NALAIYA THIRAN

IOT BASED SMART CROP PROTECTION FOR AGRICULTURE

BATCH NUMBER: B1-1M3E

TEAM ID: PNT2022TMID43224

SUBMITTED BY

TEAM MEMBERS	-	REGISTER NUMBER
DAYANA.K	-	715019106001
JOSHVA NIMALAN.R	-	715019106007
KABISHIKA.S	-	715019106008
PRIYAMANI.S	-	715019106015
SABITHA.G	-	715019106018

OF

ELECTRONICS AND COMMUNICATION ENGINEERING

OF SUGUNA COLLEGE OF ENGINEERING

COIMBATORE-641 014

INDEX

1. INTRODUCTION

1.1Project overview

- The device will 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 a 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 IBM Iot platform.
- The image will be retrieved from object storage and displayed in the web application.

- The web application is developed to visualize the soil moisture, temperature and humidity values.
- Users can also control the motors through the web application.

1.2 Purpose

An intelligent crop protection system helps the farmers in protecting the crop from the animals and birds which destroy the crops. This system also helps the 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

Most of the farmers are facing many problems nowadays due to many reasons. Our problem to solve is the invasion of various species such as birds and animals that harm the crops

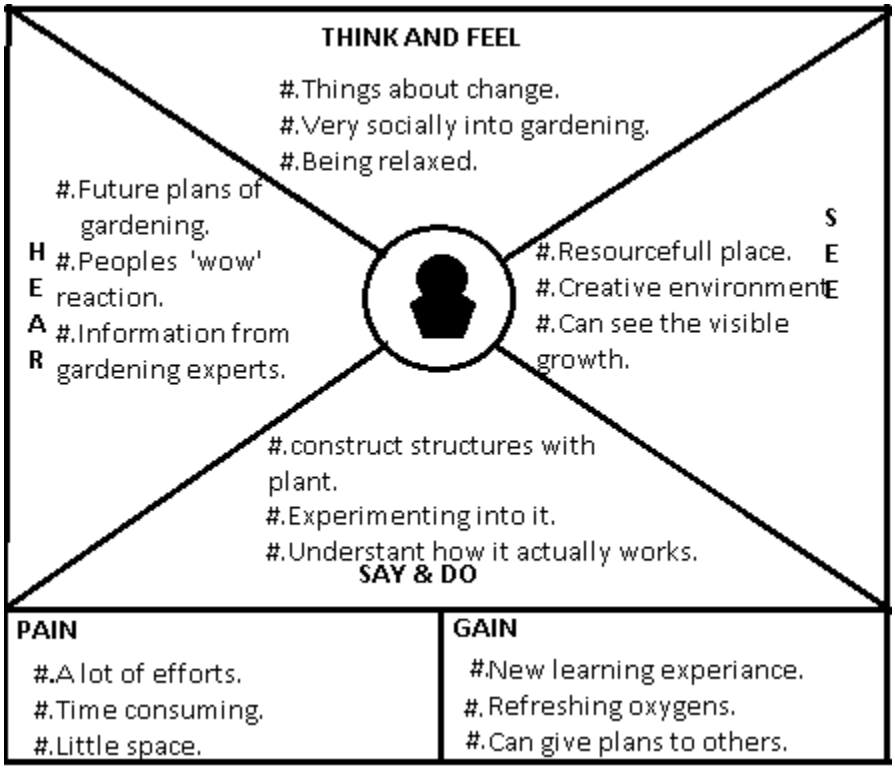
that are being cultivated .Various types of species such as birds and animals come to the cultivation field according to the crop that is being cultivated and also according to the season of cultivation. Some wild animals enter the field during night times when the field is near a forest region or when the farm cultivate some fruits and other crops that attract animals. Some animals crops the field in search of food and water and also the birds enter the field for food and they damage all the crops. When the animals enter the field they not only eat food but they also damage the entire field by walking across the crops and also by spoiling the food crops. The birds, by entering the field they come to eat seeds of the crops and also they tend to drag the crops and ruin the entire field. Some birds enter the field to eat the insects and pest in the field.

2.2 Problem Statement Definition

Most of the farmers are facing many problems nowadays due to many reasons. Our problem to solve is the invasion of various species such as birds and animals that harm the crops that are being cultivated. Various types of species such as birds and animals come to the cultivation field according to the crop that is being cultivated and also according to the season of cultivation. Some wild animals enter the field during night times when the field is near a forest region or when the farm cultivates some fruits and other crops that attract animals.

3. IDEATION P & PROPOSED SOLUTION

3.1 Empathy map canvas



3.2 Customer problem statement:

- Create app-based solution to detect soil parameters like moisture content, temperature, relative humidity, nutrient, Ph, CEC, NPK etc. and provide crop suggestions to be produced based on soil parameters & environment values
- With the help of sensors/ imaginary input create cotton crop health monitoring application which will provide various parameters related to cotton crop like moisture level, nutrient level, pest infection level, maturity/harvesting time etc. and create alert for remedial action.

I am	A digital farmer
I'am trying to	Develop affordable app-based solution for soil health monitering.
but	Animals destroy crops.
because	Cutting of forest for farming ang industrial activities.
which makes me feel	Depression.

3.3. Literature survey

S.NO	AUTHOR NAME	DESCRIPTION
1	Zhang, F. Research on applications of Internet of Things in agriculture .Springer: London, UK, 2013; pp. 69–75.	Analysis of the state-of-the-art IoT-based smart farming studies is a key challenge due to the coverage of multiple application domains, communication protocols, sensors/devices, and protocols. According to our research questions, we have gathered 67 primary studies in this section. After analyzing the selected studies, we have addressed each question according to the extracted information.
2	Tanmay Baranwal" Development of IOT based Smart Security and Monitoring Devices for Agriculture",	A system using sensors that monitor different conditions of environment like water level, humidity, temperature etc., the processor along with IC-S8817BS and wireless transceiver

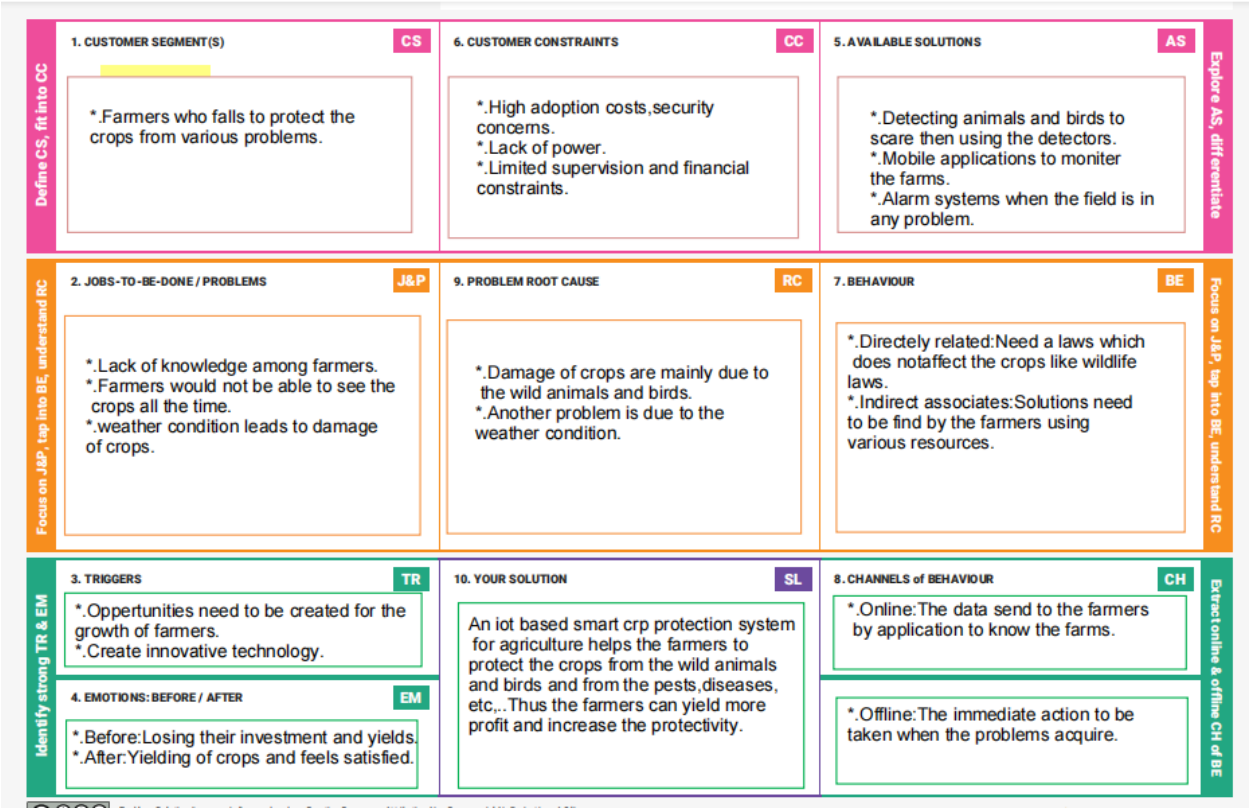
		module with zigbee protocol is used. The field condition is sent to the farmer via mobile text messages and email from the experts. With this system Sensor node failure and energy efficiency are managed.
3	S. R. Chourey, P. A. Amale	IOT tendencies are often utilized in smart farming to boost the standard of agriculture [2]. Farming the pillar of supports our country to the general commercial development. But our productivity is extremely low as associated to world standards [31]. People from rural areas drift to an urban area for other worthwhile trades.

3.4 Proposed solution

S .No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Develop system for predicting potential pest, disease, insect attacks on crop and yield prediction of crops
2.	idea\solution description	This system uses a motion sensor to detect wild animals approaching near the field and smoke sensor to detect the fire.
3.	Novelty / Uniqueness	Unlike genetic resources found in the natural world, agricultural crops are truly a human mediated form of biodiversity.

4.	Social Impact / Customer Satisfaction	Agriculture creates both jobs and economic growth. Communities also hold agricultural-based events, such as crop and livestock judging competitions and 4-H exhibits at their county fair. Many communities benefit from having Famers Markets where smaller farmers can interact directly with consumers.
5.	Business Model (Revenue Model)	On farms, IoT allows devices across a farm to measure all kinds of data remotely and provide this information to the farmer in real time. IoT devices can gather information like soil moisture, chemical application, dam levels and livestock health – as well as monitor fences, vehicles and weather.
6.	Scalability of the Solution	In future it can be enhanced by sending message directly to the fire department in case there is a mass wild animals attacks the fields. The controlling and monitoring of the soil moisture level can be automated by taking care of the crops in case of low moisture level, without notifying the farmers

3.5 Problem solution fit



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)
1.	User Registration	Install the app. Signing up with Gmail or phone number. Creating a profile. Understand the guidelines.
2.	User Confirmation	Email or phone number verification required via OTP.
3.	Accessing datasets	Data's are obtained by cloudant DB.

4.	Interface sensor	Connect the sensor and the application. When animals enter the field, the alarm is generated.
5.	Mobile application	It is used to control motors and field sprinklers.

4.2. Non- functional Requirements

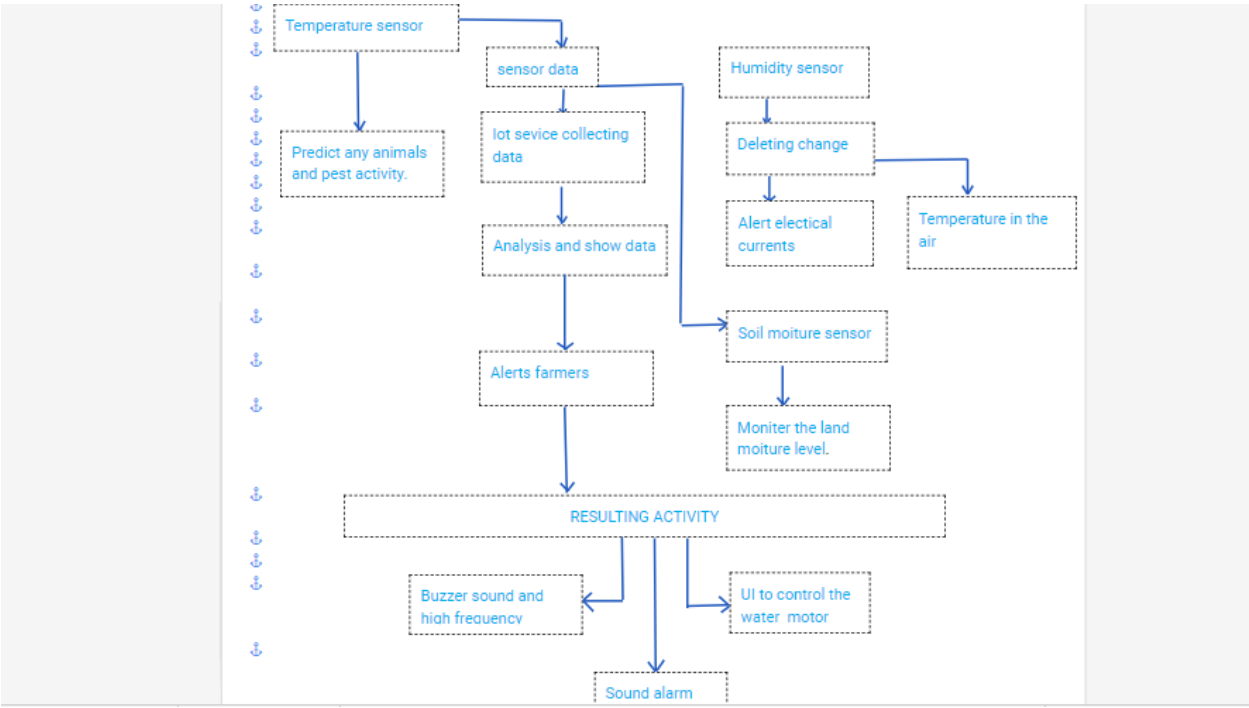
Following are the non-functional requirements of the proposed solution.

FR No.	**Non-Functional Requirement**	**Description**
1.	Usability	The project's contributes the farm protection through the smart protection system.
2.	Security	It was created to protect the crops from animals.
3.	Reliability	Farmers are able to safeguard their lands by help of this technology. They will also benefits from higher crop yields, which will improve our economic situation.
4.	Performance	When animals attempt to enter the field, IOT devices and sensors alert the farmer via message.
5.	Availability	We can defend the crops against wild animals by creating and implementing resilient hardware and software.

6.	Scalability	This system's integration of computer vision algorithms with IBM cloudant services makes it more efficient to retrieve photos at scale, enhancing scalability.

5. PROJECT DESIGN

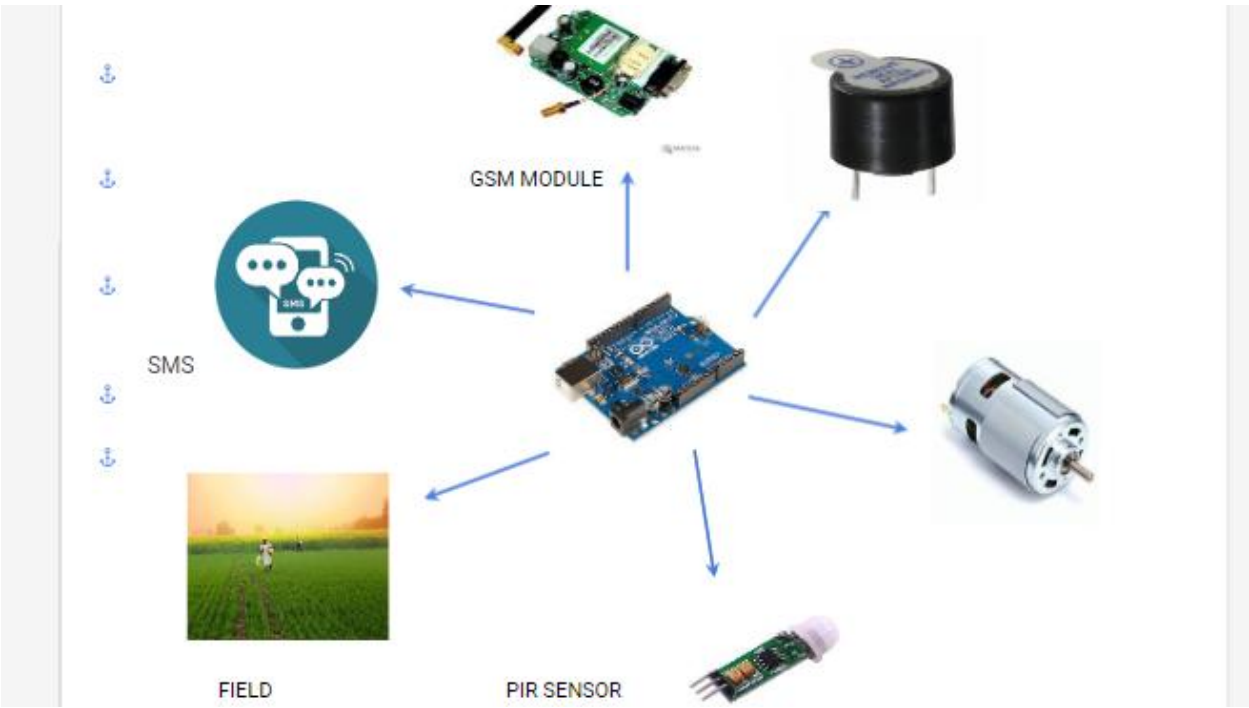
5.1 Data flow diagram



5.2 Solution architecture

Solution architecture is the process of developing solutions based on predefined processes, guidelines and best practices with the objective that the developed solution fits within the enterprise architecture in terms of information architecture, system portfolios, integration requirements.

5.3. Solution architecture diagram



6. SPRINT PLANNING AND SCHEDULING

6.1. Sprint planning and estimation

Sprint	Functional requirement (Epic)	User story Number	User story / Task	Story point	Priority	Team Members
Sprint - 1	Registration	USN-1	As a user. I can register for the application by entering my email , password , and confirming my password.	5	High	K. Dayana
Sprint - 1	Login page	USN-2	As a user , enter the username and password which is already existing	5	High	S.Priyamani k.Dayana, G.Sabitha, R.Joshva nimalan S.Kabishika
Sprint - 2	Sensing moisture condition of the soil	USN-3	As a user, We can know about soil moisture condition , controlling the motor pump for water flow	20	High	S.Kabishika K.Dayana

			by using mobile application			
Sprint - 3	Detecting the motion in certain range	USN-4	Fencing system are helpful in providing security against animals and birds.	20	High	G.Sabitha S.Priyamani
Sprint - 4	Checking the crops condition	USN-5	Here farmer needs to update the condition of crops	20	High	R.Joshva nimalan

Project Tracker, Velocity & Burn down Chart :

Sprint	Total story points	Duration	Sprint state date	Sprint end date	Story points completed	Sprint release date
Sprint 1	20	6 Days	22 Oct 2022	27 Oct 2022	20	29 Oct 2022
Sprint 2	20	6 Days	31 Oct 2022	06 Nov 2022	35	30 Oct 2022
Sprint 3	20	6 Days	07 Nov 2022	12 Nov 2022	48	06 Nov 2022
Sprint 4	20	6 Days	14 Nov 2022	19 Nov 2022	50	08 Nov 2022

7. CODING AND SOLUTION :

7.1 Python code

```
import cv2
import numpy as np
import wiot.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3
from ibm_botocore.client import Config, ClientError

#CloudantDB
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
from clarifai_grpc.grpc.api import service_pb2_grpc
stub = service_pb2_grpc.V2Stub(clarifaiChannel.get.grpc_channel())
from clarifai_grpc.grpc.api import service_pb2, resource_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2

#This is how you authenticate
metadata = (('authorization', 'key 5797d941-433e-436a-a480-680d9080a990'),)
COS_ENDPOINT = "https://s3.tok.ap.cloud-object-storage.appdomain.cloud"
```

```
COS_API_KEY_ID = "v9n8Zn4r5VpcMVz_HyRY0DrS13jSzph2IEFioVj4-vmT"
COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"
COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloud-object-
storage:global:a/3f060ee770d94e20a88f49f3da641d6d:f301cab2-2e94-48a1-a8a0-5b4968527c54::"
clientdb = cloudant("apikey-_pIeLXPoaPpnOZ7SMoVKd6tZdsjf54X9LwkFEWB1a0T6", "0165dca6-
1176-4aa5-b0fe-81473e50e35d", url="https://47643860-3553-4211-ba2a-d8e26dd17c08-
bluemix.cloudantnosqldb.appdomain.cloud")
clientdb.connect()

#Create resource
cos = ibm_boto3.resource("s3",
                                    ibm_api_key_id=COS_API_KEY_ID,
                                    ibm_service_instance_id=COS_RESOURCE_CRN,
                                    ibm_auth_endpoint=COS_AUTH_ENDPOINT,
                                    config=Config(signature_version="oauth"),
                                    endpoint_url=COS_ENDPOINT
                                    )
def = multi_part_upload(bucket_name, item_name, file_path):
      try:
          print("Starting file transfer for {0} to bucket: {1}\n".format(item_name, bucket_name))
          #set 5 MB chunks
          part_size = 1024 * 1024 * 5
          #set threadhold to 15 MB
          file_threshold = 1024 * 1024 * 15
          #set the transfer threshold and chunk size
          transfer_config = ibm_boto3.s3.transfer.TransferConfig(
                multipart_threshold=file_threshold,
                multipart_chunksize=part_size
                )
          #the upload_fileobj method will automatically execute a multi-part upload
          #in 5 MB chunks size
          with open(file_path, "rb") as file_data:
               cos.Object(bucket_name, item_name).upload_fileobj(
                     Fileobj=file_data,
                    Config=transfer_config
                     )
    print("Transfer for {0} Complete!\n".format(item_name))
    except ClientError as be:
         print("CLIENT ERROR: {0}\n".format(be))
    except Exception as e:
         print("Unable to complete multi-part upload: {0}".format(e))

def myCommandCallback(cmd):
      print("Command received: %s" % cmd.data)
      command=cmd.data['command']
      print(command)
      if(commamd=="lighton"):
          print('lighton')
      elif(command=="lightoff"):
          print('lightoff')
       elif(command=="motoron"):
```

```
            print('motoron')
        elif(command=="motoroff"):
            print('motoroff')
myConfig = {
        "identity": {
            "orgId": "chytun",
            "typeId": "NodeMCU",
            "deviceId": "12345"
             },
        "auth": {
            "token": "12345678"
             }
         }
client = wiot.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

database_name = "sample"
my_database = clientdb.create_database(database_name)
if my_dtabase.exists():
        print(f"'(database_name)' successfully created.")
cap=cv2.VideoCapture("garden.mp4")
if(cap.isOpened()==True):
        print('File opened')
else:
        print('File not found')

while(cap.isOpened()):
       ret, frame = cap.read()
       gray = cv3.cvtColor(frame, cv2.COLOR_BGR@GRAY)
        imS= cv2.resize(frame, (960,540))
        cv2.inwrite('ex.jpg',imS)
        with open("ex.jpg", "rb") as f:
              file_bytes = f.read()
       #This is the model ID of a publicly available General model. You may use any other public or custom
model ID.
        request = service_pb2.PostModeloutputsRequest(
               model_id='82eaf1c767a74869964531e4d9de5237
        inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_by
tes))
                                )])
   response = stub.PostModelOutputs(request, metadata=metadata)
        if response.status.code != status_code_pb2.SUCCESS:
               raise Exception("Request failed, status code: " + str(response.status.code))
        detect=False
        for concept in response.outputs[0].data.concepts:
              #print('%12s: %.f' % (concept.name, concept.value))
              if(concept.value>0.98):
                   #print(concept.name)
                   if(concept.name=="animal"):
                        print("Alert! Alert! animal detected")
                        playsound.playsound('alert.mp3')
```

```
                    picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
                    cv2.inwrite(picname+'.jpg',frame)
                    multi_part_upload('Umamaheswari', picname+'.jpg', picname+'.jpg')
                    json_document={"link":COS_ENDPOINT+'/'+'Umamaheswari'+'/'+picname+'.jpg'}
                    new_document = my_database.create_document(json_document)
                    if new_document.exists():
                        print(f"Document successfully created.")
                    time.sleep(5)
                    detect=True
        moist=random.randint(0,100)
        humidity=random.randint(0,100)
        myData={'Animal':detect,'moisture':moist,'humidity':humidity}
        print(myData)
        if(humidity!=None):
            client.publishEvent(eventId="status",msgFormat="json", daya=myData, qos=0,
onPublish=None)
            print("Publish Ok..")
        client.commandCallback = myCommandCallback
    cv2.imshow('frame',imS)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
client.disconnect()
cap.release()
cv2.destroyAllWindows()
```

NODE RED CODE:

TEMPERATURE:

```
msg.payload=msg.payload.”temp”

return msg;
```

HUMIDITY:

```
msg.payload=msg.payload.”Humid”

return msg;
```

MOISTURE:

```
msg.payload=msg.payload.”Moist”

return msg;
```

ANIMAL DETECTION:

```
msg.payload=msg.payload.”Animal_dect”

return msg;
```

8. TESTING:

8.1 Purpose of Document

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

8.2 Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
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	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

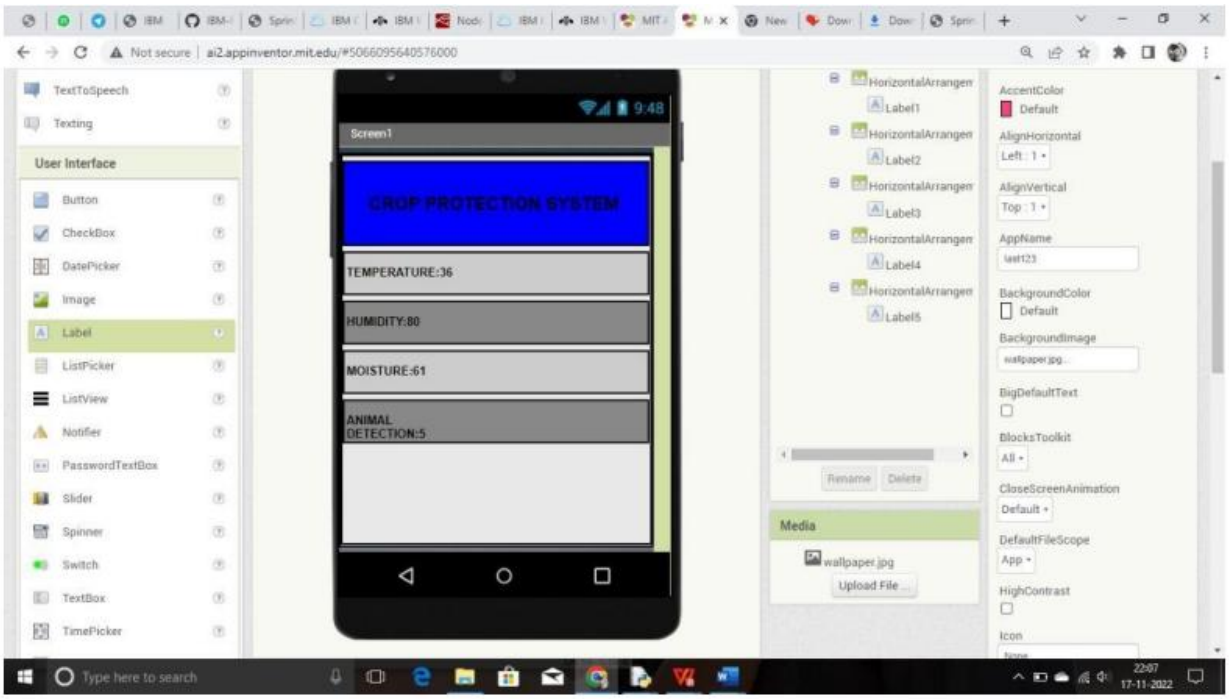
8.3 Test Case Analysis

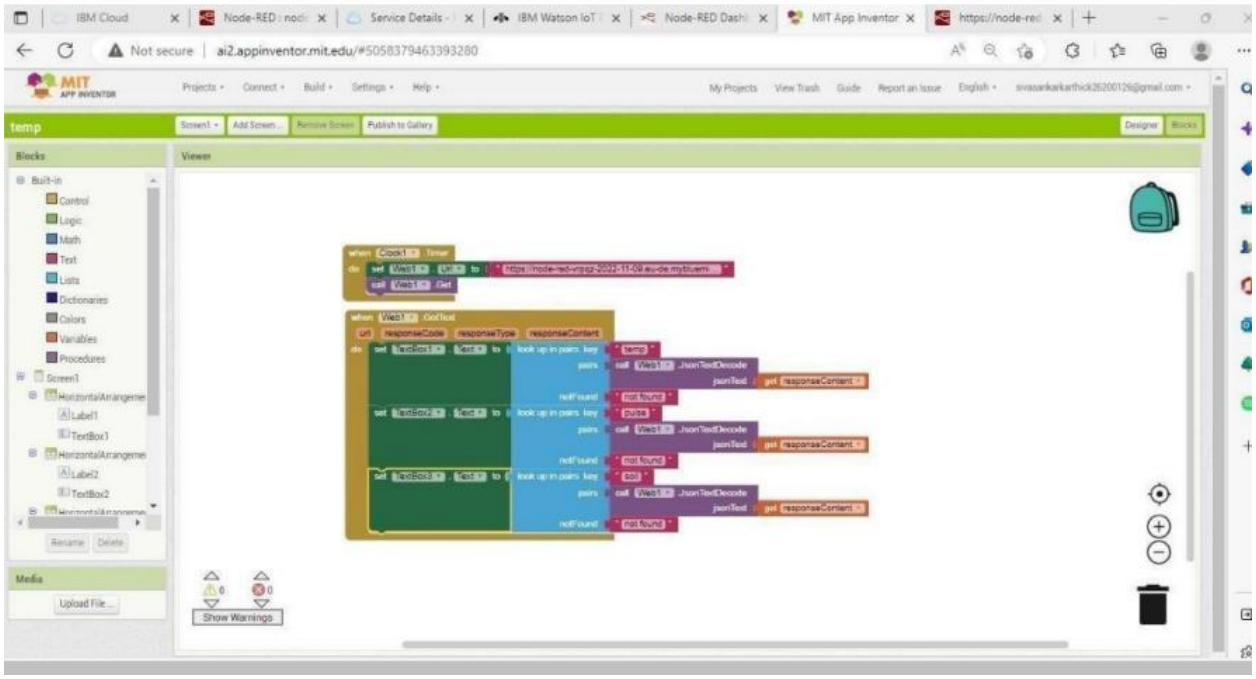
This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
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. RESULT

MIT APP INVENTOR- TO DESIGN THE APP





9.1. ADVANTAGES:

- 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
- greater number of resources through remote sensing.
- High reliance.
- Enhanced Security

9.2. DISADVANTAGES:

• Monitoring the crop field with the help of sensors (light , humidity, temperature, soil moisture, etc

• Automating the irrigation system

• Soil Moisture

• 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.

9.3. APPLICATIONS:

Monitoring (including conductivity)

10. CONCLUSION:

The problem of crop destroying by wild animals and fire has become a major social problem in current time. It requires urgent attention as no effective solution exists till date for this problem. Thus, this project carries a great social relevance as it aims to address this problem. This project will help farmers in protecting their orchards and fields and save them from significant financial losses and will save them from the unproductive efforts that they endure for the protection their fields. This will also help them in achieving better crop yields thus leading to their economic well being.

11. FUTURECODE:

Study and analysis of the developed crop protection systems for its cost effectiveness with the development of Arduino based variable frequency ultrasonic bird deterrent circuit. Outline of the crop damage caused by an particular wild animals if the features of the reduced cost in the smart phones.

APPENDIX

12. SOURCE CODE

The source code has been uploaded in the git hub.

13. GITHUB LINK:

Dayana. K(Team Leader)- https://github.com/dayanaece

Kabishika.S -https://github.com/kabishikaece

Sabitha.G - https://github.com/sabithaece

JoshvaNimalan.R- -https://github.com/joshvanimalanece

Priyamani.S - http://github.com/priyamaniece