VELTECH HIGHTECH DR.RANGARAJAN DR.SAKUNTHALA ENGINEERING COLLEGE

IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

TEAM ID: PNT2022TMID22196

TEAM LEAD: ARTHIKA.V(113019106703)

TEAM MEMBER 1: DEEKSHA.G(113019106709)

TEAM MEMBER 2: GOWRI.R(113019106711)

TEAM MEMBER 3: GEETHALAKSHMI.V(113019106715)

FACULTY MENTOR NAME: SIVS SARAVANA BABU.S

INDUSTRY MENTOR NAME: DINESH

CONTENTS

INTRODUCTION

1.1 PROJECT OVERVIEW

Today, technology has penetrated every part of human life. But the contribution of technology to the field of agriculture is considerably low when compared to the other sectors, which saw an incremental growth over the last decade.The domain of Agriculture contributes the most to the Indian economy and about1/3rd of India's population is directly dependent on agriculture for their source of income.Considering this, even a small improvement in this sector will make a huge impact on theIndian economy and on the life of farmers. This helps farmers and consumers equally as it isthe consumers in the end, who get to enjoy low priced goods without deterioration in quality.To achieve this, we have to overcome the hurdles faced by farmers, which mostlyrevolve around crop disease, improper maintenance of crops, lack of details about the qualityof soil and intervention of animals and birds. To overcome this, in this project we propose 'Anintelligent crop protection system', the main objective of which is to improve the yield andincrease the profit for farmers. An intelligent crop protection system uses data from moisture,motion, temperature, humidity sensors and updates the data in real time in IBM cross platformIOT cloud interface. The motors and the sprinkling system are activated based on the datafrom the sensors. Also when the motion sensor detects motion, the farmer is notified with thatthrough the mobile application. This helps the farmers in protecting the crop from the animalsand birds which destroy the crop.And also ease up the maintenance process. The historical data from sensors are storedin cloud, so this can also be used for soil evaluation and this also helps to plan, which type ofcrops are to be planted in the upcoming seasons so that the yield is high.

1.2 PURPOSE

A vast majority of the people are invariably affected by the production of crops.Farmers, for example, rely on them for their survival. The consumers, on the other hand,depend on the crops as it provides them with a multitude of utilities. It therefore, becomesessential to protect and maintain these crops. The project aims at improving the farmer situation by preventing them from incurring losses due to the damage of crops. Crop failure also deteriorates the quality of the yield thereby decreasing the quality of living.

LITERATURE SURVEY

2.1 EXISTING PROBLEM

In real time, it was learnt that the size of the animal is found out by using several PIR sensors. PIR sensors can be used to determine the height of the animals instead of using a camera for image processing. This reduces the processing time and power. The crop protection is majorly dependent on the moisture content of the soil, the temperature and humidity of the surrounding environment. Additionally, tracking of the damaged crops location is done and the camera is activated only at that instant in order to capture the image. From the literatures survey performed it is evident that image based animal intrusion identification is not necessary in all situations because it requires high computation power, and the cost of the installation will be high when compared to that of a typical sensor based intrusion identification.

2.2 PROBLEM STATEMENT

An intelligent crop protection system helps the farmers in protecting the crop from the animals and birds which destroy the crop. This system also helps farmers to monitor the soil moisture levels in the field and also the temperature and humidity values near the field. The motors and sprinklers in the field can be controlled using the mobile application. The significant problem which raises the requirement of this project was that the traditional agriculture method consumes time,manual labor work and is also not cost efficient.The detected signal from the soil moisture sensor is processed by aconditional comparator circuit corresponding to different levels ofactual soil moisture content. A logic circuit follows the conditional circuit with its output signals used to activate a system of relaysthat control the power circuit of the motors used for water pumping. IOT is developing rapidly and widely applied in all wireless environments. In this paper, sensor technology and wireless networks integration of IOT technology has been studied and reviewed based on the actual situation of agricultural system.The problem is that the crops in the field cannot be continuously monitored by the farmer from animals,birds,temperature,humidity .We have to provide a solution to sage guard the crops from the animals ,birds and to continuously monitor the crops from the temperature, humidity and regular check of water level and soil moisture so that the crops don’t get damaged and the productivity will be increased.

REFERENCES

1. **Automatic control of Agriculture pumps based on soil Moisture sensing**

by Beza Negash Getu,Hussain A. Attia.

2.**A model for smart Agriculture Using IOT**

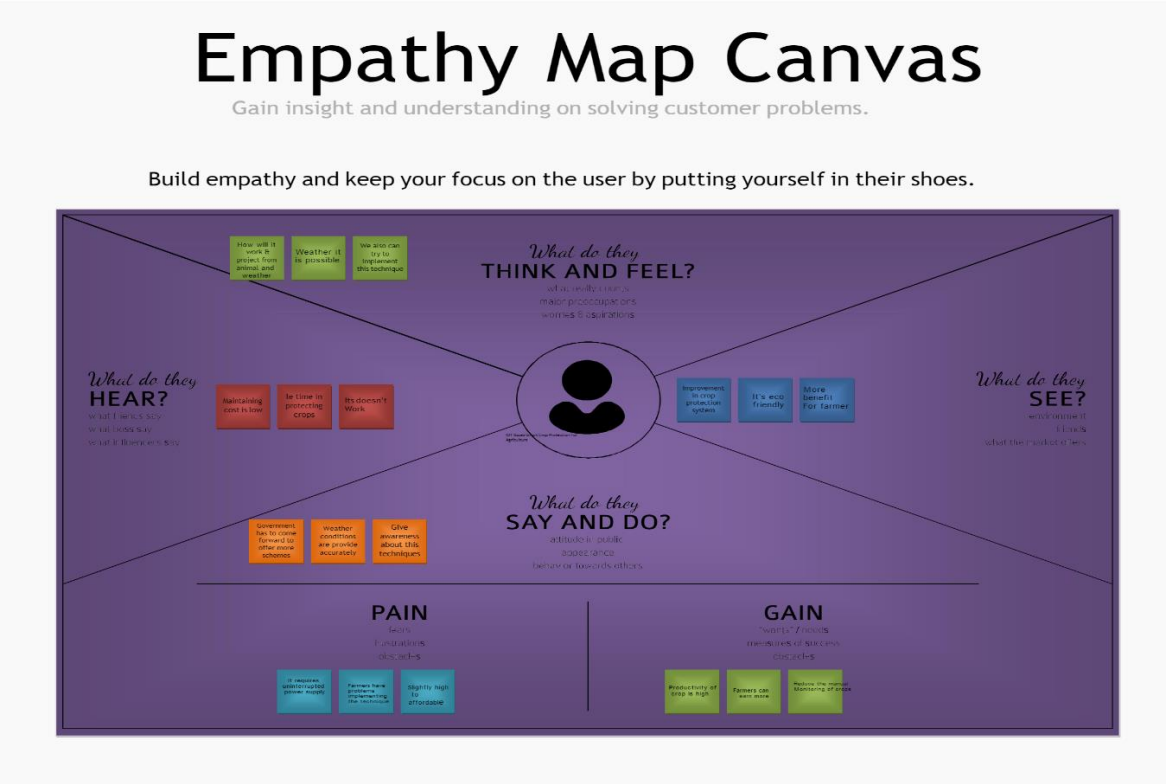
by K. A. Patil, N. R. Kale.

3.**Smart Crop Protection System from Wild Animals Using IoT**

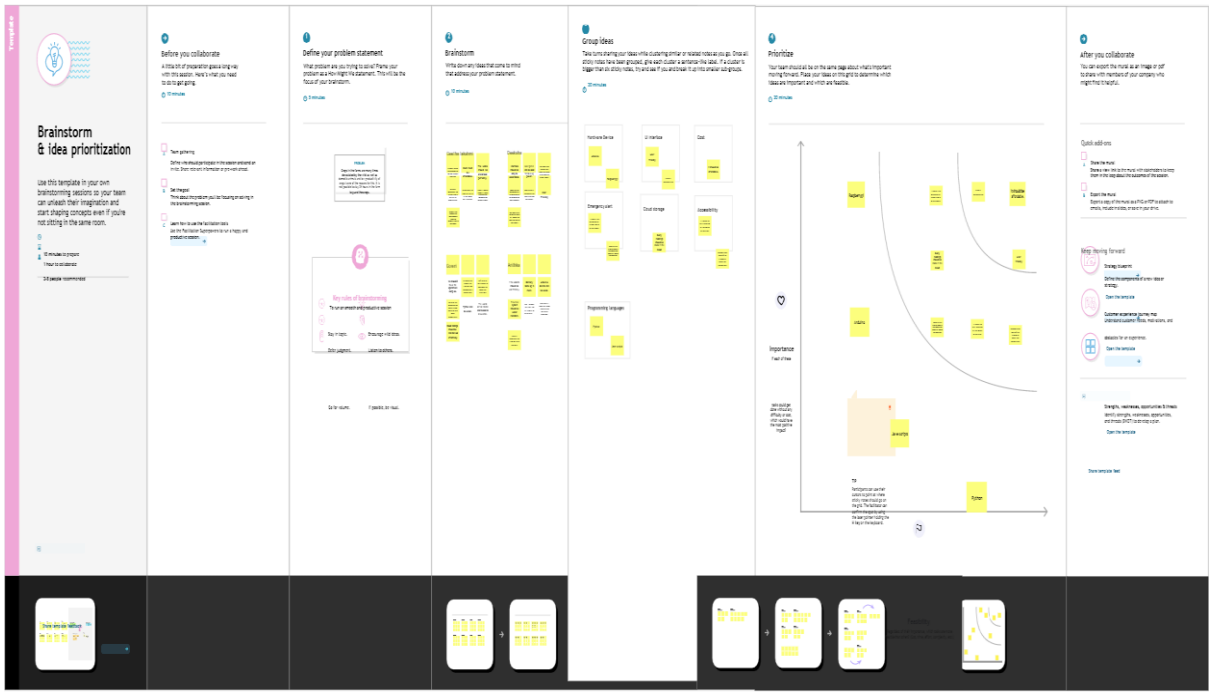
by Priyanka Deotale, Prasad Lokulwar.

IDEATION

3.1 EMPATHY MAP CANVAS



3.2 BRAINSTROMING

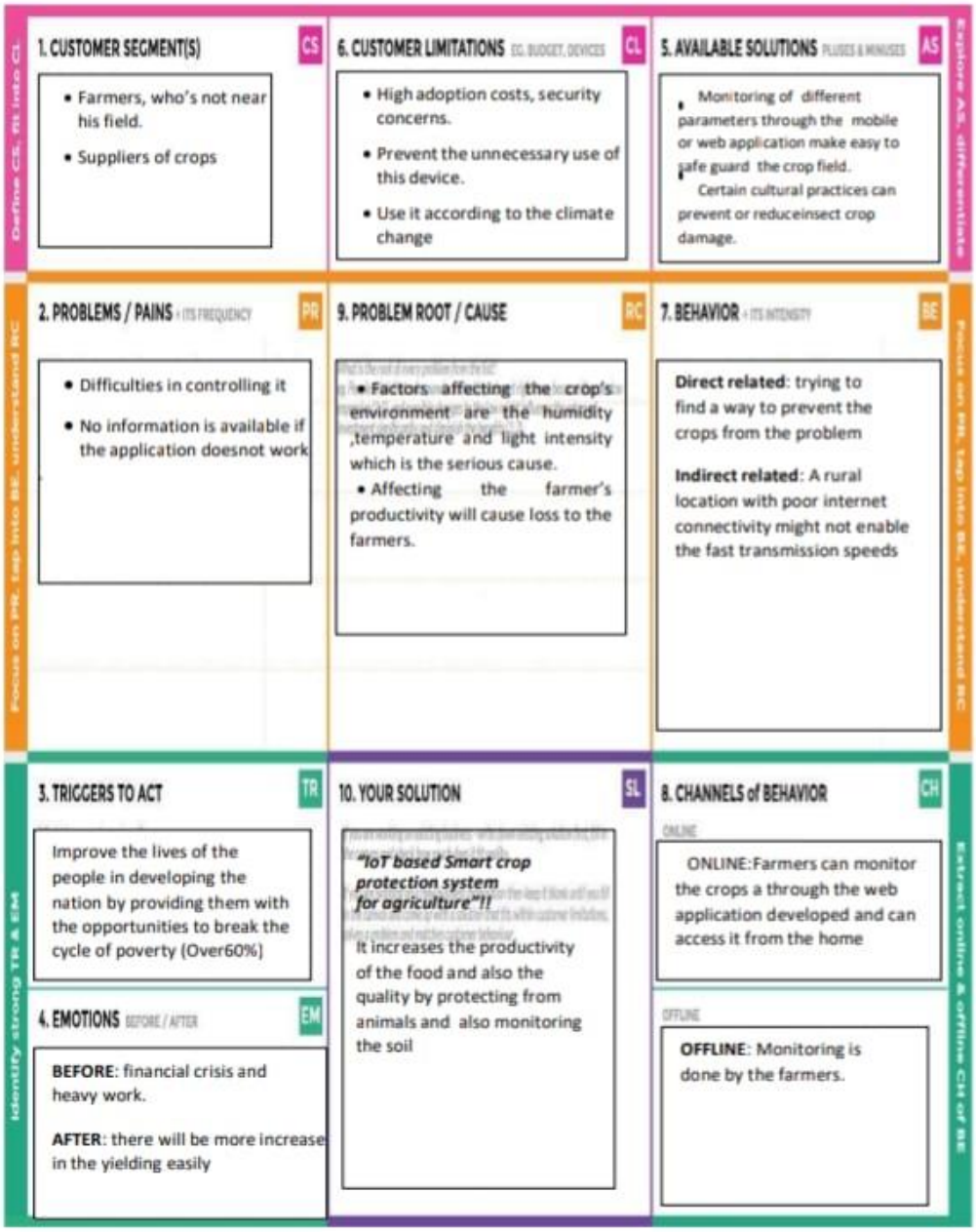


3.3 PROPOSED SOLUTION

Project teams hall fill the following information in proposed solution template.

S.No	Parameter	Description
1.	Problem Statement (Problemtobesolved)	To design an effective monitoring and alerting system which is useful for the farmers to protect their crops from animals and poor soil conditions.
2.	Idea/Solution description	➢ With help of this the field is continuously from animals and birds damage ➢ Humidity and temperature sensors are used in this product for monitoring the crops ➢ If water level in the field is high than the normal level ,it immediately detects and send the notification to the farmers through the mobile application
3.	Novelty/Uniqueness	➢ Quick intimation to the farmers ➢ Demand for quality and productivity of food increases ➢ Eco and user friendly
4.	Social Impact/Customer Satisfaction	➢ Installing and maintenance of the system is easy ➢ Can work with irrespective of fear
5.	Business Model(Revenue Model)	➢ The steps for monitoring is explained so that it is easy for the user to understand and the process is also simple ans safe ➢ The awareness and usage about this product is given through advertisements
6.	Scalability of the Solution	➢ It detects the accurate location of farmers even if there is a high interruption rate

3.4 PROBLEM SOLUTION FIT



REQUIREMENT ANALAYSIS

4.1 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

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.

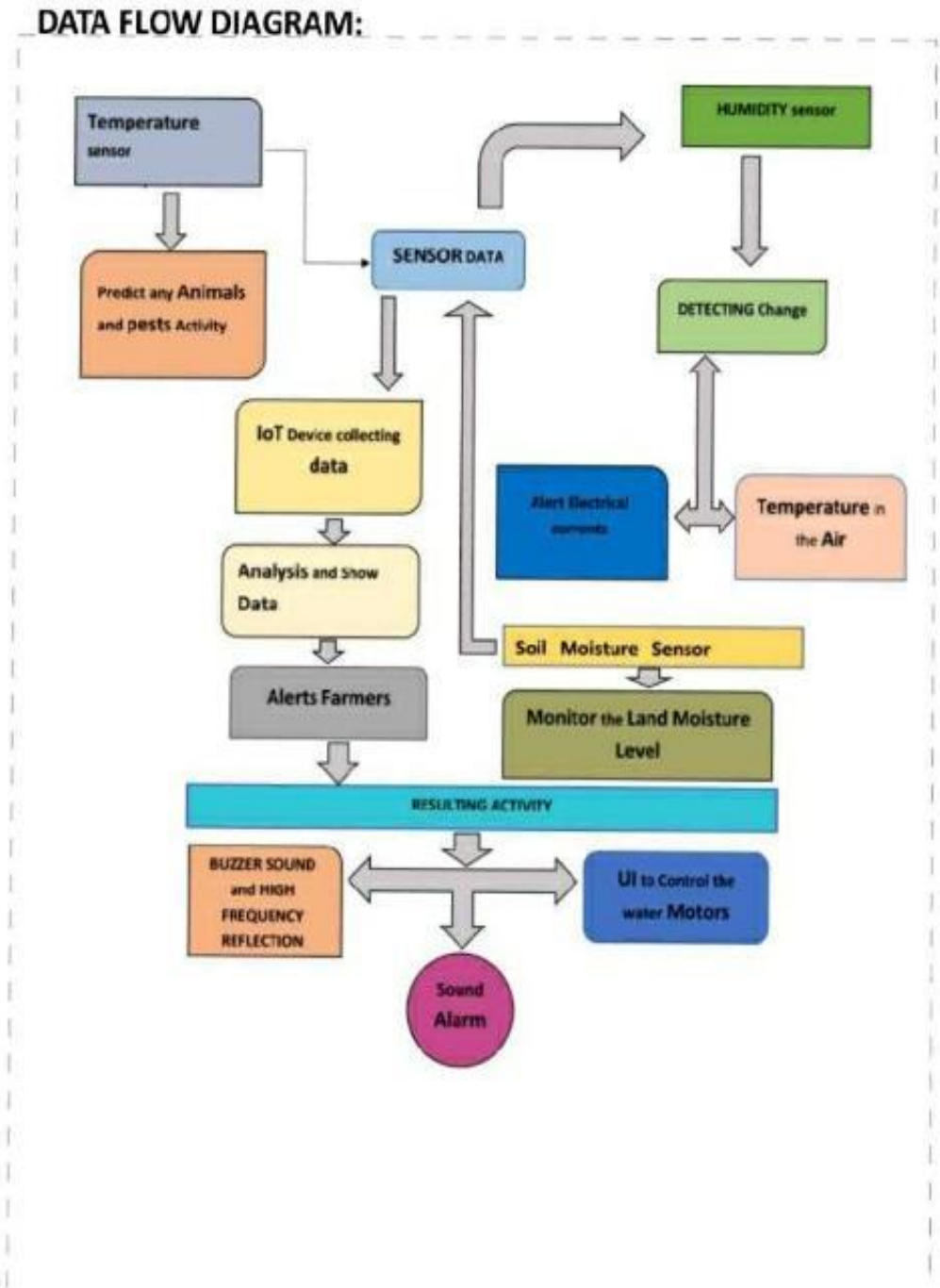
Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
1	Usability	This 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.

PROJECT DESIGN

5.1 DATA FLOW DIAGRAM



5.2 SOLUTION ARCHITECTURE

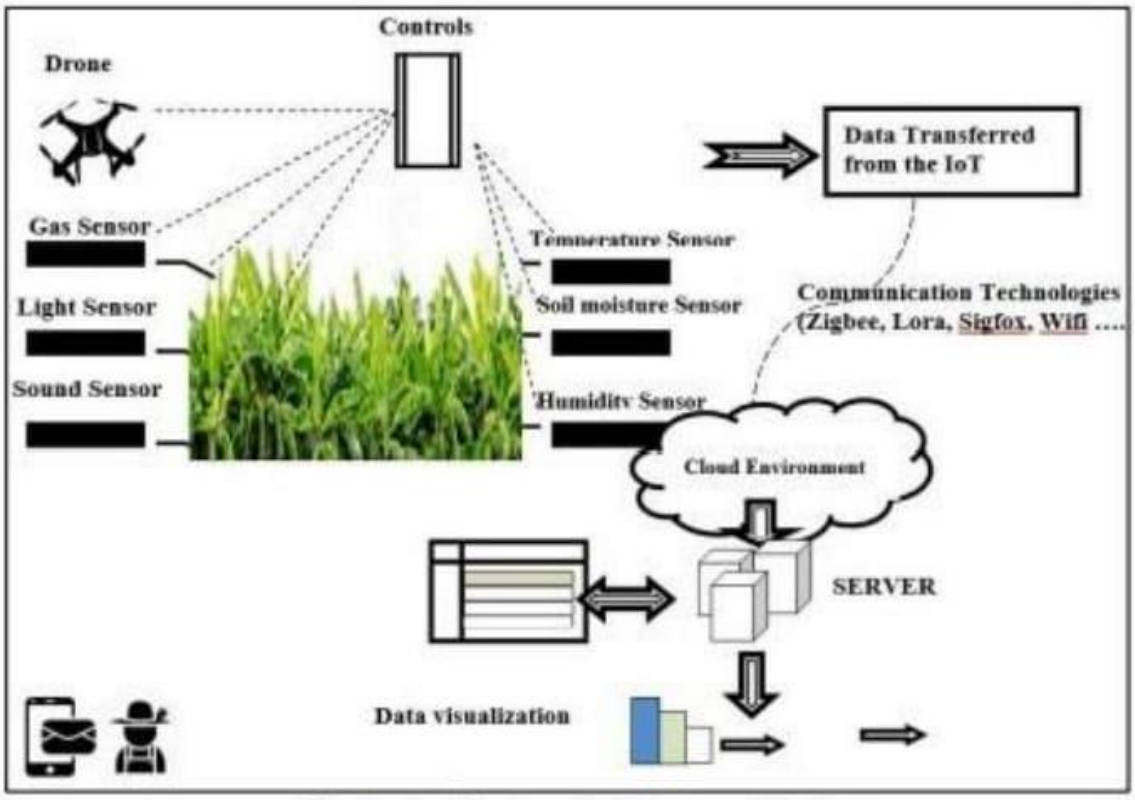


Fig. 1: Architecture for farm monitoring

5.3 TECHNICAL ARCHITECTURE

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table2.

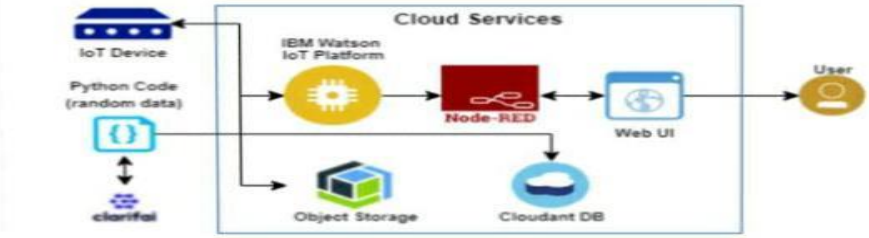


Table-1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with the Web UI	App development
2.	Application Logic-1	Logic for a process in the application	Python Objectives
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the	Node-RED service

		application	
5.	Database	Data Type	Database Cloudant DB
6.	Cloud Database	Database Service on Cloud	Cloud Object store service
7.	File Storage	File storage requirements	IBM Block Storage
8.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration:	Cloud Foundry

S.No	Characteristics	Description	Technology
1.	Open-source Frameworks	The open- source frameworks used	SAN-SAF
2.	Security Implementations	List all the security / access controls implemented	IBM cloud encryptions
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	IBM cloud Architecture
4.	Availability	Justify the availability of applications (e.g. use of load balancers, distributed servers etc.)	Web Application can even be used by the framers in the horticulture
5.	Performance	Design consideration for the performance of the application	Since the web application is high efficient, it can be used by the farmers irrespective of time

5.4 USER STORIES

Stages	Phase 1	Phase 2	Phase 3	Phase 4	Phase 5
Purpose	Motivation	Edge	Assurance	Necessity	Community Benefits
Requirements	Detection and management of threats to farm land to prevent losses.	Despite the dearth of resources, managing farmlands in terms of crop safety.	Connection to the system with sensor through app will com their trust.	Management of increasing demand of food with minimal resources	To increase the quality of farm produce with maximum utilisation of resources and low cost
Components	Prevent damages of crops while minimizing use of pesticides and dealing with droughts.	User-friendly and robust.	Should be robust and immune to the possible threats.	Being a user-friendly interface which can be operated easily.	Cooperative farming using this mechanism can improve crop yield.
Emotions	Intrigued	Gained Credibility	Gets out of dilemma regarding practical feasiblity	Impressed at positive outcomes generated.	Thinks about collaboration to benefit the entire farming community
Outcomes	Apps and devices are connected through IOT.	Devices connected ia sensors.	Buzzer sounds, notifications in mobile app.	Succesful in repelling threats and intimating farmer if threat is beyond control.	Building farmer resilience to calamities and minimum support prices for crops.
Beneficaries	Farmers	Horticulturalists and Farmers.	Farmers with lands.	Farmers even with larger lands.	Farmers nationwide

PROJECT PLANING AND SCHEDULEING

6.1 SPRINT PLAN AND DELIVERY SCHEDULE

ProductBacklog,SprintSchedule,andEstimation(4Marks)

Usethebelowtemplatetocreateproductbacklogandsprintschedule

Sprint	FunctionalRequirement(Epic)	User StoryNumber	UserStory/Task	StoryPoints	Priority	TeamMembers
Sprint-1	IBMCloud services	US-1	CreatetheIBMCloudserviceswhicharebeinguse dinthisproject.	6	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-1	IBMCloudservices	US-2	ConfiguretheIBMCloudserviceswhich arebeingusedincompletingthisproject.	4	Medium	Geethalakshmi Arithka Deeksha Gowri
Sprint-2	IBM Watson IoTplatform	US-3	IBMWatsonIoTplatformactsasthemediatortoconn ect the web application to IoT devices, so createtheIBMWatsonIoTplatform.	5	Medium	Geethalakshmi Arithka Deeksha Gowri
Sprint-2	IBM Watson IoTbplatform	US-4	In order to connect the IoT device to the IBMcloud, create a device in the IBM Watso IoTplatformandgetthedevicecredentials.	5	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-3	IBMvWatsonIoTvplatform & Node-REDservice	US-1	Configuretheconnectionsecurityandcreate APIkeys that are used in the Node-RED service foraccessing theIBMIoTPlatform.	10	High	Geethalakshmi Arithka Deeksha Gowri

Sprint	Functional Requirement(Epic)	UserStory Number	UserStory/Task	StoryPoints	Priority	TeamMembers
Sprint-3	Node-REDservice	US-2	CreateaNode-REDservice.	10	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-3	IBMIoTplatform	US-1	Developapythonscripttopublishrandomsensordata such as temperature, moisture, soil andhumiditytotheIBM IoTplatform	7	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-3	IBMIoTplatform	US-2	Afterdeveloping python code,commandsarereceived just print the statements which representthecontrolofthedevices.	5	Medium	Geethalakshmi Arithka Deeksha Gowri
Sprint-4	IBMCloudServices	US-3	Publish DatatoTheIBMCloud	8	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-4	Webpage	US-1	CreateWebUI in Node- Red	10	High	Geethalakshmi Arithka Deeksha Gowri
Sprint-4	IBMIoTplatform	US-2	ConfiguretheNode-REDflowtoreceivedatafromtheIBMIoTplatform and alsouseCloudantDB nodestostorethereceivedsensordatainthecloudantDB	10	High	Geethalakshmi Arithka Deeksha Gowri

ProjectTracker,Velocity&BurndownChart:(4Marks)

Sprint	Total StoryPoints	Duration	SprintStart Date	SprintEndDate(Planned)	StoryPoints Completed (as onPlannedEndDate)	SprintReleaseDate(Actual)
Sprint-1	20	6Days	24Oct2022	29Oct2022	20	29Oct2022
Sprint-2	20	6Days	31Oct2022	05Nov2022	20	05Nov2022
Sprint-3	20	6Days	07Nov2022	12Nov2022	20	12Nov2022
Sprint-4	20	6Days	14Nov2022	19Nov2022	20	19Nov2022

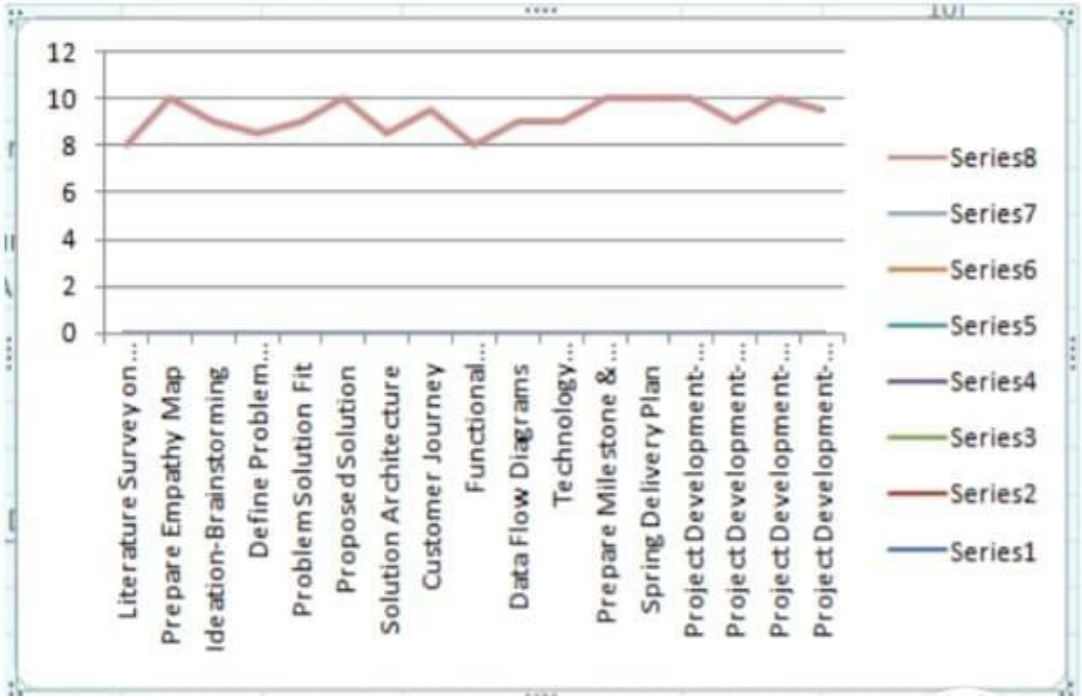
Velocity:

Imaginewehavea10-daysprintduration,andthevelocityoftheteamis20(pointspersprint).Let'scalculatetheteam'saveragevelocity(AV)periterationunit(storypoints perday)

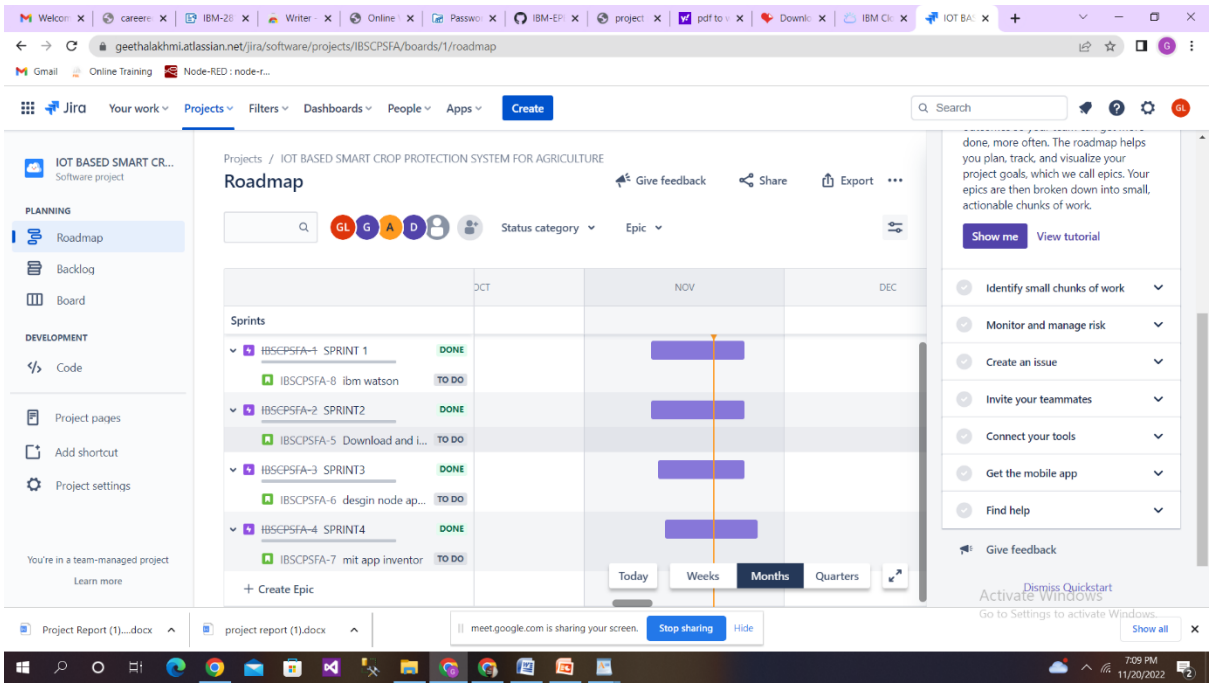
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

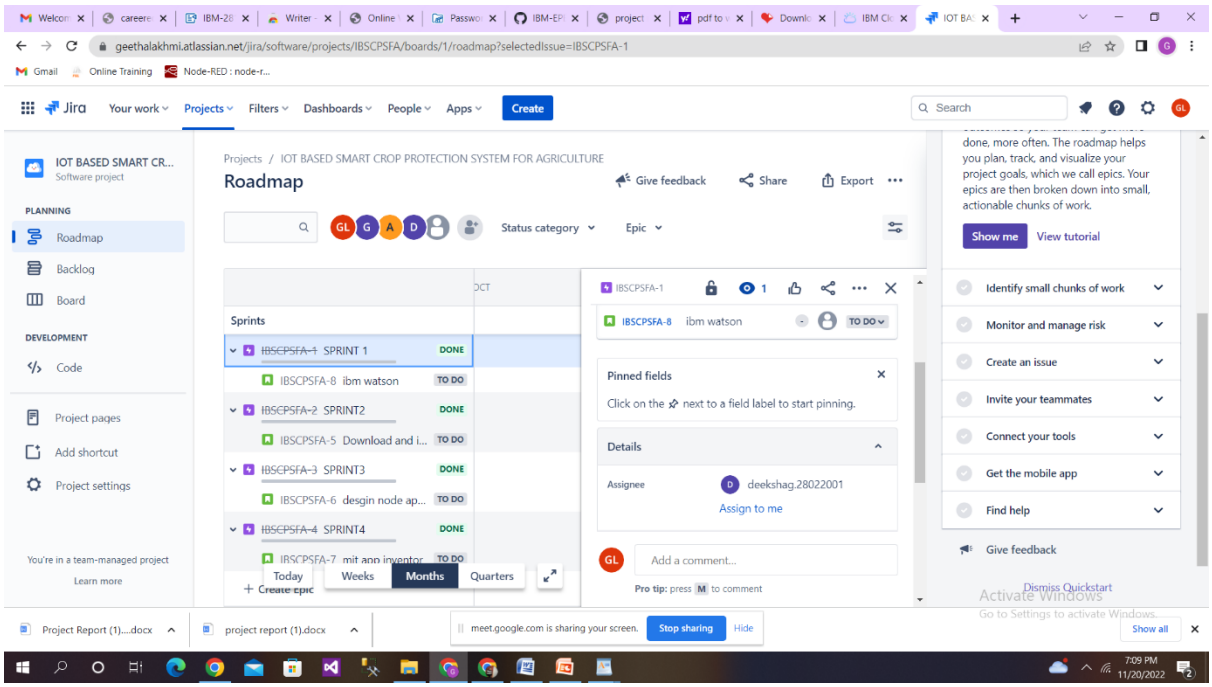
lurndownChart:

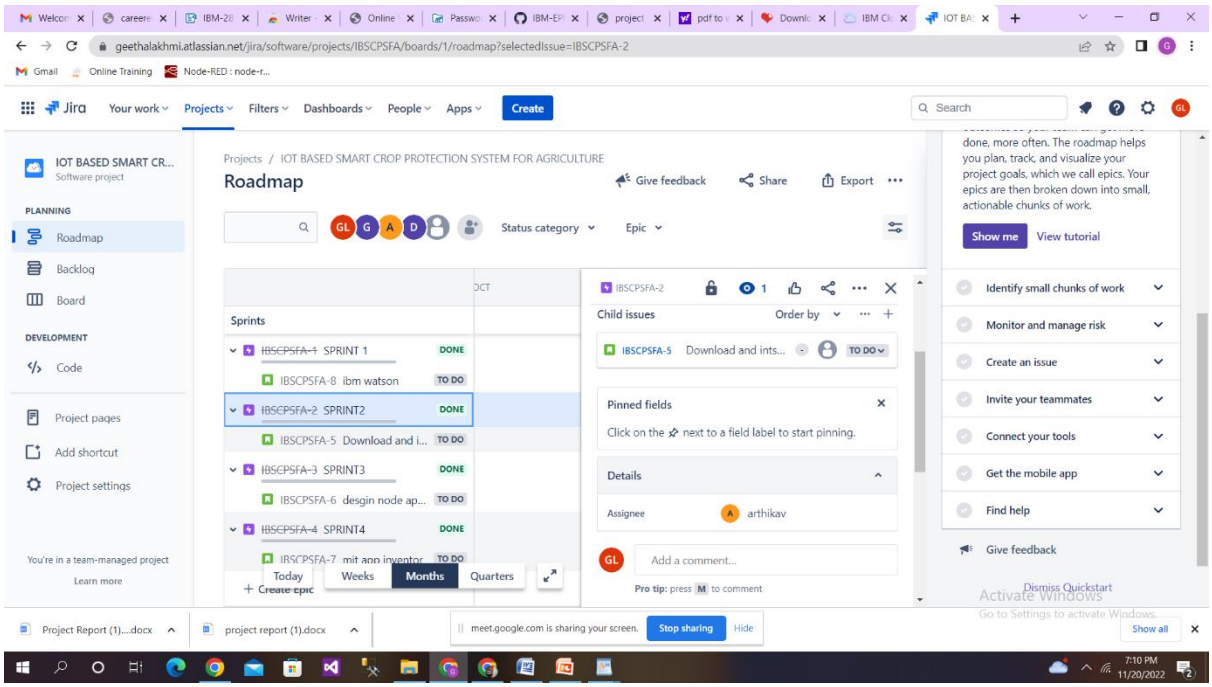
\ burndown chart is a graphical representation of work left to do versus time However, burndown charts can be applied to any project containing neasurableprogress overtime.

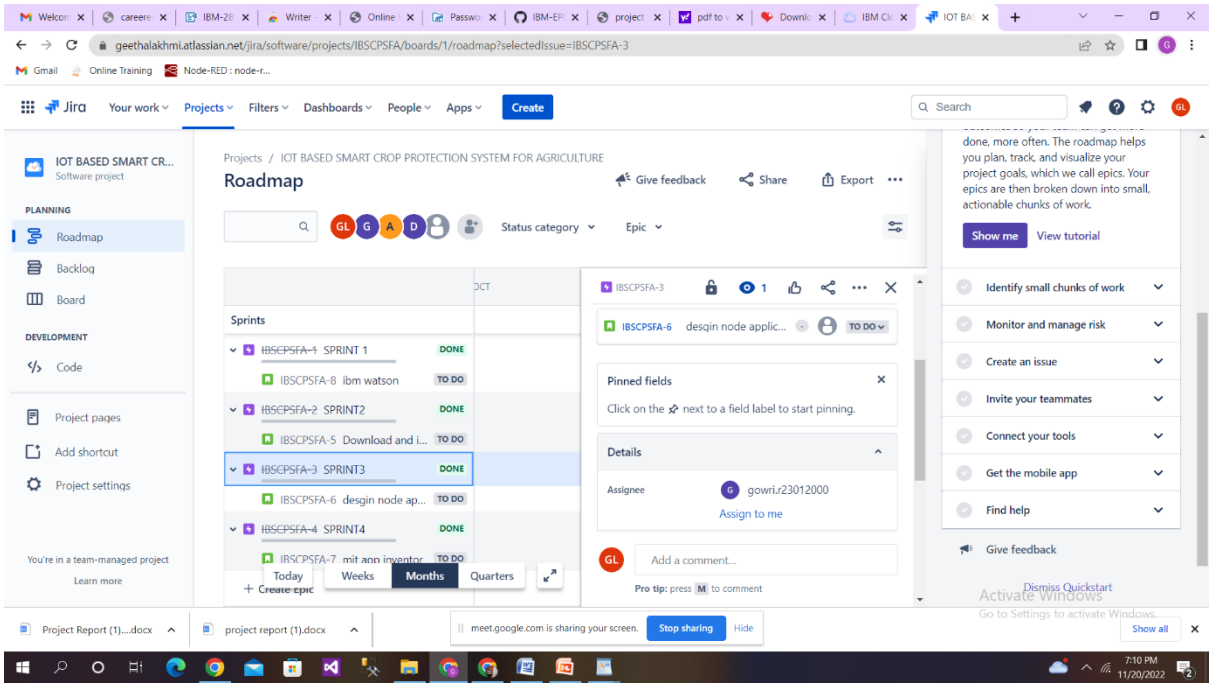


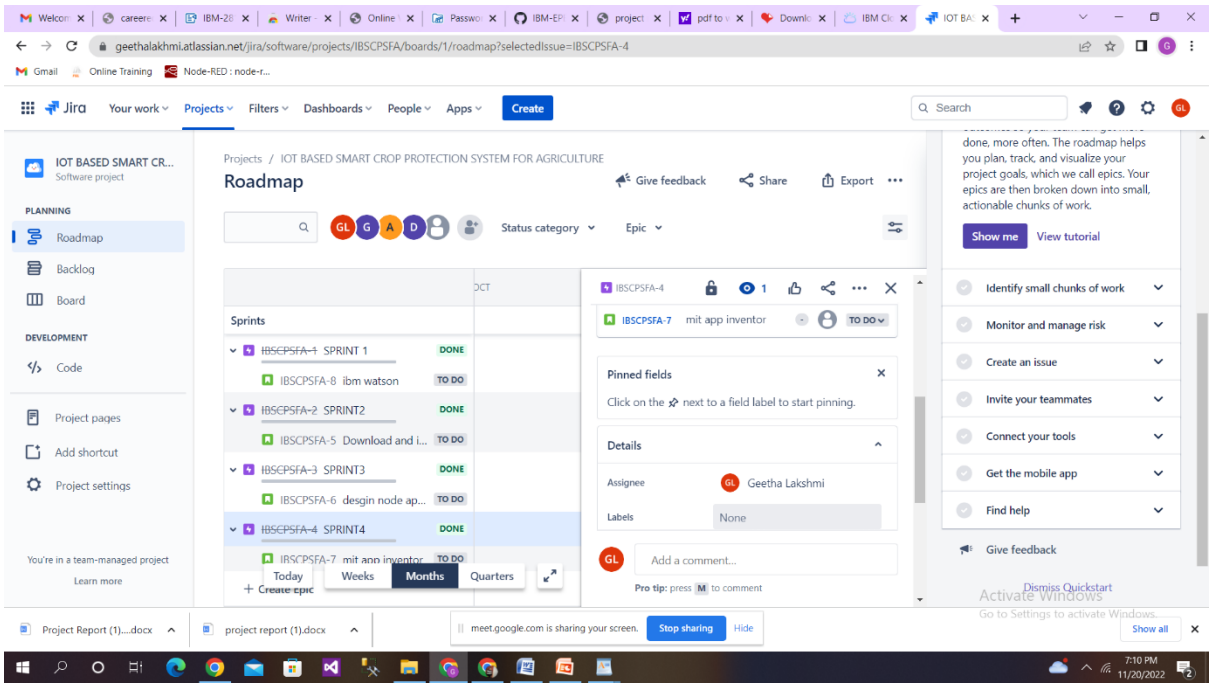
6.2 JIRA


geethalakhmi.atlassian.net/jira/software/projects/IBSCPSFA/boards/1/roadmap
Jira
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IOT BASED SMART CR...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE
Roadmap
Give feedback
Share
Export
Status category
Epic
Sprints
IBSCPSFA-1 SPRINT 1 DONE
IBSCPSFA-8 ibm watson TO DO
IBSCPSFA-2 SPRINT2 DONE
IBSCPSFA-5 Download and i... TO DO
IBSCPSFA-3 SPRINT3 DONE
IBSCPSFA-6 desgin node ap... TO DO
IBSCPSFA-4 SPRINT4 DONE
IBSCPSFA-7 mit app inventor TO DO
Create Epic
OCT
NOV
DEC
Today
Weeks
Months
Quarters
done, more often. The roadmap helps you plan, track, and visualize your project goals, which we call epics. Your epics are then broken down into small, actionable chunks of work.
Show me
View tutorial
Identify small chunks of work
Monitor and manage risk
Create an issue
Invite your teammates
Connect your tools
Get the mobile app
Find help
Give feedback
Dismiss Quickstart
Activate Windows
Go to Settings to activate Windows.
Project Report (1)....docx
project report (1).docx
meet.google.com is sharing your screen.
Stop sharing
Hide
Show all
7:09 PM
11/20/2022


geethalakhmi.atlassian.net/jira/software/projects/IBSCPSFA/boards/1/roadmap?selectedIssue=IBSCPSFA-1
Jira
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IOT BASED SMART CR...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE
Roadmap
Give feedback
Share
Export
Status category
Epic
Sprints
IBSCPSFA-1 SPRINT 1 DONE
IBSCPSFA-8 ibm watson TO DO
IBSCPSFA-2 SPRINT2 DONE
IBSCPSFA-5 Download and i... TO DO
IBSCPSFA-3 SPRINT3 DONE
IBSCPSFA-6 desgin node ap... TO DO
IBSCPSFA-4 SPRINT4 DONE
IBSCPSFA-7 mit app inventor TO DO
Create Epic
Today
Weeks
Months
Quarters
IBSCPSFA-1
IBSCPSFA-8 ibm watson TO DO
Pinned fields
Click on the next to a field label to start pinning.
Details
Assignee
deekshag.28022001
Assign to me
Add a comment...
Pro tip: press M to comment
done, more often. The roadmap helps you plan, track, and visualize your project goals, which we call epics. Your epics are then broken down into small, actionable chunks of work.
Show me
View tutorial
Identify small chunks of work
Monitor and manage risk
Create an issue
Invite your teammates
Connect your tools
Get the mobile app
Find help
Give feedback
Dismiss Quickstart
Activate Windows
Go to Settings to activate Windows.
Project Report (1)....docx
project report (1).docx
meet.google.com is sharing your screen.
Stop sharing
Hide
Show all
7:09 PM
11/20/2022

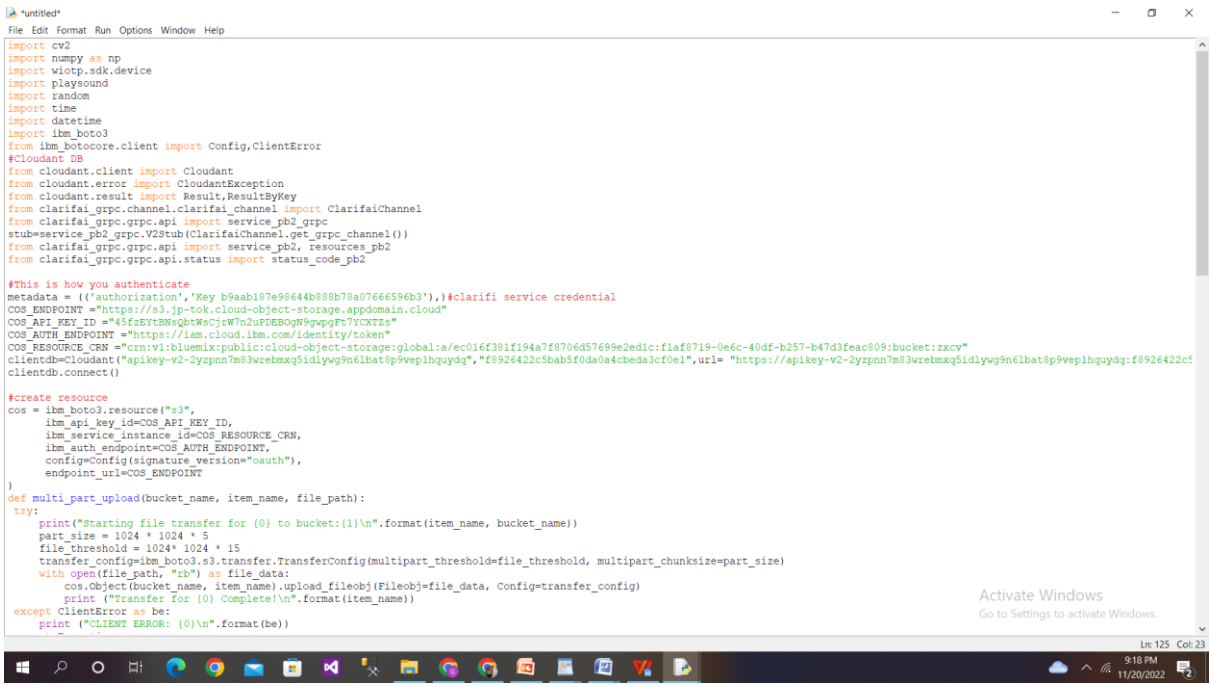

geethalakhmi.atlassian.net/jira/software/projects/IBSCPSFA/boards/1/roadmap?selectedIssue=IBSCPSFA-2
Jira
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IOT BASED SMART CR...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE
Roadmap
Give feedback
Share
Export
Status category
Epic
Sprints
IBSCPSFA-1 SPRINT 1 DONE
IBSCPSFA-8 ibm watson TO DO
IBSCPSFA-2 SPRINT2 DONE
IBSCPSFA-5 Download and i... TO DO
IBSCPSFA-3 SPRINT3 DONE
IBSCPSFA-6 desgin node ap... TO DO
IBSCPSFA-4 SPRINT4 DONE
IBSCPSFA-7 mit app inventor TO DO
Today
Weeks
Months
Quarters
Create epic
IBSCPSFA-2
Child issues
Order by
IBSCPSFA-5 Download and ints... TO DO
Pinned fields
Click on the next to a field label to start pinning.
Details
Assignee
arthikav
Add a comment...
Pro tip: press M to comment
done, more often. The roadmap helps you plan, track, and visualize your project goals, which we call epics. Your epics are then broken down into small, actionable chunks of work.
Show me
View tutorial
Identify small chunks of work
Monitor and manage risk
Create an issue
Invite your teammates
Connect your tools
Get the mobile app
Find help
Give feedback
Dismiss Quickstart
Activate Windows
Go to Settings to activate Windows.
Project Report (1)....docx
project report (1).docx
meet.google.com is sharing your screen.
Stop sharing
Hide
Show all
7:10 PM
11/20/2022

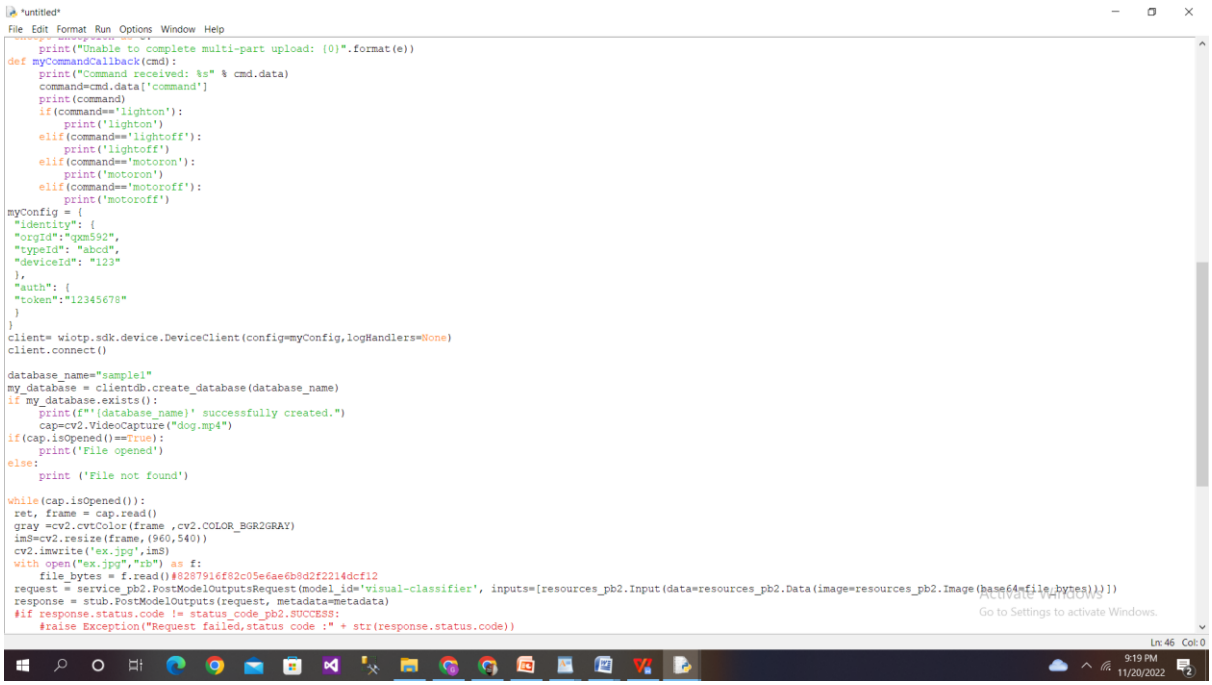

geethalakhmi.atlassian.net/jira/software/projects/IBSCPSFA/boards/1/roadmap?selectedIssue=IBSCPSFA-3
Jira
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IOT BASED SMART CR...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE
Roadmap
Give feedback
Share
Export
Status category
Epic
Sprints
IBSCPSFA-1 SPRINT 1 DONE
IBSCPSFA-8 ibm watson TO DO
IBSCPSFA-2 SPRINT2 DONE
IBSCPSFA-5 Download and i... TO DO
IBSCPSFA-3 SPRINT3 DONE
IBSCPSFA-6 desgin node ap... TO DO
IBSCPSFA-4 SPRINT4 DONE
IBSCPSFA-7 mit app inventor TO DO
Today
Weeks
Months
Quarters
Create epic
IBSCPSFA-3
IBSCPSFA-6 desgin node applic... TO DO
Pinned fields
Click on the next to a field label to start pinning.
Details
Assignee
gowri.r23012000
Assign to me
Add a comment...
Pro tip: press M to comment
done, more often. The roadmap helps you plan, track, and visualize your project goals, which we call epics. Your epics are then broken down into small, actionable chunks of work.
Show me
View tutorial
Identify small chunks of work
Monitor and manage risk
Create an issue
Invite your teammates
Connect your tools
Get the mobile app
Find help
Give feedback
Dismiss Quickstart
Activate Windows
Go to Settings to activate Windows.
Project Report (1)....docx
project report (1).docx
meet.google.com is sharing your screen.
Stop sharing
Hide
Show all
7:10 PM
11/20/2022

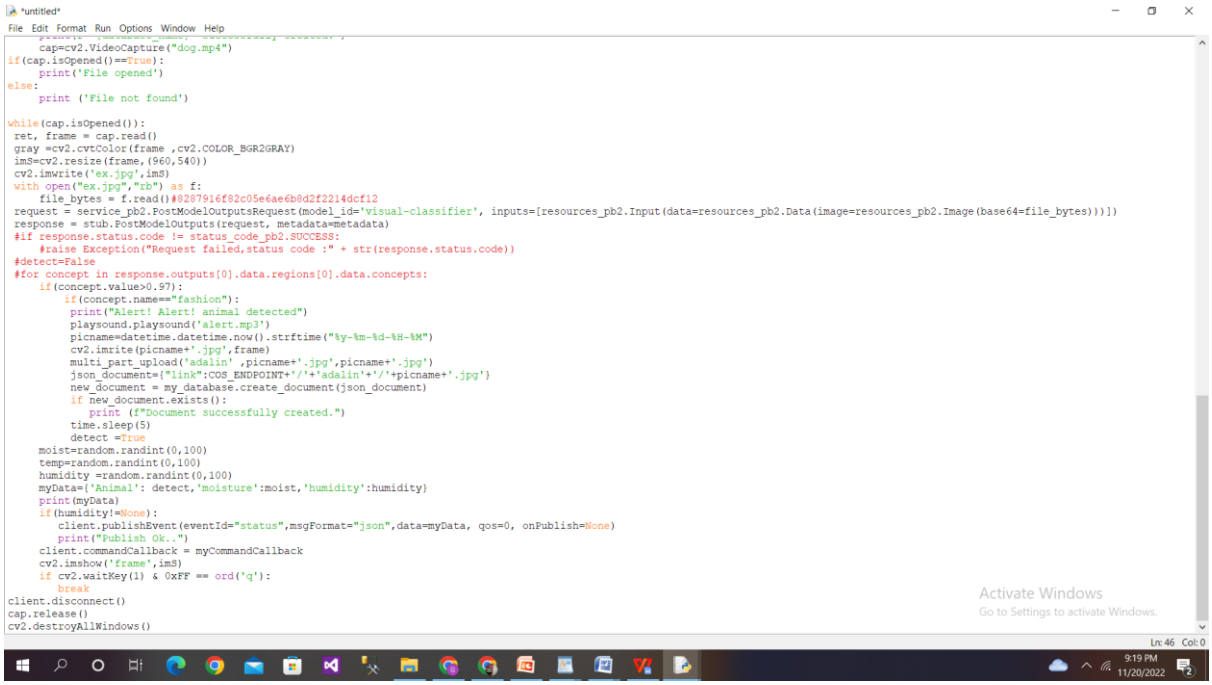


CODING AND SOLUTIONING

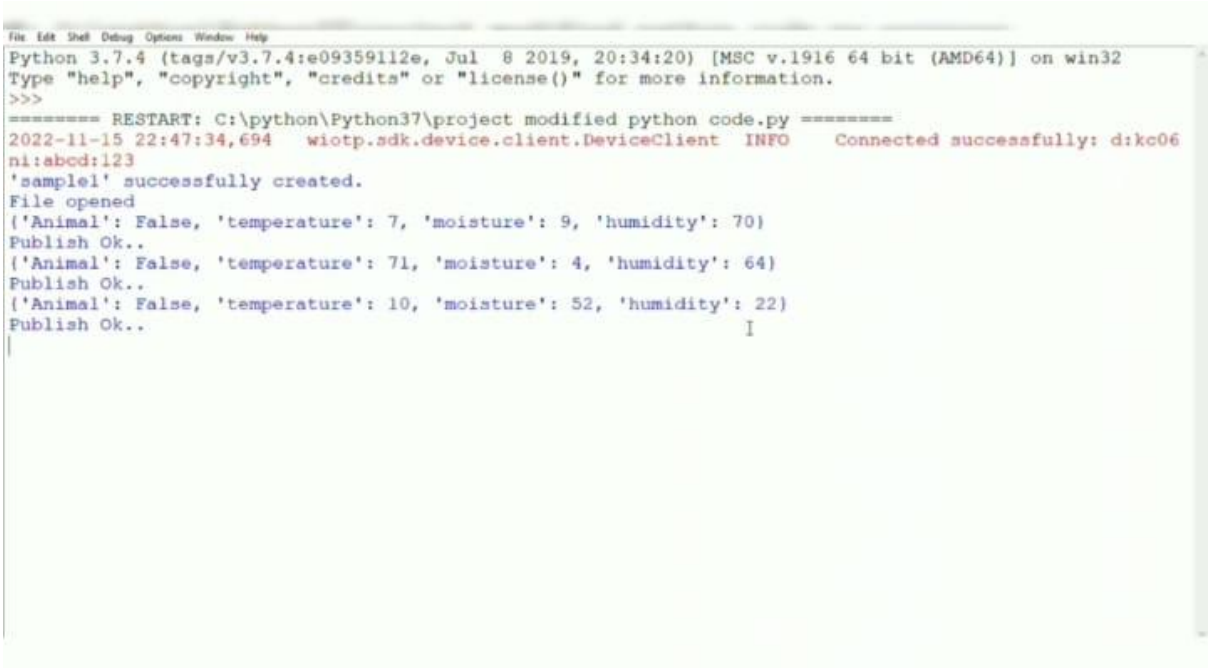
CODING IN PYTHON

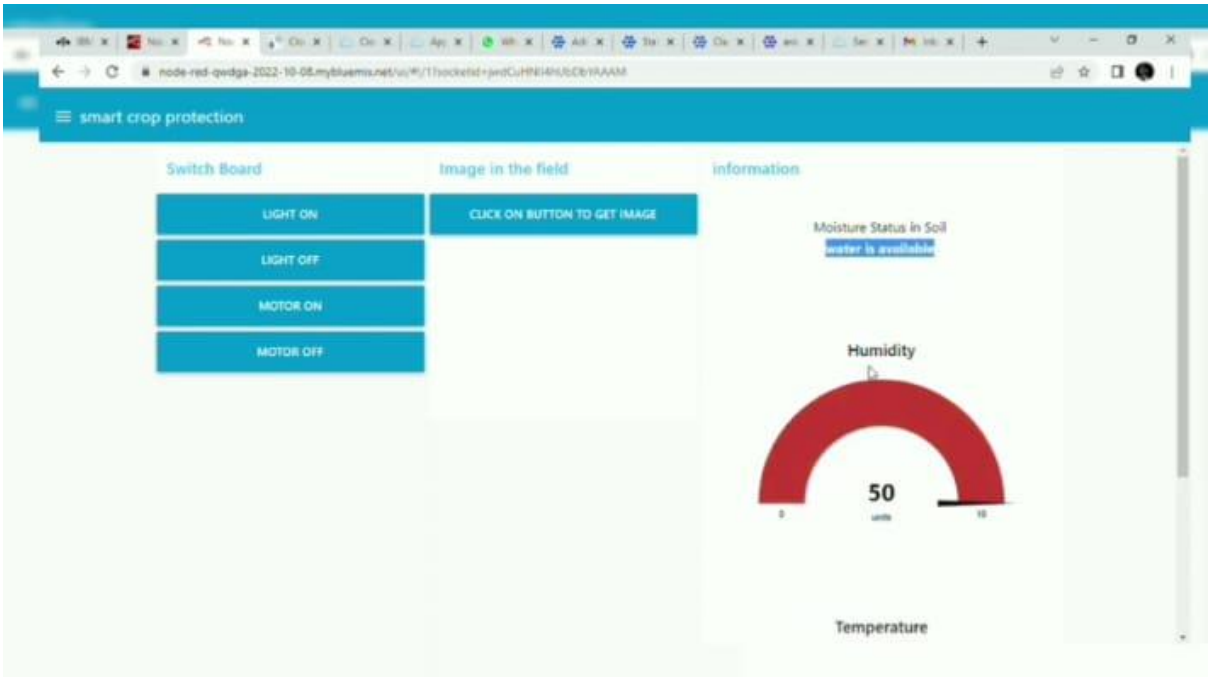


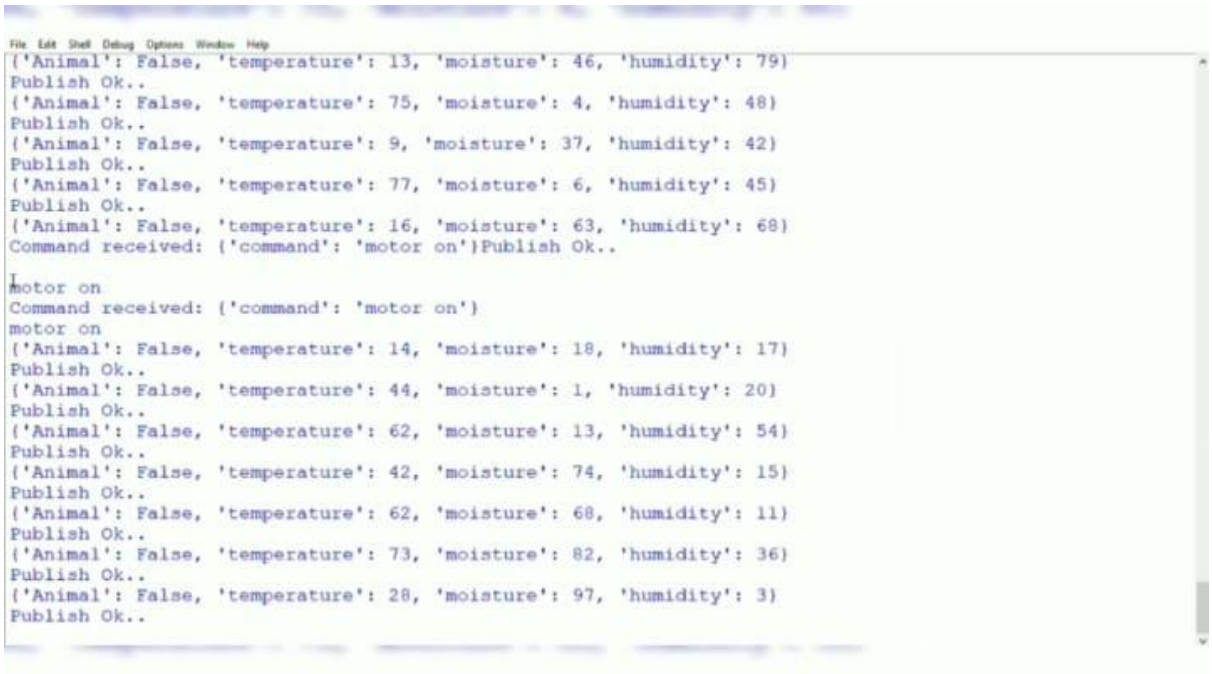

untitled
File Edit Format Run Options Window Help
print("Unable to complete multi-part upload: {0}".format(e))
def myCommandCallback(cmd):
print("Command received: %s" % cmd.data)
command=cmd.data['command']
print(command)
if(command=='lighton'):
print('lighton')
elif(command=='lightoff'):
print('lightoff')
elif(command=='motoron'):
print('motoron')
elif(command=='motoroff'):
print('motoroff')
myConfig = {
"identity": {
"orgId":"qxm592",
"typeId": "abcd",
"deviceId": "123"
},
"auth": {
"token":"12345678"
}
}
client= wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect()
database_name="sample1"
my_database = clientdb.create_database(database_name)
if my_database.exists():
print(f"'{database_name}' successfully created.")
cap=cv2.VideoCapture("dog.mp4")
if(cap.isOpened()==True):
print('File opened')
else:
print ('File not found')
while(cap.isOpened()):
ret, frame = cap.read()
gray =cv2.cvtColor(frame ,cv2.COLOR_BGR2GRAY)
im3=cv2.resize(frame,(960,540))
cv2.imwrite('ex.jpg',im3)
with open("ex.jpg","rb") as f:
file_bytes = f.read()#8287916f82c05e6ae6b8d2f2214dcf12
request = service_pb2.PostModelOutputsRequest(model_id='visual-classifier', inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes)))])
response = stub.PostModelOutputs(request, metadata=metadata)
#if response.status.code != status_code_pb2.SUCCESS:
#raise Exception("Request failed,status code :" + str(response.status.code))
Activate Windows
Go to Settings to activate Windows.
Ln: 46 Col: 0
9:19 PM
11/20/2022

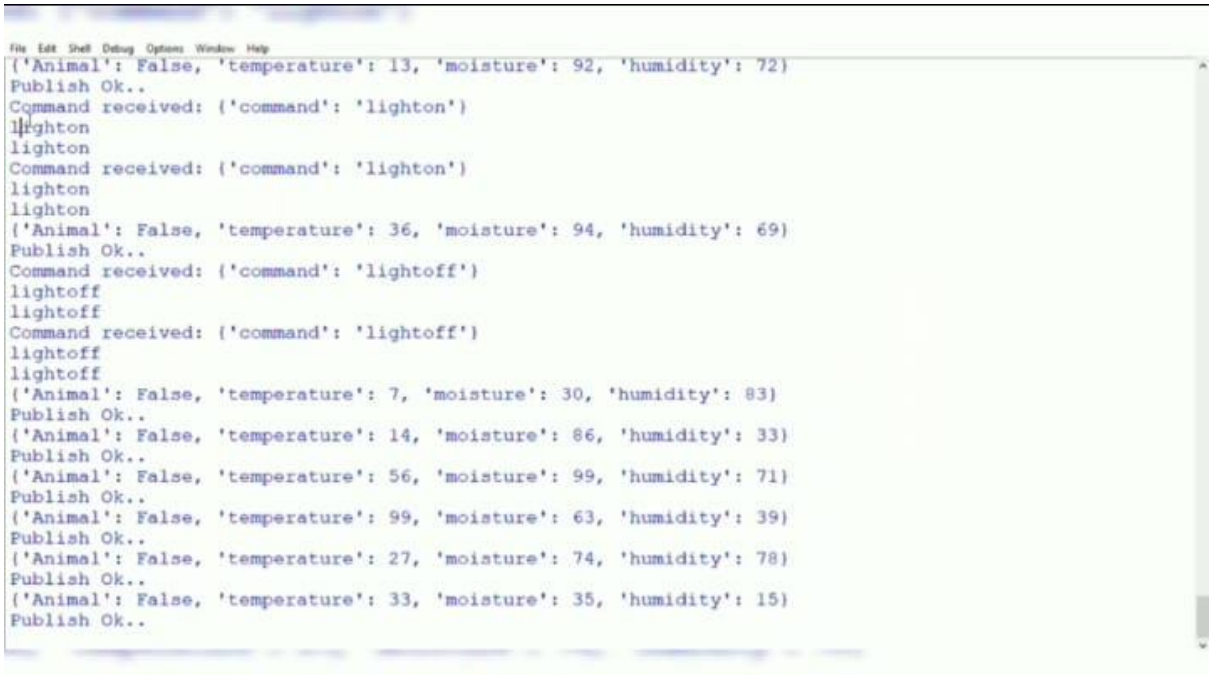

untitled
File Edit Format Run Options Window Help
cap=cv2.VideoCapture("dog.mp4")
if(cap.isOpened()==True):
print('File opened')
else:
print ('File not found')
while(cap.isOpened()):
ret, frame = cap.read()
gray =cv2.cvtColor(frame ,cv2.COLOR_BGR2GRAY)
im3=cv2.resize(frame,(960,540))
cv2.imwrite('ex.jpg',im3)
with open("ex.jpg","rb") as f:
file_bytes = f.read()#8287916f82c05e6ae6b8d2f2214dcf12
request = service_pb2.PostModelOutputsRequest(model_id='visual-classifier', inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes)))])
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.regions[0].data.concepts:
if(concept.value>0.97):
if(concept.name=="fashion"):
print("Alert! Alert! animal detected")
playsound.playsound('alert.mp3')
picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
cv2.imrite(picname+'.jpg',frame)
multi_part_upload('adalin' ,picname+'.jpg',picname+'.jpg')
json_document={"link":COS_ENDPOINT+'/'+'adalin'+'/'+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)
temp=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",data=myData, qos=0, onPublish=None)
print("Publish Ok..")
client.commandCallback = myCommandCallback
cv2.imshow('frame',im3)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
client.disconnect()
cap.release()
cv2.destroyAllWindows()
Activate Windows
Go to Settings to activate Windows.
Ln: 46 Col: 0
9:19 PM
11/20/2022

SOLUTION AND SAMPLE OUTPUT



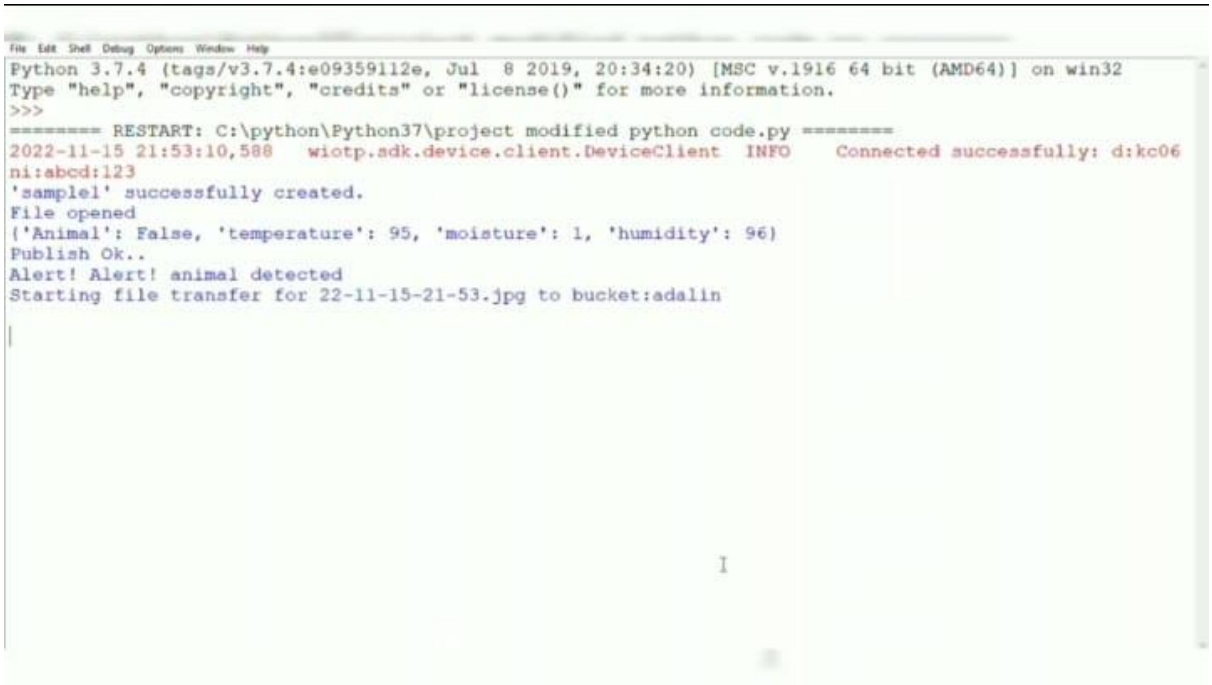



File Edit Shell Debug Options Window Help
{'Animal': False, 'temperature': 13, 'moisture': 46, 'humidity': 79}
Publish Ok..
{'Animal': False, 'temperature': 75, 'moisture': 4, 'humidity': 48}
Publish Ok..
{'Animal': False, 'temperature': 9, 'moisture': 37, 'humidity': 42}
Publish Ok..
{'Animal': False, 'temperature': 77, 'moisture': 6, 'humidity': 45}
Publish Ok..
{'Animal': False, 'temperature': 16, 'moisture': 63, 'humidity': 68}
Command received: {'command': 'motor on'}Publish Ok..
motor on
Command received: {'command': 'motor on'}
motor on
{'Animal': False, 'temperature': 14, 'moisture': 18, 'humidity': 17}
Publish Ok..
{'Animal': False, 'temperature': 44, 'moisture': 1, 'humidity': 20}
Publish Ok..
{'Animal': False, 'temperature': 62, 'moisture': 13, 'humidity': 54}
Publish Ok..
{'Animal': False, 'temperature': 42, 'moisture': 74, 'humidity': 15}
Publish Ok..
{'Animal': False, 'temperature': 62, 'moisture': 68, 'humidity': 11}
Publish Ok..
{'Animal': False, 'temperature': 73, 'moisture': 82, 'humidity': 36}
Publish Ok..
{'Animal': False, 'temperature': 28, 'moisture': 97, 'humidity': 3}
Publish Ok..


File Edit Shell Debug Options Window Help
{'Animal': False, 'temperature': 13, 'moisture': 92, 'humidity': 72}
Publish Ok..
Command received: {'command': 'lighton'}
lighton
lighton
Command received: {'command': 'lighton'}
lighton
lighton
{'Animal': False, 'temperature': 36, 'moisture': 94, 'humidity': 69}
Publish Ok..
Command received: {'command': 'lightoff'}
lightoff
lightoff
Command received: {'command': 'lightoff'}
lightoff
lightoff
{'Animal': False, 'temperature': 7, 'moisture': 30, 'humidity': 83}
Publish Ok..
{'Animal': False, 'temperature': 14, 'moisture': 86, 'humidity': 33}
Publish Ok..
{'Animal': False, 'temperature': 56, 'moisture': 99, 'humidity': 71}
Publish Ok..
{'Animal': False, 'temperature': 99, 'moisture': 63, 'humidity': 39}
Publish Ok..
{'Animal': False, 'temperature': 27, 'moisture': 74, 'humidity': 78}
Publish Ok..
{'Animal': False, 'temperature': 33, 'moisture': 35, 'humidity': 15}
Publish Ok..

TESTING

For example if a monkey is detected in the field,then the result will be



RESULT

9.1 PERFORMANCE METRICS

This system performance is good and it helps in protecting the crops from animals and weather.this inreases the food production and also increases the yield.

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- All the data like climatic conditions and changes in them, soil or crop conditions everything can be easily monitored.
- Risk of crop damage can be lowered to a greater extent.
- Many difficult challenges can be avoided making the process automated and the quality of crops can be maintained.
- The process included in farming can be controlled using the web applications from anywhere, anytime.

DISADVANTAGES

- Smart Agriculture requires internet connectivity continuously, but rural parts cannot fulfill this requirement.Any faults in the sensors can cause great loss in the agriculture, due towrong records and the actions of automated processes.
- IoT devices need much money to implement.The use of technology in farming and agriculture making it smart agriculture, is of course, a good initiative and a much-needed one with the present increasing demand in the food supply.
- In the case of equipment computer-based intelligence for running the devices, it is highly unlikely that a normal farmer will be able to possess this knowledge or even develop them.

APPLICATIONS

- Precision Farming that is farming processes can be made more controlled and accurate.
- Live monitoring can be done of all the processes and the conditions on the agricultural field.
- All the controls can be made just on the click.
- Quality can be maintained.

CONCLUSION

Smart farming is a modern farming management concept with IoT technology to increase the productivity in agriculture.With the use of smart farming, users can effectively monitor the crop field the qualityand quantity of their crops.

Users cannot be physically present on the field 24 hours a day. In addition, the farmersmay not have the knowledge to use different tools to measure the ideal environmentalconditions for their crops.

IoT provides them with the automated system, which can function without any human supervision and can notify them to make proper decision to deal with different kind of problems they may face during farming.

It has the capability to reach and notify the farmer even if farmer is not on the field,which can allow farmer to manage more farmland, thus improving their production.Thus, we can conclude that this IoT based smart crop protection system will definitely help users in farmland to effectively monitor their crops with the user-friendly platforms and alert the farm.

FUTURE SCOPE

The proposed work system is a successful working prototype that fulfils to protect crops from the intrusion of animals and birds.

This system will helps the users to monitor the temperature and to notify the weather conditions.This system assuredly assists the users to know about the soil moisture level. And the IoT based smart crop protection system implemented here brings a naval approach crop protection system from animals.

This assures the early detection and prevention of incurring losses due to the damage of crops.The following suggestions may be carried out in future implementation of the system;the smart crop prediction may be also carried out by considering the various factors like NPK content of the soil, UV radiation along with the tracking of the crop field location using GPSmodule system.

The automated pest traps also be introduced using image recognition techniques and neural networks in smart protection system.

SOURCE CODE

```
import cv2
import numpy as np
import wiotp.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3
from ibm_botocore.client import Config,ClientError
#Cloudant DB
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, resources_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2

#This is how you authenticate
metadata = (('authorization','Key b9aab187e98644b888b78a07666596b3'),)#clarifi service credential
COS_ENDPOINT ="https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID ="45fzEYtBNsQbtWsCjrW7n2uPDEBOgN9gwpgFt7YCXTZs"
COS_AUTH_ENDPOINT ="https://iam.cloud.ibm.com/identity/token"
```

```
COS_RESOURCE_CRN ="crn:v1:bluemix:public:cloud-object-storage:global:a/ec016f381f194a7f8706d57699e2ed1c:f1af8719-0e6c-40df-b257-b47d3feac809:bucket:zxcv"

clientdb=Cloudant("apikey-v2-2yzpnn7m83wrebmxq5idlywg9n6lbat8p9vep1hquydq","f8926422c5bab5f0da0a4cbeda3cf0e1",url= "https://apikey-v2-2yzpnn7m83wrebmxq5idlywg9n6lbat8p9vep1hquydq:f8926422c5bab5f0da0a4cbeda3cf0e1@30c7c015-07db-4943-8393-c2a2da703e99-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))

    part_size = 1024 * 1024 * 5

    file_threshold = 1024* 1024 * 15


transfer_config=ibm_boto3.s3.transfer.TransferConfig(multipart_threshold=file_threshold, multipart_chunksize=part_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(command=='lighton'):
        print('lighton')
    elif(command=='lightoff'):
        print('lightoff')
    elif(command=='motoron'):
        print('motoron')
    elif(command=='motoroff'):
        print('motoroff')
myConfig = {
 "identity": {
 "orgId":"qxm592",
 "typeId": "abcd",
 "deviceId": "123"
 },
 "auth": {
 "token":"12345678"
 }
}
client= wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect()

database_name="sample1"
my_database = clientdb.create_database(database_name)
```

```
if my_database.exists():
    print(f"'{database_name}' successfully created.")
    cap=cv2.VideoCapture("dog.mp4")
if(cap.isOpened()==True):
    print('File opened')
else:
    print ('File not found')


while(cap.isOpened()):
 ret, frame = cap.read()
 gray =cv2.cvtColor(frame ,cv2.COLOR_BGR2GRAY)
 imS=cv2.resize(frame,(960,540))
 cv2.imwrite('ex.jpg',imS)
 with open("ex.jpg","rb") as f:
    file_bytes = f.read()#8287916f82c05e6ae6b8d2f2214dcf12
 request = service_pb2.PostModelOutputsRequest(model_id='visual-classifier',
inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=
file_bytes)))])
 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.regions[0].data.concepts:
    if(concept.value>0.97):
        if(concept.name=="fashion"):
         print("Alert! Alert! animal detected")
         playsound.playsound('alert.mp3')
         picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
         cv2.imrite(picname+'.jpg',frame)
         multi_part_upload('adalin' ,picname+'.jpg',picname+'.jpg')
```

```
        json_document={"link":COS_ENDPOINT+'/'+'adalin'+'/'+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)
    temp=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",data=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()
```

Project Demo link-https://youtu.be/h5jkdxdNa-k