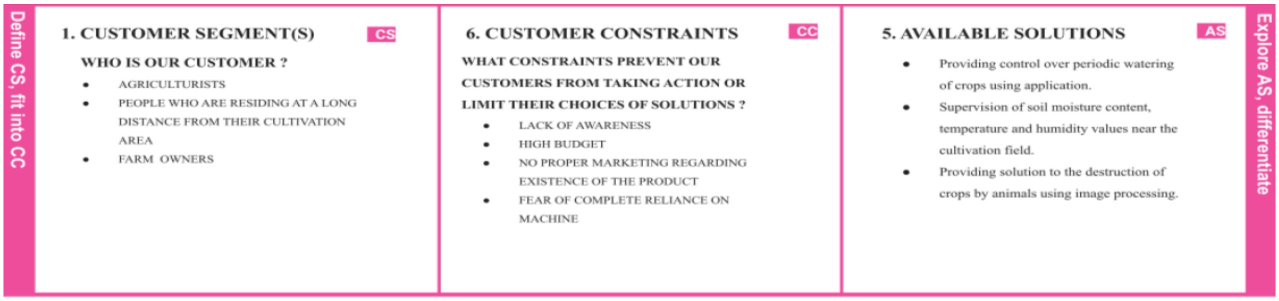

Take inputs from traditional way of farming
Awareness to the farmers regarding trustworthy of the system.
Plant disease detection
Real time sensing and control
Trying to produce high quality crop
Detailed analysis of crop characteristics before implementation
Trained system to handle varying circumstances
Greater adaptation of clean and environmentally sound technologies and industrial
Timely irrigation techniques
Monitoring soil moisture and temperature
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Try to make the product affordable
The web application must be user friendly
Using WSN and networking system
Alert to the farmers
Conducting surveys to know the user expectations.
User - system interaction must be organized
Feasibility
Regardless of their importance, which tasks are more

3.3 PROPOSED SOLUTION

PARAMETER	**DESCRIPTION**
Problem Statement (Problem to be solved)	● Low productivity of crops due to wildanimals attacking the cultivation field and bad weather condition. ● Time consumption and energy spent being high still lead to less productivity. ● Financial and production loss is high. ● Loss in productivity leads to food scarcity for the growing population.
Idea / Solution description	● 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.
Novelty / Uniqueness	● Responsible Consumption and production. ● End of poverty increasing access to natural resources and new technology. ● Climate-related negative and positive aspects should be investigated utilizing IoT based sensors.
Social Impact / Customer Satisfaction	Double the agriculture productivity and incomes of small scale food producers
Business Model (Revenue Model)	The business model will be a freemium model with an add-on subscription. The Freemium model brings in customers who get used to basic services like tracking personnel diet.

Scalability of the Solution	Upgrading technological capabilities across industrial sectors

3.4 PROBLEM SOLUTION FIT



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

WHO IS OUR CUSTOMER ?

- AGRICULTURISTS
- PEOPLE WHO ARE RESIDING AT A LONG DISTANCE FROM THEIR CULTIVATION AREA
- FARM OWNERS

6. CUSTOMER CONSTRAINTS CC

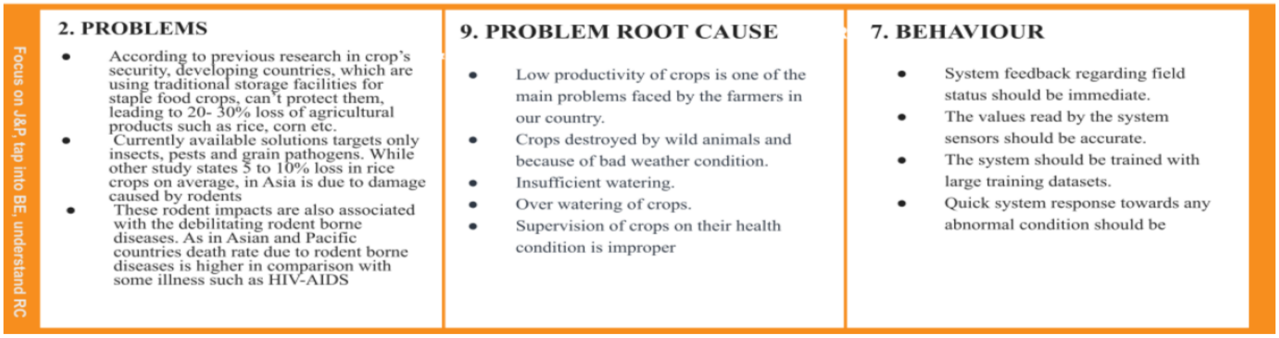
WHAT CONSTRAINTS PREVENT OUR CUSTOMERS FROM TAKING ACTION OR LIMIT THEIR CHOICES OF SOLUTIONS ?

- LACK OF AWARENESS
- HIGH BUDGET
- NO PROPER MARKETING REGARDING EXISTENCE OF THE PRODUCT
- FEAR OF COMPLETE RELIANCE ON MACHINE

5. AVAILABLE SOLUTIONS AS

- Providing control over periodic watering of crops using application.
- Supervision of soil moisture content, temperature and humidity values near the cultivation field.
- Providing solution to the destruction of crops by animals using image processing.

Explore AS, differentiate



Focus on J&P, tap into BE, understand RC

2. PROBLEMS

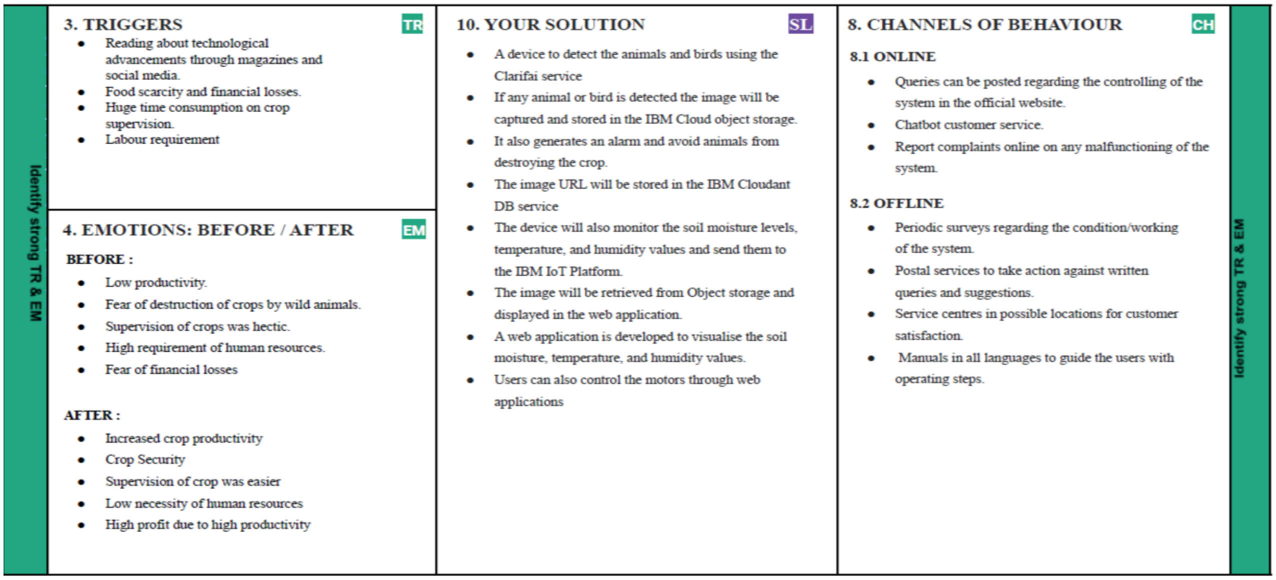
- According to previous research in crop's security, developing countries, which are using traditional storage facilities for staple food crops, can't protect them, leading to 20- 30% loss of agricultural products such as rice, corn etc.
- Currently available solutions targets only insects, pests and grain pathogens. While other study states 5 to 10% loss in rice crops on average, in Asia is due to damage caused by rodents
- These rodent impacts are also associated with the debilitating rodent borne diseases. As in Asian and Pacific countries death rate due to rodent borne diseases is higher in comparison with some illness such as HIV-AIDS

9. PROBLEM ROOT CAUSE

- Low productivity of crops is one of the main problems faced by the farmers in our country.
- Crops destroyed by wild animals and because of bad weather condition.
- Insufficient watering.
- Over watering of crops.
- Supervision of crops on their health condition is improper

7. BEHAVIOUR

- System feedback regarding field status should be immediate.
- The values read by the system sensors should be accurate.
- The system should be trained with large training datasets.
- Quick system response towards any abnormal condition should be



Identify strong TR & EM

3. TRIGGERS TR

- Reading about technological advancements through magazines and social media.
- Food scarcity and financial losses.
- Huge time consumption on crop supervision.
- Labour requirement

4. EMOTIONS: BEFORE / AFTER EM

BEFORE :

- Low productivity.
- Fear of destruction of crops by wild animals.
- Supervision of crops was hectic.
- High requirement of human resources.
- Fear of financial losses

AFTER :

- Increased crop productivity
- Crop Security
- Supervision of crop was easier
- Low necessity of human resources
- High profit due to high productivity

10. YOUR SOLUTION SL

- A device to detect the animals and birds using the Clarifai service
- If any animal or bird is detected the image will be captured and stored in the IBM Cloud object storage.
- It also generates an alarm and avoid animals from destroying the crop.
- The image URL will be stored in the IBM Cloudant DB service
- The device will also monitor the soil moisture levels, temperature, and humidity values and send them to the IBM IoT Platform.
- The image will be retrieved from Object storage and displayed in the web application.
- A web application is developed to visualise the soil moisture, temperature, and humidity values.
- Users can also control the motors through web applications

8. CHANNELS OF BEHAVIOUR CH

8.1 ONLINE

- Queries can be posted regarding the controlling of the system in the official website.
- Chatbot customer service.
- Report complaints online on any malfunctioning of the system.

8.2 OFFLINE

- Periodic surveys regarding the condition/working of the system.
- Postal services to take action against written queries and suggestions.
- Service centres in possible locations for customer satisfaction.
- Manuals in all languages to guide the users with operating steps.

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	1. Registration through Form 2. Registration through Gmail 3. Registration through LinkedIN
FR-2	User Confirmation	1. Confirmation via Email 2. Confirmation via OTP
FR-3	App features	1. Simple to use 2. Can be used in all operating systems 3. Regular updates for the app 4. Asking feedback from users to add in updates
FR-4	Speed	1. The whole device and sensors need to be connected to the internet 2. The device must be able to update values as soon as possible for better crop management.
FR-5	Data management and analysis	1. **Data preprocessing** - This will help to improve accuracy and efficiency of the subsequent mining. 2. **Data reduction** - is used to encode the data to a smaller reduced representation,so the integrity of original data was preserved. 3. **Data modeling** - It extracts the knowledge from the prepared data. Data modeling applies intelligent methods to identify patterns in the data.

FR-6	Authentication	1. Data and control of the field status can be accessed only by the concerned / authorized user. 2. Authorized access involves access over Motor control and visualizing the temperature , soil moisture and humidity values.

4.2 NON FUNCTIONAL REQUIREMENTS

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

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	1. The user must be able to understand ,learn new features and use them . 2. Language used is English which is the standard medium for communication. It depends on :- 1. Efficiency of user intuitiveness 2. Low perceived workload
NFR-2	**Security**	1. Prevents hackers from stealing personal data of customer for identity theft 2. Prevents interception of sensitive information travelling over the network
NFR-3	**Reliability**	1. Periodically notified regarding the field status 2. Long distant field and crop management 3. System is trained to reduce the probability of errors in crop monitoring.

NFR-4	**Performance**	1. The real time information from IoT devices was used to control on-off switching water sprinklers automatically. 2. Initially, we collected IoTs information for 5 months (170 days) and performed yield analysis with this data. 3. The obtained IoTs information consists of temperature, humidity, and soil moisture, and was collected every 20 min, but for analysis the daily averages were used.
NFR-5	**Availability**	1. Need internet connection for updating the values and low maintenance needed for the operation. 2. Can be accessed remotely from anywhere in the world .
NFR-6	**Scalability**	Extending functionality and features of the system on a regular basis based on customer feedback.

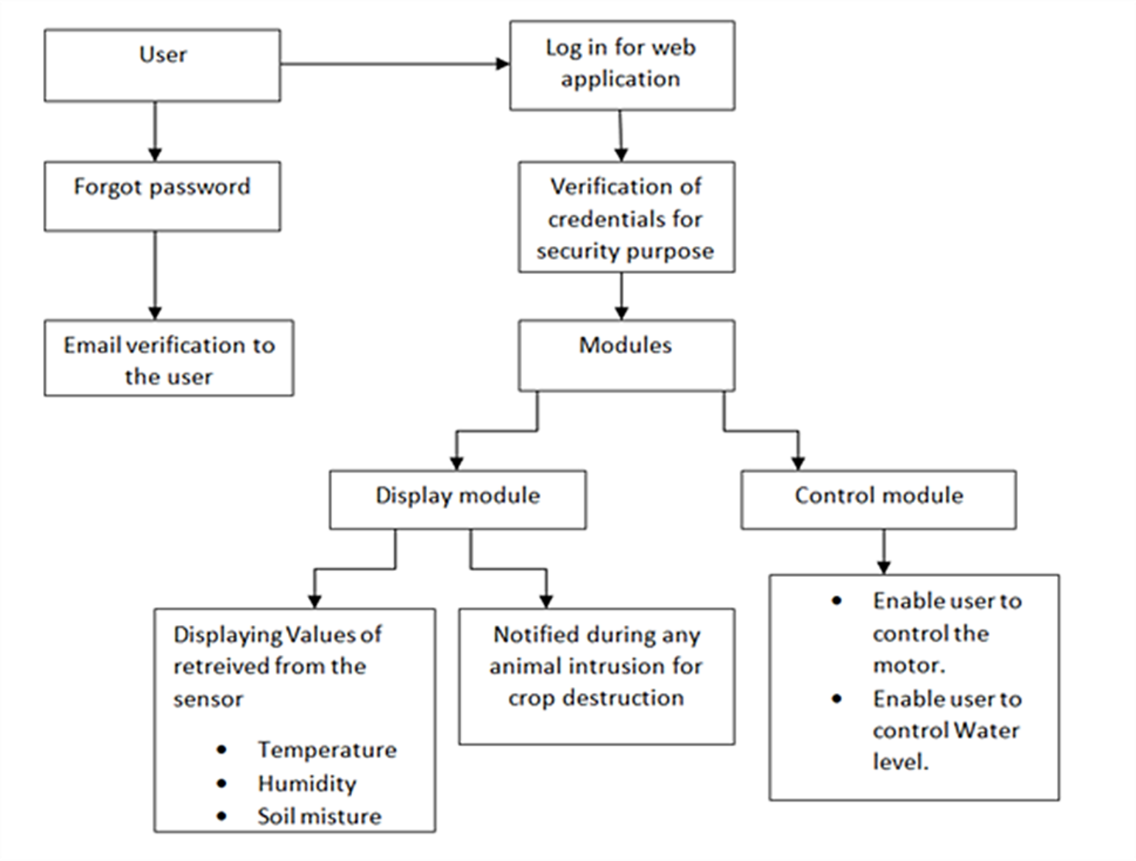
5. PROJECT DESIGN

5.1. DATAFLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.

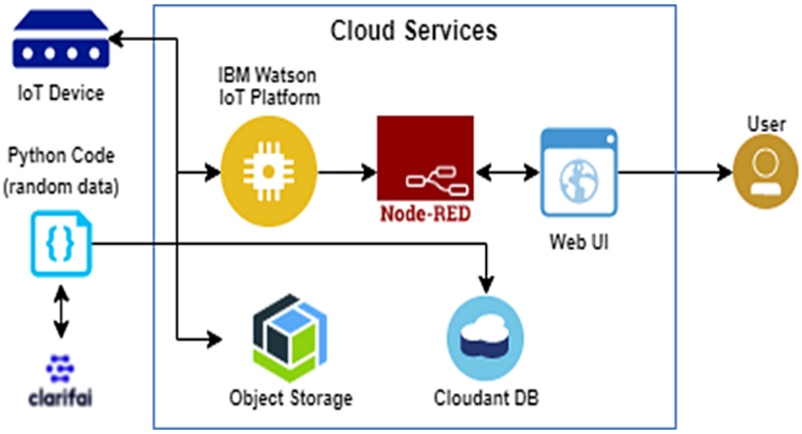
A neat and clear DFD can depict the right amount of the system requirement graphically.

It shows how data enters and leaves the system, what changes the information, and where data is stored.\

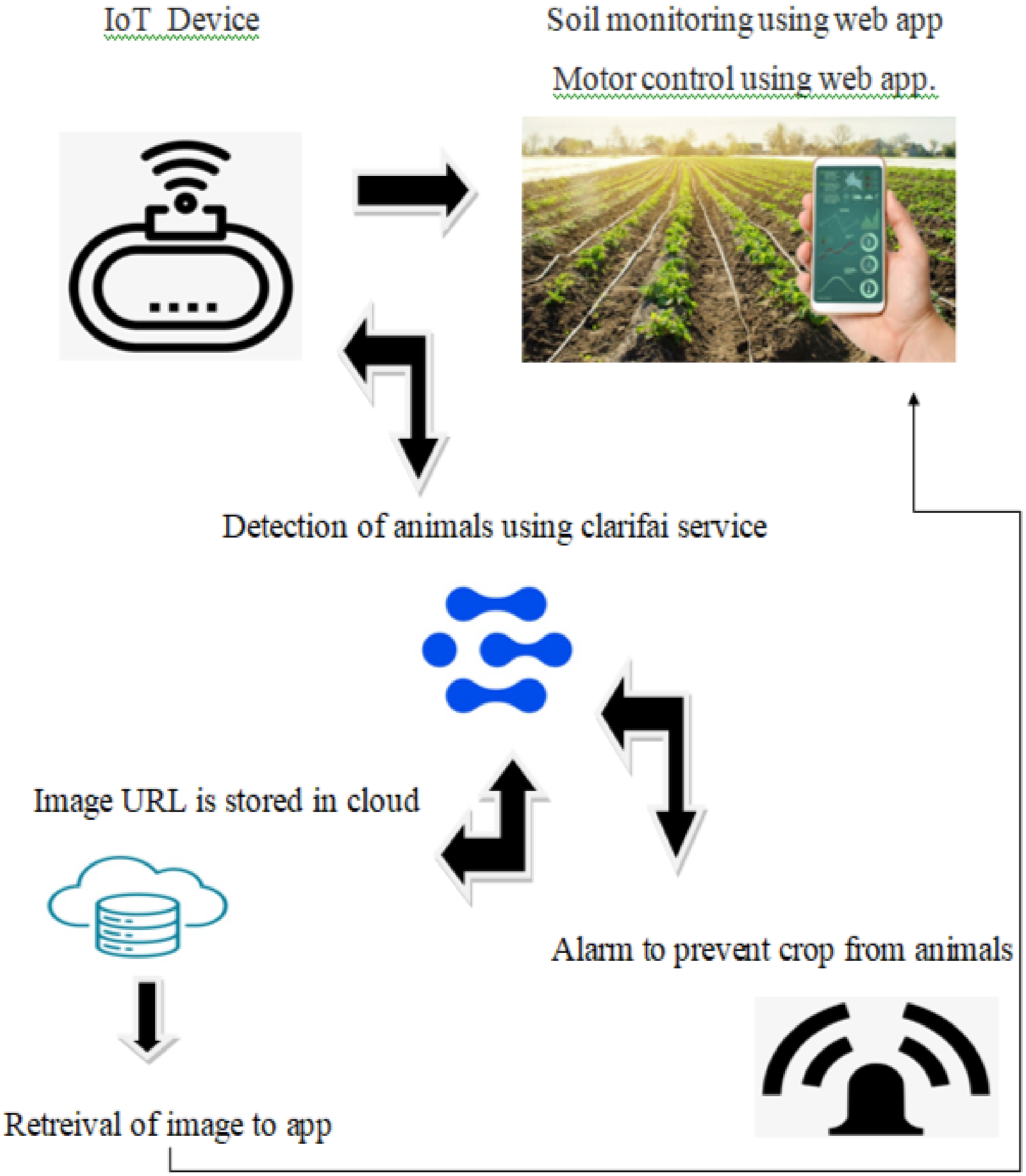

User
Log in for web application
Forgot password
Verification of credentials for security purpose
Email verification to the user
Modules
Display module
Control module
Displaying Values of retreived from the sensor
• Temperature
• Humidity
• Soil misture
Notified during any animal intrusion for crop destruction
• Enable user to control the motor.
• Enable user to control Water level.

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

TECHNICAL ARCHITECTURE



SOLUTION ARCHITECTURE


IoT Device
Soil monitoring using web app
Motor control using web app.
Detection of animals using clarifai service
Image URL is stored in cloud
Alarm to prevent crop from animals
Retreival of image to app

5.3 USER STORIES

User stories discuss the tasks thet has to be completed and the gain being attained by both consumer and developer on completion of the task.

Functional Requirement (Epic)	USN	User Story/ Task	Acceptance criteria	Priority	Release
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
Dashboard	USN-3	As a user,I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2

	USN-4	As a user,I can register for the application through Gmail	I can register & access the dashboard with Gmail	Medium	Sprint-1
Login	USN-5	As a user, I can log into the application by entering email & password	I can login into my account anytime based on demands	High	Sprint-1
Dashboard	USN -6	The application should be able to display the desired results to the user.	I can confirm the temperature and gas levels in my surroundings	High	Sprint - 1

Registration	USN - 7	As a web user I can go through social media websites and register for the application	I can access my dashboard	High	Sprint - 1

Login	USN - 8	As a web user I can link my google accounts to login in to the web application	I can log in and access the dashboard	High	Sprint -2
Dashboard	USN -9	As a user, I can track and analyze available data using Dashboard	I can analyze data	High	Sprint - 3
Monitoring field	USN - 10	24/7 Live monitoring of field /crop conditions for high reliability.	I can monitor from long distance	High	Sprint - 2
Instant notification	USN - 11	I can react instantaneously to the problems being faced.	I can solve issues without delay	High	Sprint - 3
IBM Watson	USN - 12	As a user I can request developers for help in case of failure of service or unanswered queries	I can ensure smooth experience	Medium	Sprint - 4

Have an overview on the entire process and ensure smooth workflow	USN -13	We must ensure perfect service from our team in order to satisfy the customer needs and neglect or reduce the errors and provide .	I can complete the work without any errors	High	Sprint - 4

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	**Functional Requirement (Epic)**	**USN**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Create and configure IBM Cloud services	USN-1	As a user, I can control and monitor the IoT device using IBM Watson	5	High	Alben Richards Abishek
		USN-2	As a developer, I can create a web application using Node red	5	High	Harinandini Maheswari

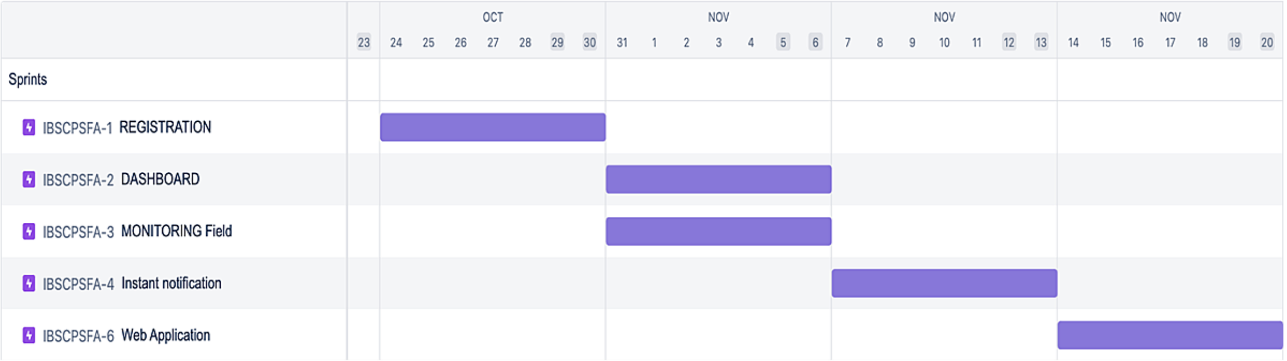
		USN-3	As a developer, I can create a databae to store image URL	5	Low	Alben Richards Maheswari
		USN-4	As a developer, I can create a bucket to store images	5	Low	Abishek Harinandini
Sprint-2	Development of python script	USN-5	As a user, I can tracktemperature , moisture and humidity values	10	Low	Alben Richards Harinandini
		USN-6	24/7 Live monitoring of field/crop conditions for high reliability.	10	Medium	Abishek Maheswari
Sprint-3	Clarifai service for detection.	USN-7	As a user , I can track the intrusion of any animal	10	High	Alben Richards Abishek
		USN-8	I can react instantaneously to the problems being faced.	10	Low	Harinandini Maheshwari
Sprint-4	Web Application using Node Red Service	USN-9	As a user I can request for help in case of failure of service or unanswered	10	Low	Alben Richards Maheswari

			queries			
		USn -10	We must ensure perfect service from our team in order to satisfy the customer needs and neglect errors.	10	High	Abishek Harinandini

6.2 SPRINT DELIVERY SCHEDULE

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

6.3 REPORTS FROM JIRA



Sprints

IBSCPSFA-1 REGISTRATION

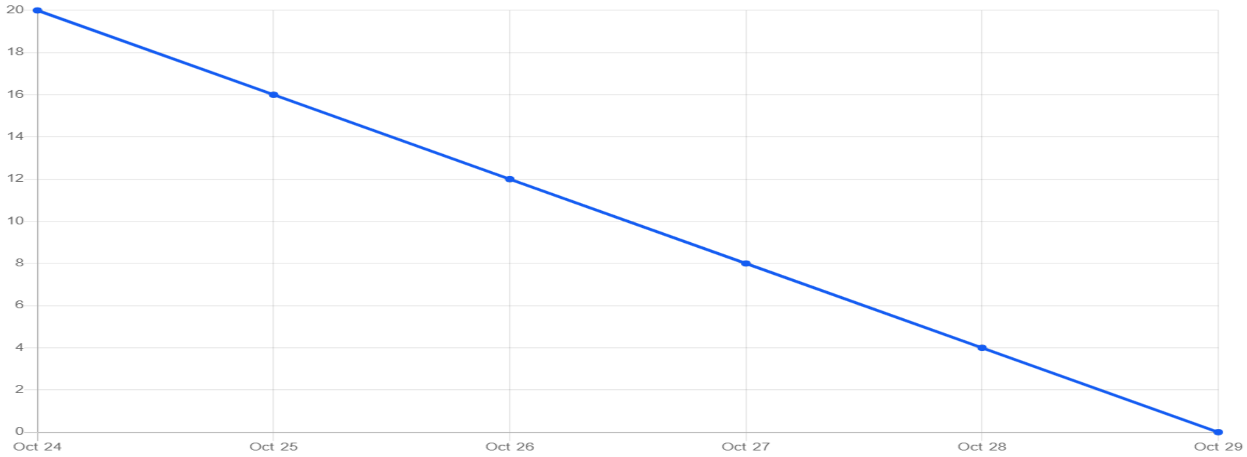
IBSCPSFA-2 DASHBOARD

IBSCPSFA-3 MONITORING Field

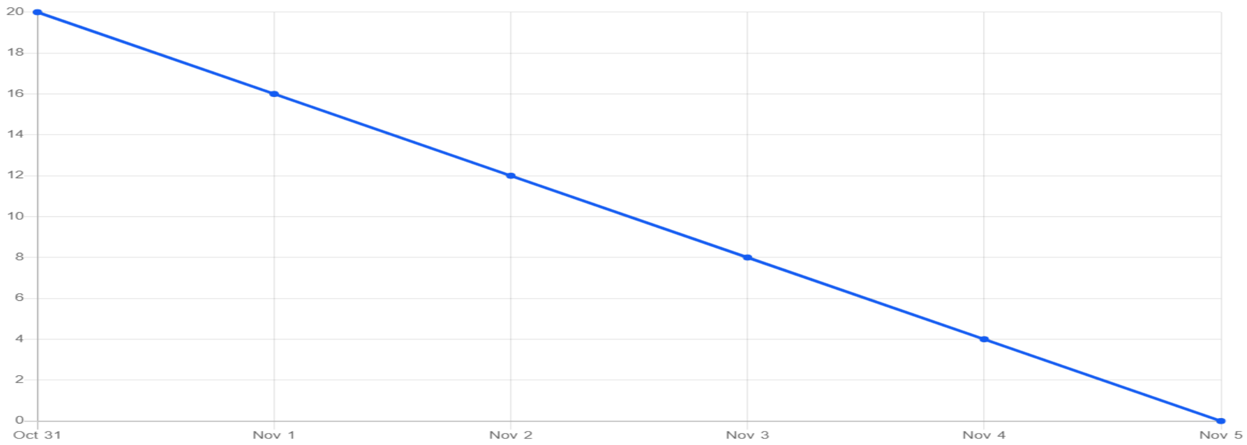
IBSCPSFA-4 Instant notification

IBSCPSFA-6 Web Application

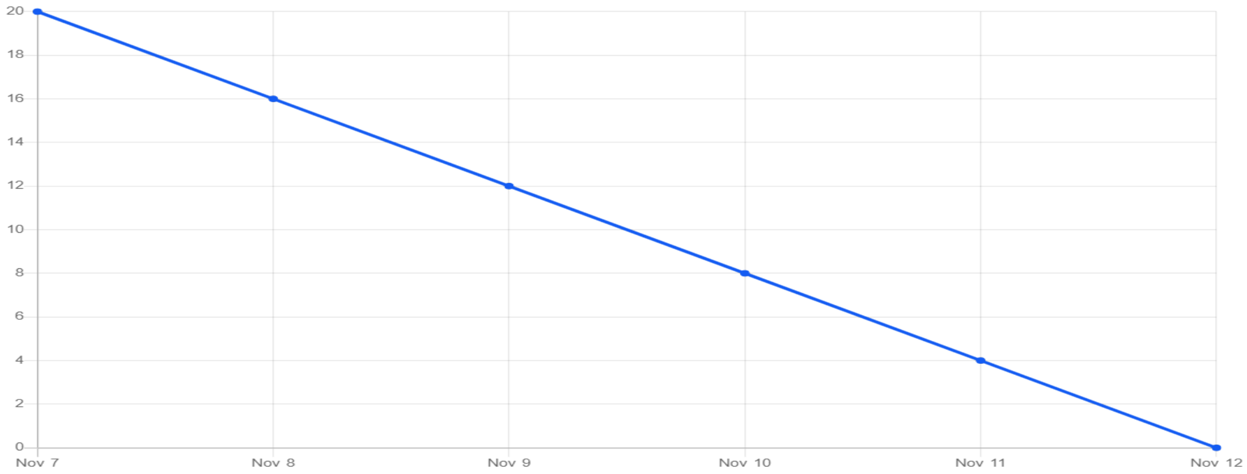
SPRINT - 1



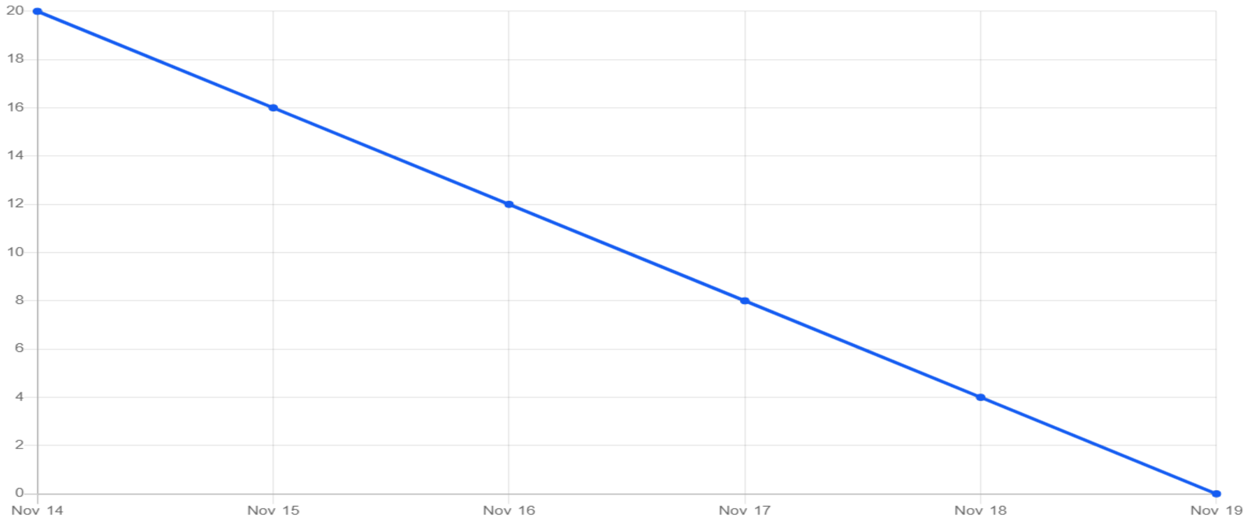
SPRINT - 2



SPRINT - 3



SPRINT - 4



7. CODING & SOLUTIONING

7.1 IoT SIMULATOR

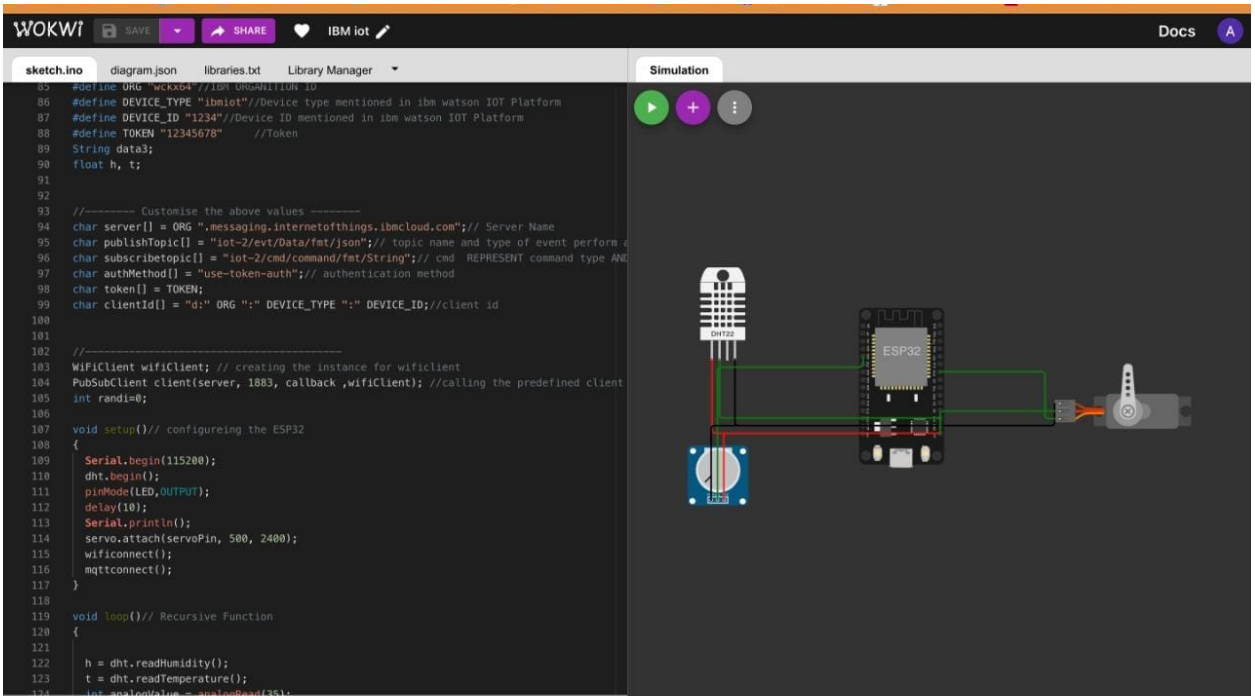
In our project instead of sensors we used IoT sensor simulator which gives random readings to the connected cloud. We have used temperature sensor-

DHT22, pH sensor and servo motor for motor control.

In case of any threshold mismatches, the motor needs to be turned on. Hence, in the simulation the is turned at an angle of 180 degree which indicates that the motor is in on condition.

The final results can be viewed in the IBM IoT Watson platform.

The link to simulator: https://wokwi.com/projects/348588724291895890



```
#define ORG "wckx64"//IBM ORGANITION ID
#define DEVICE_TYPE "ibmiot"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "1234"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //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 a
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND
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
int randi=0;

void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  servo.attach(servoPin, 500, 2400);
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  int analogValue = analogRead(35);
```

7.2 Connecting IOT simulator to IBM watson IOT platform

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

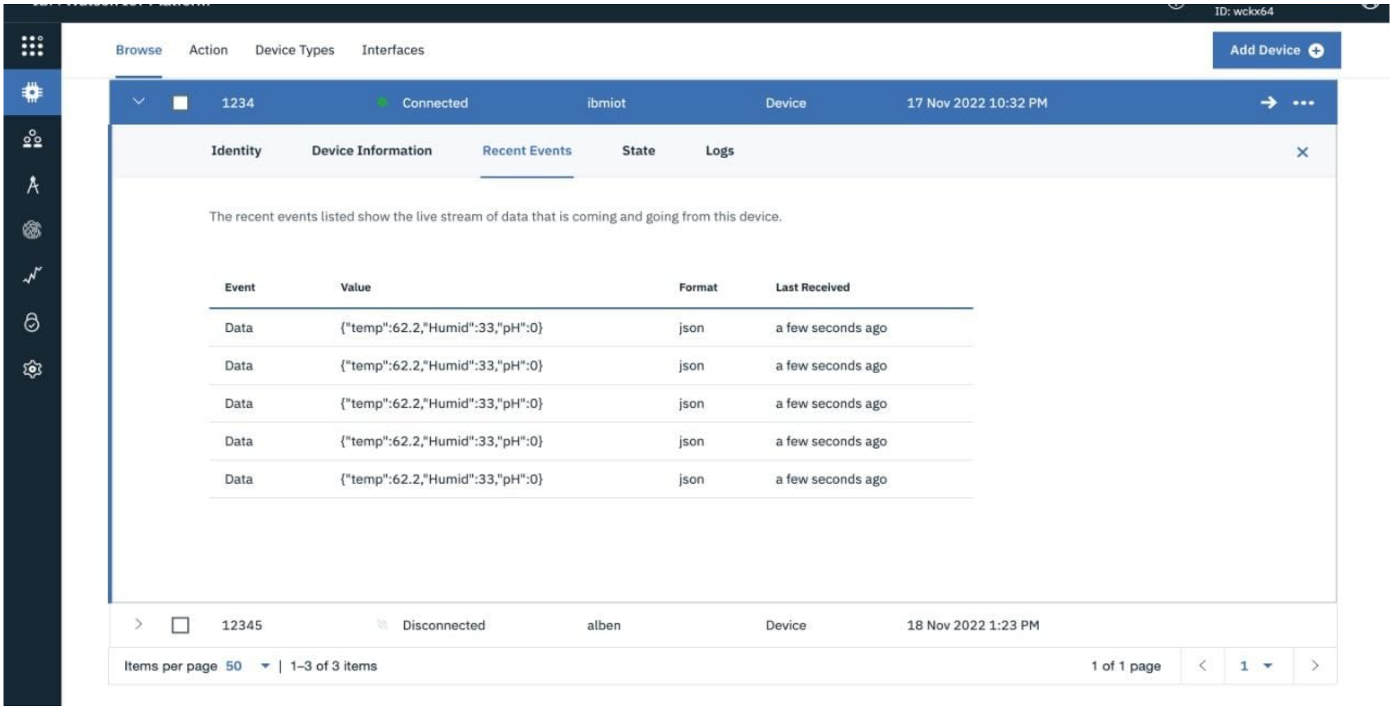
My credentials given to simulator are:

ORG **"wckx64"**

DEVICE_TYPE "**ibmiot**"

DEVICE_ID **"1234"**

TOKEN **"12345678"**



1234	Connected	ibmiot	Device	17 Nov 2022 10:32 PM

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

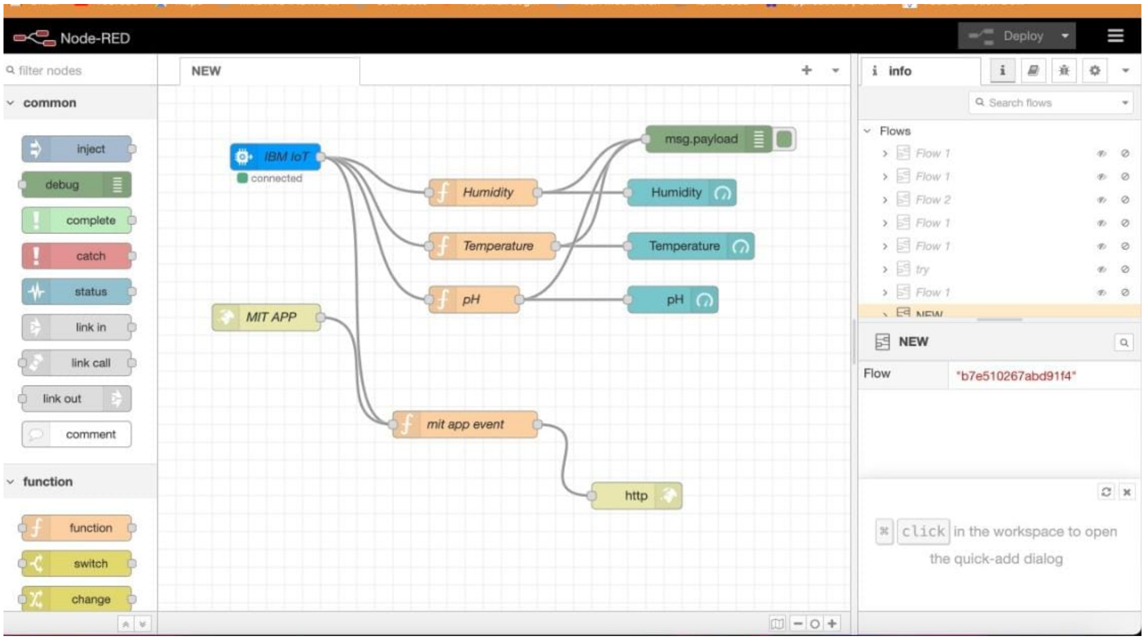
Event	Value	Format	Last Received
Data	{"temp":62.2,"Humid":33,"pH":0}	json	a few seconds ago
Data	{"temp":62.2,"Humid":33,"pH":0}	json	a few seconds ago
Data	{"temp":62.2,"Humid":33,"pH":0}	json	a few seconds ago
Data	{"temp":62.2,"Humid":33,"pH":0}	json	a few seconds ago
Data	{"temp":62.2,"Humid":33,"pH":0}	json	a few seconds ago

12345	Disconnected	alben	Device	18 Nov 2022 1:23 PM

Items per page 50 | 1–3 of 3 items 1 of 1 page 1

7.3 Configuration of Node-Red to collect IBM cloud data

Each values obtained from Iot watson platform is considered as a single node. Then the appropriate device credentials obtained earlier are entered into the node to connect and fetch device telemetry to Node-Red.



Flow "b7e510267abd91f4"

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

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

```
msg.payload=msg.payload.d.temperature
return msg
msg.payload=msg.payload.d.humidity
return msg
msg.payload=msg.payload.d.pH
return msg
```

Finally connect Gauge nodes from dashboard to see the data in UI.

8. TESTING

8.1 TEST CASES

Parameters: Depending on temperature, humidity and pH values the motor can be controlled by the user.

- **Test case 1:** Temperature below 50F: Motor in OFF state
  Humidity level below 70%: Motor in OFF state
  pH level between 5.5 to 7.5: Motor in OFF state
  Soil moisture level below 60%: Motor in OFF state
- **Test case 2:**
  Temperature below 60F: Motor in ON state
  pH level below 5.5 and above 7.5: Motor in ON state
  Soil moisture level above 60% :Motor in ON state

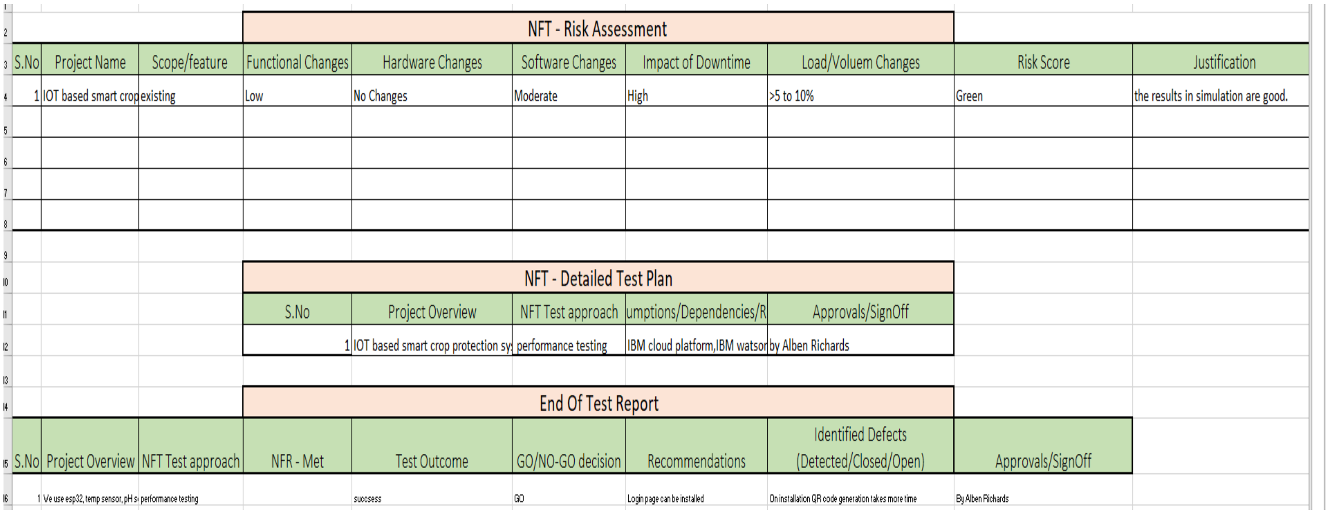
8.2 USER ACCEPTANCE TESTING

Users are able to see the login/sign up credentials and when the parameter values exceed or go below a threshold , a pop-up message is shown.

Providing control over periodic watering of crops , supervision of soil moisture content and measures the temperature and humidity values to compare it with a threshold. This design is used to make high productivity and can earn high profit.

9. RESULTS

9.1 PERFORMANCE METRICS



NFT - Risk Assessment

S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score	Justification
1	IOT based smart crop	existing	Low	No Changes	Moderate	High	>5 to 10%	Green	the results in simulation are good.

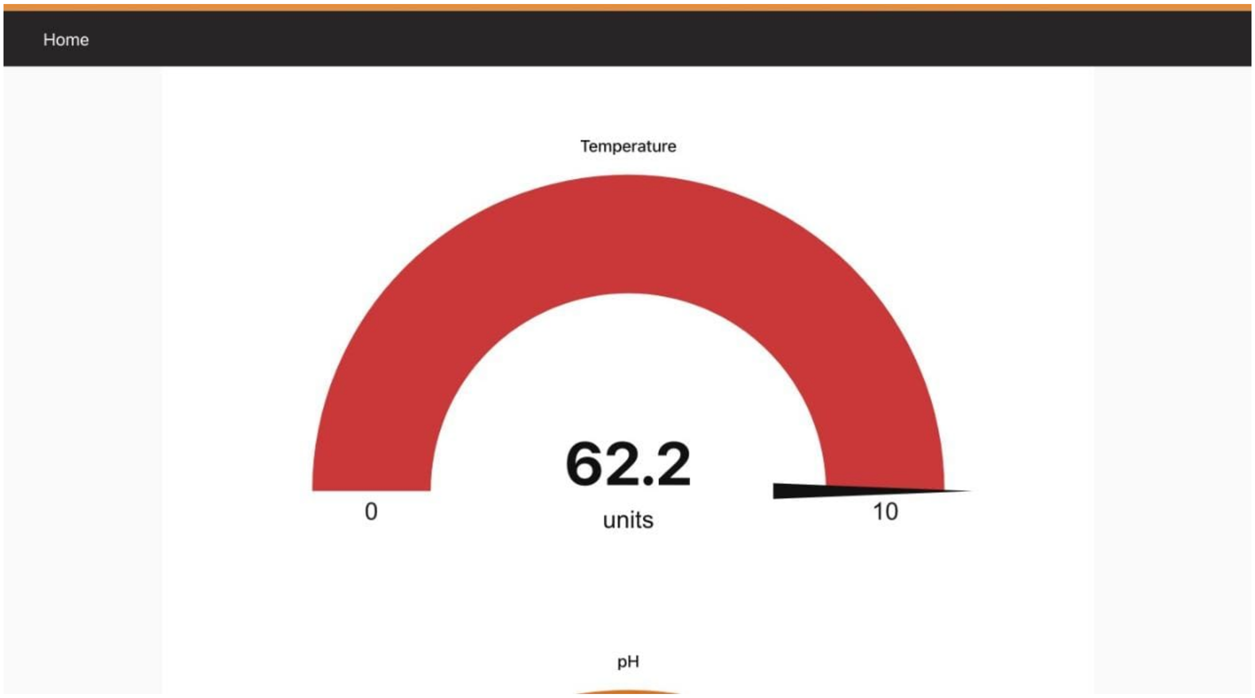
NFT - Detailed Test Plan

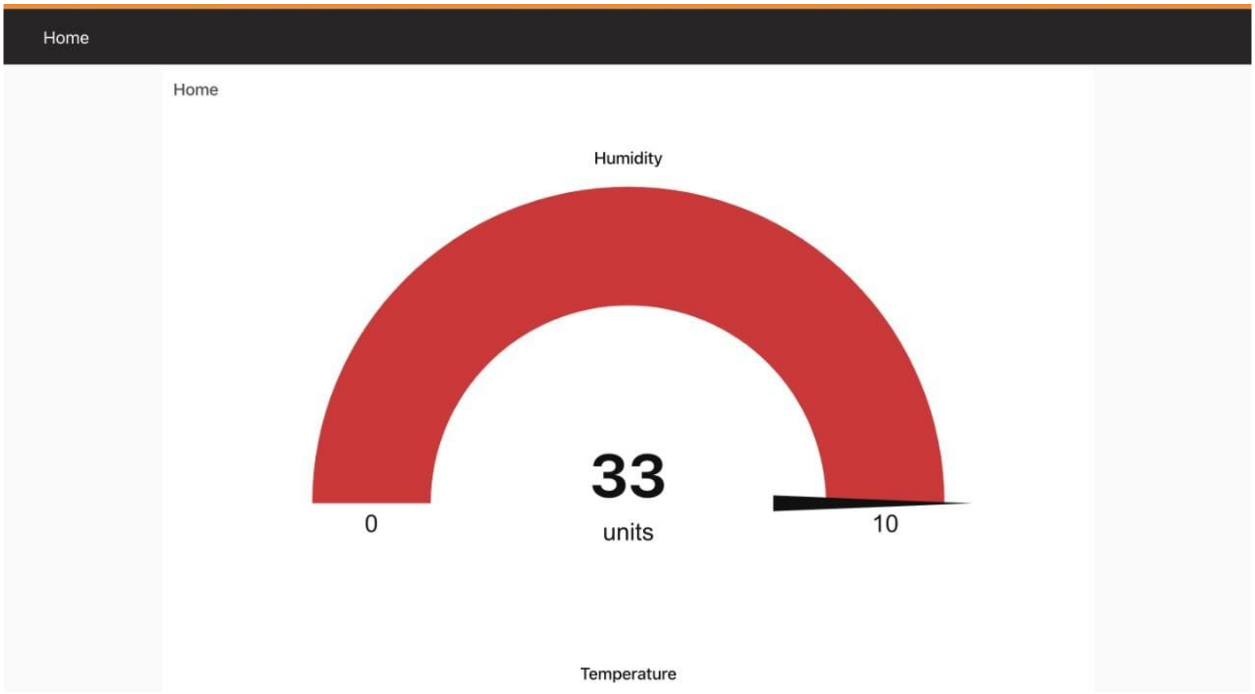
S.No	Project Overview	NFT Test approach	umptions/Dependencies/R	Approvals/SignOff
1	IOT based smart crop protection sy	performance testing	IBM cloud platform,IBM watsor	by Alben Richards

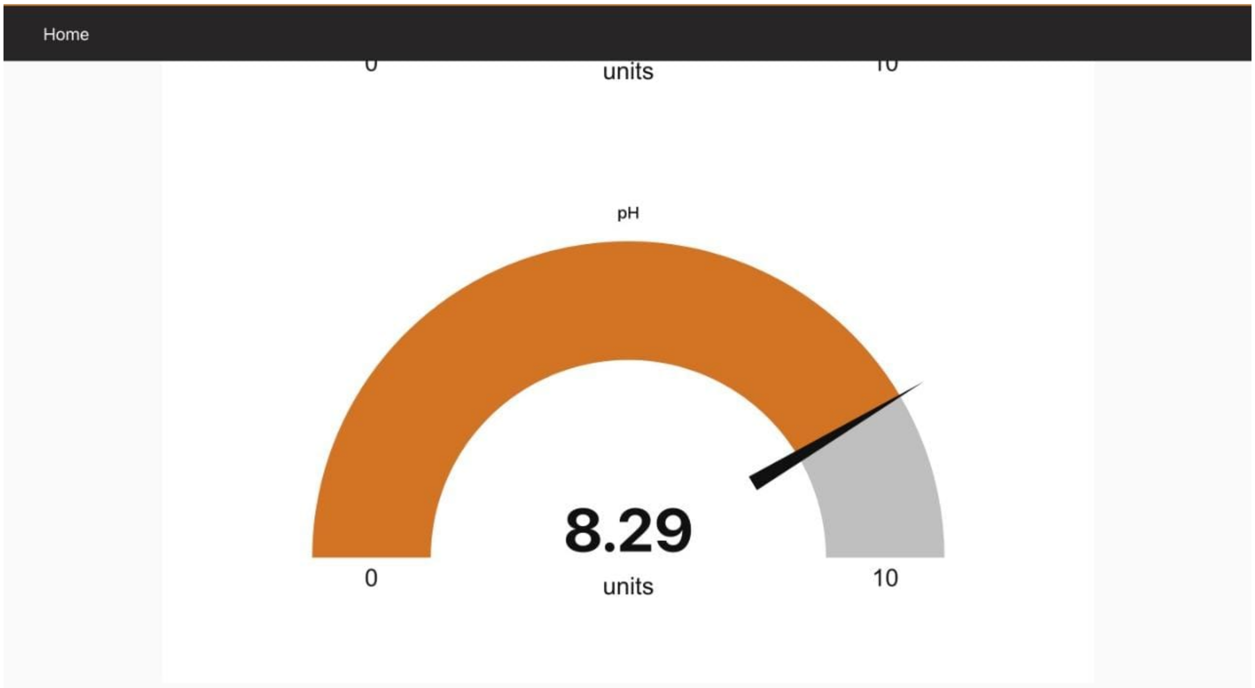
End Of Test Report

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff
1	We use esp32, temp sensor, pH s	performance testing		succsess	GO	Login page can be installed	On installation QR code generation takes more time	By Alben Richards

9.2 WEB APP UI

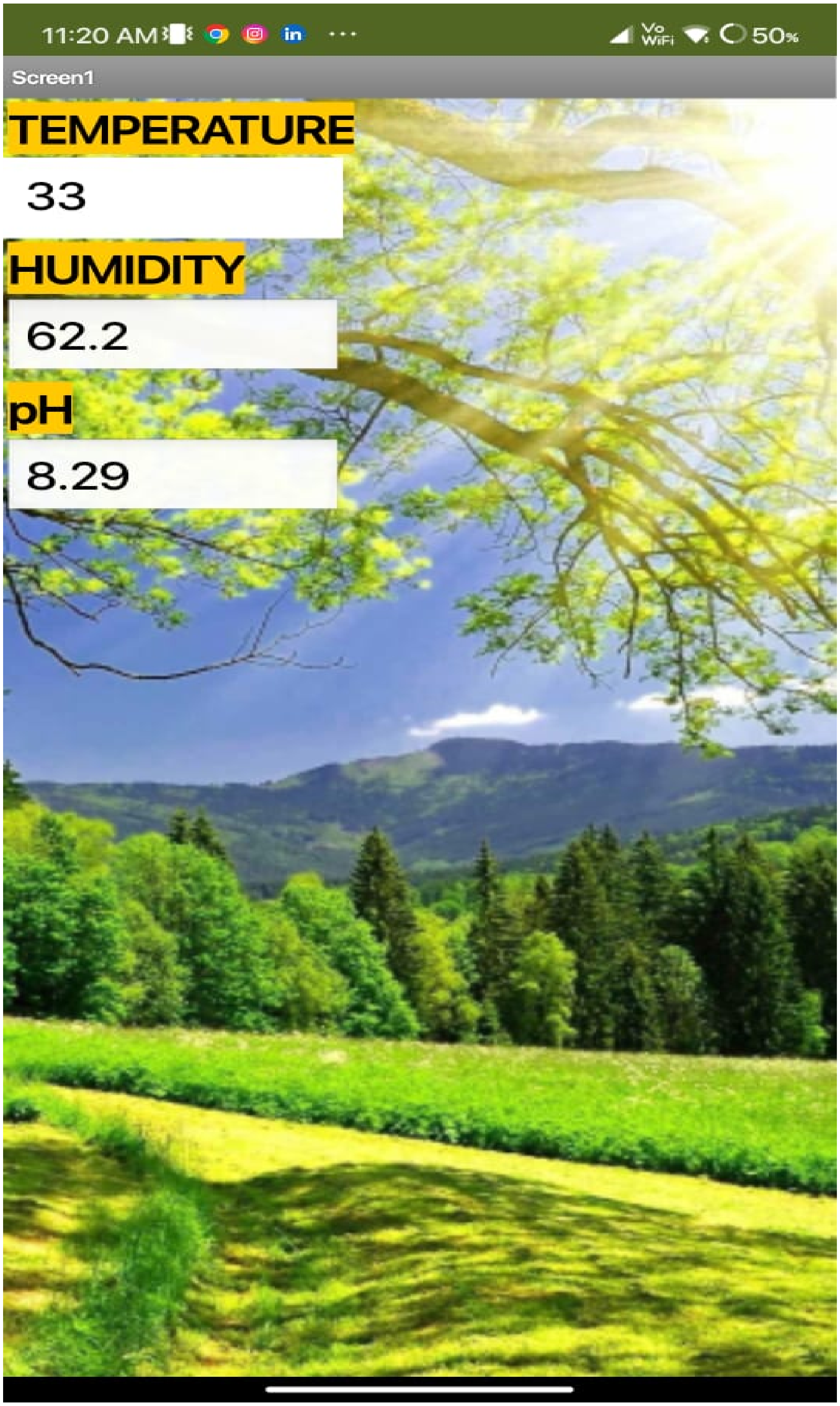

Home
Temperature
62.2
units
0
10
pH


Home
Home
Humidity
33
units
0
10
Temperature



9.3 MIT APP INVENTOR

LINK: http://169.51.200.193:32631/ui/#!/0?socketid=t9TCMsqPDED2o63SAACT



TEMPERATURE

33

HUMIDITY

62.2

pH

8.29

10. ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Increased agility of the processes.Farmers can quickly respond to any significant change in weather, humidity and wild animals attack.
- Increased productivity.
- Reduced financial losses.
- 24/7 crop monitoring and protecting it from wild animals is made easy and completely automated.
- Reduced man power requirement.
- Provides path to meet increased food demand in our country.
- It also prevents the crop being damaged due to heavy climatic conditions.
- Prevents crop from insufficient watering and also watering the crops when there is no necessity.

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 equipments require farmers to understand and learn the use of technology. This is major challange in adopting smart agriculture farming at large scale across the countries.The cost of maintenance becomes high whether there is a repair or not.
- In the case of agriculture, most of the process is dependent on weather conditions. It is a natural phenomenon which in spite of the updated technology can become unpredictable.
- Since technology involves a lot of machines, there are chances for faulty data processing equipment or sensors then which may lead to incorrect actions leading to financial losses.
- Lack of infrastructure and lack of security also considered to be an important drawback.

11. CONCLUSION

Crop protection from animal intrusion and changing environmental conditions is important for the successful cultivation of the crops which can be achieved using IoT. The development of agricultural sector will always be a priority especially given the dynamics of the world today. This testing phases of the project justifies that this project can be used in a real time farming environment. Also, the project was developed after studying the market requirement which makes it extremely suitable in the context of present scenarios. The post survey result provides that the system is useful in real time scenario and end users are interested in using this system. Therefore, using IoT in agriculture has a big promising future as a driving force of efficiency, sustainability, and scalability in this industry.

12. FUTURE SCOPE

The performance of the system can be further improved in term of the operating speed, memory capacity and instruction cycle period of the microcontroller by using another high-end controller. The number of channels can be increased to interface a greater number of sensors which is possible by using advanced versions of controllers.This device can be made to perform better by providing the power supply with the help of renewable sources. Time bound administration of fertilizer, insecticides and pesticides can be introduced. A water meter can be installed to estimate the amount of water used for irrigation and thus giving a cost estimation and a solenoid valve can be used for varying then volume of water flow. This project can be made based on Image processing in which wild animal and fire can be detected by cameras and if it comes towards farm then system will be directly activated through wireless networks. Wild animals can also be detected ng by usiwireless networks such as laser wireless sensors and by sensing this laser or sensor’s security system will be activated.

13. APPENDIX

13.1 source code

```
/*#include "DHTesp.h"
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 35     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2

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 "wckx64"//IBM ORGANITION ID
#define DEVICE_TYPE "ibmiot"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "1234"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //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/command/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);


const int DHT_PIN = 15;


DHTesp dhtSensor;

float floatMap(float x, float in_min, float in_max, float out_min, float out_max) {

  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;

}




void setup() {

  Serial.begin(115200);

  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);

}


void loop() {
  int analogValue = analogRead(35);

  float voltage = floatMap(analogValue, 0, 4095, 0, 14);

  Serial.print(" ph value ");

  Serial.println(voltage);

  delay(1000);

  TempAndHumidity  data = dhtSensor.getTempAndHumidity();

  Serial.println("Temp: " + String(data.temperature, 2) + "°C");

  Serial.println("Humidity: " + String(data.humidity, 1) + "%");

  Serial.println("---");

  delay(1000);

}*/

#include <WiFi.h>//library for wifi

#include <PubSubClient.h>//library for MQtt

#include "DHT.h"// Library for dht11

#include<stdlib.h>

#define DHTPIN 15     // what pin we're connected to

#define DHTTYPE DHT22   // define type of sensor DHT 11

#define LED 2

float floatMap(float x, float in_min, float in_max, float out_min, float out_max) {

  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
```

```
}
#include <ESP32Servo.h>

const int servoPin = 18;
int pos = 0;

Servo servo;

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 "wckx64"//IBM ORGANITION ID
#define DEVICE_TYPE "ibmiot"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "1234"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //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/command/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
int randi=0;


void setup()// configureing the ESP32
```

```
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  servo.attach(servoPin, 500, 2400);
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  int analogValue = analogRead(35);
  float voltage = floatMap(analogValue, 0, 4095, 0, 14);
  delay(1000);
  Serial.print("temp:");
  Serial.println(t);
  Serial.print("Humid:");
  Serial.println(h);
  Serial.print(" ph value ");
  Serial.println(voltage);
   randi = rand()%10 ;
  Serial.println(randi);
  PublishData(t,h,voltage);
  if(t>40 && randi >60)
  {
    servo.write(180);
  }
  else{
    servo.write(0);
  }
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}
```

```
/....................................retrieving to Cloud............................../

void PublishData(float temp,float humid,float ph)
 {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */

 String payload = "{\"temp\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
   payload += "," "\"pH\":";
  payload += ph;

  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 defination for wificonnect
{
  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 GIT HUB PROJECT AND DEMO LINK

Wowki :- https://wokwi.com/projects/348588724291895890

Node Red :- http://169.51.200.193:32631/ui/#!/0?socketid=TIB-n3ao90veUz4dAACV

Mit App :- http://ai2.appinventor.mit.edu/#4605951492358144