**SMART FARMER: IOT ENABLED SMART FARMING**

**APPLICATION**

**PROJECT REPORT**

**SUBMITTED BY**

**ANUPRIYA.J (712819106701)**

**VASUKI.V (712819106704)**

**SUDALAI.M (712819106706)**

**MADHAVASHANMUGAM.D (712819106703)**

**INPARTIAL FULFILLMENT FOR THE**

**AWARD OF THE DEGREE**

***of***

# BACHELOR OF ENGINEERING

in

# ELECTRONICS AND COMMUNICTION ENGINEERING

# RVS COLLEGE OF ENGINEERING AND

# TECHNOLOGY

# 

# COIMBATORE-641402

ANNA UNIVERSITY-CHENNAI 600 025

NOVEMBER-2022

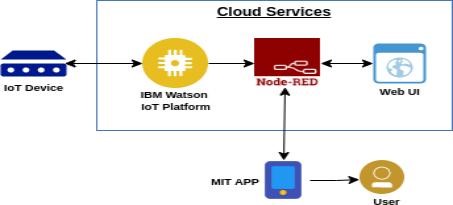
# 

# SMART FARMING

1.**INTRODUCTION:**

**1.1 PROJECT OVERVIEW:**

* IoT-based agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, Temperature, humidity using some sensors.
* This is system that enables framers to monitor and control their forms with a web-based application build with Node-RED.
* It uses the IBM IOT Watson cloud platform as its Backend.



**1.2 PURPOSE**:

Smart Farming reduce the ecological foot print of farming. Minimized

Or SiteSpecific application of inputs, such as fertilizers and pesticides, in precision agriculture systems will mitigate leaching problems as well as the emission of greenhouse gases.

1. **LITERATURE SURVEY:**

2.1 EXISTING PROBLEM:

The biggest challenges faced by IoT in the agricultural sector are lack of information, high adoption costs, and security concerns etc. Most of the farmers are not aware of the implementation of IoT in agriculture.

To successfully deploy a smart agriculture system, consider setting up a communications network that can integrate a limited number of sensors across a large area of farmland. This will require third-party network provisioning or setting up a private network consisting of access points and uplinks to a private backhaul network, which channels all the data traffic to centralized monitoring software or an analytics head-end system • It is not a secure system.

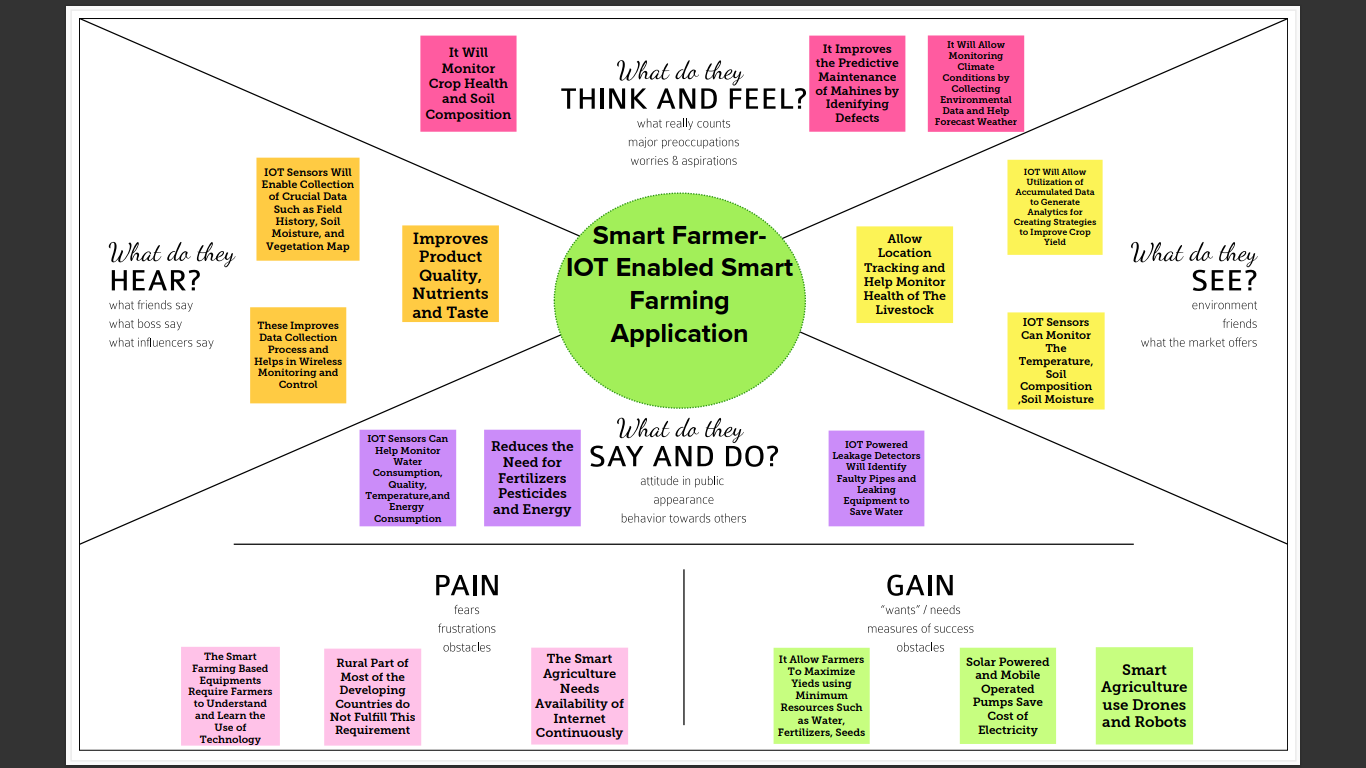
* There is no motion detection for protection of agriculture field.
* Automation is not available.
  1. **REFERENCES:**

It is the application of modern ICT (Information and Communication Technologies) into agriculture. In IOT- based smart farming, a system is built for monitoring the crop field with the help of sensors (light, humidity, temperature, soil moisture, etc.). The farmers can monitor the field conditions from anywhere.

* 1. **PROBLEM STATEMENT DEFINITION:**

Overuse of pesticides and fertilizer in agricultural fields leads to destruction of the crop as well as reduces the efficiency of the field increasing the soil vulnerability toward pest. IoT applications may be used to update the farmer/user about type & quantity of pesticide required by the crop.

1. **IDEATION & PROPOSED SOLUTION**:
   1. **EMPATHY MAP CANVAS:**



* 1. **IDEATION AND BRAINSTORMING:**

Introduction on Internet of Things (IoT), application of IoT in

agricultural field to improve the yield and quality by reducing the cost is provided. The sensors which are used in the architecture are discussed briefly and the process of transmission of data from the agriculture field to the central system is explained. The proposed system advantages are included. In addition, open research issues, challenges, and future of IoT in agricultural field are highlighted. The concept is basically developed on an idea, where there are numerous things or objects - such as Arduino, sensors, GSM models, LCD display, etc., that are connected with the Internet. Each of the objects has a different address and is able to interact with other items. The things or objects co-operate with each other to reach a common goal.

We are going to construct a smart agricultural monitoring

system which can collect crucial agricultural data and send it to an IoT platform called Thing speak in real time where the data can be logged and analyzed. The logged data on Thing speak is in graphical format, a botanist or a reasonably knowledged farmer can analyze the data (from anywhere in the world) to make sensible changes in the supplied resources (to crops) to obtain high quality yield.

Smart agriculture monitoring system or simply smart farming is

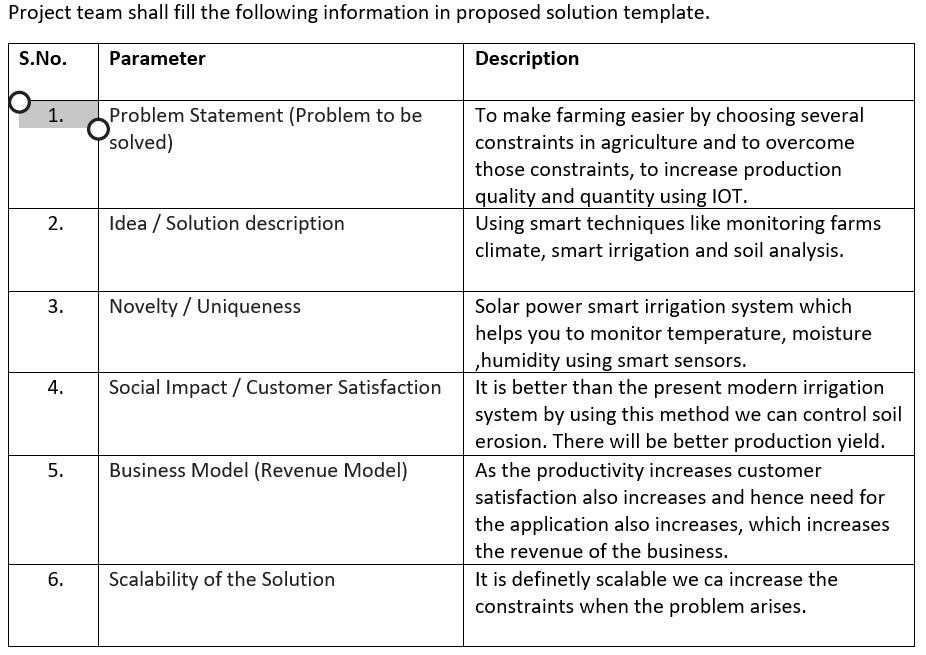
an emerging technology concept where data from several agricultural fields ranging from small to large scale and its surrounding are collected using smart electronic sensors. The collected data are analyzed by experts and local farmers to draw short term and long-term conclusion on weather pattern, soil fertility, current quality of crops, amount of water that will be required for next week to a month etc.

We can take smart farming a step further by automating

several parts of farming, forexample smart irrigation and water management. We can apply predictive algorithms on microcontrollers or SoC to calculate the amount of water that will be required today for a particular agriculture field. Say, if there was rain yesterday and the quantity of water required today is going to be less. Similarly, if humidity was high the evaporation of water at upper ground level is going to be less, so water required will be less than normal, thus reducing water usage.

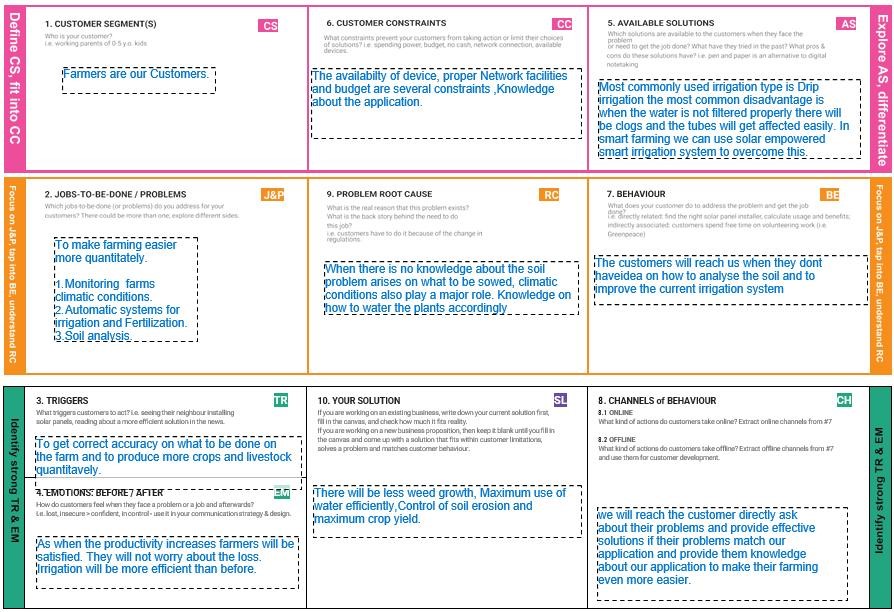
**3.3 PRPOSED SOLUTION:**

* To develop a Smart Agricultural System based on IOT which can give real time data and can help farmers in a very efficient manner.
* Soil Moisture can be checked by using the sensors that can sense the soil condition and send the data (moisture content in the soil) over the cloud services to the web application.



**3.4 PROBLEM SOLUTION FIT:**

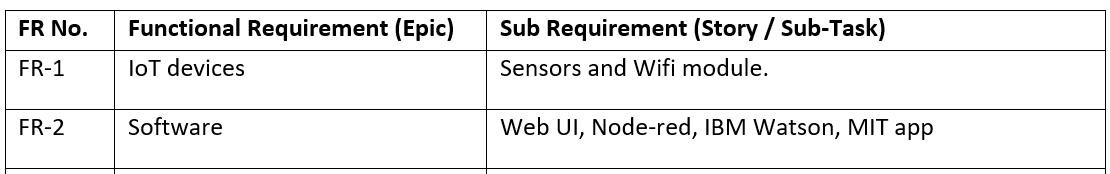
* With the help of the IoT devices, you can know the real-time status of the crops by capturing the data from sensors.
* Using predictive analytics, you can get an insight to make better decisions related to harvesting.



**4.REQUIREMENT ANALYSIS:**

* 1. **FUNCTIONAL REQUIREMENTS:**

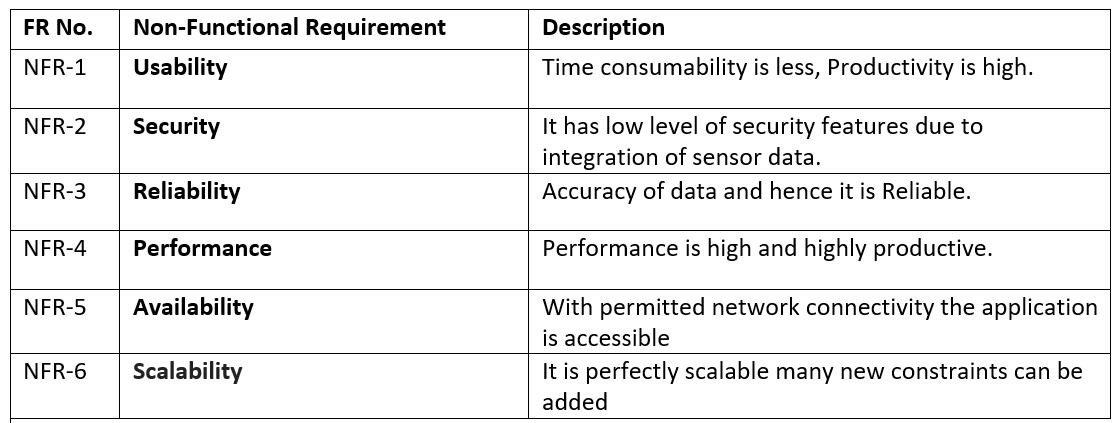
Following are the functional requirements of the proposed solution.



* 1. **NON -FUNCTIONAL REQUIREMENTS:**

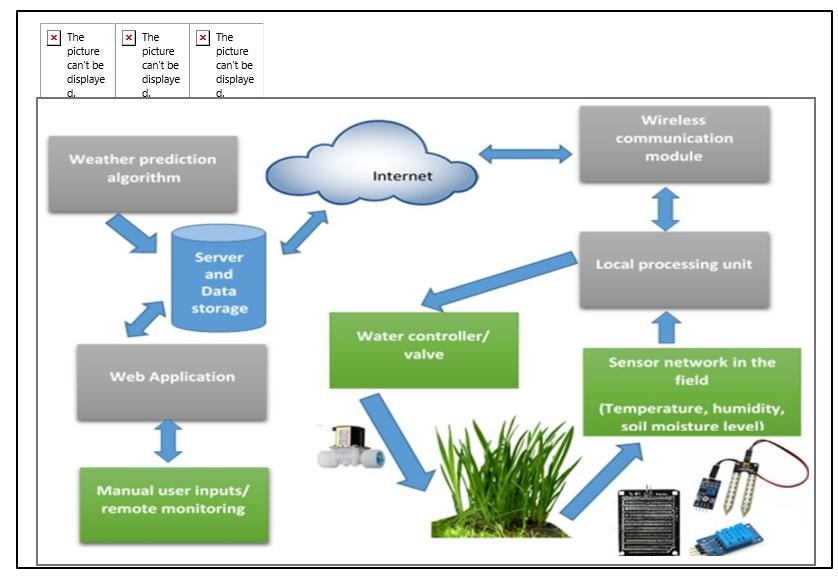
Following are the non-functional requirements of the proposed

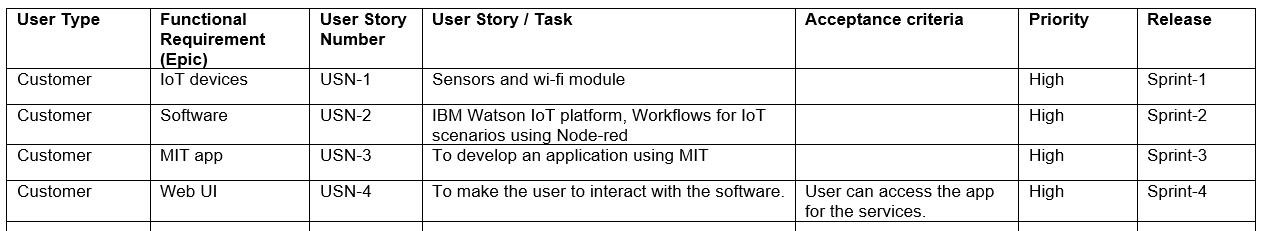
solution.



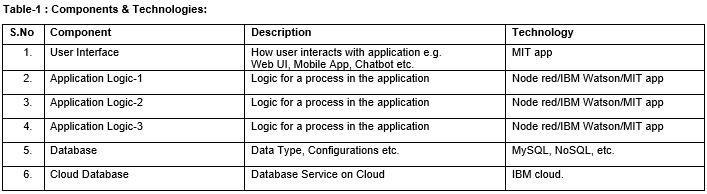
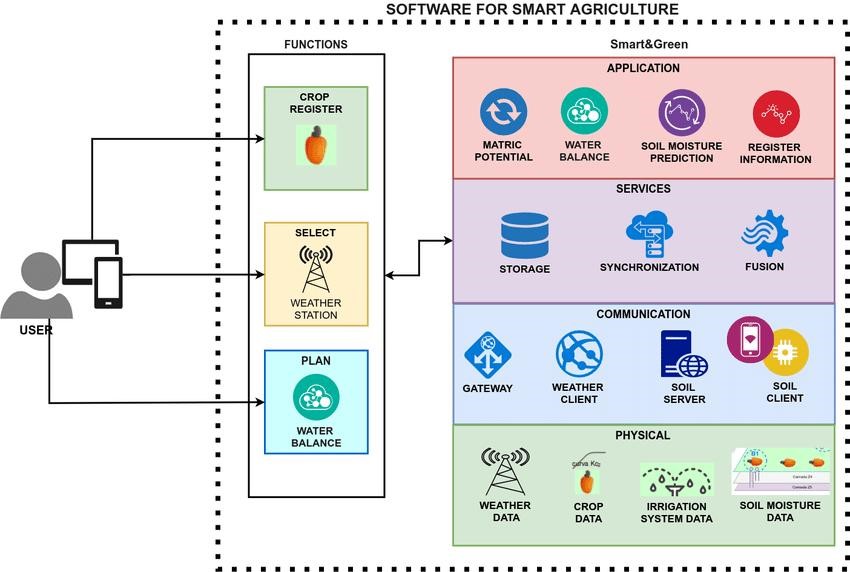
1. **PROJECT DESIGN:**
   1. **DATA FLOW DIAGRAM:**

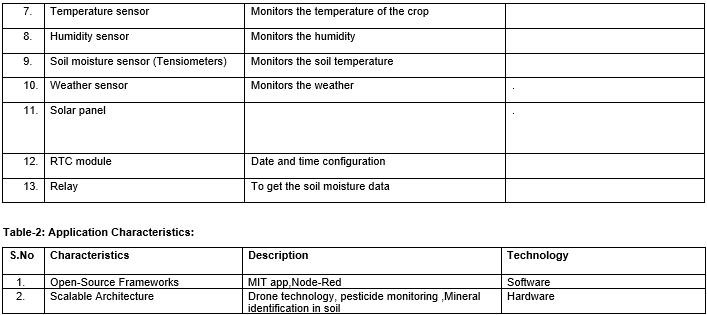
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.





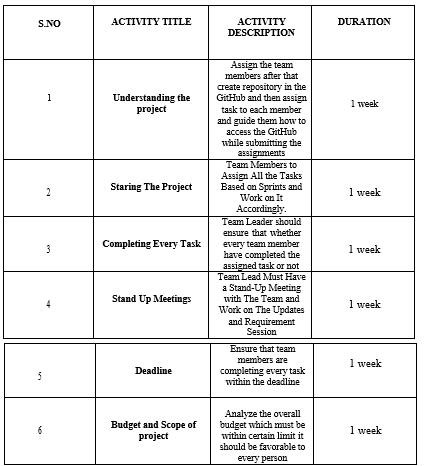
* 1. **SOLUTION AND TECHNICAL ARCHITECTURE:**



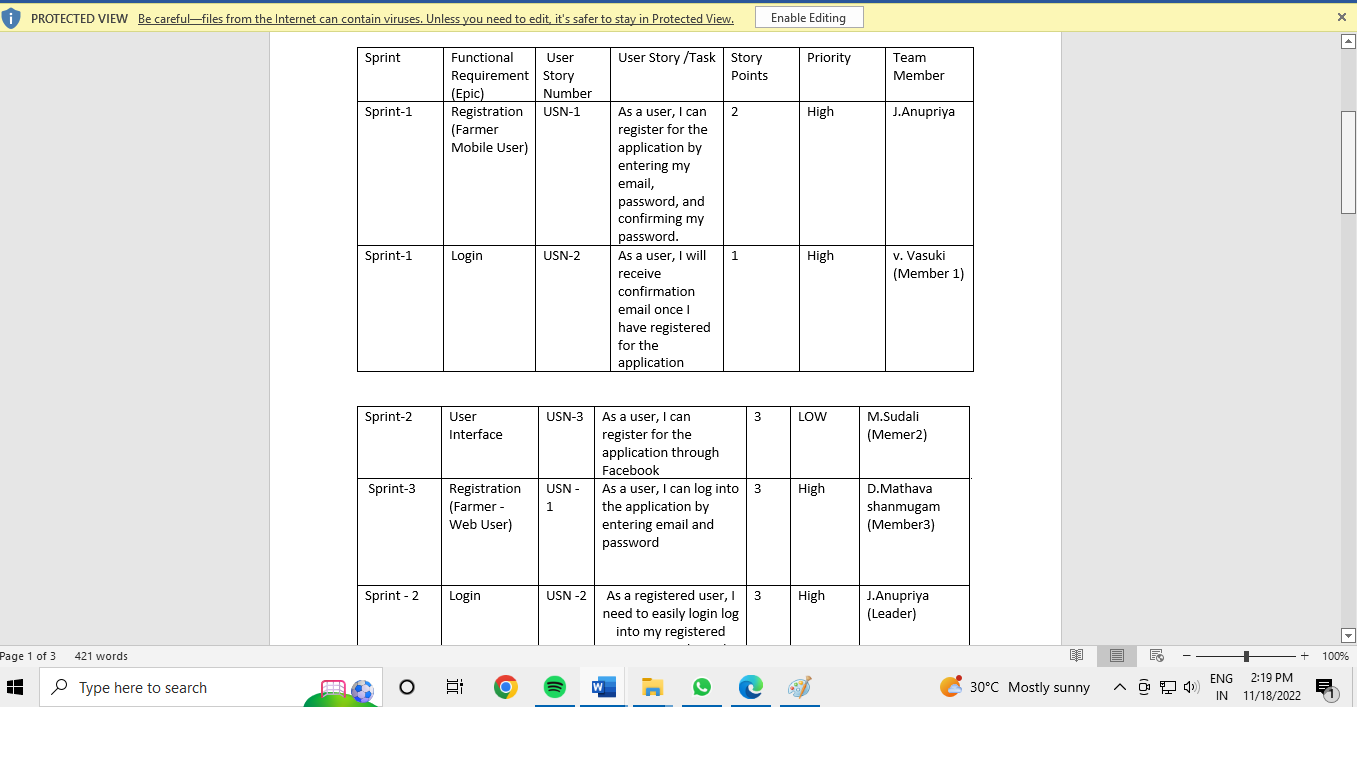


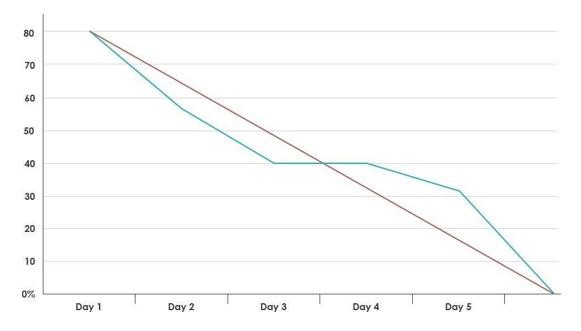
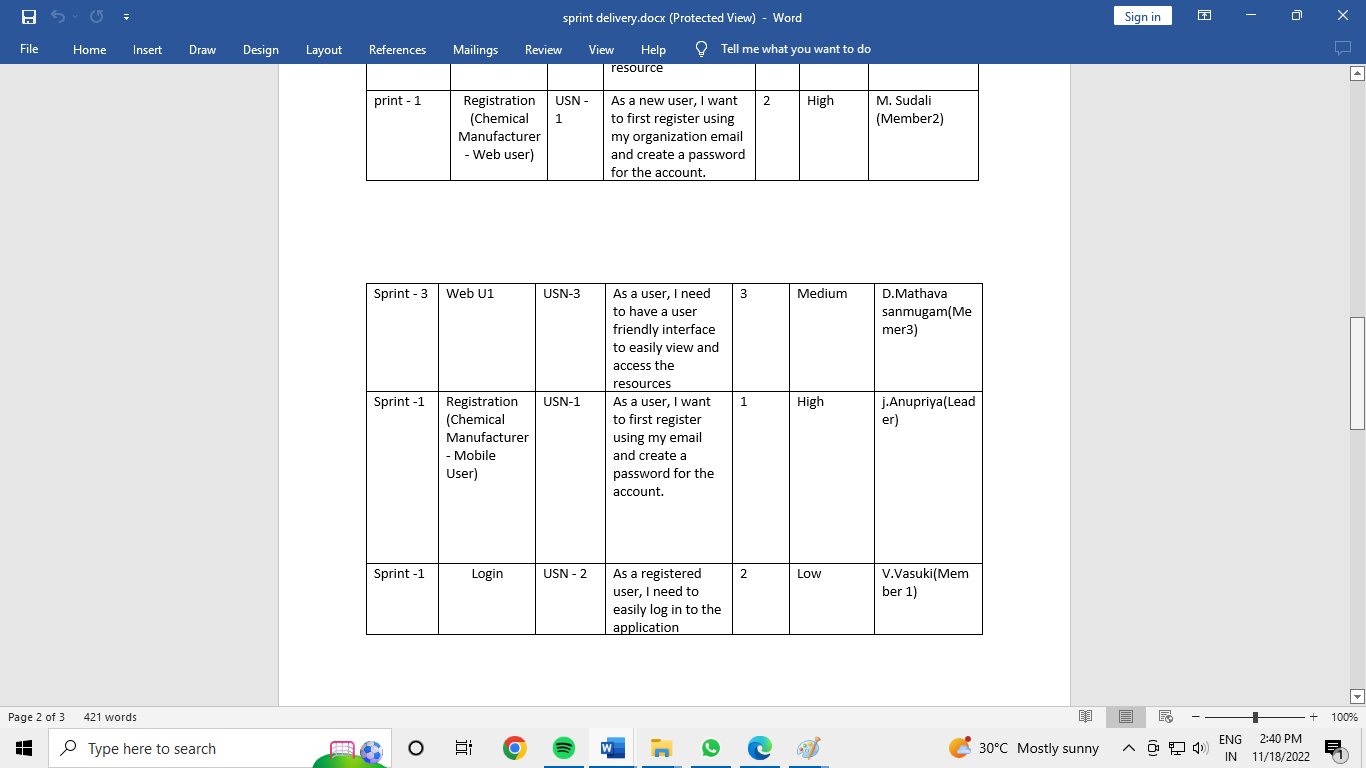
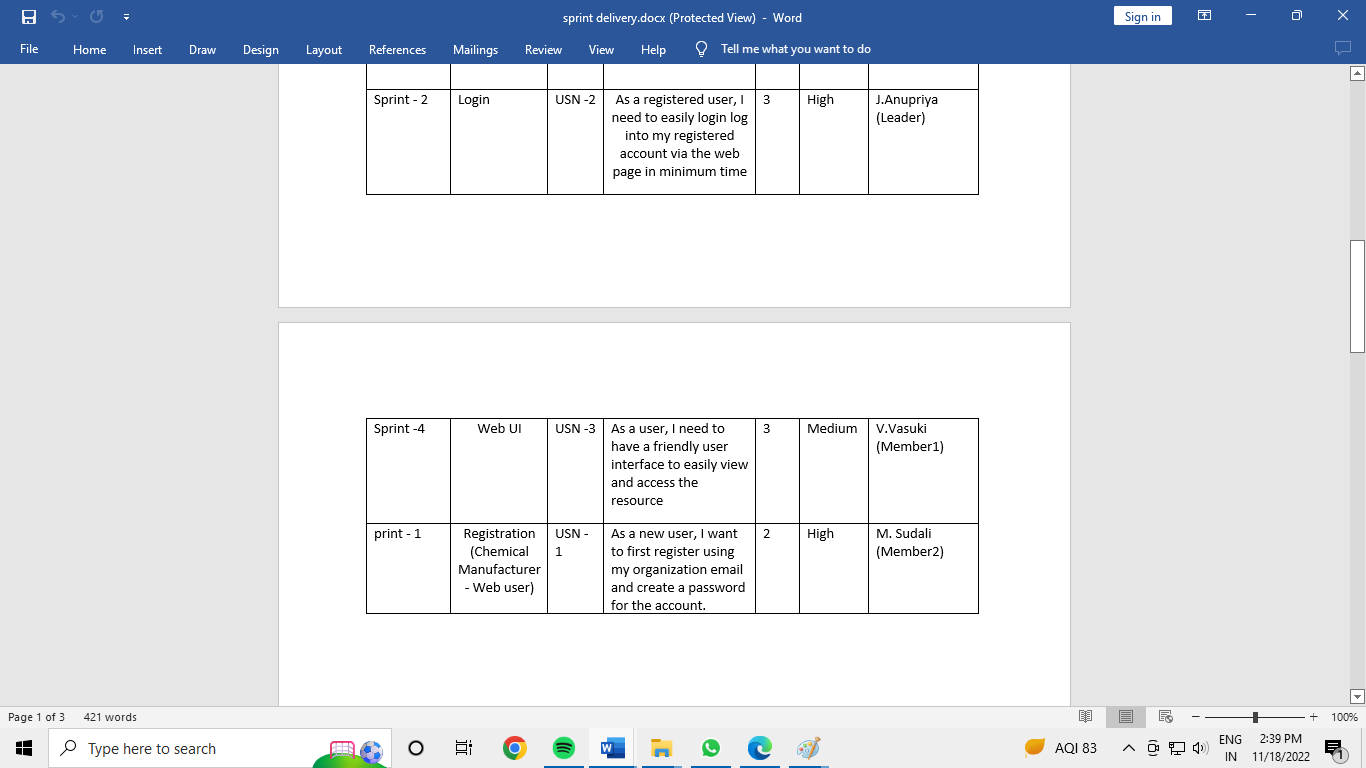
**6.PROJECT PLANNING AND SCHEDULING :**

* 1. **SPRINT PLANNING AND ESTIMATION**:



* 1. **SPRINT DELIVERY SCHEDULE:**





1. **CODING AND SOLUTIONING:**

**7.1 CODING:**

import time

import sys import

IBMTOT f. Application import

IBMIOT f. Device import random

#Provide your IBM Watson Device organization = "r8cpvf" Device Type = "farming" device Id = "12345" auth Method = "token" auth Token ="87654321" # Initialize GPIO def my Command Callback (c md):

print ("Common received: %s" % cmd. data['command']) status=cmd. data['command'] if status=="m o t o r on":

print ("motor is on")

e l if status == "m o t o r off":

print ("motor is off")

else: print ("please send proper command")

try:

device Options = {"org": organization, "type": device Type, "id": device Id,

"auth-method":

auth Method, "auth-token": auth Token}

device C li = IBMIOT f. device. Client (device Options)

#.............................................. 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 device C li. Connect () while True:

#Get Sensor Data fromDHT11

temp=random. Rand int (90,110) Humid=random. Rand int (60,100) Mo is=random. Rand int (20,120) data = {'temp': temp, 'Humid': Humid, 'Mo is': Mo is}

#Print data def

my On Publish Callback ():

print ("Published Temperature = %s C" % temp, "Humidity

= %s %%" %Humid, "Moisture =%s deg c" % Mo is, "to IBM Watson") success = device li. Publish Event ("IoT Sensor", "j son",

data, q s=0, on\_ publish=my On Publish Callback) if not success

print ("Not connected to Io TF")

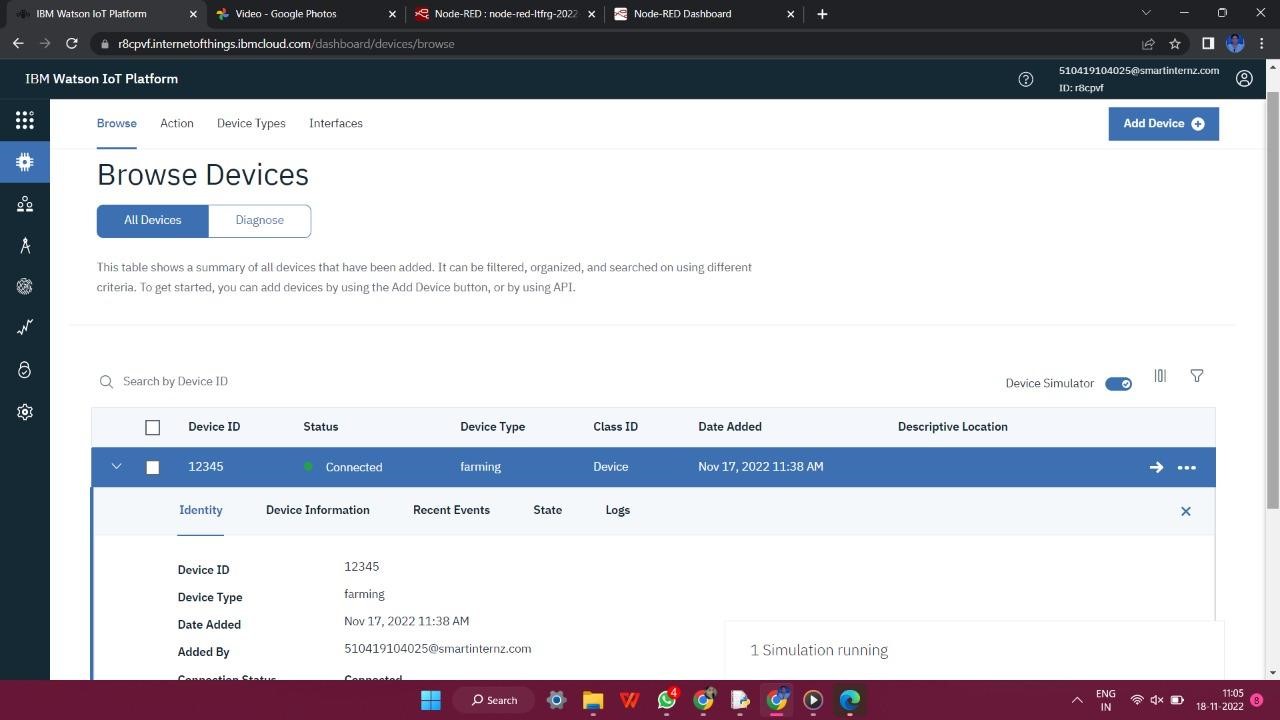
time. Sleep (10) device C li. Command Callback = my Command Callback

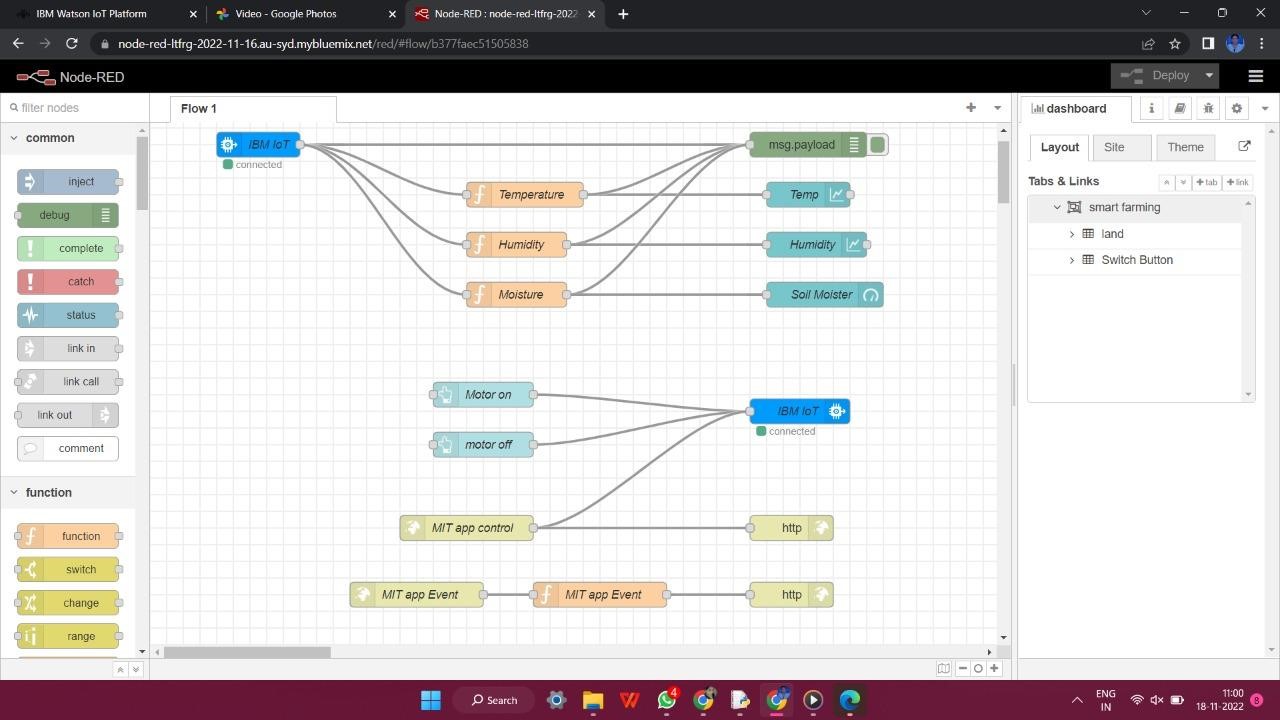
#Disconnect the device and application from the cloud device C li. Disconnect ()

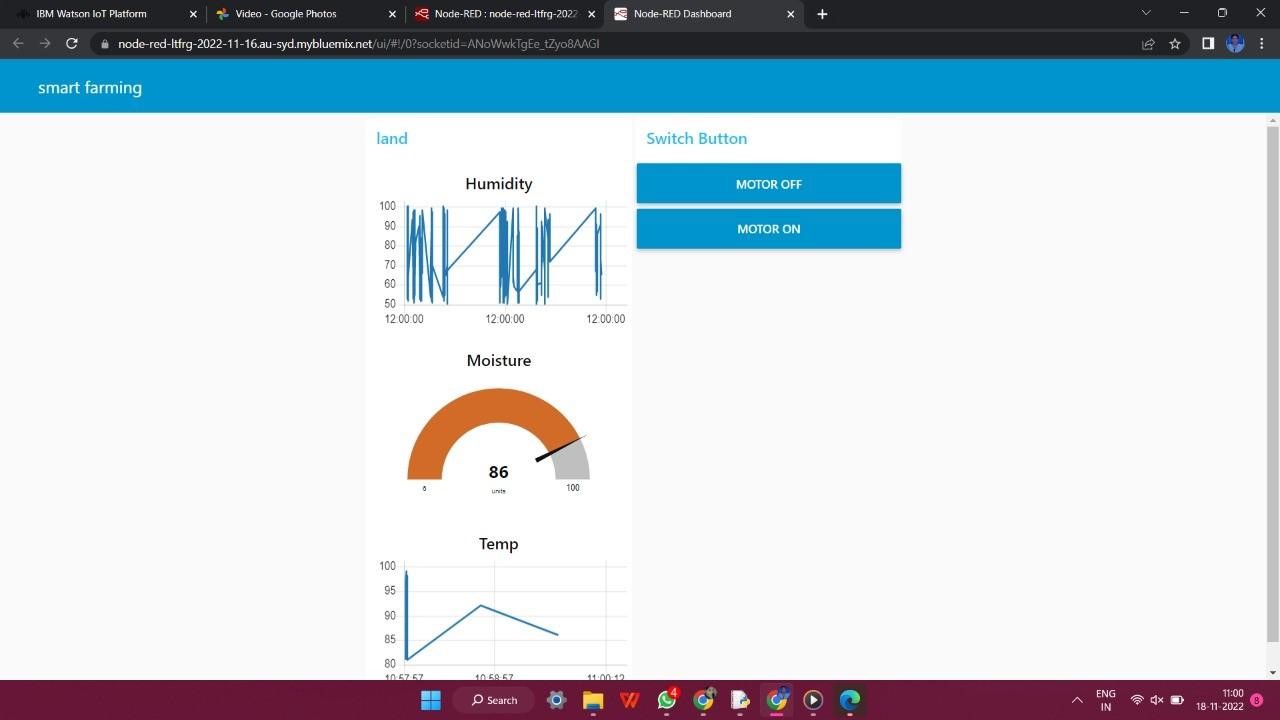
**7.2 SOLUTION:**



1. **TESTING:**

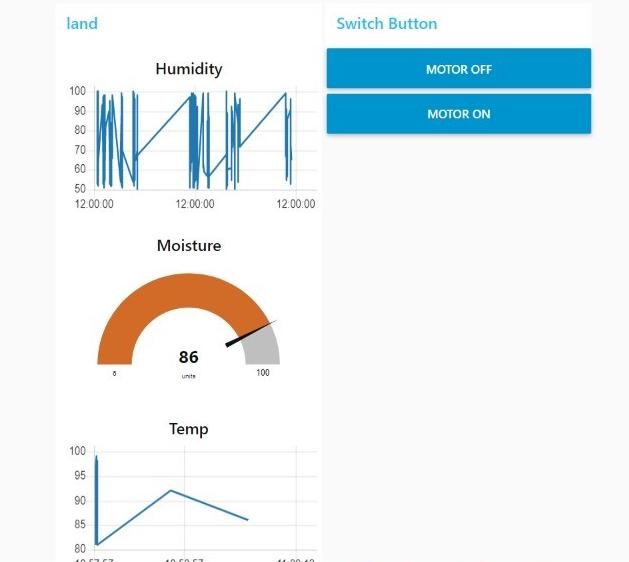






1. **RESULTS:**

We have successfully build a web based UI and integrated all the services using Node-RED.



1. **ADVANTAGES AND DISADVANTAGES:**
   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 extend
      * 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.
   2. **DISADVANTAGES:**
      * Smart Agriculture requires internet connectivity continuously, but rural parts cannot fulfill this requirement.
      * IoT devices need much money to implement.
2. **CONCLUSION:**

An IOT based smart agriculture system using Watson IOT Platform, Watson simulation, IBM cloud and Node-RED.

1. **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 resources like electricity and water IoT can be implemented in most of the places.

1. **APPENDIX :**
   1. SOURCE CODE:

import time import sys import ibmiotf.application import IBM IOT f. device import random

#Provide your IBM Watson Device organization = "r8cpvf" device Type = "farming" device Id = "12345" auth Method = "token" auth Token ="87654321" # Initialize GPIO def my Command Callback (c md):

print ("Command received: %s" % cmd. data['command']) status=cmd. data['command'] if status=="m o t o r on":

print ("motor is on")

e l if status == "m o t o r off":

print ("motor is off")

else:

print ("please send proper command")

try:

device Options = {"org": organization, "type": device Type, "id": device Id,

"auth-method": auth Method, "auth-token": auth Token}

device C li = IBMIOT f. device. Client (device Options)

#.............................................. 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 device C li. Connect () while True:

#Get Sensor Data fromDHT11

temp=random. Rand int (90,110)

Humid=random. rand int (60,100)

Mo is=random. Rand int (20,120) data = {'temp': temp, 'Humid': Humid, 'Mo is': Mo is}

# Print data def

my On Publish Callback ():

print ("Published Temperature = %s C" % temp, "Humidity

= %s %%" %Humid, "Moisture =%s deg c" % Mo is, "to IBM Watson") success = device C li. publish Event ("IoT Sensor", "j son",

data, q s=0, on \_publish=my On Publish Callback) if not success:

print ("Not connected to IoT F")

time. Sleep (10) device C li. Command Callback = my Command Callback

#Disconnect the device and application from the cloud device C li. Disconnect ()