IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

Category:INTERNET OF THINGS

A PROJECT REPORT

Submittedby

SRI HARINI

SANTHOSH KUMAR

CINDRELLA PASCAL

MANOJ

PRADEEPA

FROM

St.annes college of engineering and technology

Infulfillment of projectin IBM-NALAYATHIRAN2022

TeamId:PNT2022TMID39047

PROJECT GUIDES

Industry Mentor: Mr. Dinesh

INDEX

1. INTRODUCTION

1.1 ProjectOverview:

Smart crop protection system

Smart crop protection system solutions use sensors placed in crop yields to measure humidity ,temperarture , moisture and to notify farmers when crops are ready to be emptied. Over time, historical data collected by sensors can be used to identify crop patterns . The cost of these sensors is steadily decreasing, making IoT crop protection more feasible to implement and more attractive to farmer.

1.2 Purpose:

- At present, we can see crop are being damaged due to many reasons. Our primary goal is to protect the crop from being damaged .
- Due to damage in crops, many farmers left farming and started doing other jobs because of loss they faced in agriculture. So our crop protection should prevent crop from being damaged and produce better yield.
- In agriculture fields crops are being damaged by birds, animals, insects, climate, disease, excess water, etc. Our crop protection system should stop these from damaging the crops .
- So, our problem statement is to design a system based on IOT application for protecting crops from birds, animals, insects, climate, disease, excess water, etc and provide high yield in agriculture to make farmers happy and people enjoy the healthy food.

2. LITERATURESURVEY

2.1 ExistingProblem:

- At present, we can see crop are being damaged due to many reasons. Our primary goal is to protect the crop from being damaged.

- Due to damage in crops, many farmers left farming and started doing other jobs because of loss they faced in agriculture. So our crop protection should prevent crop from being damaged and produce better yield.

- In agriculture fields crops are being damaged by birds, animals, insects, climate, disease, excess water, etc. Our crop protection system should stop these from damaging the crops

- So, our problem statement is to design a system based on IOT application for protecting crops from birds, animals, insects, climate, disease, excess water, etc and provide high yield in agriculture to make farmers happy and people enjoy the healthy food.

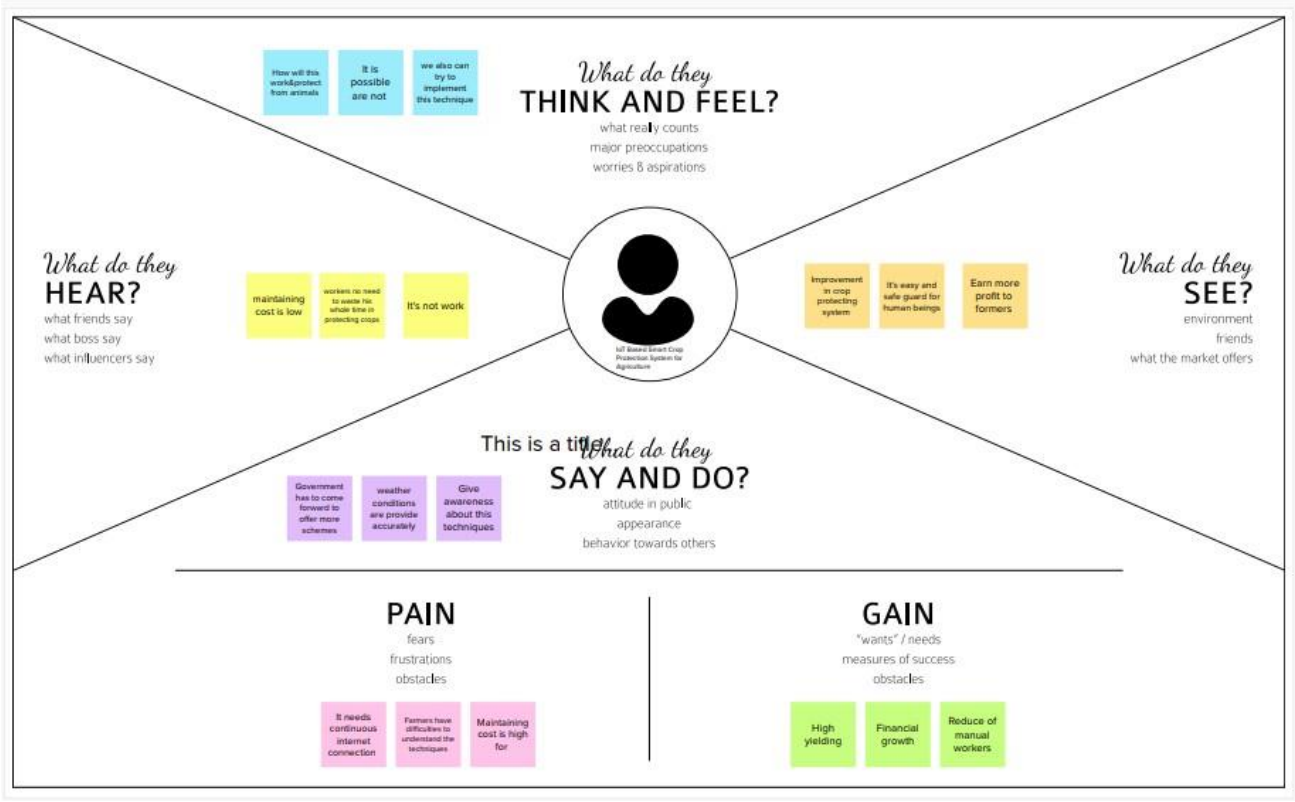
2.2 ProblemStatement Definition:

- Due to damage in crops, many farmers left farming and started doing other jobs because of loss they faced in agriculture. So our crop protection should prevent crop from being damaged and produce better yield .In agriculture fields crops are being damaged by birds, animals, insects, climate, disease, excess water, etc. Our crop protection system should stop these from damaging the crops .So, our problem statement is to design a system based on IOT application for protecting crops from birds, animals, insects, climate, disease, excess water, etc and provide high yield in agriculture to make farmers happy and people enjoy the healthy food.

IDEATION&PROPOSEDSOLUTION

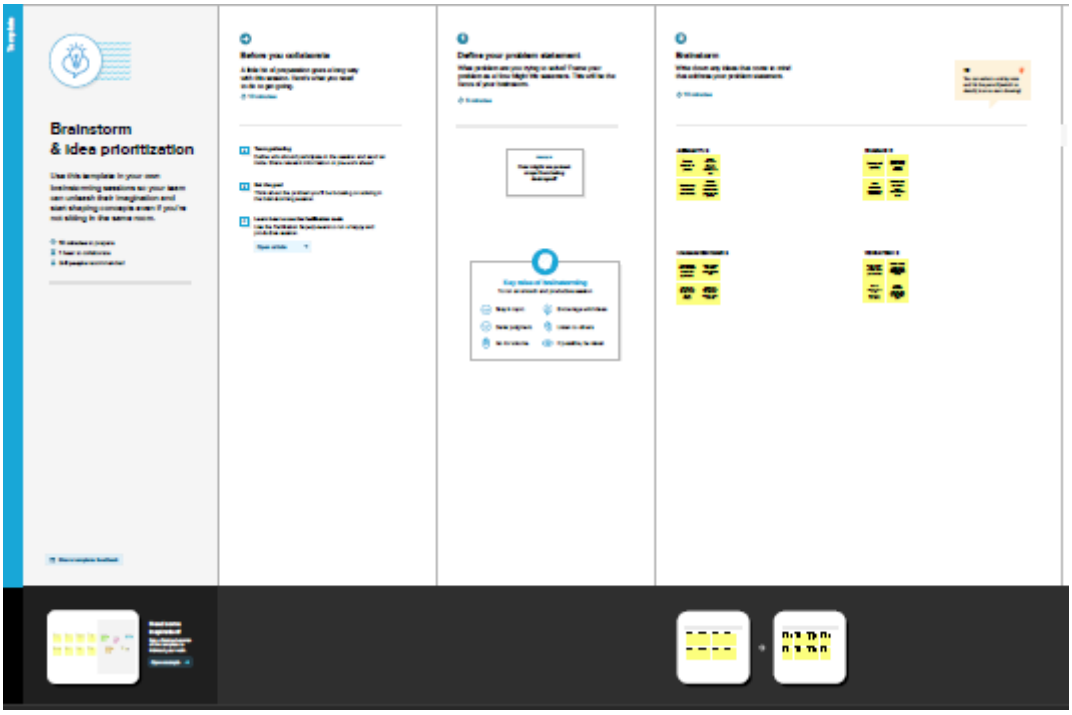
3.1 EmpathyMapCanvas:

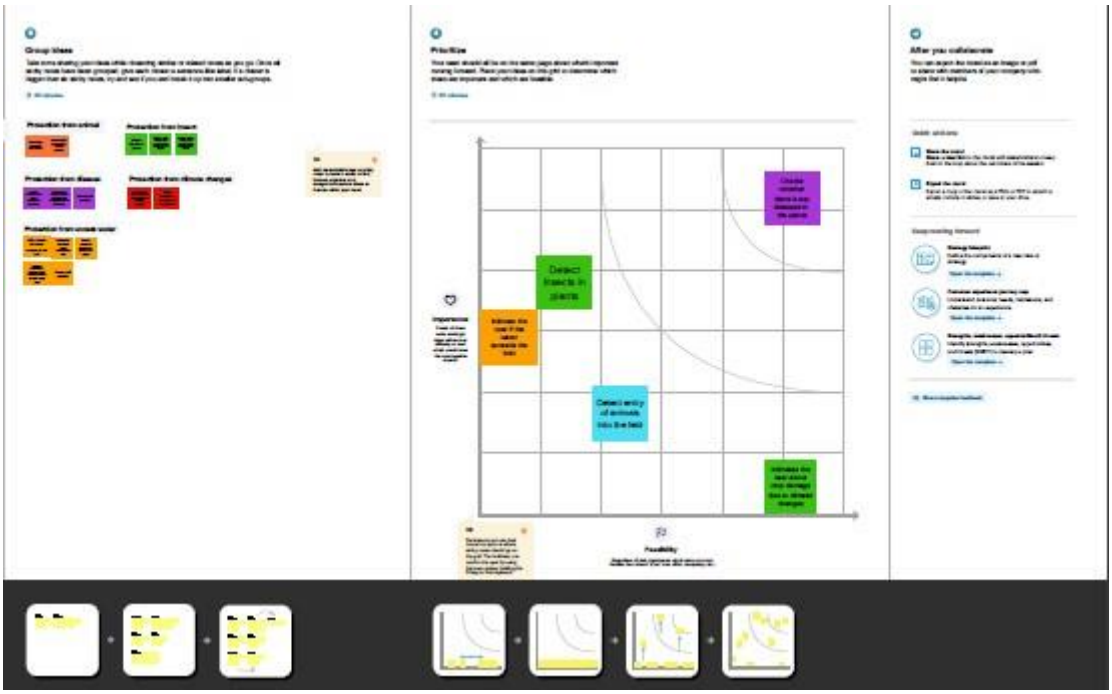
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user'sbehavioursandattitudes.Itisausefultooltohelpsteamsbetterunderstandtheirusers.Creating an effective solution requires understanding the true problem and the personwho is experiencing it. The exercise of creating the map helps participants consider thingsfromtheuser'sperspectivealongwithhisorhergoalsandchallenges.



3.2 Ideation&Brainstorming:

Ideation and Brainstorming Ideation is often closely related to the practice of brainstorming, a **specific technique that is utilized to generate new ideas**. A principal difference between ideation and brainstorming is that ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity.



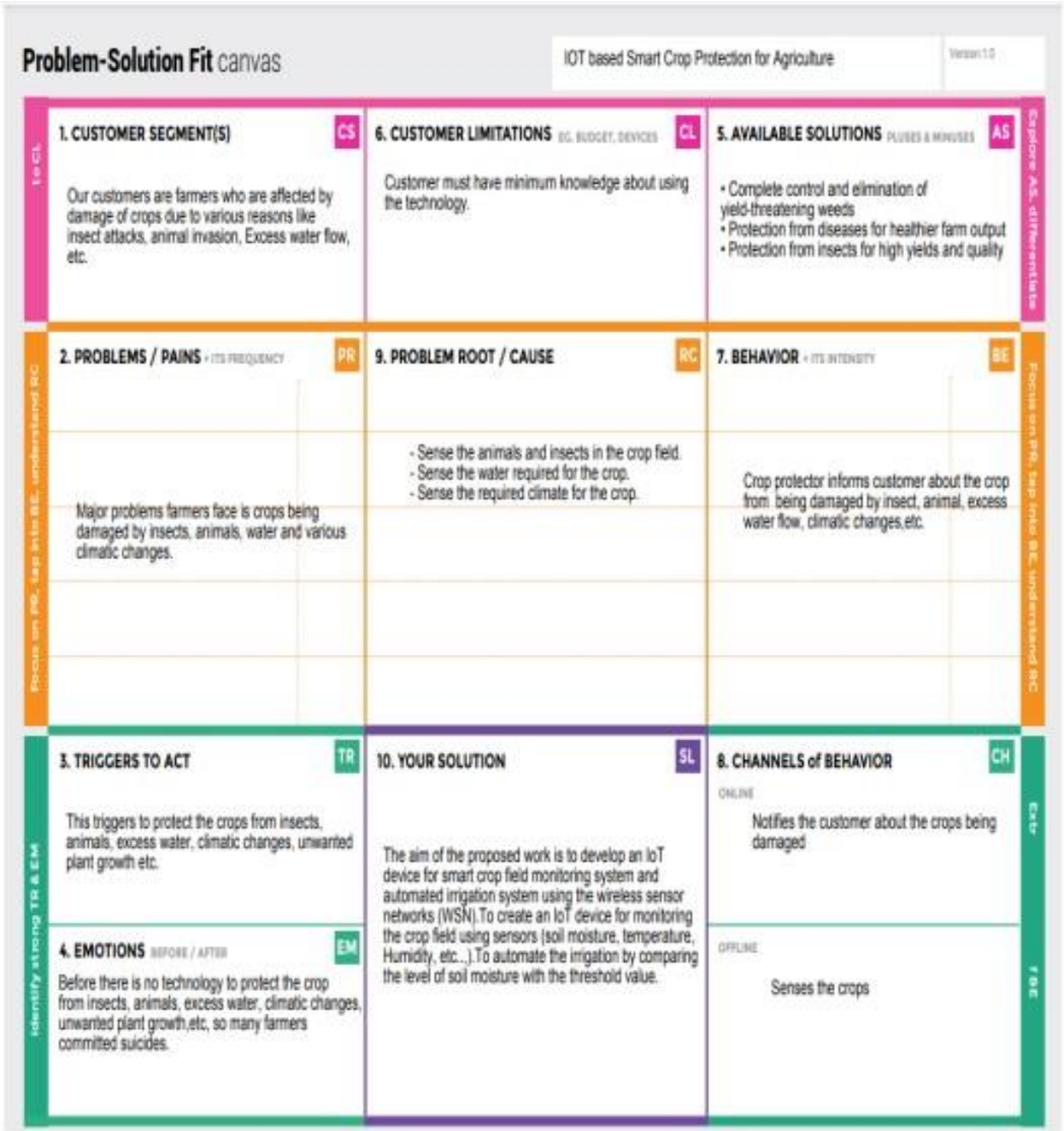

Group ideas
Prioritize
After you collaborate
Detect insects in plants
Detect entry of animals into the field

3.3 ProposedSolution:

S.No.	Parameter	Description
1.	ProblemStatement(Problemtobesolved)	IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE
2.	Idea/Solutiondescription	The aim of the proposed work is to develop an IOT device for smart crop field monitoring system and automated irrigation system using the wireless sensor networks(WSN) . To create an IOT device for monitoring the crop field using sensors (soil moisture ,temperature ,Humidity ,etc.,) To automate the irrigation by comparing the level of soil moisture with the threshold value .
3.	Novelty/Uniqueness	Daily update about the Condition of the land send to the farmers via mail
4.	Social Impact /CustomerSatisfaction	• Cost effective to the society • Modernaization to the society • High protection and High yield

5.	Business Model(Revenue Model)	Outcome based model Data based model Platform based model
6.	ScalabilityoftheSolution	Start small and build out

3.4 PROBLEMSOLUTION FIT:


Problem-Solution Fit canvas
IOT based Smart Crop Protection for Agriculture
1. CUSTOMER SEGMENT(S)
CS
Our customers are farmers who are affected by damage of crops due to various reasons like insect attacks, animal invasion, Excess water flow, etc.
6. CUSTOMER LIMITATIONS
CL
Customer must have minimum knowledge about using the technology.
5. AVAILABLE SOLUTIONS
AS
• Complete control and elimination of yield-threatening weeds
• Protection from diseases for healthier farm output
• Protection from insects for high yields and quality
2. PROBLEMS / PAINS
PR
Major problems farmers face is crops being damaged by insects, animals, water and various climatic changes.
9. PROBLEM ROOT / CAUSE
RC
- Sense the animals and insects in the crop field.
- Sense the water required for the crop.
- Sense the required climate for the crop.
7. BEHAVIOR
BE
Crop protector informs customer about the crop from being damaged by insect, animal, excess water flow, climatic changes,etc.
3. TRIGGERS TO ACT
TR
This triggers to protect the crops from insects, animals, excess water, climatic changes, unwanted plant growth etc.
4. EMOTIONS
EM
Before there is no technology to protect the crop from insects, animals, excess water, climatic changes, unwanted plant growth,etc, so many farmers committed suicides.
10. YOUR SOLUTION
SL
The aim of the proposed work is to develop an IoT device for smart crop field monitoring system and automated irrigation system using the wireless sensor networks (WSN).To create an IoT device for monitoring the crop field using sensors (soil moisture, temperature, Humidity, etc...).To automate the irrigation by comparing the level of soil moisture with the threshold value.
8. CHANNELS of BEHAVIOR
CH
ONLINE
Notifies the customer about the crops being damaged
OFFLINE
Senses the crops

4REQUIREMENTANALYSIS

4.1 FunctionalRequirements:

Followingarethefunctionalrequirementsoftheproposedsolution.

FRNo.	FunctionalRequirement (Epic)	SubRequirement(Story/Sub-Task)
FR-1	UserRegistration	Registration through FormRegistrationthroughGmail .
FR-2	UserConfirmation	ConfirmationviaEmail .
FR-3	Interfacingwithhardware	Interfacethesensorswiththesoftwareapplicationsoastoalertthefarmersincaseofanyharmforcrops .
FR-4	DatabaseConnection	DatabasesareretrievedfromIBMCloudant .
FR-5	MobileApplication	Alarmandmotorscanbeaccessedfromthemobileapp .

4.2 ***Non-functionalRequirements:***

Followingarethenon-functionalrequirementsoftheproposedsolution.

FRNo.	Non-FunctionalRequirement	Description
NFR-1	**Usability**	Thesmartcropprotectionalertsthefarmersincaseofanyobstaclesandhelpsinprotectingthecrops
NFR-2	**Security**	SmartAgriculturecanimprovethefarmingpracticesandmaintainsustainableproductionofcrops especially by preventing the animals into theagriculturallandsthroughIoTenableddevices
NFR-3	**Reliability**	With a proper power supply, SD card andprogrammingtheprocessorshouldbeabletorun24/7 for years. The SD card and power supply willlikely wear out faster than the Pi. The possiblereasonsbehindRaspberryPifailurecanbepower breakdowns,SDcardfailures,andineligibleenvironments.
NFR-4	**Performance**	UsageofanSDcardmodulethathelpstostoreaspecifiedsoundtoscaretheanimals. Cropdamageduetoanimalattackcanbesensed.NetworkandDesignEvaluation
NFR-5	**Availability**	Agriculturefordifferentvarietyofcropsisbasedonthe monsoon changes, indoor and outdoor climatictemperatures,availabilityofrainfallandirrigation methods.
NFR-6	**Scalability**	The product shall be made available to everyoneespecially in remote areas for better efficiency ofcropyieldwiththebettersafetyofcropsa

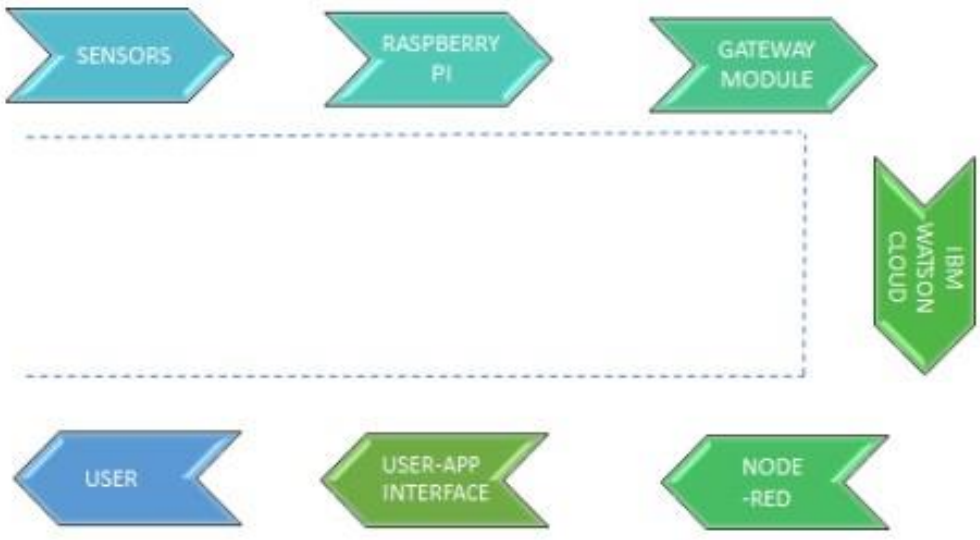
		swell asthefarmers.

5PROJECTDESIGN

5.1 DataFlowDiagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the informationflows within a system. A neat and clear DFD can depict the right amount of the systemrequirement graphically. It shows how data enters and leaves the system, what changes theinformation,andwheredataisstored.

DATA FLOW DIAGRAM:

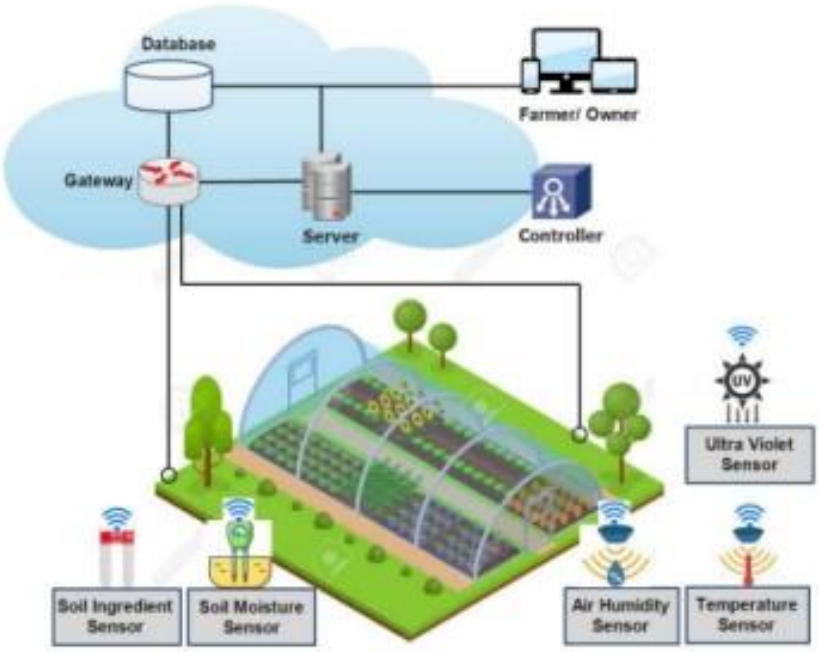


5.2 SOLUTIONANDTECHNICALARCHITECTURE

1. Summary:

The smart crops are constructed based on the sensor application and raspberry pi . It can also act as a transceiver since it is connected to the mobile phone of the user . The overall process of the sensors and raspberry pi is monitored using real time monitor which can help data transmission.This is stored and formulated using cloud data . Through which the admin can access the data and then track the location from GPS .

SOLUTION ARCHITECTURE :



2. Components&Technologies:

S.No	Component	Description	Technology
1.	UserInterface	Howuserinteractswith applicationе.g. WebUI,MobileApp, Chatbotetc.	HTML,CSS,JavaScript/AngularJs/ReactJsetc.
2.	Application Logic-1	Logicforaprocessintheapplication	Java/Python
	Database	DataType,Configurationsetc.	MySQL,NoSQL,etc
4.	IoT	To collect the data and alert the users	IBM Watson IoT Platform, Node Red.
5.	Cloud Database	Database ServiceonCloud	Cloudant DB

3. ApplicationCharacteristics:

S.No	Characteristics	Description	Technology
1.	Open Source framework	Listtheopen-sourceframeworksused	TechnologyofOpensourceframework
2.	Security implementation	Listallthesecurity/accesscontrolsimplemented,useoffirewallsetc .,	e.g. SHA-256, Encryptions,IAMControls,OWASPetc.
3.	Scalable Architecture	Justifythescalabilityofarchitecture(3 –tier,Micro-services)	Technologyused
4.	Availability	Justifytheavailabilityofapplication(e.g.use of load balancers, distributedserversetc.)	Technologyused
5.	Performance	Design consideration for theperformance of the application(numberofrequestspersec,useof Cache,useofCDN's)etc.	Technologyused

5.3 UserStories

Usethebelowtemplatetolistalltheuserstoriesfortheproduct.

UserType	FunctionalRequirement(Epic)	UserStory Number	UserStory/Task	Acceptance criteria	Priority	Release
Customer(Farmer)	Maintain fields	USN-1	Asauser,Icanmonitorthegrowthofcropsandprotectthecropsagainstanimals	Icanmaintainthefieldswithlesslabor	High	Sprint-1
	Analyzing problem	USN-2	As auser,Icollectthere quired information aboutthepro blemsonagr iculturefiel ds	Icanaskmyfieldownerdirectly.	low	Sprint-2
ProjectDesigners	Identifying theproblema nd providesolutions	USN-3	Asauser,Icanse nsethewaterlevelandflame in the field using sensor andmonitorusingIOT	IcanperformthisactionsviaIoT.	Medium	Sprint-1
Customer field Maintainer	Problem solution	USN-4	Asauser,areascanbemonitoredfromaremoteplace	CheckingProcess	Medium	Sprint-3
	Final process	USN-5	Thispropos edsmartIOT - basedcroppr otectiondev iceisfoundt obe cost-effectiveandefficient	Icantakenecessaryactionifrequired	High	Sprint-4

6. PROJECTPLANNINGANDSCHEDULING

6.1 SPRINTPLANNING&SCHEDULING:

TITLE	DESCRIPTION	DATE
LiteratureSurvey &InformationGathering	Literature survey on the selected project isdonebygatheringinformationaboutrelatedetailsontechnical papers and webbrowsing.	28 SEPTEMBER 2022
Prepare EmpathyMap	PreparedEmpathy Map Canvastocapture the user Pains & Gains list of problem statements.	24 SEPTEMBER 2022
Ideation	List the organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility and importance.	25 SEPTEMBER 2022
ProposedSolution	Preparedtheproposedsolution document, whichincludesthenovelty, feasibilityofidea,business model,socialimpact,scalabilityofsolution,etc.	23 SEPTEMBER 2022
ProblemSolutionFit	Prepared problem - solutionfitdocument.	30 SEPTEMBER 2022
Solution Architecture	Prepare solution architecture document.	28 SEPTEMBER 2022

6.2 SPRINTDELIVERYSCHEDULE

ProductBacklog,SprintSchedule,andEstimation

Usethe below template to create product backlog and sprint schedule.

Product Backlog, Sprint Schedule, and 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	Umamaheswari
Sprint-1		USN-2	As a user, I will receive confirmation email and the SMS once I have registered for the application	2	High	Umamaheswari
Sprint-2	Cloud services	USN-3	As a user, I can register for the application through Facebook or any social media	1	Low	Abimanyu
Sprint-4		USN-4	As a user, I can register for the application through Gmail/web service	2	Medium	Dharani
Sprint-3	Login	USN-5	As a user, I can log into the application network by entering email & password	4	High	Venkatesh
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	Dharani
Sprint-1	Collecting Dataset	USN-7	To collect various sources of animal threats and keep developing a dataset.	3	Medium	Umamaheswari
Sprint-4	Integrating	USN-8	To integrate the available dataset and keep improving the accuracy of finding animals	2	High	Abimanyu
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	Venkatesh
Sprint-2		USN-10	Request Saveetha Engineering College to deploy the project in our campus and test	1	Low	Dharani
Sprint-1	Training	USN-11	As programmer, we need to train our data perfectly so that the program runs smoothly	3	High	Umamaheswari
Sprint-3		USN-12	Train the data using out available services and IBM dataset from server and improve that	2	Medium	Venkatesh
Sprint-4	Coding	USN-13	To modify the code according to our program and improve the efficiency of that code	4	High	Abimanyu
Sprint-2		USN-13	To improve performance	1	Low	Venkatesh
Sprint-2	Record	USN-5	To record the data and plot the graph to show the characteristics officially	4	High	Venkatesh
Sprint-1	Planning	USN-4	Plan the programming language and feasibility	3	Medium	Umamaheswari Abimanyu
Sprint-4		USN-14	Demonstrate the working and improve accuracy overall	2	Low	Abimanyu

ProjectTracker,Velocity&BurndownCharts

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Plannd)	Story Points Completed(as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	5Days	20Oct2022	24Oct2022	20	21Oct2022
Sprint-2	20	5Days	25Oct2022	29Oct2022	20	27Oct2022
Sprint-3	20	5Days	31Oct 2022	4Nov2022	20	02Nov2022
Sprint-4	20	7Days	5Nov 2022	11Nov2022	20	08Nov2022

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	5 Days	20 Oct 2022	24 Oct 2022	20	21 Oct 2022
Sprint-2	20	5 Days	25 Oct 2022	29 Oct 2022	20	27 Oct 2022
Sprint-3	20	5 Days	31 Oct 2022	4 Nov 2022	20	2 Nov 2022
Sprint-4	20	7 Days	5 Nov 2022	11 Nov 2022	20	8 Nov 2022

Velocity:

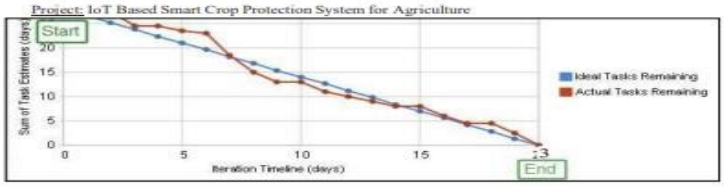
We have a 23-day sprint duration, and the velocity of the team is 20 (points per sprint).

To Find: Calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{23}{20} = 1.15$$

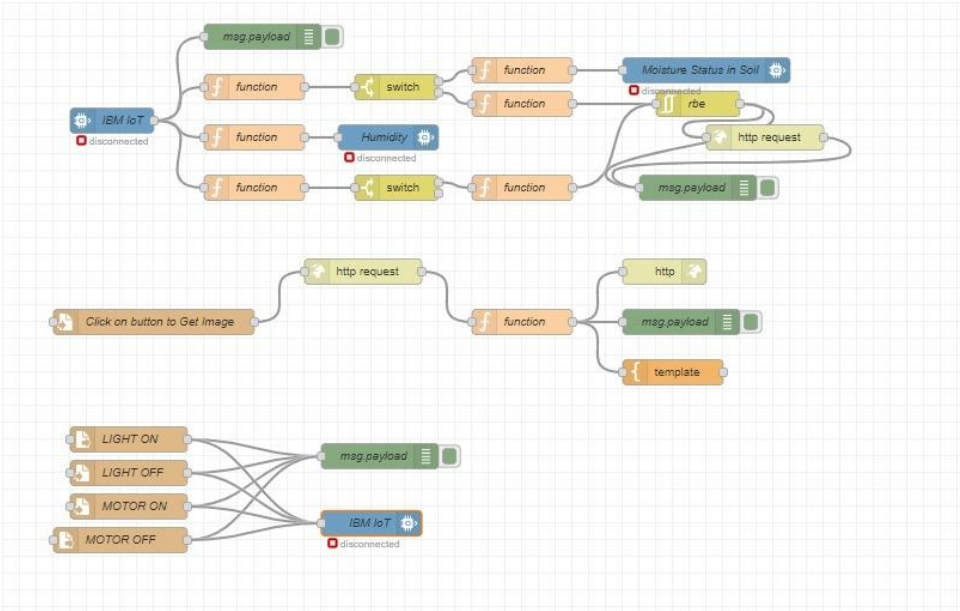
Burndown Chart:

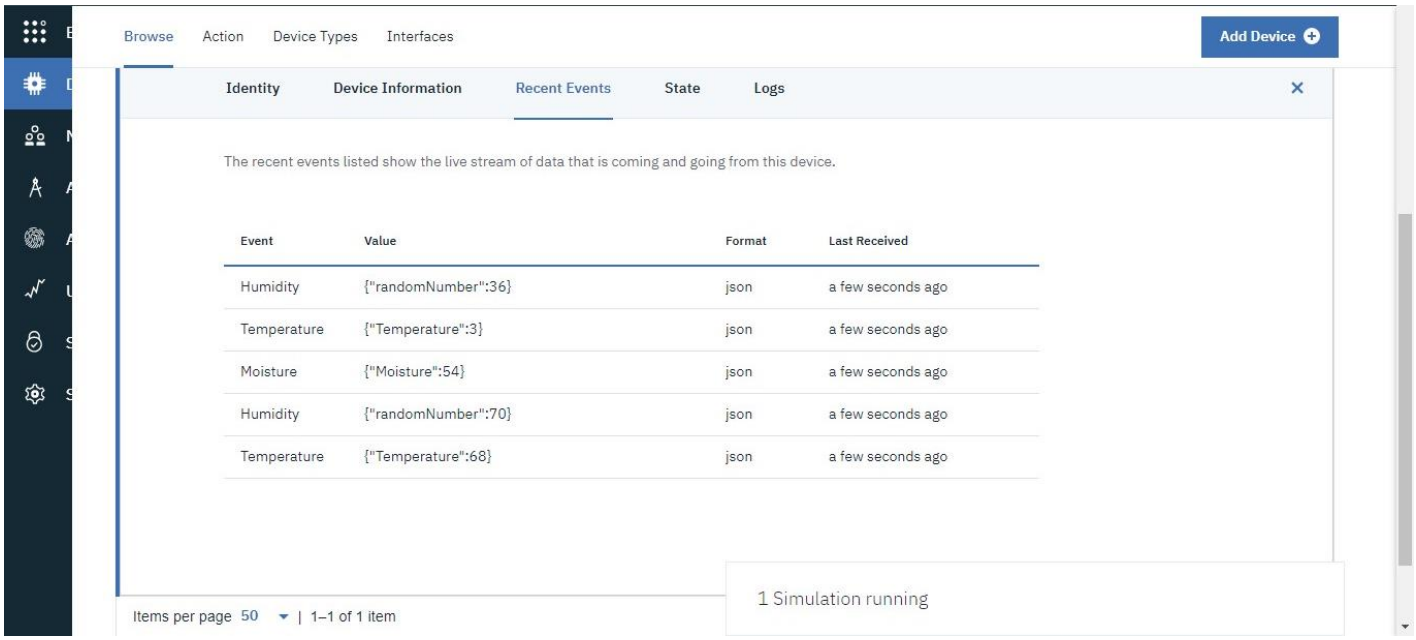
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



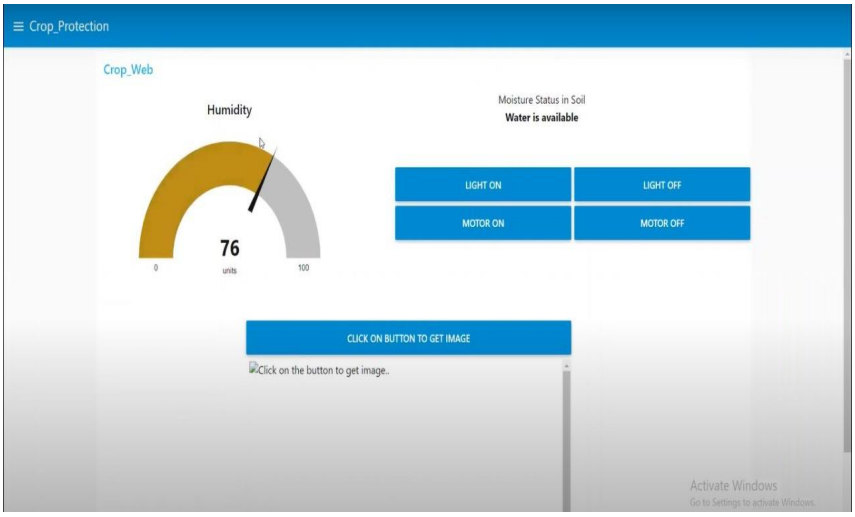
7. CODINGANDSOLUTIONING

NODE RED SERVICE ASSOCIATED WITH IBM CLOUD:





Node red Dashboard:



9. CONCLUSION

We presented an intelligent Smart crop protection system. The system is based on IoT sensors. It is responsible formeasuring the waste level in the smart crop. When the smart crop gets affected almost there will be information received by the admin, Since the admin can access the data and location of the crop. Later send this data (through Internet) to a server for storage and processing.

This data helps to compute the optimized collection routes for the workers. In future, we would like to enhance the system for different kind of crop management system .

10. FUTURESCOPE

The advantage of this work is its contribution in making a Smart crop. Among the many challenges that a city faces, crop protection management is of utmost importance. This is because, it is directly related tofood of people living in the area. We are further extending his work to address problems of seggragating different kind of crops (e.g.,paddy ,wheat ,etc.,), and identifying different agricultural department for collecting it. The optimization algorithms may bed evised accordingly depending on the requirements. In future, we would like to enhance the system for different kind of crops .

11.APPENDIX

11.1 SOURCECODE:

PYTHONCODETOPUBLISHDATA

```
import cv2
importnumpy as np
importwiot.sdk.device
importplaysound
import random
import time
importdatetime
import ibm_boto3
fromibm_botocore.client import Config, ClientError

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

#This is how you authenticate
metadata = (('authorization', 'key 0620e202302b4508b90eab7efe7475e4'),)
COS_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID = "g5d4qO8EIgv4TWUCJj4hfEzgalqEjrDbE82AJDWlAOHo"
COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"
```

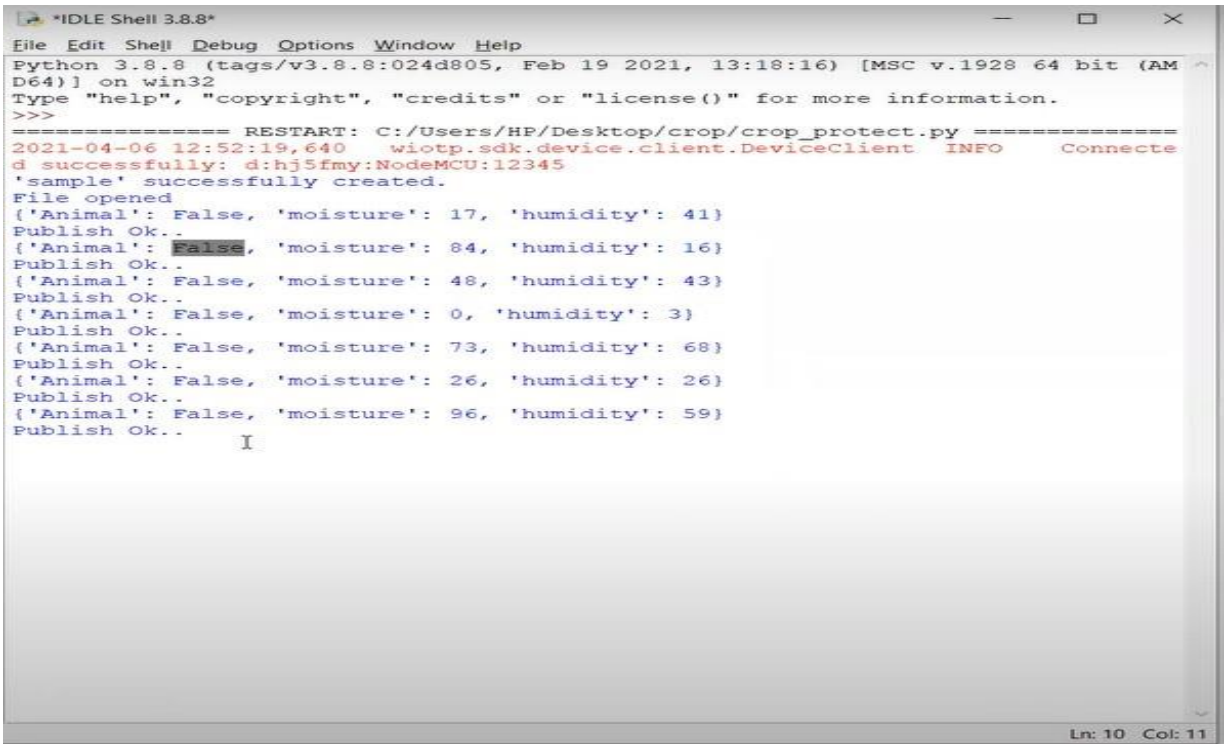
```
COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloud-object-
storage:global:a/c2fa2836eaf3434bbc8b5b58fefff3f0:62e450fd-4c82-4153-ba41-ccb53adb8111::"
clientdb = cloudant("apikey-W2njldnwtjO16V53LAVUCqPwc2aHTLmlj1xXvtdGKJBn",
"88cc5f47c1a28afbfb8ad16161583f5a", url="https://d6c89f97-cf91-48b7-b14b-c99b2fe27c2f-
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
```

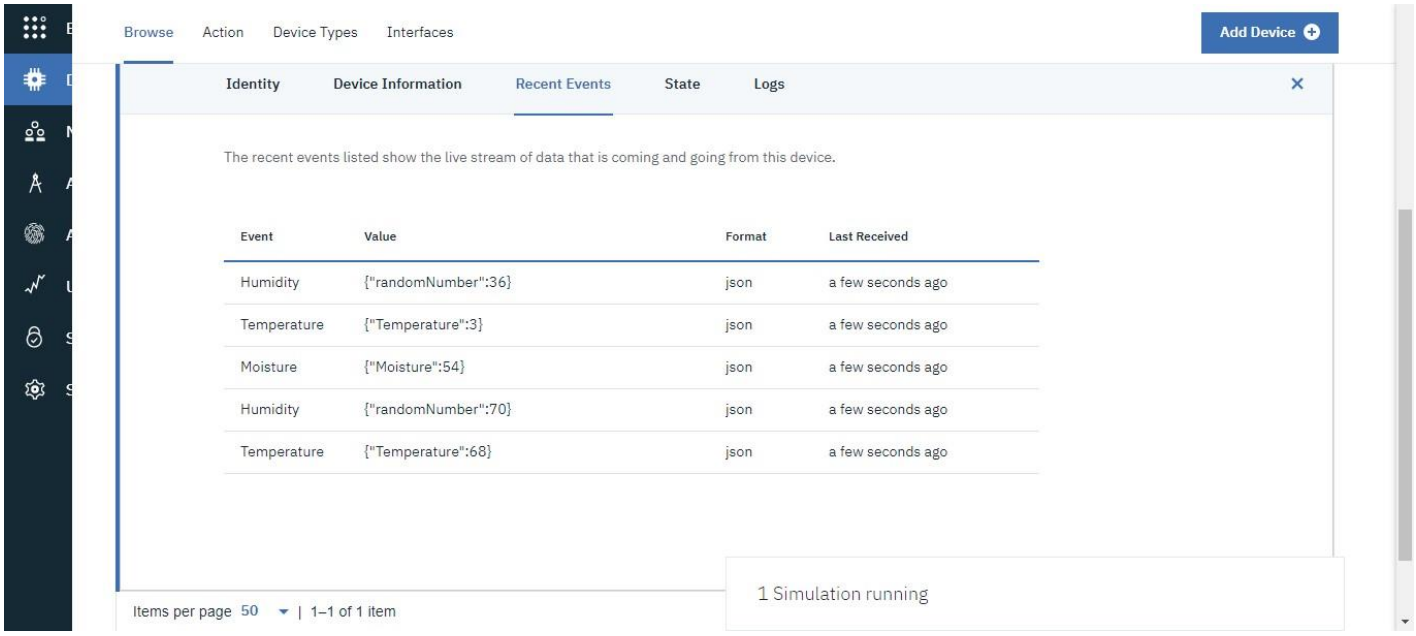
```
            )
print("Transfer for {0} Complete!\n".format(item_name))
exceptClientError as be:
print("CLIENT ERROR: {0}\n".format(be))
except Exception as e:
print("Unable to complete multi-part upload: {0}".format(e))

defmyCommandCallback(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"
```

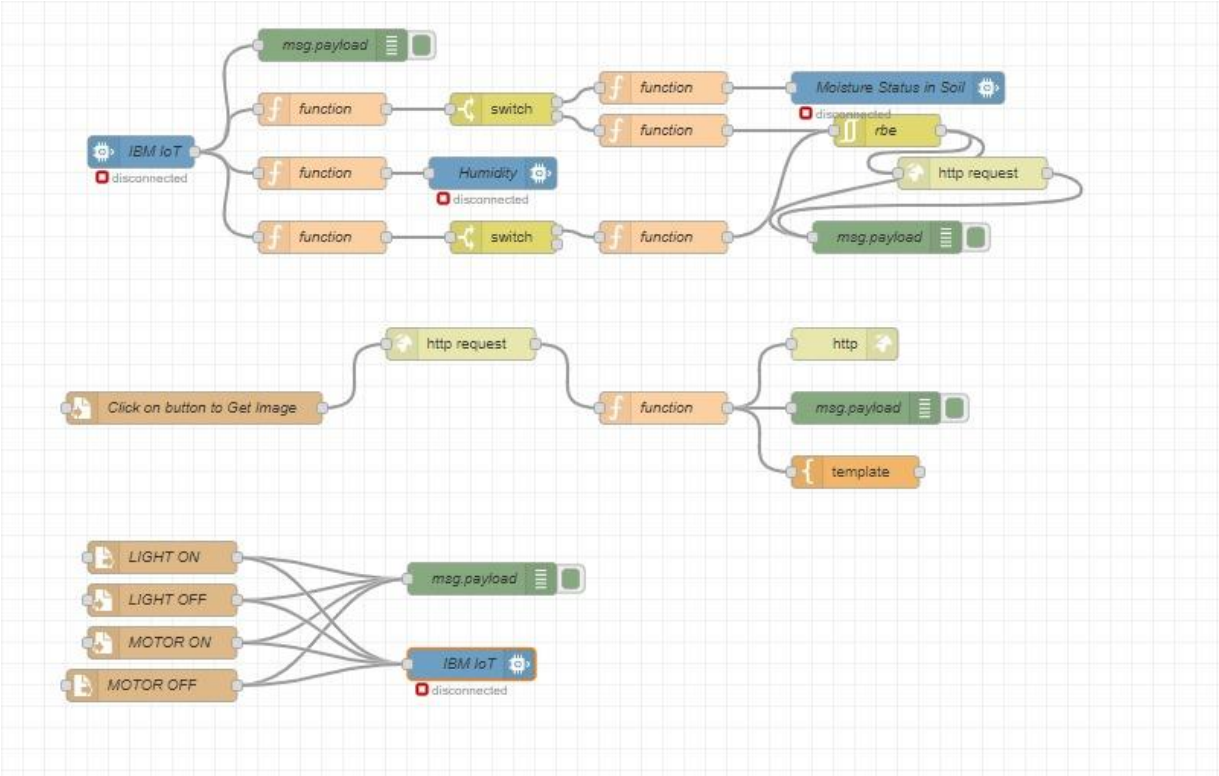
```
my_database = clientdb.create_database(database_name)
ifmy_dtabase.exists():
print(f"'(database_name)' successfully created.")
cap=cv2.VideoCapture("garden.mp4")
if(cap.isOpened()==True):
print('File opened')
else:
print('File not found')

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='e9359dbe6ee44dbc8842ebe97247b201',
inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_byt
es))
                                )])
response = stub.PostModelOutputs(request, metadata=metadata)
ifresponse.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('Dhakshesh', picname+'.jpg', picname+'.jpg')
        json_document={"link":COS_ENDPOINT+'/'+'Dhakshesh'+'/'+picname+'.jpg'}
new_document = my_database.create_document(json_document)
ifnew_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()
```

OUTPUT

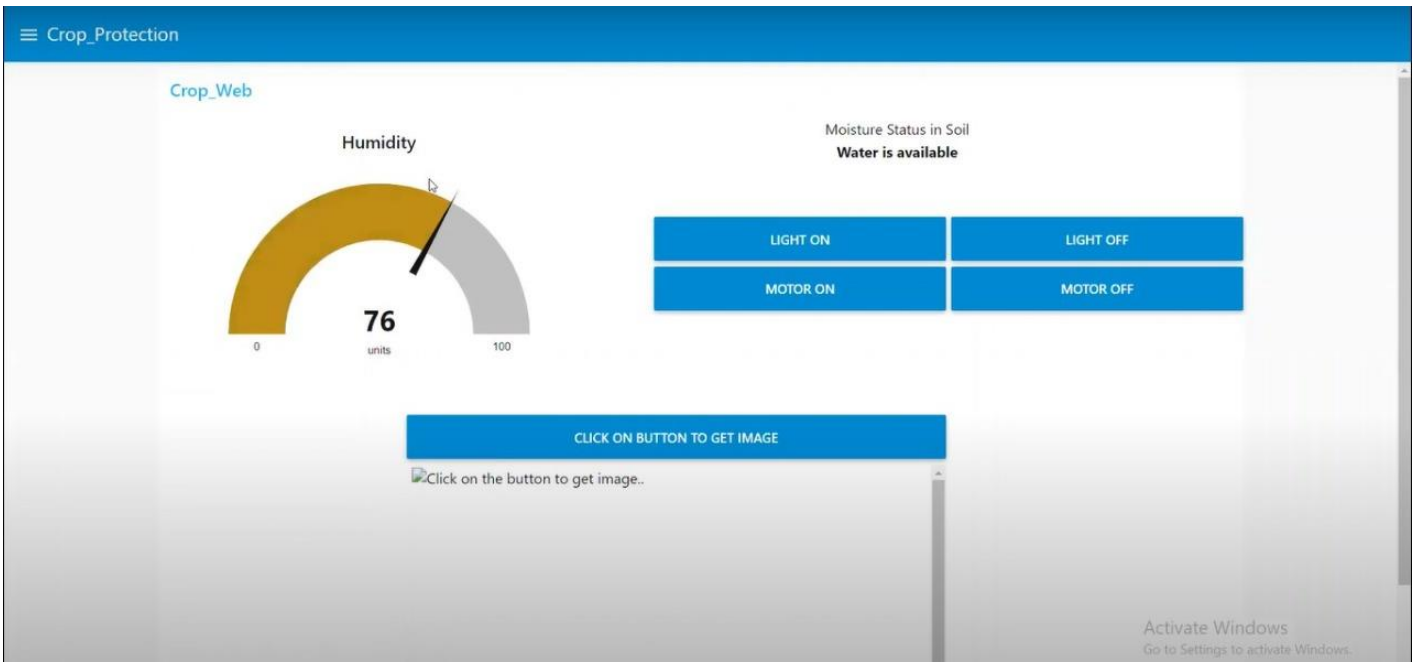

IDLE Shell 3.8.8
File Edit Shell Debug Options Window Help
Python 3.8.8 (tags/v3.8.8:024d805, Feb 19 2021, 13:18:16) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:/Users/HP/Desktop/crop/crop_protect.py ==========
2021-04-06 12:52:19,640 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:hj5fmy:NodeMCU:12345
'sample' successfully created.
File opened
{'Animal': False, 'moisture': 17, 'humidity': 41}
Publish Ok..
{'Animal': False, 'moisture': 84, '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..
Ln: 10 Col: 11


Browse
Action
Device Types
Interfaces
Add Device
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
Humidity {"randomNumber":36} json a few seconds ago
Temperature {"Temperature":3} json a few seconds ago
Moisture {"Moisture":54} json a few seconds ago
Humidity {"randomNumber":70} json a few seconds ago
Temperature {"Temperature":68} json a few seconds ago
Items per page 50 | 1–1 of 1 item
1 Simulation running



TECH TO SPEECH:

```
fromibm_watson import TextToSpeechV1
fromibm_cloud_sdk_core.authenticators import IAMAuthenticator
importplaysound

authenticator = IAMAuthenticator('v9n8Zn4r5VpcMVz_HyRY0DrS13jSzph2IEFioVj4-vmT')
text_to_speech = TextToSpeechV1(
authenticator=authenticator
    )

text_to_speech.set_service_url('https://api.eu-gb.text-to-speech.watson.cloud.ibm')

with open('alert.mp3', 'wb') as audio_file:
audio_file.write(
text_to_speech.synthesize(
'Alert! Alert! Animal Detected.',
voice='en-US_ALLisonV3Voice',
accept='audio/mp3'
        ).get_result().content)
playsound.playsound('alert.mp3')
```


Crop_Protection
Crop_Web
Humidity
76
units
0
100
Moisture Status in Soil
Water is available
LIGHT ON
LIGHT OFF
MOTOR ON
MOTOR OFF
CLICK ON BUTTON TO GET IMAGE
Click on the button to get image..
Activate Windows
Go to Settings to activate Windows.