PROJECT REPORT

IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

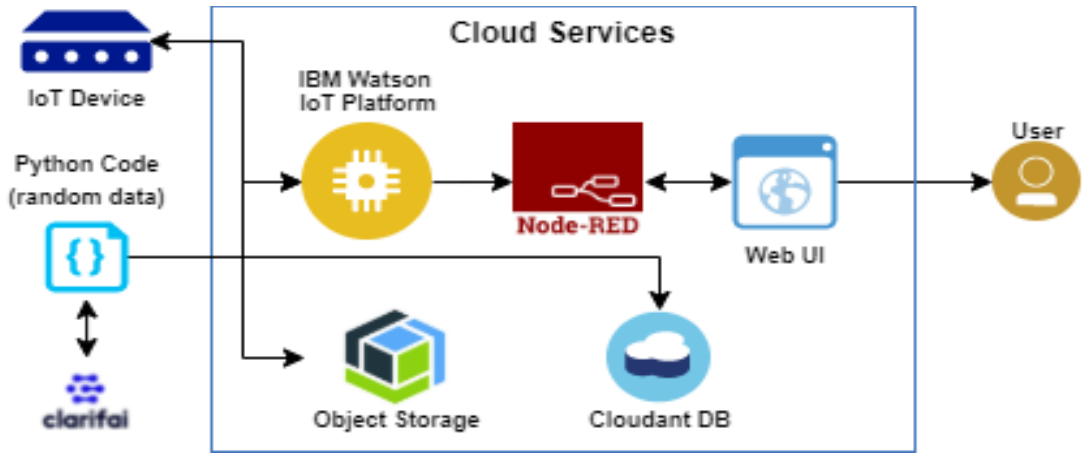
TEAM ID: PNT2022TMID17772

ROLL NUMBER	NAME
19EC124	VISHWAJITH.R
19EC070	NAVEEN.R
19EC085	RAJEEV.M
19EC118	VENKATESH.K

ABSTRACT

Crops are in farms are many times ravaged by local animals like buffaloes, cows, goats, birds etc. This leads to huge losses for the farmer. Due to over population, it occurs a deforestation this results in shortage of food, water nd shelter in forest areas. So,animal's interference in residential areas is increasing day by day which affects human life and property causes human animal conflict but as per nature's rule every living creature on this earth has important role in eco-system.Elephants and other animals coming in to contact with humans,impact negatively in various means such as by depredation of crops,damaging grain stores,water supplies,houses and other assets,injuring and death of humans.

So here we propose automatic crop protection system from animals.This is a microcontroller-based system using PIC family microcontroller.These system use a motion sensor to detect wild animal approaching near the field.In such a case the sensor signal the microcontroller to take action.Traditional methods used by farmers are given below.



INTRODUCTION

A system using sensors that monitor different conditions of environment like humidity, temperature etc., the processor and GUI module is used. The field condition is sent to the farmer via mobile text messages. With this system Soil moisture, humidity and energy efficiency are managed. A system is proposed for intelligent agriculture monitoring system based on IOT technology. The main aim of this project is to help farmers to automate their farms by providing them with a Web App through which they can monitor the parameters of the field like Temperature, humidity etc. and control the equipment like water motor and other devices remotely via internet without their actual presence in the field.

Crops in the farms are many times devastated by the wild as well as domestic animals and low productivity of crops is one of the reasons for this. It is not possible to stay 24 hours in the farm to sentinel the crops. So to surmount this issue an automated perspicacious crop aegis system is proposed utilizing Internet of Things (IOT). The system consists of esp8266 (node MCU), soil moisture sensor, dihydrogen monoxide sensor, GPRS and GSM module, servo motor, dihydrogen monoxide pump, etc. to obtain the required output. As soon as any kineticism is detected the system will engender an alarm to be taken and the lights will glow up implemented at every corner of the farm. This will not harm any animal and the crops will stay forfended.

PROJECT OVERVIEW

1. This project is based on Internet Of Things (IoT), that can measure soil moisture, Humidity and temperature conditions for agriculture and crop protection using Watson IoT services. IoT is network that connects physical objects or things embedded with electronics, software and sensors through network connectivity that

collects and transfers data using cloud for communication. Data is transferred through internet without human to human or human to computer interaction.

2. In this project we have not used any hardware. Instead of real soil moisture, Humidity and Temperature data obtained from sensors we make use of IBM IoT Simulator which can transmit these parameters as required.

PURPOSE:

3. An intelligent crop protection system helps the farmers in protecting the crop from the animals and birds which destroy the crop.
4. 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.

2. LITERATURE SURVEY:

EXISTING PROBLEM:

1. Agriculture is a field which forms the basis of our economy. Yet it faces a lot of problems in terms of availability of resources, Irrigation, increasing rate of Pesticides, Climatic disasters, Insects which ruin the crops and makes a huge loss this sector.
2. In agriculture water is needed for the crops for their growth. If the Soil gets dry it is necessary to supply water. But sometime if the farmer doesn't visit the field it is not possible to know the condition of soil.
3. Sometimes over supply of water or less supply of water affects the growth of crops.
4. Sometimes if the weather/temperature changes suddenly it is necessary to take certain actions.
5. Specific crops grow better in specific conditions, they may get damaged due to bad weather.

REFERENCES:

6. https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf
7. https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf
8. https://openweathermap.org/
9. https://smartinternz.com/assets/docs/Sending%20Http%20request%20to%20Open%20weather%20map%20web site%20to%20get%20the%20weather%20forecast.pdf
10. https://www.youtube.com/watch?v=cicTw4SEdxk

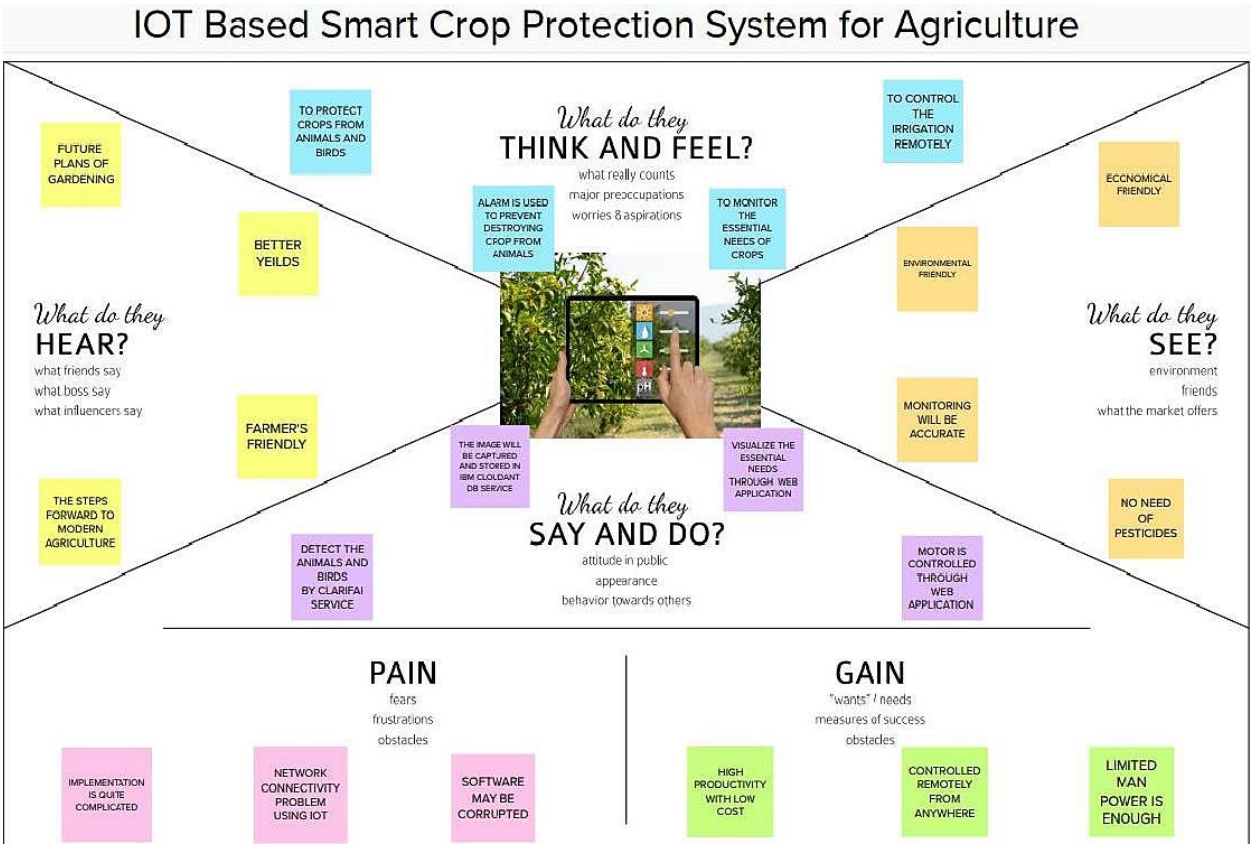
11. https://smartinternz.com/assets/docs/Smart%20Home%
20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf
12. https://github.com/rachuriharish23/ibmsubscribe

PROBLEM STATEMENT DEFINITION:

13. Smart Crop Protection System based on IoT can monitor soil moisture and climatic conditions to grow and yield a good crop.
14. The farmer can also get the realtime weather forecasting data by using external platforms like Open Weather API.
15. Farmer is provided a mobile app using which he can monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.
16. Based on all the parameters he can water his crop by controlling the motors using the mobile application.
17. Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.
18. Here we are using the Online IoT simulator for getting the Temperature, Humidity and Soil Moisture values.

3. IDEATION & PROPOSED SOLUTION:

Empathy Map Canvas



Brain Storming & Ideation

Idea 1:

Crops in the farms are many times devastated by the wild as well as domestic animals and low productivity of crops is one of the reasons for this. It is not possible to stay 24 hours in the farm to sentinel the crops. So to surmount this issue an automated perspicacious crop aegis system is proposed utilizing Internet of Things (IOT). The system consists of esp8266 (nodeMCU), soil moisture sensor, dihydrogen monoxide sensor, GPRS and GSM module, servo motor, dihydrogen monoxide pump, etc. to obtain the required output. As soon as any kineticism is detected the system will engender an alarm to be taken and the lights will glow up implemented at every corner of the farm. This will not harm any animal and the crops will stay forfended.

Idea 2:

The Smart protection system defines that this project help to farmer for the protection of a farm. We have designed this project for the only secure from animals but this project have the provision to secure from the human begins also. This can be achieved by the help of IOT device. The SCPS work on the battery so that this project can be easily portable and also we are added solar panels and converter modules. This can help the battery to charge from solar energy. The IOT device is used to indicate the farmer by a message while someone enter into the farm and we are used SD card module that helps to store a specified sound to fear the animals.

Idea 3:

A centralizing method in the area of IIoT (Industrial Internet of Things) contrived for understanding agriculture which is preceding the arrangements low-power devices . This project yields a monitoring procedure for farm safety against animal attacks and climate change conditions. IIoT advances are frequently used in smart farming to emphasize the standard of agriculture. It contains types of sensors, controllers. On behalf of WSN, the ARM Cortex-A board which consumes 3W is the foremost essence of the procedure . Different sensors like DHT 11 Humidity & Temperature Sensor, PIR Sensor, LDR sensor, HC-SR04 Ultrasonic Sensor, and camera are mounted on the ARM Cortex-A board. The PIR goes high on noticing the movement within the scope, the camera starts to record, and the data will be reserved onboard and in the IoT cloud, instantaneously information will be generated automatically towards the recorded quantity using a SIM900A unit to notify about the interference with the information of the weather conditions attained by DHt11. If a variance happens, the announcement of the threshold rate will be sent to the cell number or to the website. The result will be generated on a catalog of the mobile of the person to take the necessary action.

Brainstorming

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil (switch to sketch) icon to start drawing!

Harish Ragavendran

(1) Incase of emergency,the system should play an eggective role
(2) cost must be affortable
(3) The waste should be discarded perfectly

Arun kumar

(1) interface should be easy to understand
(2) Eco-Friendly
(3) solar panels can be used for back up power

Kamalanathan

(1) Motors can be connected with this devices whic can be controlled from anywhere
(2) More pictroal represents should be use than words

Karthick

(1) Final outcomes should be user friendly.
(2) Arduino device can be used.
(3) Battery power back is important.

Mohamed Sulaiman

(1) The whole system should be water resistant
(2) Operate with also both Solar and electrical sources
(3) solar panels will be included in advanced devices

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

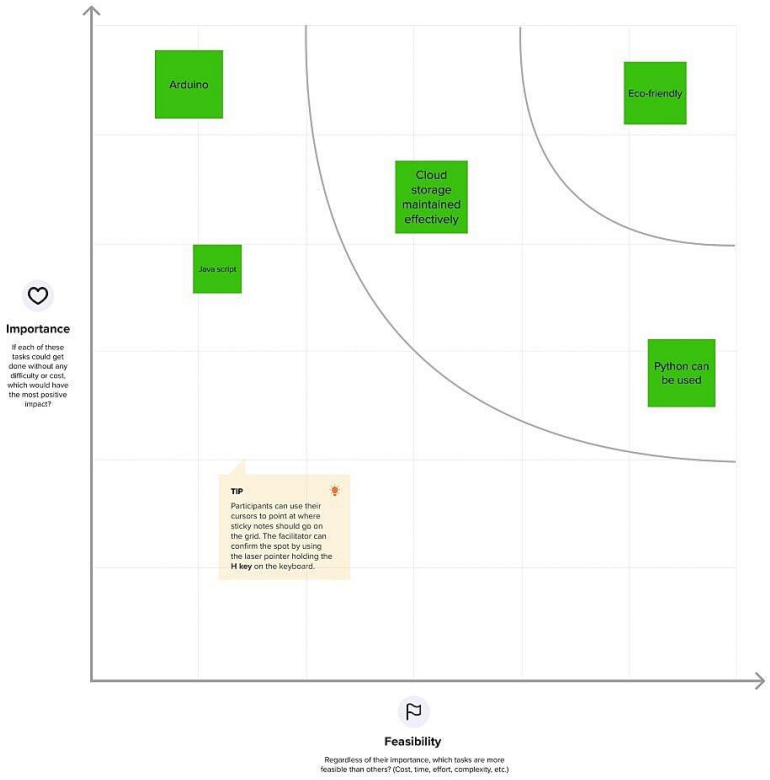
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes



After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A **Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B **Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

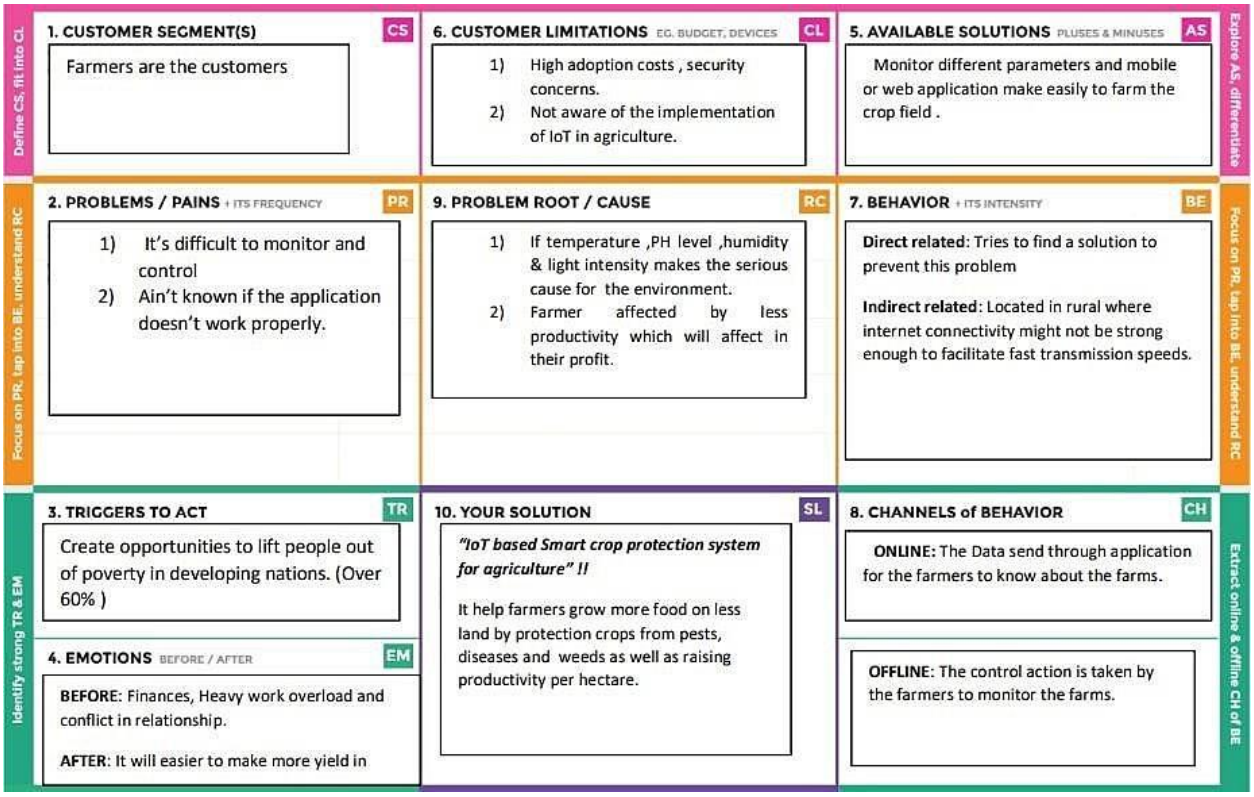
Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

Share template feedback

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Crops in farms are many times ravaged by local animals like buffaloes, cows, goats, birds, and fire etc. This leads to huge losses for the farmers. It is not possible for farmers to barricade entire fields or stay on field 24 hours and guard it.
2.	Idea / Solution description	Here we propose an automatic crop protection system from animals and fire. This is an arduino Uno based system using microcontroller. This system uses a motion sensor to detect wild animals approaching near the field and smoke sensor to detect the fire.
3.	Novelty / Uniqueness	Fastest alert to the farmers through SMS.
4.	Social Impact / Customer Satisfaction	Real time data and production insight. Remote monitoring.
5.	Business Model (Revenue Model)	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 wellbeing.
6.	Scalability of the Solution	Alerts the farmers immediately through an SMS.

i. **Proposed Solution**

ii. **Problem Solution Fit**



1. **<u>REQUIREMENT ANALYSIS</u>**:

    i. **Functional Requirements**

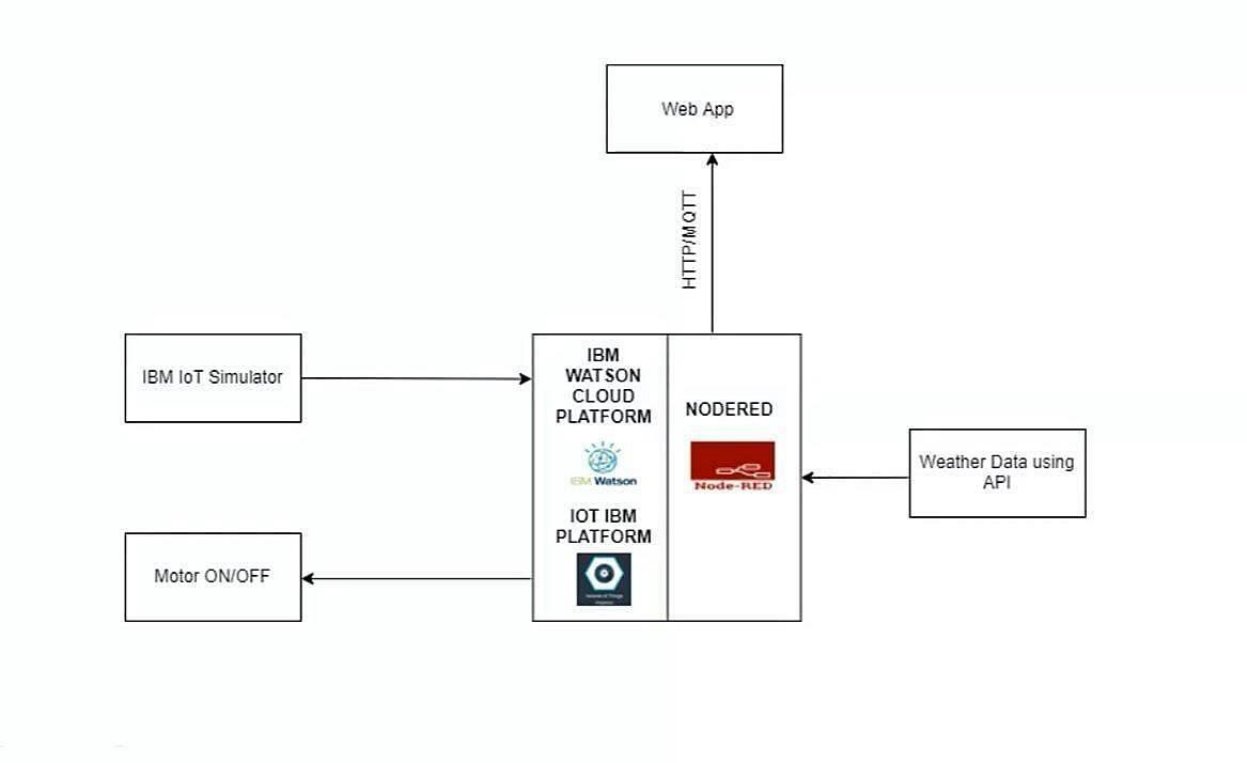
- **Functional Requirements:**

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Install the app. Signing up with Gmail or phone numbers. Creating a new profile. Understand the guidelines which we given
FR-2	User Confirmation	Email or phone number verification required via OTP.
FR-3	Accessing datasets	The data like values of temperature, data sensor, humidity, soil moisture are received by alert SMS.
FR-4	Interface sensor	Connect the sensor and the application When animals enter the field, the alarm is generated.
FR-5	User action	The user needs to take action like detecting through crop rotation, fertilizer, strip cropping.

ii. **Non Functional Requirements**

2. **PROJECT DESIGN**:

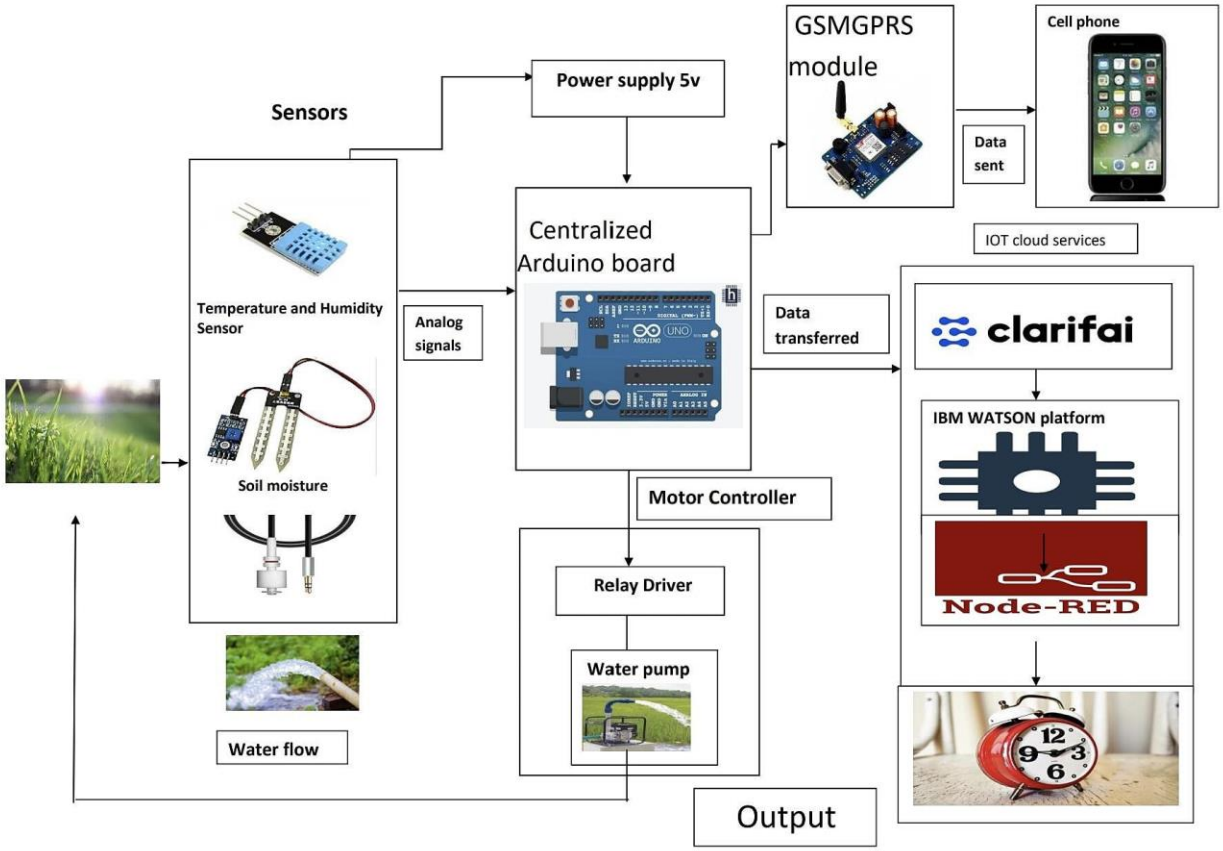


1. **Data Flow Diagram**

2. Solution & Technical Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

1. Find the best tech solution to solve existing business problems.
2. Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
3. Define features, development phases, and solution requirements.
4. Provide specifications according to which the solution is defined, managed, and delivered.



Explanation for the Architecture Diagram:

1. The device will detect the animals and birds using the Clarifai service.
2. If any animal or bird is detected the image will be captured and stored in the IBM Cloud object storage.
3. It also generates an alarm and avoid animals from destroying the crop.
4. It also generates an alarm and avoid animals from destroying the crop.
5. The image URL will be stored in the IBM Cloudant DB service.

6. The device will also monitor the soil moisture levels, temperature, and humidity values and send them to the IBM IoT Platform.
7. The image will be retrieved from Object storage and displayed in the web application.
8. A web application is developed to visualize the soil moisture, temperature, and humidity values.
9. Users can also control the motors through web applications

3.User Stories

User Type	Functional requirement(Epic)	User Story number	User Story/Task	Acceptancecriteria	Priority	Release
Customer (Mobil user)	Registration	USN-1	User can enter into the web application	I can accessmy account /dashboard	High	Sprint 1
		USN-2	User can register their credentials likeemail id and password	I can receive confirmation email & click confirm	High	Sprint 1
	Login	USN-3	User can log into theapplication by entering email & password	I can login tomy account	High	Sprint 1
	Dashboard	USN-4	User can view the temperature	I can view the data given by the device	High	Sprint 2
		USN-5	User can view thelevel of sensor monitoring value	I can view the data given by the device	High	Sprint 2
Customer(Web user)	Usage	USN-1	User can view the web page and get theinformation	I can view the data given by the device	High	Sprint 3
Customer	Working	USN-1	User act according to the alert given by thedevice	I can get thedata work according to it	High	Sprint 3

3. PROJECT PLANNING & SCHEDULING.

a. Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1		US-1	Create the IBM Cloud services which are being used in this project.	6	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-1		US-2	Configure the IBM Cloud services which are being used in completing this project.	4	Medium	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-2		US-3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the IBM Watson IoT platform.	5	Medium	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-2		US-4	In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credentials.	5	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-3		US-1	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	10	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K

Sprint-3		US-2	Create a Node-RED service.	10	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-3		US-1	Develop a python script to publish random sensor data such as temperature, moisture, soil and humidity to the IBM IoT platform	7	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-3		US-2	After developing python code, commands are received just print the statements which represent the control of the devices.	5	Medium	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-4		US-3	Publish Data to The IBM Cloud	8	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-4		US-1	Create Web UI in Node- Red	10	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K
Sprint-4		US-2	Configure the Node-RED flow to receive data from the IBM IoT platform and also use Cloudant DB nodes to store the received sensor data in the cloudant DB	10	High	Vishwajith.R Naveen.R Rajeev.M Venkatesh.K

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	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6. CODING & SOLUTIONING:

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "hrodmj" #replace the ORG ID
deviceType = "NODEMCU1"#replace the Device type
wi
deviceId = "12345"#replace Device ID
authMethod = "token"
authToken = "kp1234" #Replace the authtoken

def myCommandCallback(cmd): # function for
    Callback print("Command received: %s" %
    cmd.data) if cmd.data['command']=='motoron':
        print("Motor On IS RECEIVED")

    elif cmd.data['command']=='motoroff':

    print("Motor Off IS RECEIVED") if

    cmd.command == "setInterval":

        if 'interval' not in cmd.data:
            print("Error - command is missing required information: 'interval'")
        else:
            interval = cmd.data['interval']
    elif cmd.command == "print":
        if 'message' not in cmd.data:
            print("Error - command is missing required information: 'message'")
            else:
          output=cmd.data['message']
            print(output)
try:
      deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}
      deviceCli = ibmiotf.device.Client(deviceOptions)
      #..... except Exception as e:
      print("Caught exception connecting device: %s" %
      str(e)) sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an event of
type "greeting" 10 times deviceCli.connect() while True:
```

```
deviceCli.commandCallback = myCommandCallback # Disconnect the device and application from the cloud deviceCli.disconnect()
```

1.

1. **RESULT:**

We have successfully built an IOT Based Smart Crop Protection System for Agriculture and integrated all the services using Node-RED.

2. **ADVANTAGES & DISADVANTAGES:**

Advantages

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

Disadvantages

5. Smart Crop Protection requires internet connectivity continuously, but rural parts can not fulfill this requirement.
1. Any faults in the sensors can cause great loss in the agriculture, due to wrong records and the actions of automated processes.
2. IoT devices need much money to implement.

a. **CONCLUSION:**

IoT based smart Crop Monitoring System for Agriculture for Live Monitoring of Temperature and Soil Moisture and to control motor and light remotely has been proposed using Node Red and IBM Cloud Platform. The System has high efficiency and accuracy in fetching the live data of temperature and soil moisture. The IoT based smart farming System being proposed via this project will assist farmers in increasing the agriculture yield and take efficient care of food production as the System will always provide helping hand to farmers for getting accurate live feed of environmental temperature and soil moisture with more than 99% accurate results. Therefore, the project proposes a thought of consolidating the most recent innovation into the agrarian field to turn the customary techniques for water system to current strategies in this way making simple profitable and temperate trimming.

b. **FUTURE SCOPE**:

Agriculture domains encounters with many challenges starting from soil parameters, seed sowing, crop growth and its quality, weed handling, disease management till harvesting and storing crop. Artificial intelligence driven techniques along with other available tools and automation can address these challenges and proven the revolution in agriculture. Most popular AI application in agriculture is use of Robot and Drones, they perform almost all task like humans even at a faster rate with accuracy. From literature review it is clear that precision farming is probable by integrating sensors, cameras, data analytics, GPS and remote sensing. Image recognitions software's, IoT sensors can be used for disease recognition at primary stages and hence crop health can be supervised which increases superior quality production with minimum loss. Table 1 demonstrate the various applications in view of Smart Agriculture for improved evolution as well as superiority. Still there are several challenges associated with AI and IoT application in smart agriculture which is the promising future to be explored area for researchers. Some of major challenges are: Awareness issues Hardware implementation challenges Cost of software and hardware Network management Energy management Privacy issues Security challenges Interoperability of systems with the induction of Computer vision, Deep learning, Big data also agriculture sector has influenced a lot. Researchers can integrate IoT sensors along with smart systems and computational optimization algorithms to overcome the limitations/shortcomings. Smart Agriculture has a budding potential towards productivity, precision, optimization, adaptive resource management and intelligent food

traceability. It will contribute to environment also in terms of efficient use of water, prevent disease contamination and precise use of pesticides.

c. **APPENDIX**:

```
import random
        import ibmiotf.device          from
time import sleep
                    import sys
                    #IBM Watson Device Credentials.
        organization = "op701j"          deviceType
= "Lokesh"          deviceId = "Lokesh89"
authMethod = "token"          authToken =
"1223334444"          def
myCommandCallback(cmd):
                    print("Command received: %s" % cmd.data['command'])
 status=cmd.data['command']                  if
status=="sprinkler_on":       print ("sprinkler
is ON")   else :
                      print ("sprinkler is OFF")
                    #print(cmd)



                                                                                try:
                    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}    deviceCli =
ibmiotf.device.Client(deviceOptions)  except Exception as e:
                      print("Caught exception connecting device: %s" % str(e))
                    sys.exit()
                    #Connecting to IBM watson.
        deviceCli.connect()          while
True:
                    #Getting values from sensors.
                     temp_sensor = round( random.uniform(0,80),2)
                     PH_sensor = round(random.uniform(1,14),3)
```

```
                camera = ["Detected","Not Detected","Not Detected","Not Detected","Not
        Detected","Not Detected",]     camera_reading =
random.choice(camera)
                flame = ["Detected","Not Detected","Not Detected","Not Detected","Not
        Detected","Not Detected",]     flame_reading =
random.choice(flame)          moist_level =
round(random.uniform(0,100),2)
                water_level = round(random.uniform(0,30),2)



               #storing the sensor data to send in json format to cloud.



         temp_data = { 'Temperature' : temp_sensor }           PH_data
= { 'PH Level' : PH_sensor }          camera_data = { 'Animal attack' :
camera_reading}          flame_data = { 'Flame' : flame_reading }
moist_data = { 'Moisture Level' : moist_level}            water_data =
{ 'Water Level' : water_level}



               # publishing Sensor data to IBM Watson for every 5-10 seconds.  success =
               deviceCli.publishEvent("Temperature sensor", "json", temp_data,
               qos=0)
        sleep(1)           if
success:
                  print (" .............................publish ok..... ")
            print ("Published Temperature = %s C" % temp_sensor, "to IBM Watson")



              success = deviceCli.publishEvent("PH sensor", "json", PH_data, qos=0)
        sleep(1)           if
success:
               print ("Published PH Level = %s" % PH_sensor, "to IBM Watson")



             success = deviceCli.publishEvent("camera", "json", camera_data, qos=0)
        sleep(1)           if
success:
```

```
                print ("Published Animal attack %s " % camera_reading, "to IBM
            Watson")
             success = deviceCli.publishEvent("Flame sensor", "json", flame_data,
            qos=0)  sleep(1)  if success:
              print ("Published Flame %s " % flame_reading, "to IBM Watson")


             success = deviceCli.publishEvent("Moisture sensor", "json", moist_data,
            qos=0)
        sleep(1)          if
success:
                print ("Published Moisture Level = %s " % moist_level, "to IBM
            Watson")


             success = deviceCli.publishEvent("Water sensor", "json", water_data,
            qos=0)
        sleep(1)          if
success:
          print ("Published Water Level = %s cm" % water_level, "to IBM Watson")
             print ("")
             #Automation to control sprinklers by present temperature an to send alert
            message to IBM Watson.

                                                                    if (temp_sensor > 35):
                print("sprinkler-1 is ON")
             success = deviceCli.publishEvent("Alert1", "json",{ 'alert1' :
           "Temperature(%s) is high, sprinkerlers are turned ON" %temp_sensor }
       ,              qos=0)
sleep(1)          if success:
                print( 'Published alert1 : ', "Temperature(%s) is high, sprinkerlers are
             turned ON" %temp_sensor,"to IBM Watson")
       print("")          else:
       print("sprinkler-1 is OFF")    print("")


             #To send alert message if farmer uses the unsafe fertilizer to crops.
```

```
                                                    if (PH_sensor > 7.5 or PH_sensor < 5.5):
                    success = deviceCli.publishEvent("Alert2", "json",{ 'alert2' : "Fertilizer PH
                level(%s) is not safe,use other fertilizer" %PH_sensor } ,
                qos=0)
                sleep(1)  if
                success:
                print('Publi
                shed alert2
                : ' ,
                "Fertilizer
                PH
                level(%s)
                is not
                safe,use
                other
                fertilizer" %PH_sensor,"to IBM Watson")
                 print("")


                 #To send alert message to farmer that animal attack on crops.


                                                        if (camera_reading == "Detected"):
                    success = deviceCli.publishEvent("Alert3", "json", { 'alert3' : "Animal
                attack on crops detected" }, qos=0)
          sleep(1)            if
success:
                    print('Published alert3 : ' , "Animal attack on crops detected","to IBM
        Watson","to IBM Watson")     print("")
                 #To send alert message if flame detected on crop land and turn ON the
                splinkers to take immediate action.


                                                       if (flame_reading == "Detected"):
                    print("sprinkler-2 is ON")
                 success = deviceCli.publishEvent("Alert4", "json", { 'alert4' : "Flame is
                detected crops are in danger,sprinklers turned ON" }, qos=0)
          sleep(1)            if
success:
                    print( 'Published alert4 : ' , "Flame is detected crops are in
```

```
                    danger,sprinklers turned ON","to IBM Watson")


                    #To send alert message if Moisture level is LOW and to Turn ON Motor-1
for irrigation.   if (moist_level < 20):      print("Motor-1 is ON")
                    success = deviceCli.publishEvent("Alert5", "json", { 'alert5' : "Moisture
 level(%s) is low, Irrigation started" %moist_level }, qos=0)           sleep(1)
if success:
                      print('Published alert5 : ' , "Moisture level(%s) is low, Irrigation started"
                    %moist_level,"to IBM Watson" )
                    print("")
                    #To send alert message if Water level is HIGH and to Turn ON Motor-2 to
                    take water out.  if (water_level > 20):     print("Motor-2 is ON")
                    success = deviceCli.publishEvent("Alert6", "json", { 'alert6' : "Water
                    level(%s) is high, so motor is ON to take water out "
                    %water_level }, qos=0)
        sleep(1)            if
success:
                      print('Published alert6 : ' , "water level(%s) is high, so motor is ON to take
       water out " %water_level,"to IBM Watson" )              print("")
 #command recived by farmer  deviceCli.commandCallback =
myCommandCallback
                  # Disconnect the device and application from the cloud deviceCli.disconnect()
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-43615-1660718462