PROJECT REPORT

IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

TEAM ID: PNT2022TMID36755

Team Leader

HARI PRIYA (210219106012)

Team Members
KAVERI
(210219106022)

Bharathi
Srimathi
Jaicinth salom

In

Bachelor of Engineering

Apollo engineering college, Anna University, Chennai.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
1	**INTRODUCTION** 1.1 Project Overview 1.2 Purpose	4
2	**LITERATURE SURVEY** 2.1 Existing problem 2.2 References 2.3 Problem Statement Definition	5
3	**IDEATION & PROPOSED SOLUTION** 3.1 Empathy Map Canvas 3.2 Ideation & Brainstorming 3.3 Proposed Solution 3.4 Problem Solution fit	7
4	**REQUIREMENT ANALYSIS** 4.1 Functional requirement 4.2 Non-Functional requirements	11
5	**PROJECT DESIGN** 5.1 Data Flow Diagrams 5.2 Solution & Technical Architecture 5.3 User Stories	13
6	**PROJECT PLANNING & SCHEDULING**	15

	6.1 Sprint Planning & Estimation 6.2 Sprint Delivery Schedule 6.3 Reports from JIRA	
7	**CODING & SOLUTIONING** 7.1 Feature 1 7.2 Feature 2	19
8	**TESTING**	22
9	**RESULTS**	23
10	**ADVANTAGES & DISADVANTAGES**	23
11	**CONCLUSION**	24
12	**FUTURE SCOPE**	24
13	**APPENDIX**	25

1. INTRODUCTION:

1.1 Project Overview

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 we this project have the provision to secure from the human begins also. This can achieve by the help of IOT device that we are discuss in this paper. The SCPS work on the battery so that this project can be easily portable and also we are add 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. This project is smart crop protection system for protect the farm from animals as well as unknown person. This projects contents ardiuno UNO, Nodemcu, LCD display, PIR sensor, flame sensor, sd card module, solar panel, solar charges converter. This whole project is work on 12v dc supply from battery. We used solar panel to charge the battery. 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. 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.

2.2 Purpose

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.

2. LITERATURE SURVEY:

2.1 Existing Problem

• 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.

• 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.

• Sometimes over supply of water or less supply of water affects the growth of crops.

• Sometimes if the weather/temperature changes suddenly it is necessary to take certain actions.

• Specific crops grow better in specific conditions, they may get damaged due to bad weather.

2.2 References

• https://smartinternz.com/assets/docs/Smart%20Home%
20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

• https://smartinternz.com/assets/docs/Smart%20Home%
20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

• https://smartinternz.com/assets/docs/Sending%2 0Http%
20request%20to%20Open%20weather%20map%20 web
site%20to%20get%20the%20weather%20forecast.pdf

• https://www.youtube.com/watch?v=cicTw4SEdxk

• https://smartinternz.com/assets/docs/Smart%20Home%
20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

2.3 Problem Statement Definition

- Smart Crop Protection System based on IoT can monitor soil moisture and climatic conditions to grow and yield a good crop.

- The farmer can also get the real time weather forecasting data by using external platforms like Open Weather API.

- Farmer is provided a mobile app using which he can monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.

- Based on all the parameters he can water his crop by controlling the motors using the mobile application.

- 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.

- Here we are using the Online IoT simulator for getting the Temperature, Humidity and Soil Moisture values.

3. IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map Canvas

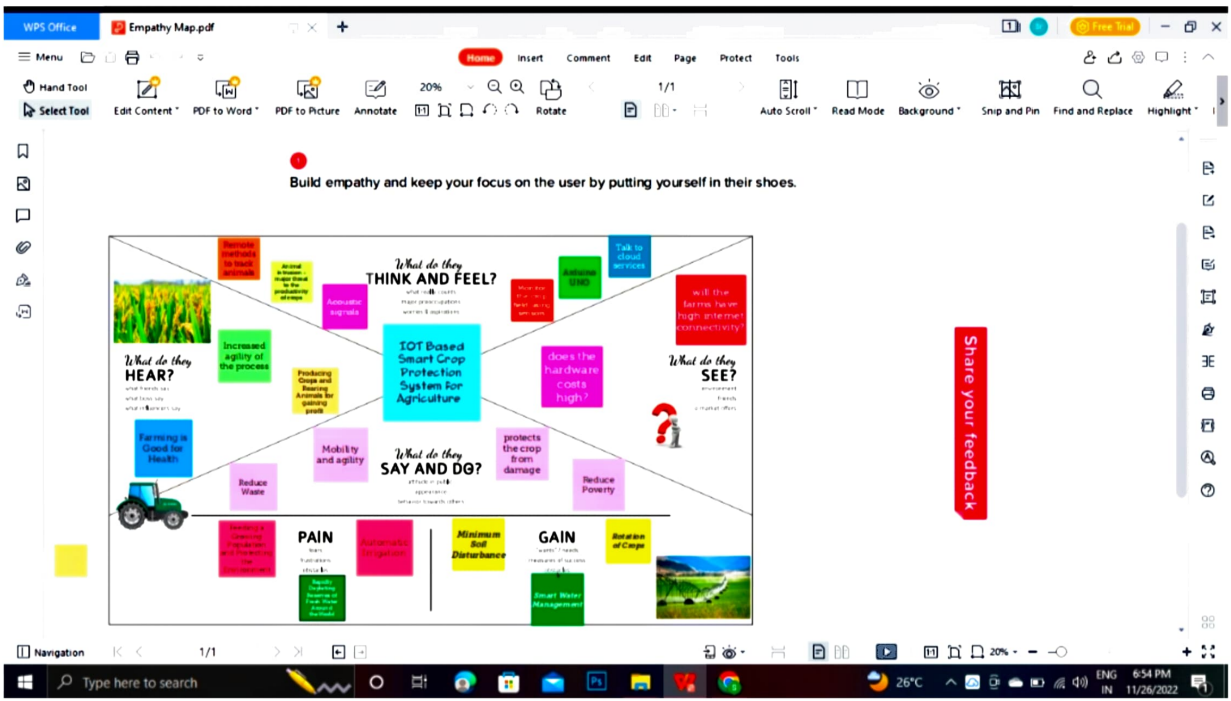
Edit this template
Right-click to unlock

Empathy Map Canvas

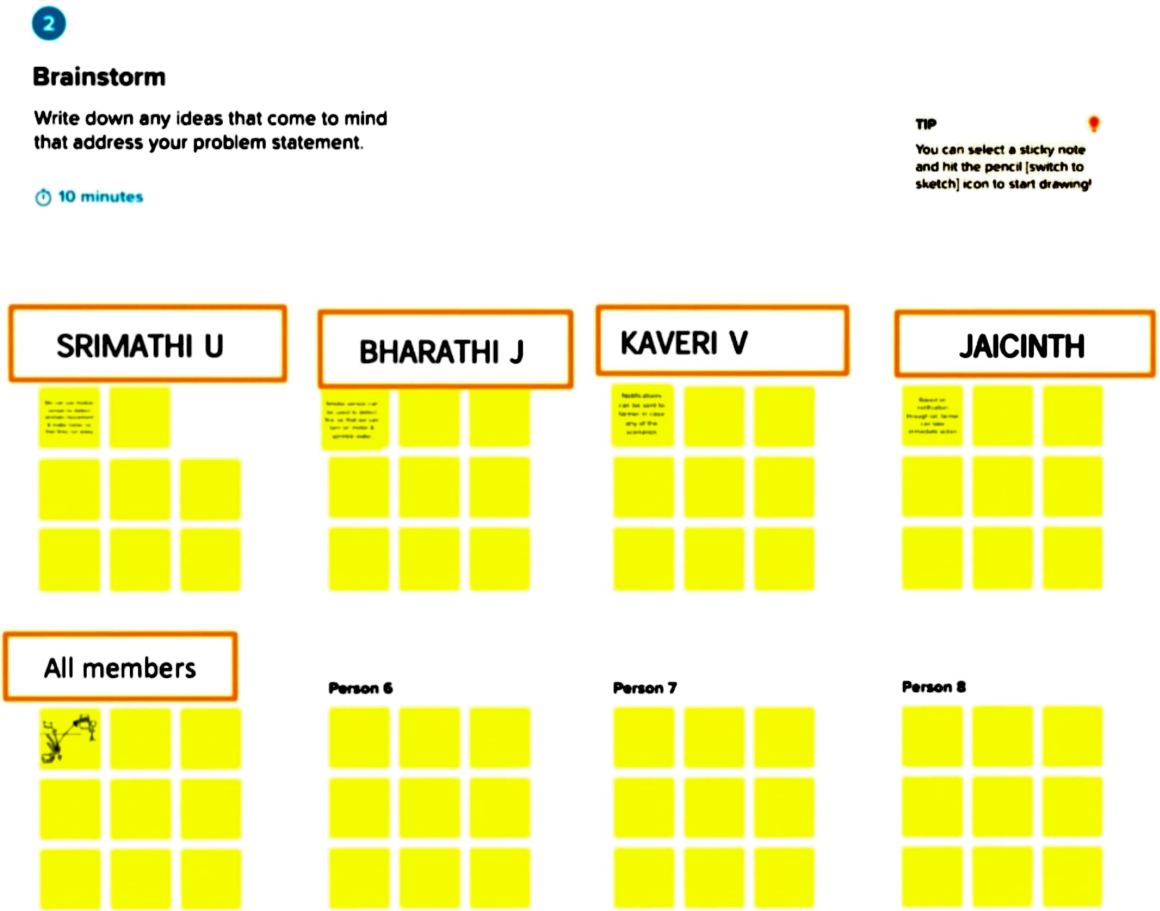
Gain insight and understanding on solving customer problems.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.



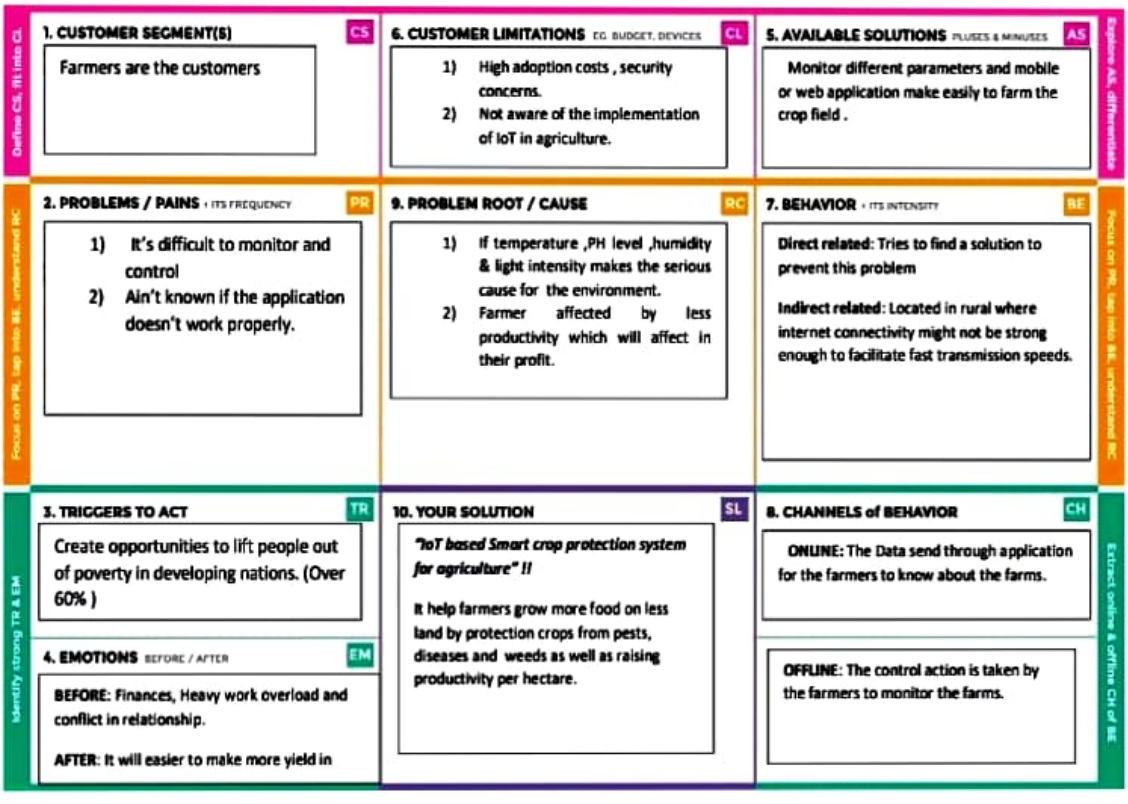
3.2 Ideation & Brain Storming



3.3 Proposed Solution

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.

3.4 Problem Solution Fit



4.REQUIREMENT ANALYSIS:

4.1.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.

4.2.Non Functional Requirements

- **Non-functional Requirements:**

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	This project's contributors to the farm protection through the smart protection system and use new technologies and also increase the quality of its crop.

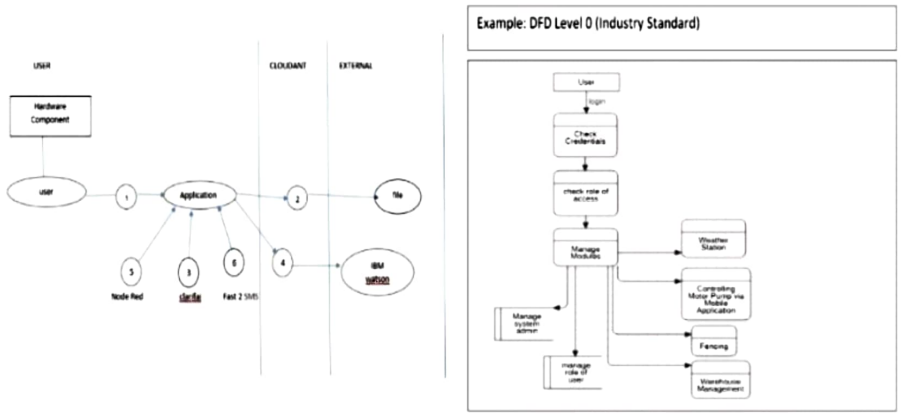
NFR-2	**Security**	It was created to protect the crops from animals.
NFR-3	**Reliability**	Farmers are able to safeguard their lands by help of this technology. They get some good benefits from higher crop yields.
NFR-4	**Performance**	When animals attempt to enter the crop field, IOT devices and sensors alert the farmer via message and maintain good yields.
NFR-5	**Availability**	Agriculture fences are quite an effective wild animal protection system.
NFR-6	**Scalability**	The develop system will not harmful and injurious to animals as well as human beings through the system.

5.PROJECT DESIGN:

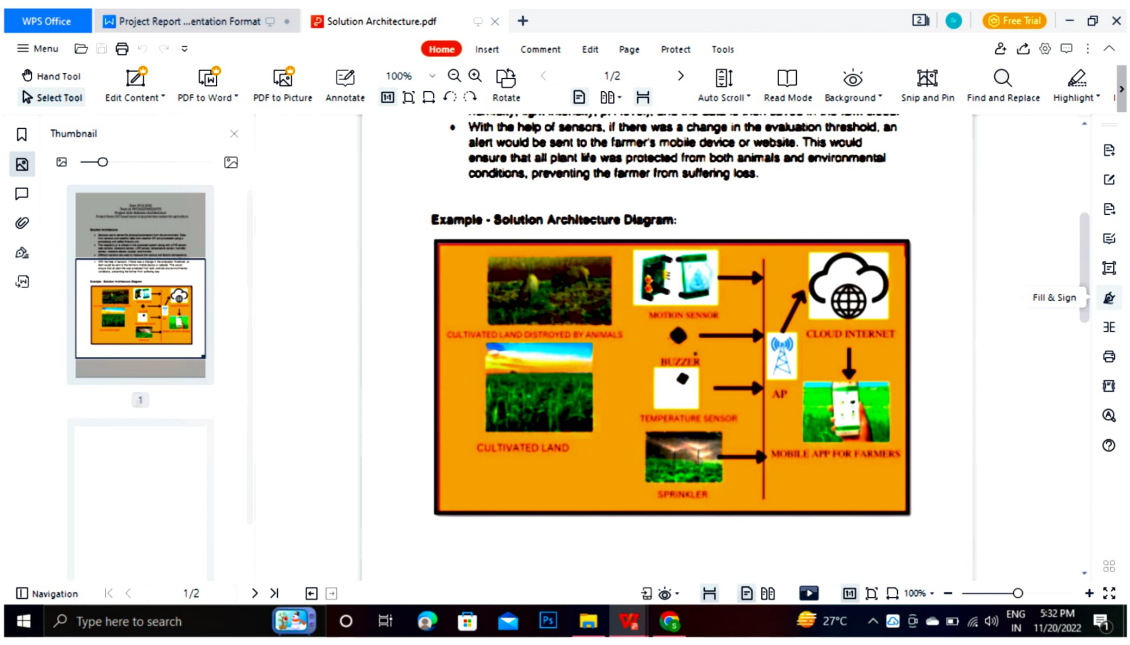
5.1 Data Flow Diagram

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2.Solution & Technical Architecture



13.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
 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('Published 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
 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()
```

9.RESULT:

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

10.ADVANTAGES & DISADVANTAGES:

10.1 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 automatedand the quality of crops can be maintained.
- The process included in farming can be controlled using the web applications from anywhere, anytime.

10.2 Disadvantages

Smart Crop Protection requires internet connectivity continuously, but rural parts cannot fulfill this requirement.

1. Any faults in the sensors can cause great loss in the agriculture, due to wrong record and the actions of automated processes.
2. IoT devices need much money to implement.

11 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 CloudPlatform. 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.

12.FUTURE SCOPE:

In future due to more demand of good and more farming in less time, for betterment of the crops and reducing the usage of extravagant resource like electricity and water IoT can be implemented in most of the places.