

# **Project Report Documentation**

**Team ID:PNT2022TMID45173**

## **1. INTRODUCTION**

### **1.1 Project Overview**

The title of our project is “**IOT based Smart Crop Protection**”. The Overview of our Project is to Safeguard the farm from climatic changes like soil erosion, landslide, and birds, animals etc.,.So, that we are making a IOT based project to protect the farm from climatic changes and haunting animals. For that we are just making a cloud based project and placing IOT based sensor. over which it will produce sounds and notification and provide results on IOT- MIT app. From which we can protect our farm. And it will provide better yield for us.

### **1.2 Purpose**

- ⇒ The way to protect the farm from climatic changes animals, birds , pests.
- ⇒ And to make the crop to grow better and provide better yield.

## **2. LITERATURE SURVEY**

### **2.1 Existing problem**

- ⇒ “Food” is the important thing, which is needed for everyone to survive in this world. For that farmers are doing their own part in an effective manner, during which they have to face some problems such as:
- ⇒ There are increasing pressures from climate change, soil erosion and biodiversity loss and from consumers’ changing tastes in food and concerns about how it is produced.
- ⇒ And the natural world that farming works with – plants, pests and diseases – continue to pose their own challenges beyond that, they have to ⇒ Stay resilient against global economic factors.
- ⇒ Inspire young people to stay in rural areas and become future farmers

- ⇒ The effects of climate change affect farmers' ability to grow the food we all need.
- Increasingly volatile weather and more extreme events – like floods and droughts – change growing seasons, limit the availability of water, allow weeds, pests and fungi to thrive, and can reduce crop productivity.

## 2.2 **References**

- ⇒ [https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20\(1\).pdf](https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service%20s%20(1).pdf)
- ⇒ [https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20\(1\).pdf](https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf)
- ⇒ <https://openweathermap.org/>
- ⇒ <https://smartinternz.com/assets/docs/Sending%20Http%20request%20to%20Open%20weather%20map%20web site%20to%20get%20the%20weather%20forecast.pdf>
- ⇒ <https://www.youtube.com/watch?v=cicTw4SEdxk>
- ⇒ [https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20\(1\).pdf](https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf)
- ⇒ <https://github.com/rachuriharish23/ibmsubscribe>

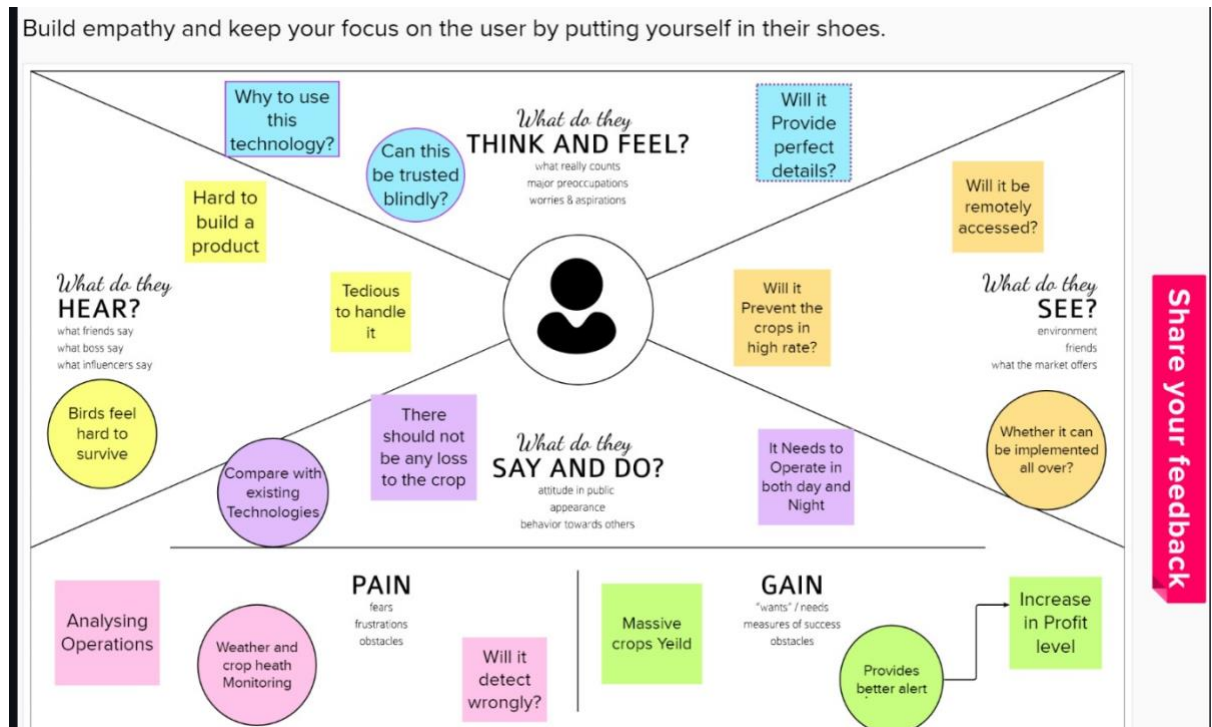
## 2.3 **Problem Statement Definition**

- ⇒ Agriculture is one of the Area which required urgent attention and advancement for high yield and efficient utilization of resources.
- ⇒ In this paper an approach of smart crop monitoring is presented through Internet of things (IOT).
- ⇒ A 4 Level framework is proposed namely sensing devices, sensor data level, base station level, edge computing and cloud data level for smart crop monitoring.
- ⇒ In this project, farm is going to get protected from humidity, temperature, and animals. With the help of IOT cloud module.

- ⇒ The agricultural form is been monitored with the help of MIT app and then, the data will be collected and stored it in cloud.
- ⇒ It will monitor and sense the humidity level and movement of animals and will send the message as notification to the user.

### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas



#### 3.2 Ideation & Brainstorming

##### What do they think and feel?

As its name may imply, smart farming is the use of technology in animal agriculture, and it's something that's been around since the Industrial Revolution. The biggest difference between then and now, though? "Motorized devices are being replaced with IOT".

### **What do they hear?**

Smart farming is about using the new technologies which have arisen at the dawn of the Fourth Industrial Revolution in the areas of agriculture and cattle production to increase production quantity and quality, by making maximum use of resources and minimizing the environmental impact.

### **What do they see?**

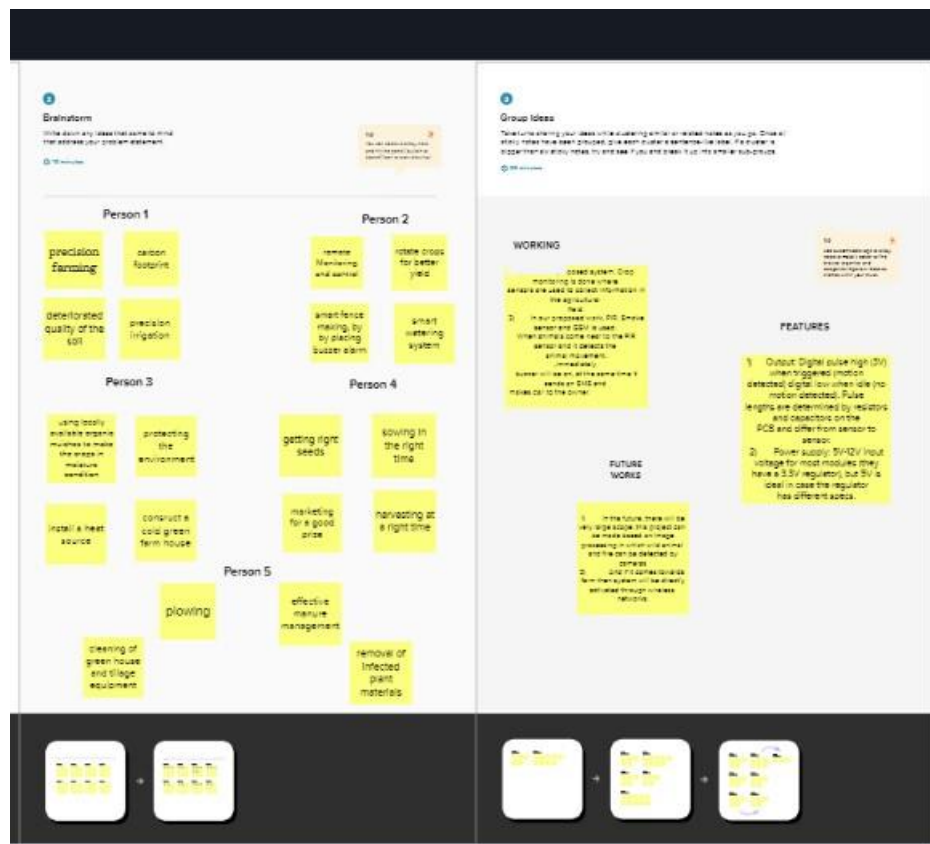
Smart farming is a management concept focused on providing the agricultural industry with the infrastructure to leverage advanced technology – including big data, the cloud and the internet of things (IoT) – for tracking, monitoring, automating and analyzing operations.

### **What do they say and do?**

- ⇒ The aim of this technology is to **make the most of all the data** collected by various tools, by converting them into real **sources of information** in order to then define ways of simplifying agricultural work. It also allows for **accurate and predictive analysis** of all situations that may affect the farms, such as weather conditions (temperature, humidity, etc.) and sanitary or economic situations, for example. This makes it easier to organize the supply of energy, water, livestock feed and fertilizer.
- ⇒ In its most advanced form, smart farming facilitates the exchange of information between different farms, creating a real network of connected farms accessible from a smartphone or a computer.

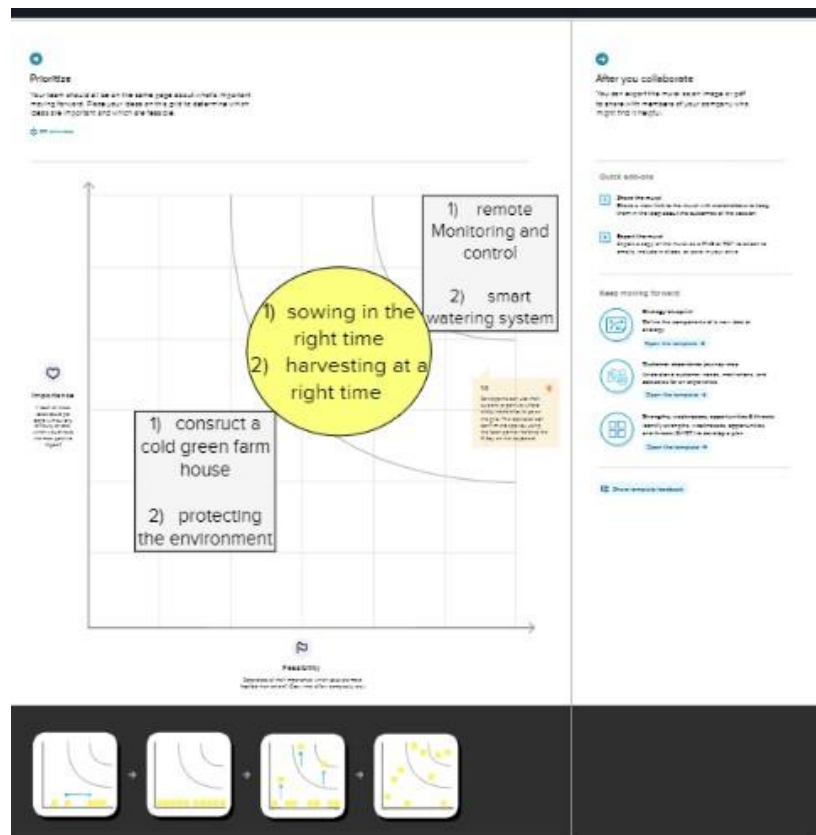
### **BRAINSTORM:**

## BRAINSTORM



## PRIORITIZATON:

## PRIORITIZE :



### 3.3 **Proposed Solution**

#### **Proposed Solution Template:**

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Crop protection from animals and pests, for better yield of crops.
2.	Idea / Solution description	Making a fence, which helps in identifying the animals and pests, by buzzer alarm.
3.	Novelty / Uniqueness	Improve productivity, crop variety improvement, crop protection management.
4.	Social Impact / Customer Satisfaction	good export of products, high profit, increase in brand loyalty.
5.	Business Model (Revenue Model)	high production, livestock and crops, direct sales, and advertising.
6.	Scalability of the Solution	by incorporating integrated pests and insect management, by creating smart fence with buzzer alarm, and irrigation.

### 3.4 Problem Solution fit

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> Who is your customer? i.e. working parents of 0-5 y.o. kids  Farmers  <b>CS:</b> It is the process, by which we divide our customers into segments.	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices  Budget  <b>CC:</b> The average visit of duration or the last date of visit of customers.	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> 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? i.e. pen and paper is an alternative to digital notetaking  Smart fence - buzzer alarm  <b>AS:</b> Today, the recent technology which helps in crop protection is IOT based fence, whereas in past days it was not implemented, because of which, lot many crops get destroyed, by birds and animals.	Explore AS, differentiate	
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.  Creation of smart fence  <b>J&amp;P:</b> The smart fence is the one in which, its designed in a way, so that it can make a caution alarm to the nearing or attacking preys and prevent the crop from damages.	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span> What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations.  For food haunting  <b>RC:</b> Due to the haunt made by birds and animals for their survival, the yields of crop protection gets lost / down. so, in order to overcome form this, we have to jump into this latest technology called smart crop protection system.	<b>7. BEHAVIOUR</b> <span>BE</span> What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)  Better yield  <b>BE:</b> 1) In order to protect the crops from pests, birds and animals, the one ad only thing is making or installing a smart crop protection system with the help of IOT. 2) Farmer's should do proper maintenance.		Focus on J&P, tap into BE, understand RC
	<b>3. TRIGGERS</b> <span>TR</span> What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.  <b>TR:</b> On by seeing and hearing Neighbour's works made in their farm.	<b>10. YOUR SOLUTION</b> <span>SL</span> 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. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.  <b>SL:</b> The perfect output of my problems solution is designing a IOT based fence project, which is normally made up of steel fence, inside which its been incorporated with the help of buzzer alarm, to monitor and predict the crops from animals.	<b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span> <b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7 <b>CH:</b> A quick response to queries.  <b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.  Brings announcement and community discussions.		
<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design. <b>EM:</b> BEFORE: Insecure, heavy loss AFTER: proud of their work, happy					

## 4 REQUIREMENT ANALYSIS



## 4.1 Functional requirement

### 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 Form Registration through Gmail Registration through Linked IN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Certification Requirements	Regulation Rules Profession wide.
FR-4	Authorization	Healthcare provider User group.
FR-5	Business rules	Decision making Marketing.
FR-6	External interfaces	Wide Area Network (WAN) Screen layouts.

## 4.2 Non-Functional requirements

**Non-functional Requirements:**

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

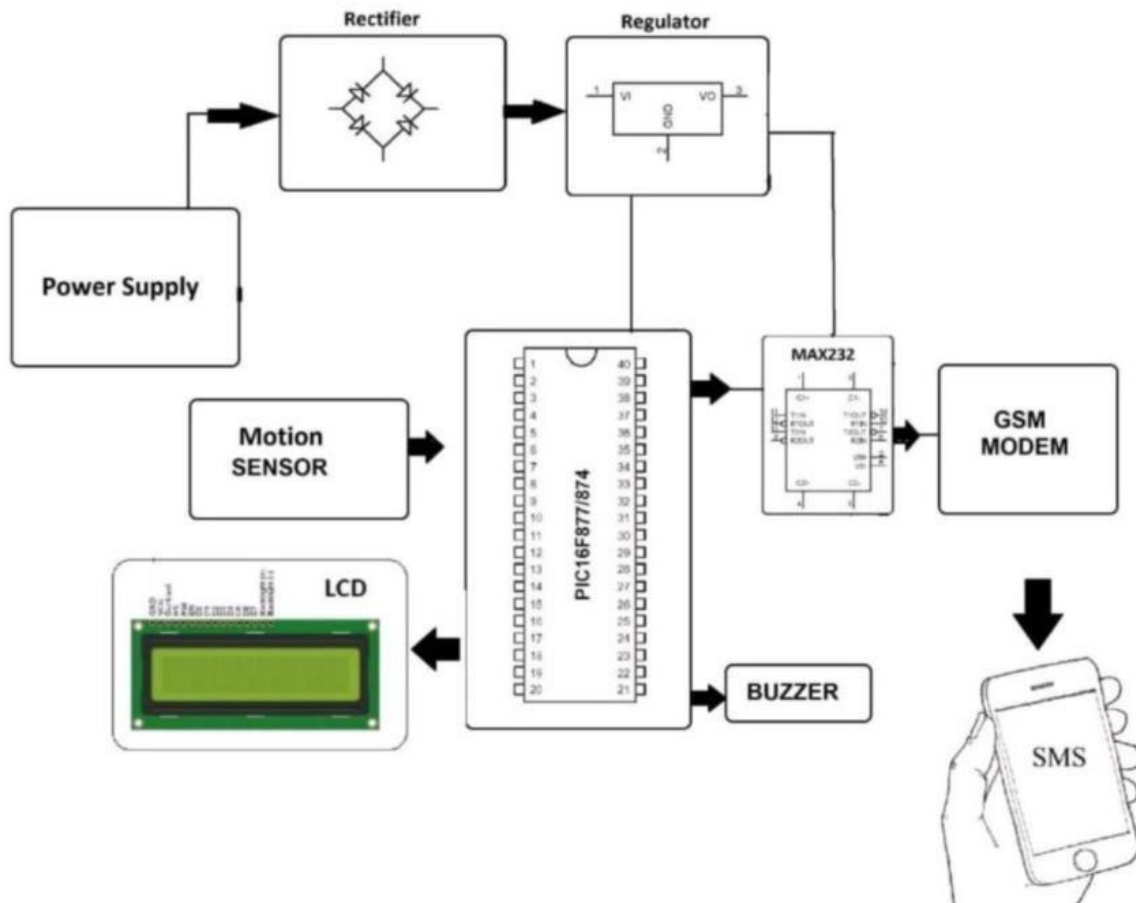
FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	Use of fertilizers , Irrigation and scheduled planting operation.
NFR-2	<b>Security</b>	Crops could be protected from these diseases using pesticides and biocontrol agents.
NFR-3	<b>Reliability</b>	Reducing deforestation , conserving natural resources and curbing soil erosion.
NFR-4	<b>Performance</b>	Agricultural productivity depends on the quality of seeds with which farmers sow their

		fields.
NFR-5	<b>Availability</b>	Farming methods requires growers appropriate plant protection strategy and training.
NFR-6	<b>Scalability</b>	Application of sensors and automated irrigation practices can help monitor agricultural land.

## 5. PROJECT DESIGN

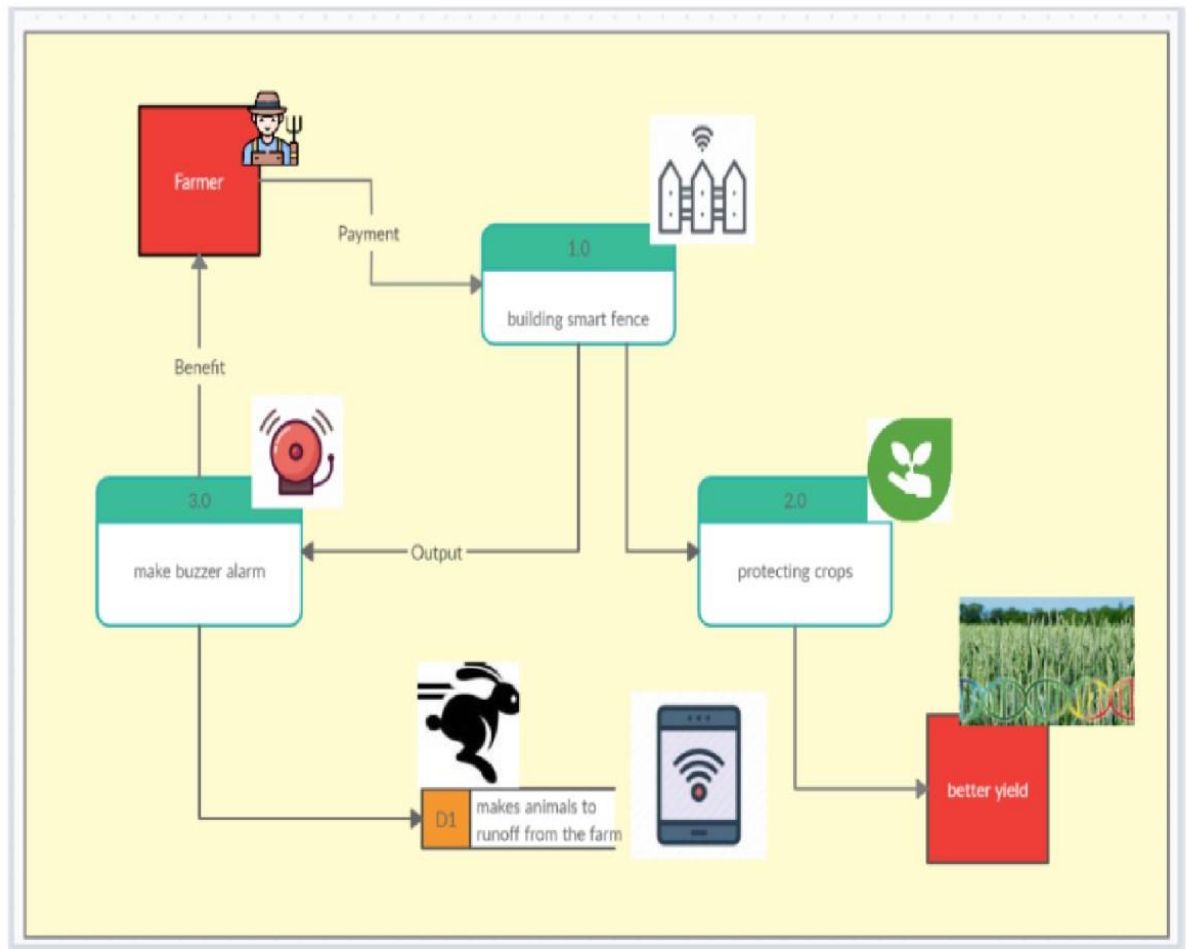
## 5.1 Data Flow Diagrams

Flow diagram of Smart Crop Protection in IOT:

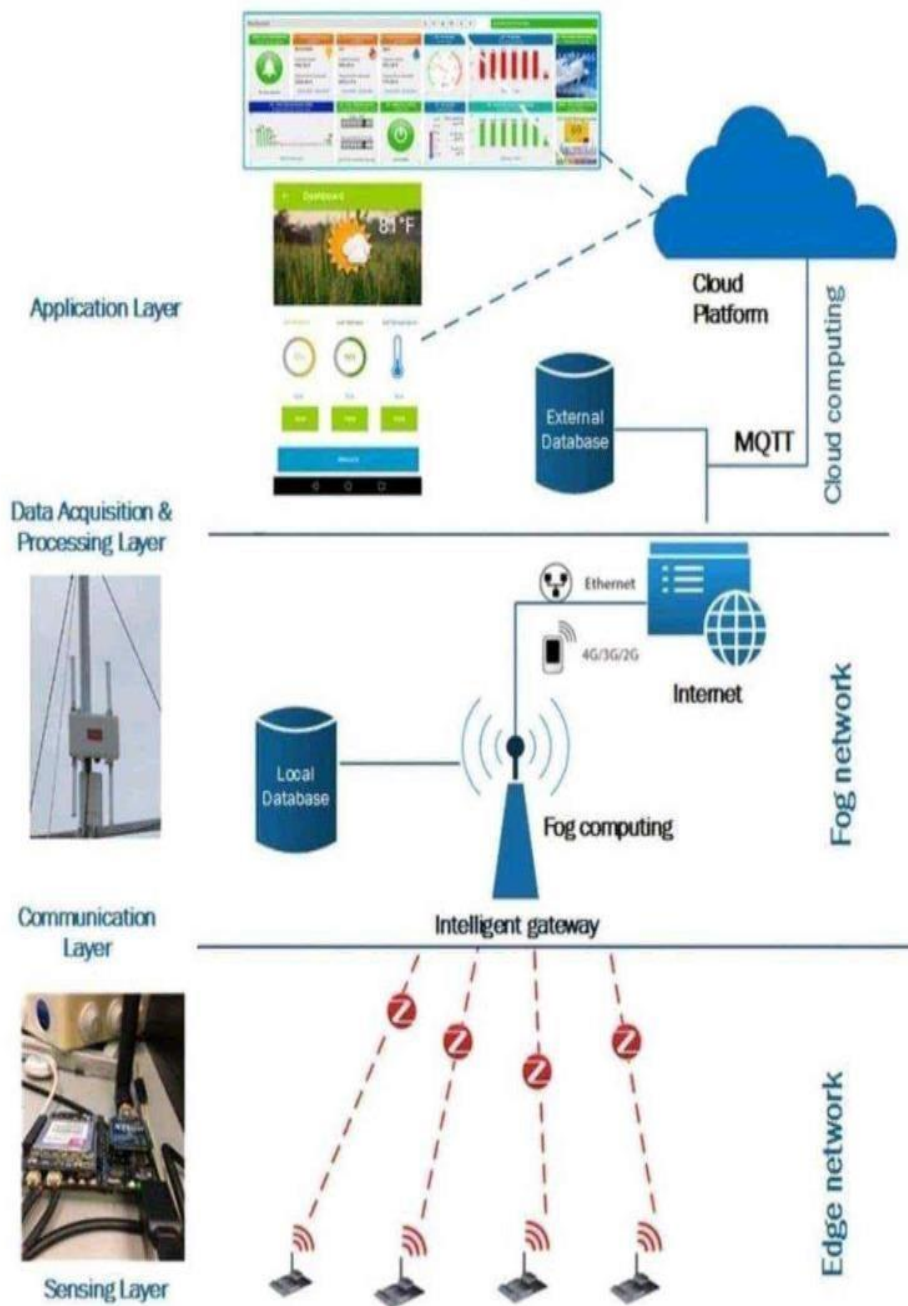


## 5.2 Solution & Technical Architecture :

## SOLUTION ARCHITECTURE:



## TECHNICAL ARCHITECTURE



### 5.3 User Stories

**USER STORIES:**

User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance criteria	priority
Customer (Mobile user)	Download the database	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
	Register	USN-2	As a user I can register for the application by entering my email, password and confirming my password.	I can receive confirmation email and click confirm	High
	Login	USN-3	As a user I will receive confirmation email once I have registered for the application.	I can register and access the dashboard with Facebook login	Low
	Upload the image	USN-4	As a user I must upload the image to identify the problem and works on it.		Medium
Customer (Web user)	The functional requirements are same as	Same as mobile user	Same as mobile user.	Same as mobile user	High when compare

## **6. PROJECT PLANNING & SCHEDULING**

S.NO	ACTIVITY TITLE	ACTIVITY DESCRIPTION	DURATION
s 1	<b>Understanding the Project Requirement</b>	1)Assign the team members and create repository in the GitHub. 2)Assign the task to each Members and teach how to use and access the GitHub and	1 week

<b>2</b>	<b>Starting of Project</b>	1)Advice students to attend classes of IBM portal. 2)Create and develop a rough diagram based on project description. 3)Team leader assign task to each member of the project.	1 week
<b>3</b>	<b>Attend Classes</b>	Team members and team lead must watch and learn from classes provided by IBM and Nalaiya Thiran.	3 weeks
<b>4</b>	<b>Prerequisites</b>	1) Create an account in clarifai. 2)Register in IBM Cloud Services. 3)Software installed	1 week

<b>5</b>	<b>Task Assigned</b>	1)Develop the python Script. 2)Develop a web Application Using Node RED Service. 3)Ideation Phase 4)Project Design Phase-I. 5)Project Design Phase-II. 6)Project Planning Phase. 7)Project Development Phase.	5 weeks
<b>6</b>	<b>Scope Of the Project</b>	It helps the farmers grow more food on less land by protecting crops from pests, diseases and weeds as well as raising productivity per hectare.	1 week

## 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points (40)	Priority (Low to High)	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the required dataset by entering my email, password, and confirming my password.	3	High	A Angalaparameshwari
Sprint-1		USN-2	As a user, I will receive confirmation email and the SMS once I have registered for the application	2	High	A Angalaparameshwari
Sprint-2	Cloud services	USN-3	As a user, I can register for the application through Facebook or any social media	1	Low	M Vanathi
Sprint-4		USN-4	As a user, I can register for the application through Gmail/web service	2	Medium	M Shiyam
Sprint-3	Login	USN-5	As a user, I can log into the application network by entering email & password	4	High	S Sudhakar
Sprint-2	Pre processing	USN-6	As a farmer, the user must be able to find the system easy to access so pre-processes and other task must be perfect.	3	High	M Vanathi
Sprint-1	Collecting Dataset	USN-7	To collect various sources of animal threats and keep developing a dataset.	3	Medium	A Angalaparameshwari
Sprint-4	Integrating	USN-8	To integrate the available dataset and keep improving the accuracy of finding animals	2	High	S Sudhakar
Sprint-3		USN-9	To find and use appropriate compiler to run and test the data so that we can implement our program	1	Low	M Shiyam
Sprint-2		USN-10	Request Saveetha Engineering College to deploy the project in our campus and test	1	Low	M Vanathi
Sprint-1	Training	USN-11	As programmer, we need to train our data perfectly so that the program runs smoothly	3	High	A Angalaparameshwari
Sprint-3		USN-12	Train the data using out available services and IBM dataset from server and improve that	2	Medium	S Sudhakar
Sprint-4	Coding	USN-13	To modify the code according to our program and improve the efficiency of that code	4	High	M Vanathi
Sprint-2		USN-13	To improve performance	1	Low	M Shiyam
Sprint-2	Record	USN-5	To record the data and plot the graph to show the characteristics officially	4	High	A Angalaparameshwari
Sprint-1	Planning	USN-4	Plan the programming language and feasibility	3	Medium	M Vanathi



## 6.2 Sprint Delivery Schedule

SPRINT	TOTAL STORY POINTS	DURATION (DAYS)	Sprint Start Date	SPRINT END DATE (PLANNED)	STORY POINTS COMPLETED	SPRINT RELEASE DATE
Sprint 1	20	6	24 Oct 2022	29 Oct 2022	25	29 Oct 2022
Sprint 2	20	6	31 Oct 2022	05 Nov 2022 2	15	30 Oct 2022
Sprint 3	20	6	7 Nov 2022	12 NOV 2022	14	6 NOV 2022
Sprint 4	20	6	14 NOV 2022	19 NOV 2022	20	7 Nov 2022

Velocity:

AV for sprint 1= Sprint Duration /velocity =20/6=3.3

AV for sprint 2= Sprint Duration/Velocity=6/6=3.3

AV for Sprint 3=Sprint Duration/Velocity=6/6=3.3

AV for Sprint 4=Sprint Duration/Velocity=6/6=3.3

### BURNDOWN CHART:

## Burndown Chart



### 6.3 Reports from JIRA

[illegible]

Your work

Projects

Filters

Dashboards

More

+

Search

Projects / IOT based crop protection system for agriculture

APICS board

GROUP BY

None

IN PROGRESS

IN REVIEW

DONE 5 ISSUES

+ Create issue

As a user , I can register for the application by entering my email, password ,and confirming my password

REGISTRATION

APICS-1

As a user , I can register for the application through Facebook

REGISTRATION

APICS-3

As a user , I can register for the application through Gmail

REGISTRATION

APICS-4

As a user , I will receive confirmation email once I have registered for the application

REGISTRATION

APICS-2

As a user , I can log into the application by entering email & password

REGISTRATION

APICS-5

## 7. CODING & SOLUTIONING

### 7.1 Feature 1

```
1 Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep 5 2022, 14:08:36) [MSC v.1933 64 bit (AMD64)] on win32
2 Type "help", "copyright", "credits" or "license()" for more information.
3 import cv2
4 import numpy as np
5 import wiot.sdk.device
6 import playsound
7 import random
8 import time
9 import datetime
10 import ibm_boto3
11 from ibm_botocore.client import Config, ClientError
12
13 #CloudantDB
14 from cloudant.client import Cloudant
15 from cloudant.error import CloudantException
16 from cloudant.result import Result, ResultByKey
17 from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
18 from clarifai_grpc.grpc.api import service_pb2_grpc
19 stub = service_pb2_grpc.V2Stub(ClarifaiChannel.get_grpc_channel())
20 from clarifai_grpc.grpc.api import service_pb2, resource_pb2
21 from clarifai_grpc.grpc.api.status import status_code_pb2
22
23 #This is how you authenticate
24 metadata = (('authorization', 'key 0620e202302b4508b90eab7efe7475e4'),)
25 COS_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
26 COS_API_KEY_ID = "g5d4q08EIgv4TMUCJj4hfzgalqEjrDbE82AJDWIAOho"
27 COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"
28 COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloud-object-storage:global:a/c2fa2836eaf3434bbc8b5b58fefff3f0:62e450fd-4c82-4153-ba41-ccb53adb8111::"
29 clientdb = cloudant(("apikey-W2njldwtj016V53LAVUCqPwc2aHTLmlj1xXvtdGKJ8n", "88cc5f47c1a28afbfb8ad16161583f5a", url="https://d6c89f97-cf91-48b7-b14b-c99b2fe27c2f-bluemix.clouda
30 clientdb.connect()
31
32 #Create resource
33 cos = ibm_boto3.resource("s3",
34                         ibm_api_key_id=COS_API_KEY_ID,
35                         ibm_service_instance_id=COS_RESOURCE_CRN,
36                         ibm_auth_endpoint=COS_AUTH_ENDPOINT,
37                         config=Config(signature_version="oauth"),
```

```

37         config=Config(signature_version="oauth"),
38         endpoint_url=COS_ENDPOINT
39     )
40     def multi_part_upload(bucket_name, item_name, file_path):
41         try:
42             print("Starting file transfer for {0} to bucket: {1}\n".format(item_name, bucket_name))
43             #set 5 MB chunks
44             part_size = 1024 * 1024 * 5
45             #set threshold to 15 MB
46             file_threshold = 1024 * 1024 * 15
47             #set the transfer threshold and chunk size
48             transfer_config = ibm_boto3.s3.transfer.TransferConfig(
49                 multipart_threshold=file_threshold,
50                 multipart_chunksize=part_size
51             )
52             #the upload_fileobj method will automatically execute a multi-part upload
53             #in 5 MB chunks size
54             with open(file_path, "rb") as file_data:
55                 cos.Object(bucket_name, item_name).upload_fileobj(
56                     Fileobj=file_data,
57                     Config=transfer_config
58                 )
59             print("Transfer for {0} Complete!\n".format(item_name))
60         except ClientError as be:
61             print("CLIENT ERROR: {0}\n".format(be))
62         except Exception as e:
63             print("Unable to complete multi-part upload: {0}".format(e))
64
65     def myCommandCallback(cmd):
66         print("Command received: %s" % cmd.data)
67         command=cmd.data['command']
68         print(command)
69         if(command=="lighton"):
70             print('lighton')
71         elif(command=="lightoff"):
72             print('lightoff')
73         elif(command=="motoron"):
74             print('motoron')

```

```

74     print('motoron')
75     elif(command=="motoroff"):
76         print('motoroff')
77     myConfig = {
78         "identity": {
79             "orgId": "chytun",
80             "typeId": "NodeMCU",
81             "deviceId": "12345"
82         },
83         "auth": {
84             "token": "12345678"
85         }
86     }
87     client = wiot.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
88     client.connect()
89
90     database_name = "sample"
91     my_database = clientdb.create_database(database_name)
92     if my_database.exists():
93         print(f'({database_name})' successfully created.")
94     cap=cv2.VideoCapture("garden.mp4")
95     if(cap.isOpened()==True):
96         print('File opened')
97     else:
98         print('File not found')
99
100     while(cap.isOpened()):
101         ret, frame = cap.read()
102         gray = cv3.cvtColor(frame, cv2.COLOR_BGR@GRAY)
103         imS= cv2.resize(frame, (960,540))
104         cv2.imwrite('ex.jpg',imS)
105         with open("ex.jpg", "rb") as f:
106             file_bytes = f.read()
107         #This is the model ID of a publicly available General model. You may use any other public or custom model ID.
108         request = service_pb2.PostModelOutputsRequest(
109             model_id='e9359dbe6ee44dbc8842ebe97247b201',
110             inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))

```

```

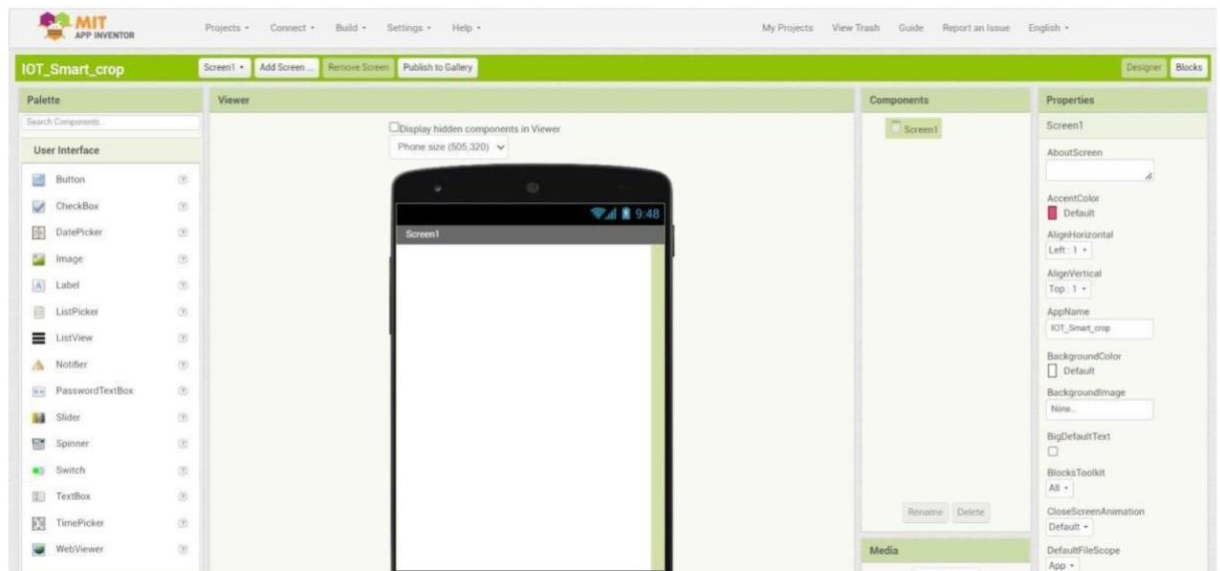
110 ...         inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))
111 ...             ))
112 ...         response = stub.PostModelOutputs(request, metadata=metadata)
113 ...         if response.status.code != status_code_pb2.SUCCESS:
114 ...             raise Exception("Request failed, status code: " + str(response.status.code))
115 ...         detect=False
116 ...         for concept in response.outputs[0].data.concepts:
117 ...             #print('%12s: %.f' % (concept.name, concept.value))
118 ...             if(concept.value>0.98):
119 ...                 #print(concept.name)
120 ...                 if(concept.name=="animal"):
121 ...                     print("Alert! Alert! animal detected")
122 ...                     playsound.playsound('alert.mp3')
123 ...                     picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
124 ...                     cv2.imwrite(picname+'.jpg',frame)
125 ...                     multi_part_upload('Dhakshesh', picname+'.jpg', picname+'.jpg')
126 ...                     json_document={"link":COS_ENDPOINT+'/'+'+Dhakshesh'++'/'+picname+'.jpg'}
127 ...                     new_document = my_database.create_document(json_document)
128 ...                     if new_document.exists():
129 ...                         print(f"Document successfully created.")
130 ...                         time.sleep(5)
131 ...                         detect=True
132 ...                     moist=random.randint(0,100)
133 ...                     humidity=random.randint(0,100)
134 ...                     myData={'Animal':detect,'moisture':moist,'humidity':humidity}
135 ...                     print(myData)
136 ...                     if(humidity!=None):
137 ...                         client.publishEvent(eventId="status",msgFormat="json", daya=myData, qos=0, onPublish=None)
138 ...                         print("Publish Ok..")
139 ...                     client.commandCallback = myCommandCallback
140 ...                     cv2.imshow('frame',imS)
141 ...                     if cv2.waitKey(1) & 0xFF == ord('q'):
142 ...                         break
143 ...                     client.disconnect()
144 ...                     cap.release()

```

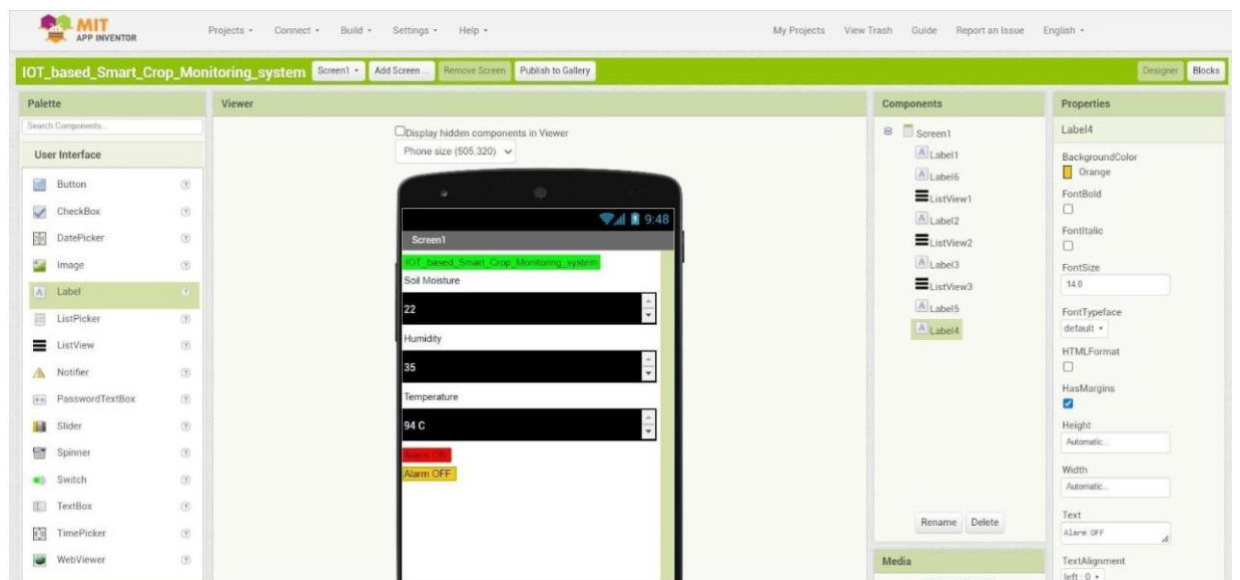
```
*IDLE Shell 3.8.8*
File Edit Shell Debug Options Window Help
Python 3.8.8 (tags/v3.8.8:024d805, Feb 19 2021, 13:18:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/HP/Desktop/crop/crop_protect.py =====
2021-04-06 12:52:19,640 wiotp.sdk.device.client.DeviceClient INFO Connecte
d successfully: d:hj5fmy:NodeMCU:12345
'sample' successfully created.
File opened
{'Animal': False, 'moisture': 17, 'humidity': 41}
Publish Ok..
{'Animal': False, 'moisture': 64, 'humidity': 16}
Publish Ok..
{'Animal': False, 'moisture': 48, 'humidity': 43}
Publish Ok..
{'Animal': False, 'moisture': 0, 'humidity': 3}
Publish Ok..
{'Animal': False, 'moisture': 73, 'humidity': 68}
Publish Ok..
{'Animal': False, 'moisture': 26, 'humidity': 26}
Publish Ok..
{'Animal': False, 'moisture': 96, 'humidity': 59}
Publish Ok..
I
Ln: 10 Col: 11
```

## 7.2 Feature 2

### MIT app inventor to design the app:



### Customizing the app interface to display the values:





## 8 TESTING

### 8.2 Test Cases

#### ▴ Defect Analysis

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

+	Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
	By Design	11	4	2	2	19
	Duplicate	1	1	2	0	4
	External	2	3	0	1	6
	Fixed	10	2	3	20	35
	Not Reproduced	0	0	2	0	2
	Skipped	0	0	2	1	3
	Won't Fix	0	5	2	1	8
	Totals	24	15	13	25	77

#### | 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	5	0	1	4
Client Application	47	0	2	45

Security	3	0	0	3
Outsource Shipping	2	0	0	2
Exception Reporting	11	0	2	9
Final Report Output	5	0	0	5
Version Control	3	0	1	2

## **9 RESULTS**

Thus the IOT based Smart Crop Protection has been build successfully with the help of MIT app, Node.Js, and node red. And the output has been tested and verified using MIT app.

### **ADVANTAGES :**

- ⇒ Sensors in Agriculture invented to meet the increasing demand for food with minimum resources such water, fertilizers and seeds.
- ⇒ They are easy to operate and use and easy to maintain.
- ⇒ Sensors are cheaper in price and best in quality.
- ⇒ They can used for measuring pollution and global warming for their fields and crops.

### **DISADVANTAGES:**

- ⇒ Farms are located in remote areas and are far from access to the internet.
- ⇒ A farmer needs to have access to crop data reliably at any time from any location, so connection issues would cause an advanced monitoring system to be useless.
- ⇒ High Cost: Equipment needed to implement IoT in agriculture is expensive.

## **10 CONCLUSION:**

Smart farming reduces the ecological footprint of farming. Minimized or site-specific application of inputs, such as fertilizers and pesticides, in precision agriculture systems will mitigate leaching problems as well as the emission of greenhouse gases

## **11 FUTURE SCOPE**

IoT smart agriculture products are designed to help monitor crop fields using sensors and by automating irrigation systems. As a result, farmers and associated brands can easily monitor the field conditions from anywhere without any hassle.

## **12 APPENDIX**

**GitHub Link** <https://github.com/IBM-EPBL/IBM-Project-43986-1660721022>

**Source Code** <https://github.com/IBM-EPBL/IBM-Project-43986-1660721022/tree/main/Final%20Deliverables/Final%20Code>

**Demo Link** <https://github.com/IBM-EPBL/IBM-Project-43986-1660721022/tree/main/Final%20Deliverables/Demonstration%20Video>

Prepared by:

- ANGALAPARAMESHWARI A
- VANATHI M
- SUDHAKAR S
- SHIYAM M

**THANK YOU!**