MAHENDRA ENGINEERING

COLLEGE WOMEN

Category:INTERNETOFTHINGS

A PROJECT REPORT

Submittedby

T. Sneka (TEAM LEADER)

S. Sathiyabama

P. Sivasakthi

C. Tamilarasi

ECHNOLOGTECHN

COMPUTER SCIENCE AND ENGINEERING

NALAIYATHIRAN2022TeamId:PNT2022TMID30395

PROJECTGUIDES

IndustryMentor: Mr. Dinesh

FacultyMentor:J P KAVITHA

INDEX

OLUTION

a. **EmpathyMapCanvas**

b. **Ideation&Brainstorming**

c. **ProposedSolution**

d. **Problem**

Solution

fit4.REQ

UIREME

NTANAL

YSIS

a. **Functionalrequirement**

b. **Non-Functional**

requirements5.PR

OJECTDESIGN

a. **DataFlowDiagrams**

b. **Solution&TechnicalArchitecture**

c. **UserStories**

i. INTRODUCTION

a. 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.

b. Purpose:

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

2. 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.

3. 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 .

4. 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.

ii. LITERATURESURVEY

a. ExistingProblem:

5. At present, we can see crop are being damaged due to many reasons. Our primary

goal is to protect the crop from being damaged.

a. 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.

b. 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

c. 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.

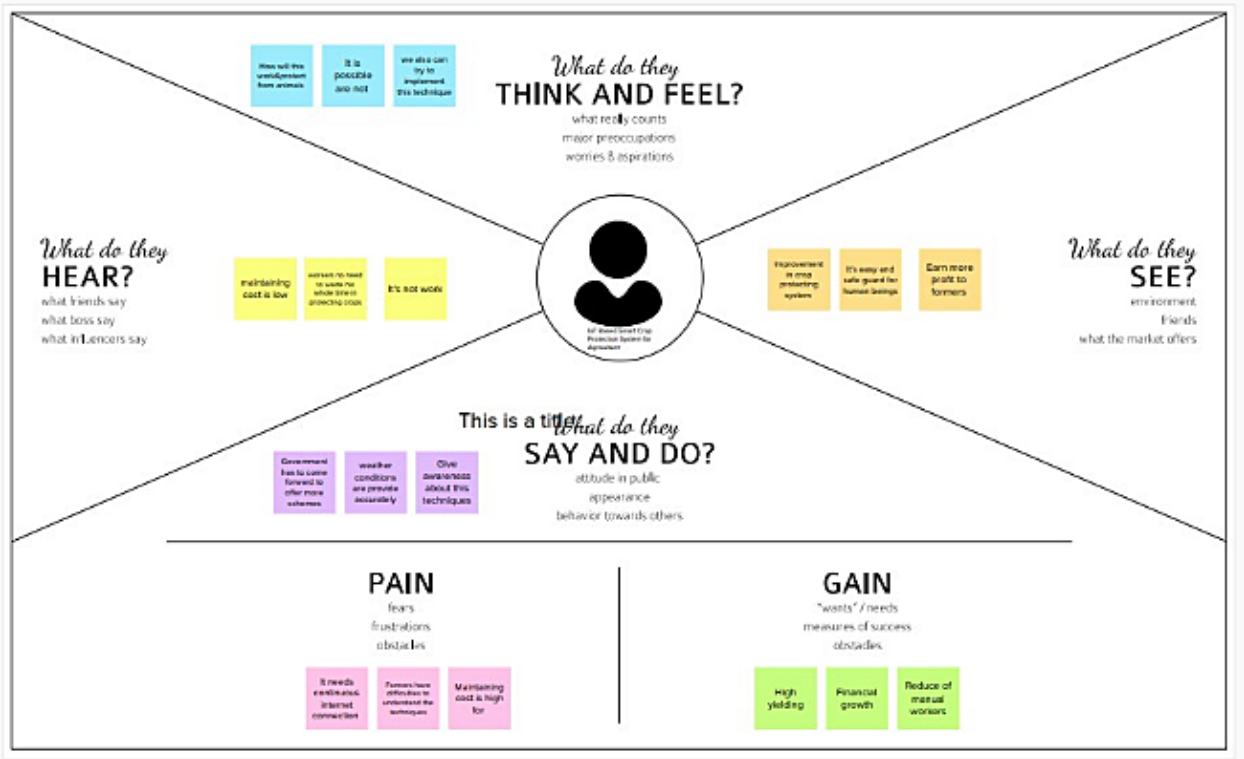
b. ProblemStatement Definition:

6. 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

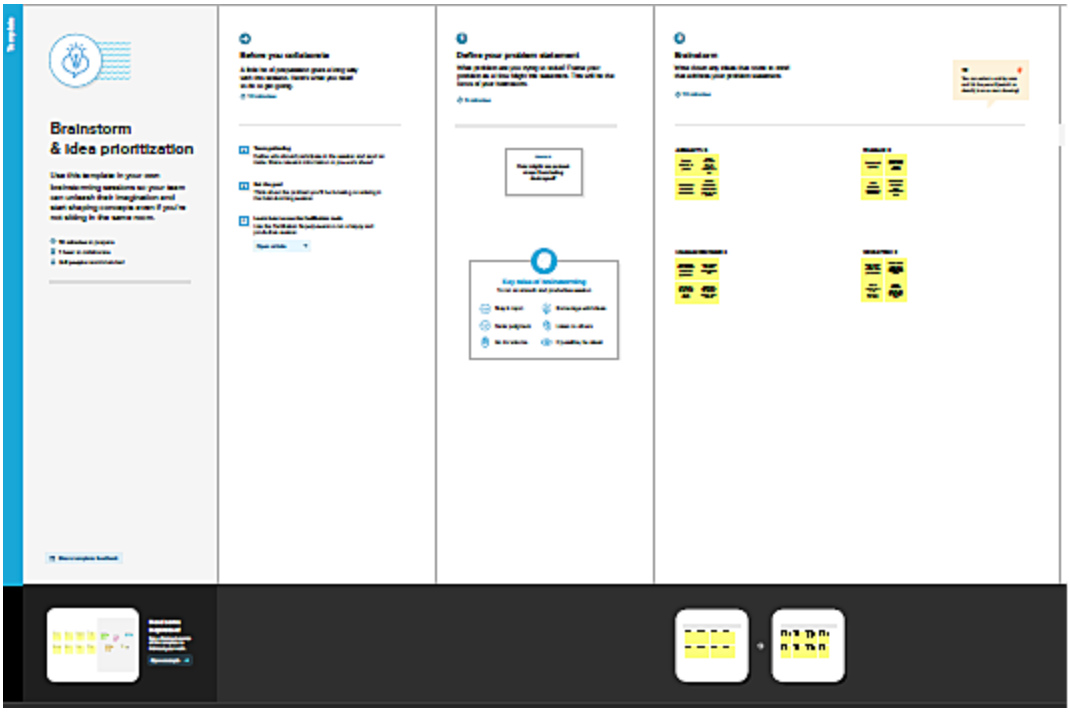
a. EmpathyMapCanvas:

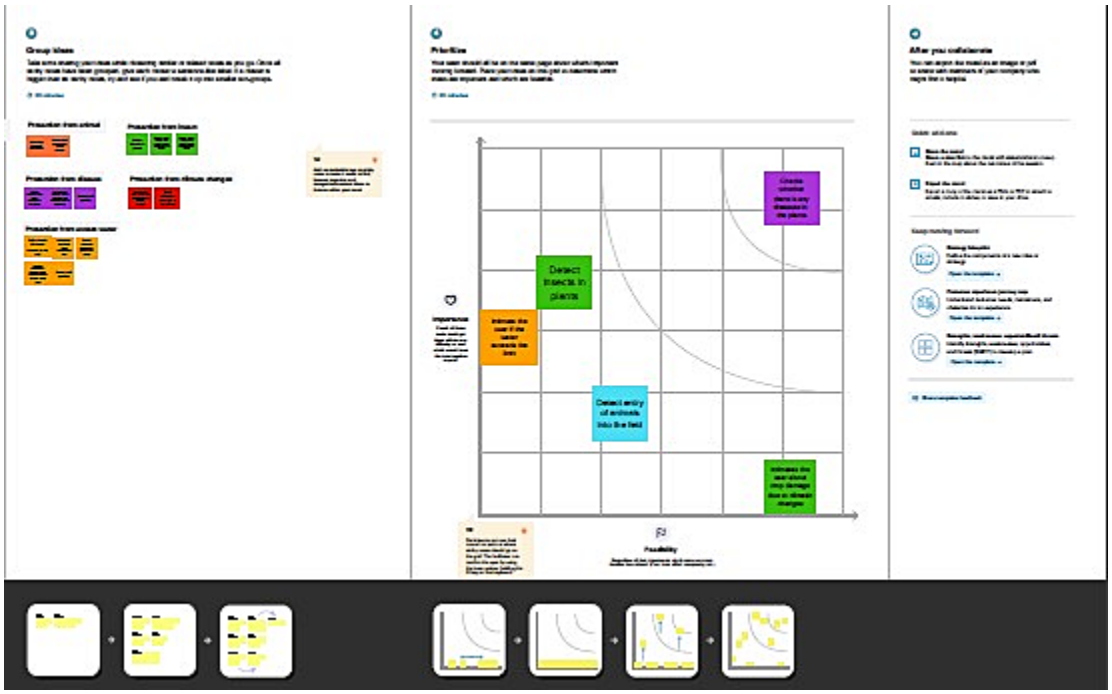
An empathy map is a simple, easy-to-digest visual that captures knowledge about auser'sbehavioursandattitudes.Itisausefultooltohelpsteamsbetterunderstandtheirusers.Creating an effective solution requires understanding the true problem and the personwho is experiencing it. The exercise of creating the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



b. 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.





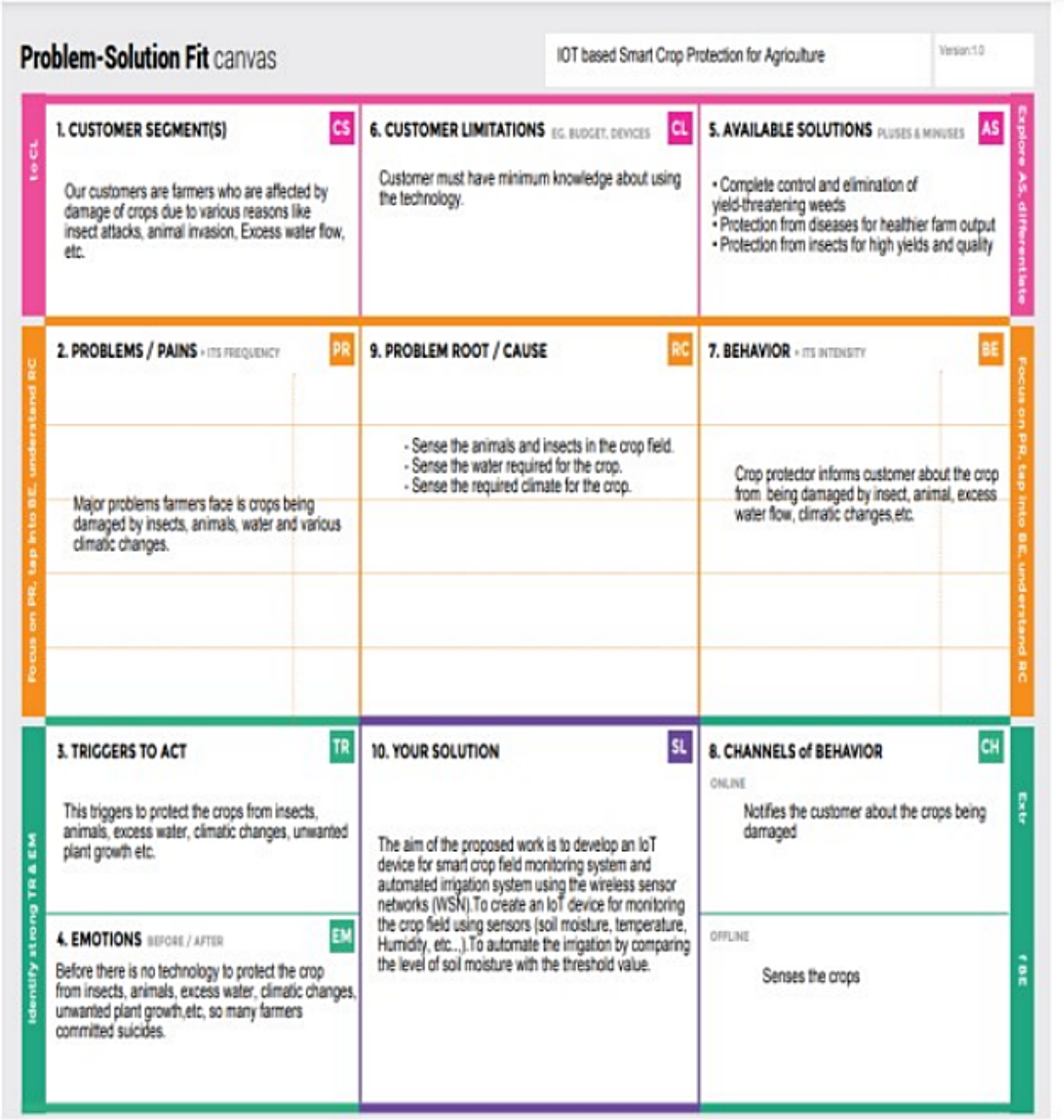
C. ProposedSolution:

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.	SocialImpact/CustomerSatisfacti on	• Cost effective to the society • Modernaization to the society • High protection and High yield

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

d. PROBLEMSOLUTION FIT:



4REQUIREMENTANALYSIS

a. ***FunctionalRequirements:***

Followingarethefunctionalrequirementsoftheproposedsolution.

FRNo.	**FunctionalRequirement (Epic)**	**SubRequirement(Story/Sub-Task)**
FR-1	UserRegistration	Registration through FormRegistrationthroughGmail .
FR-2	UserConfirmation	ConfirmationviaEmail .
FR-3	Interfacingwithhardware	Interfacethesensorswiththesoftwareapplicationsoastoalertthefarmersin ps .
FR-4	DatabaseConnection	DatabasesareretrievedfromIBMCloudant .

FR-5	MobileApplication	Alarmandmotorscanbeaccessedfromthemobileapp .

b. ***Non-functionalRequirements:***

Followingarethenon-functionalrequirementsoftheproposedsolution.

FRNo.	Non-FunctionalRequirement	Description
NFR-1	**Usability**	Thesmartcropprotectionalertsthefarmersincaseofanyobstaclesandhelpsinprotecting
NFR-2	**Security**	SmartAgriculturecanimprovethefarmingpracticesandmaintainsustainableproductio ps 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.

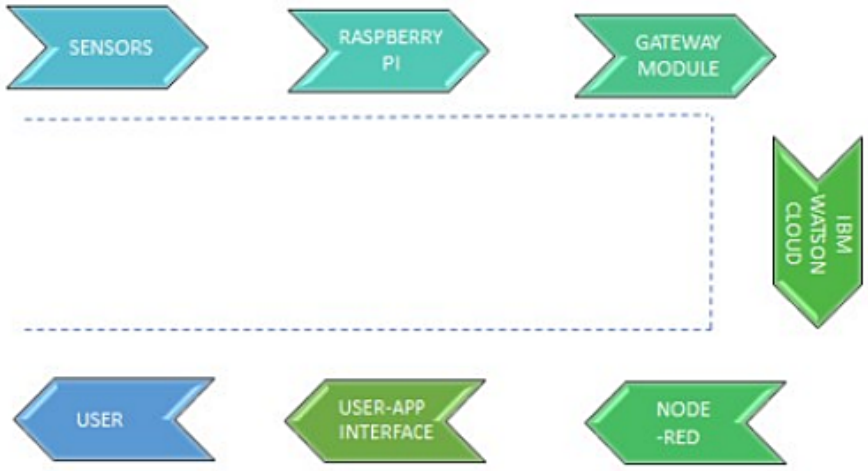
NFR-4	**Performance**	UsageofanSDcardmodulethathelpstostoreaspecifiedsoundtoscaretheanimals. Cropdamageduetoanimalattackcanbesensed.NetworkandDesignEvaluation
NFR-5	**Availability**	Agriculturefordifferentvarietyofcropsisbasedonthe monsoon changes, indoor outdoor climatictemperatures,availabilityofrainfallandirrigation methods.
NFR-6	**Scalability**	The product shall be made available to everyoneespecially in remote areas for bett efficiency ofcropyieldwiththebettersafetyofcropsaswell asthefarmers.

5PROJECTDESIGN

a. 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:

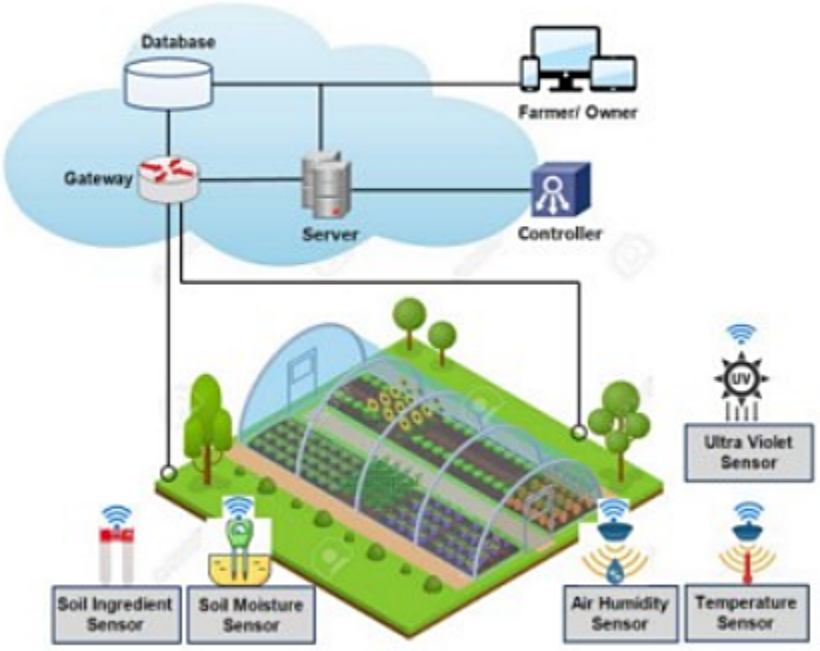


b. SOLUTIONANDTECHNICALARCHITECTURE

i. 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 :



ii. Components&Technologies:

S. No	Component	Description	Technology
1.	UserInterface	Howuserinteractswithapplicatione.g. WebUI,MobileApp, Chatbotetc.	HTML,CSS,JavaScript/AngularJs/ReactJsetc.
		Logicforaprocessintheapplication	Java/Python

2.	Application Logic-1		
	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

iii. **ApplicationCharacteristics:**

S. No	Characteristics	Description
1.	Open Source framework	Listtheopen-sourceframeworksused
2.	Security implementation	Listallthesecurity/accesscontrolsimplemented,useoffirewall tc .,

3.	Scalable Architecture	Justifythescalabilityofarchitecture(3 –tier,Micro-services)
4.	Availability	Justifytheavailabilityofapplication(e.g.use of load balancers distributedserversetc.)
5.	Performance	Design consideration for theperformance of the application(numberofrequestspersec,useof Cache,useofCDN's)etc.

C. UserStories

Usethebelowtemplatetolistalltheuserstoriesfortheproduct.

UserType	FunctionalRequirement(Epic)	UserStory Number	UserStory/Task
Customer(Farmer)	Maintain fields	USN-1	Asauser,Icanmonitorthegrowthof cropsandprotectthecropsagainstanimals

	Analyzing problem	USN-2	As auser,Icollecttherequired informationabouttheproblemsonagriculturefields
ProjectDesigners	Identifying theproblemand providesolutions	USN-3	Asauser,Icansensethewaterlevelandflame in the field using sensor andmonitorusingIOT
Customer field Maintainer	Problem solution	USN-4	Asauser,areascanbemonitoredfromaremoteplace
	Final process	USN-5	ThisproposedsmartIOT-basedcropprotectiondeviceisfoundtobe cost-effectiveandefficient

1. PROJECTPLANNINGANDSCHEDULING

a. SPRINTPLANNING&SCHEDULING:

TITLE	DESCRIPTION
LiteratureSurvey &InformationGathering	Literature survey on the selected project isdonebygatheringinformationaboutrela webbrowsing.
Prepare EmpathyMap	PreparedEmpathy Map Canvastocapture the user Pains & Gains list of problem stat

Ideation	List the organizing the brainstorming session and prior feasibility and importance.
ProposedSolution	Preparedtheproposedsolution whichincludesthenovelty,feasibilityofidea,businessmod c.
ProblemSolutionFit	Prepared problem -solutionfitdocument.
Solution Architecture	Prepare solution architecture document.

b. **SPRINTDELIVERYSCHEDULE**

ProductBacklog,SprintSchedule,andEstimation

Usethe below template to create product backlog and sprint schedule.

ProjectTracker,Velocity&BurndownCharts

Sprint	TotalStoryPoints	Duration	SprintStartDate	SprintEndDate(Planned)	StoryPointsConed EndDate)
Sprint-1	20	5Days	20Oct2022	24Oct2022	20
Sprint-2	20	5Days	25Oct2022	29Oct2022	20
Sprint-3	20	5Days	31Oct 2022	4Nov2022	20
Sprint-4	20	7Days	5Nov 2022	11Nov2022	20

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:

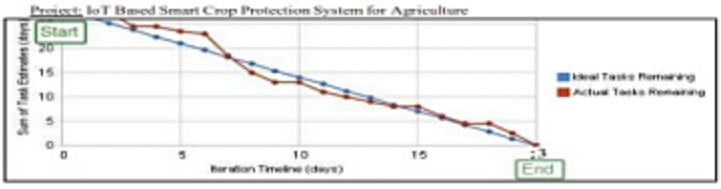
We have a 23-day sprint duration, and the velocity of the team is 20 (points per sprint).

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

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

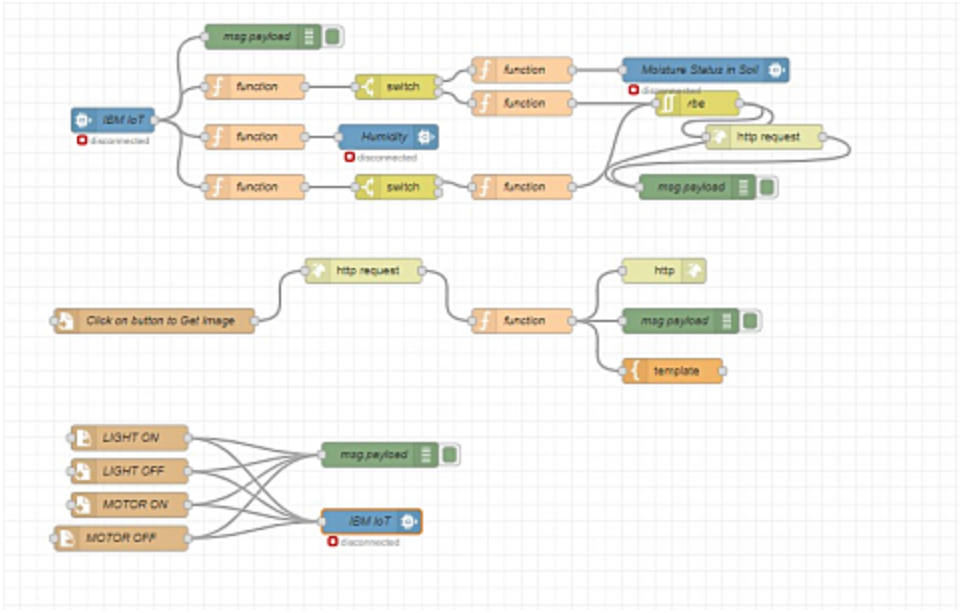
Burndown Chart:

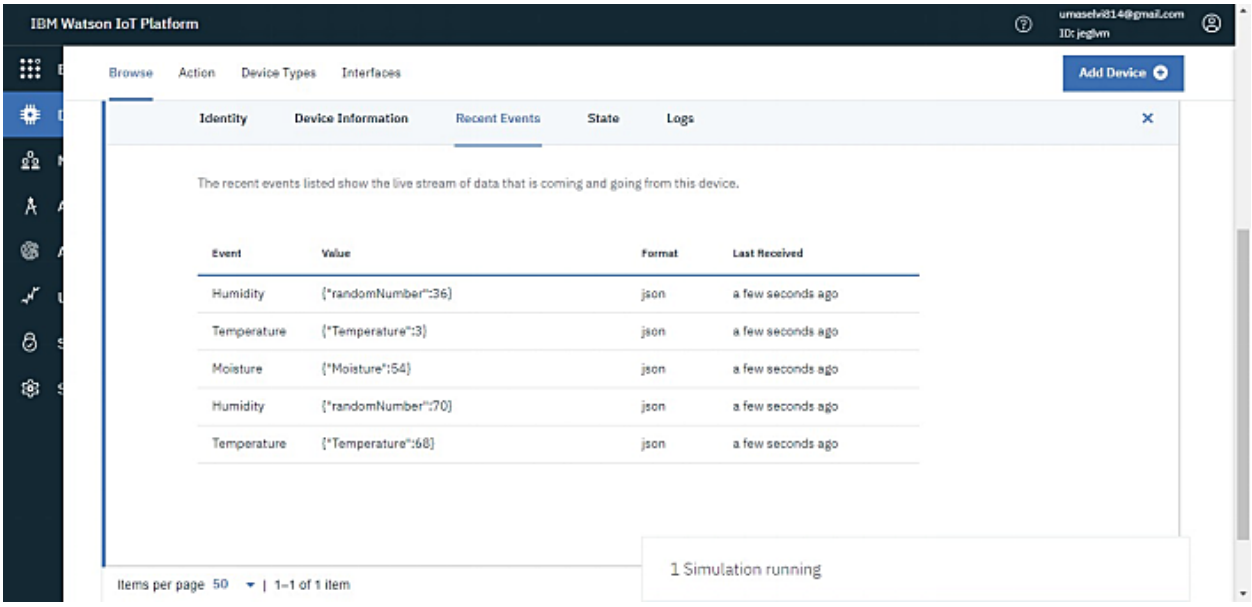
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



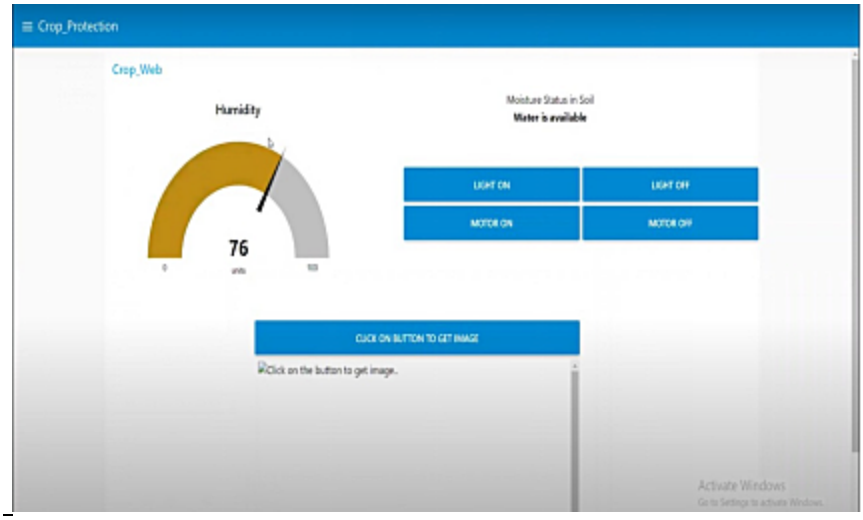
2. CODINGANDSOLUTIONING

NODEREDSERVICEASSOCIATEDWITHIBMCLOUD:





NoderedDashboard:



a. 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 thisdata (through Internet) to a server for storage and processing.This data helps to compute the optimized collection routes forthe workers. In future, we would like to enhance the systemfor different kind of crop management system .

b. FUTURESCOPE

The advantage of thiswork is its contribution in making a Smart crop. Among themany challenges that a city faces, crop protection management is ofutmost importance. This is because, it is directly related tofood of people living in the area. We are further extendingthis work to address problems of seggragating different kindof crops (e.g.,paddy ,wheat ,etc.,), and identifying differentagricultural department for collecting it. The optimization algorithms may bedevised accordingly depending on the requirements.In future, we would like to enhance the systemfor different kind of crops .

11.APPENDIX

a. 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:
```

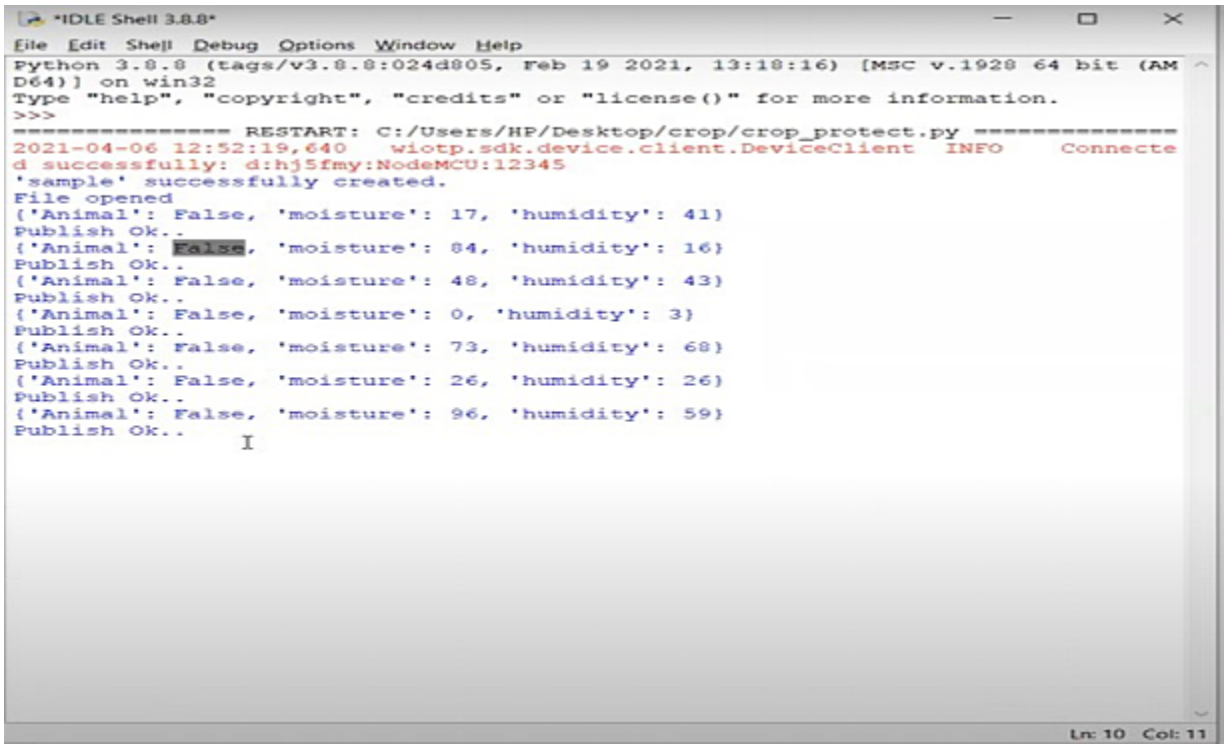
```
cos.Object(bucket_name, item_name).upload_fileobj(
Fileobj=file_data,
Config=transfer_config
            )
print("Transfer for {0} Complete!\n".format(item_name))
exceptClientError as be:
print("CLIENT ERROR: {0}\n".format(be))
except Exception as e:
print("Unable to complete multi-part upload: {0}".format(e))

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"
```

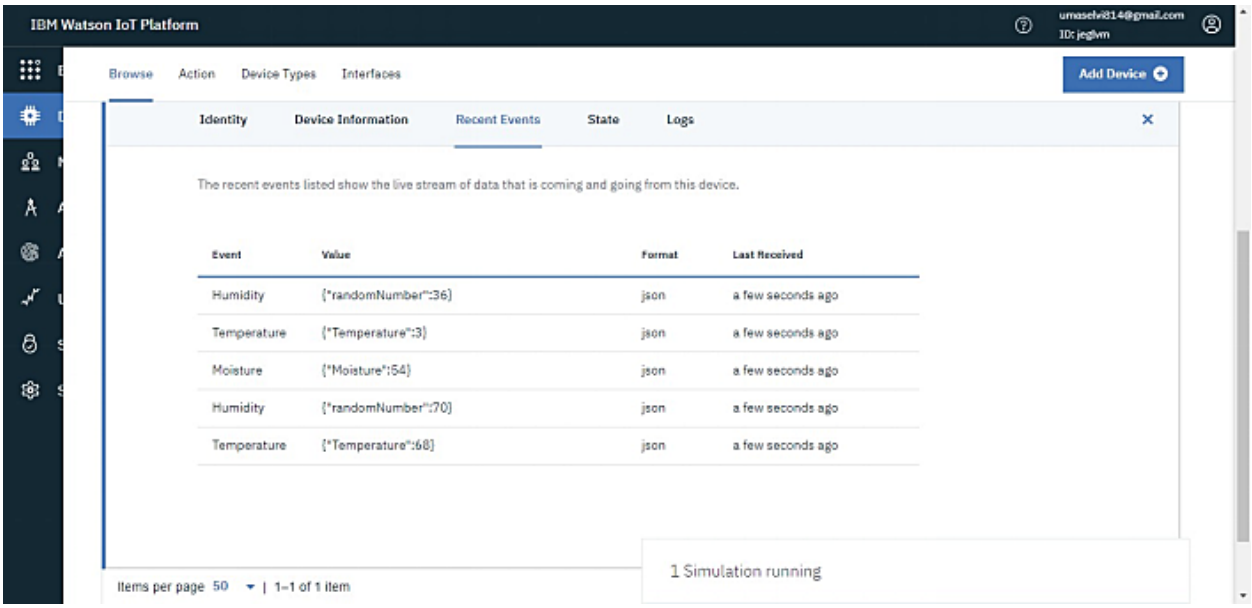
```
    },
  "auth": {
    "token": "12345678"
    }
  }
client = wiot.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

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.
```

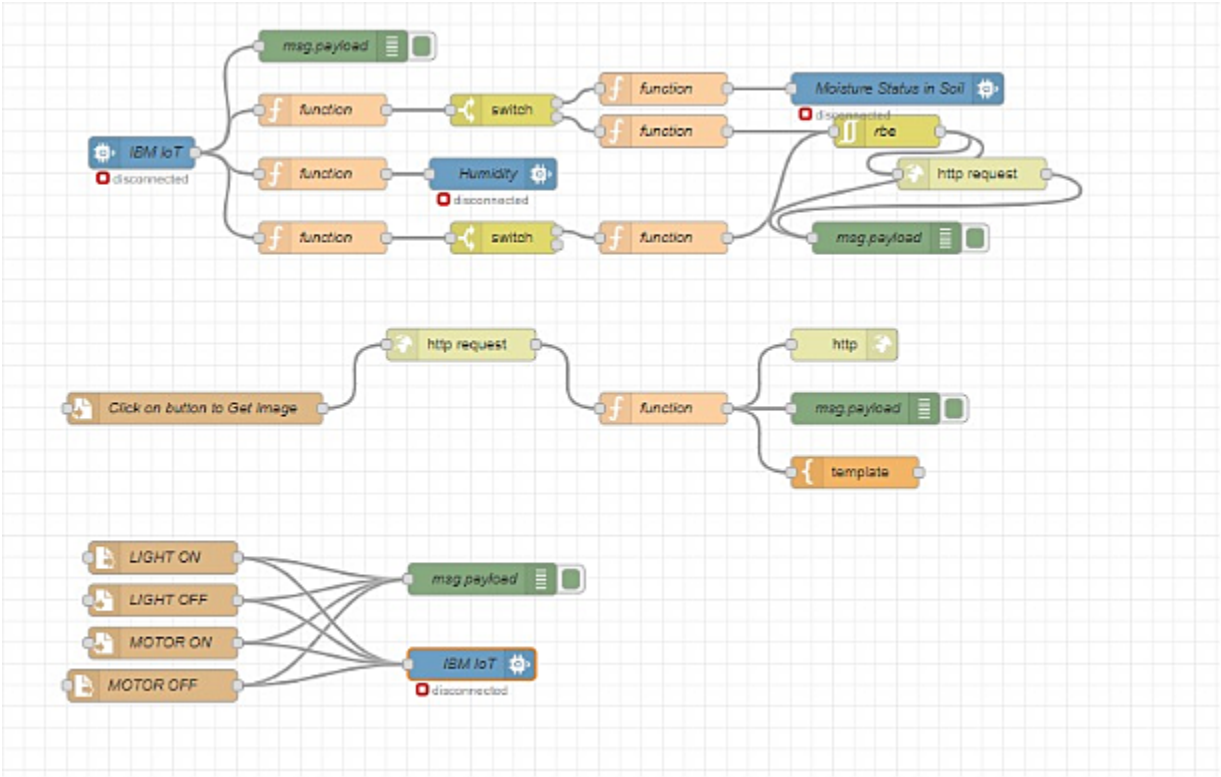
```
request = service_pb2.PostModeloutputsRequest(
model_id='e9359dbe6ee44dbc8842ebe97247b201',
inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base
64=file_bytes))
                                )])
response = stub.PostModelOutputs(request, metadata=metadata)
ifresponse.status.code != 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


IBM Watson IoT Platform
umasehi814@gmail.com
ID: jeglvm
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')
```