VEL TECH HIGH TECH Dr. RANGARAJAN
Dr. SAKUNTHALAENGINEERING COLLEGE

PROJECT REPORT

Project Name:SMART FARMER- IOT ENABLEDSMART FARMING APPLICATION.

Team ID: PNT2022TMID22142

Team:

NAVEEN.V.M(TEAM LEAD)	113019106050
GHAJENDHIRAN.R(TEAM MEMBER)	113019106018
KOUSHIK.P.H(TEAM MEMBER)	113019106034
GANESA MOORTHI .M (TEAM MEMBER)	113019106019

1. **INTRODUCTION**
    a. Project Overview
    b. purpose

2. **LITERATURE SURVEY**
    a. Existing problem
    b. References
    c. Problem Statement Definition

3. **IDEATION & PROPOSED SOLUTION**
    a. Empathy Map Canvas
    b. Ideation & Brainstorming
    c. Proposed Solution
    d. Problem Solution fit

1. **REQUIREMENT ANALYSIS**
    a. Functional requirement
    b. Non-Functional requirements

2. **PROJECT DESIGN**
    a. Data Flow Diagrams& User Stories
    b. Solution & Technical Architecture

3. **PROJECT PLANNING & SCHEDULING**
    a. Sprint Planning& Estimation
    b. Sprint Delivery Schedule

4. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    a. Feature
    b. Database Schema (if Applicable)

5. **TESTING**
    a. Test Cases
    b. User Acceptance Testing

6. **RESULTS**
    a. Performance Metrics

7. **ADVANTAGES & DISADVANTAGES**

8. **CONCLUSION**

SMART FARMING

INTRODUCTION

1. PROJECT OVERVIEW:

Agriculture plays a important role in country's economy and provides a large-scale employment to the people. However, agriculture is highly dependent upon weather and climate. For example, changes in temperature, soil moisture, carbon dioxide may result in low yield of crops. It is Significant to monitor environmental parameters in order to manage crop growth and increase the agricultural production yield. The sensed information is not only important for decision makingbut also for evaluating impacts of agricultural practiceson environment. Nowadays, it is more necessary than ever to increase the crop yields food grain production. Cloud connected, wireless system aid in this crop yield maximization, which automates day-to-day agricultural tasks and real time monitoring for smart decision-making.

2. PURPOSE:

Need for technology to monitor important parameters like soil moisture, temperature, Humidityetc. to improve the cultivation process. Need for technology to monitor weather of particular area with reliable sourceto save the crops at the time of natural calamities like flood, cycloneetc. Development of certain techniques to reduce the workforce, energy and time for cultivation. Development of a feasible method to control the electrical equipment in the farm from any part of the world.

3. LITERATURE SURVEY:
   a. EXISTING PROBLEM:

The biggest challenges faced by IoT in the agriculturalsector are lack of information, high adoption costs , and security concers , etc. Most of the farmers are not aware of the implementationof IoT in agriculture.

   b. REFERENCES:

Itis the application of modern ICT (Information and Communication Technologies) into agriculture. In IOT- based

smart farming, asystem 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.

c. PROBLEM STATEMENT DEFINITION:

Overuse of pesticides and fertilizer in agricultural fields leads to destruction ofthe 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.

4. IDEATION & PROPOSED SOLUTION:

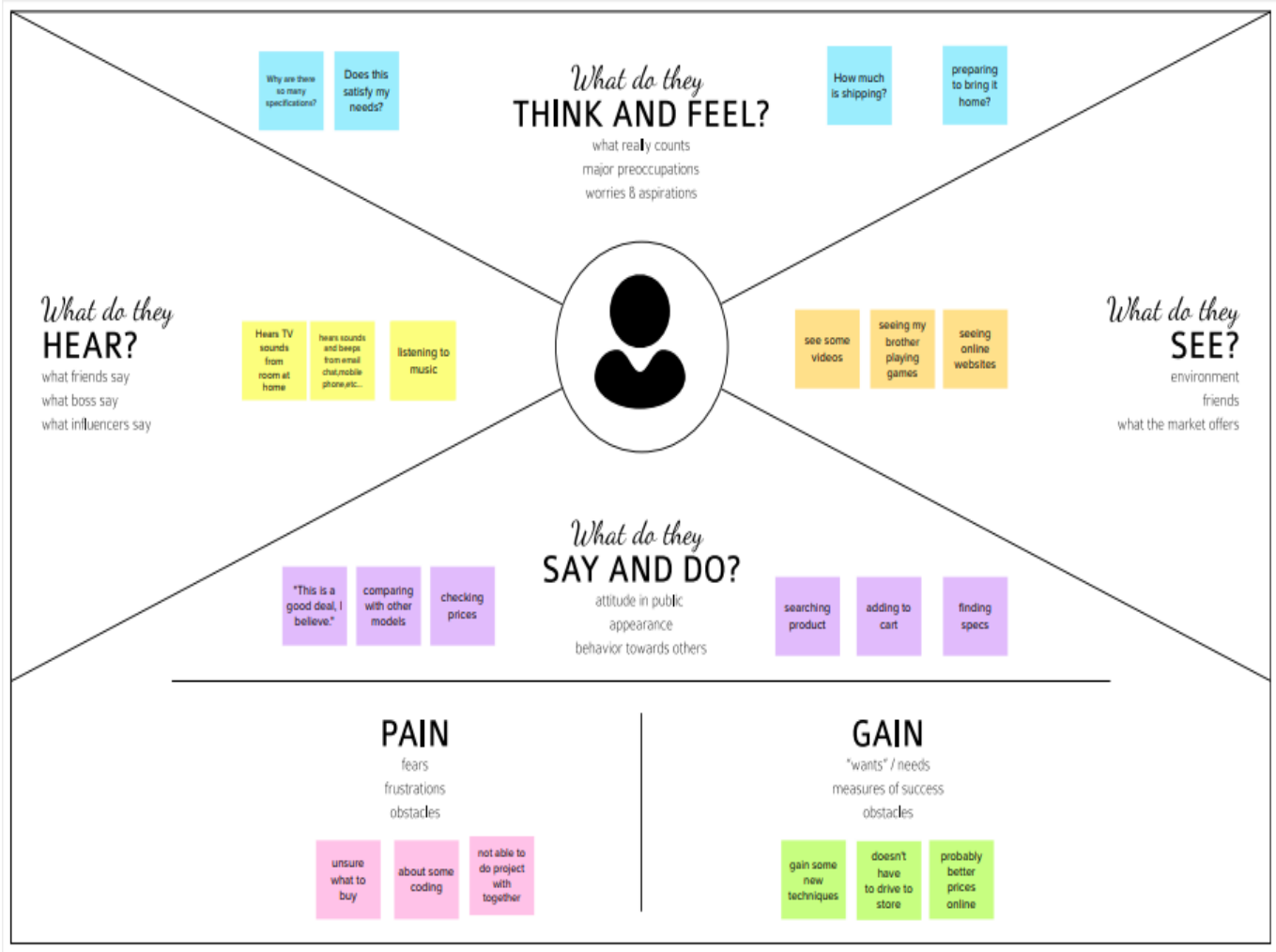
a. EMPATHY MAP CANVAS:

Empathy Map Canvas

Gain insight and understanding on solving customer problems.

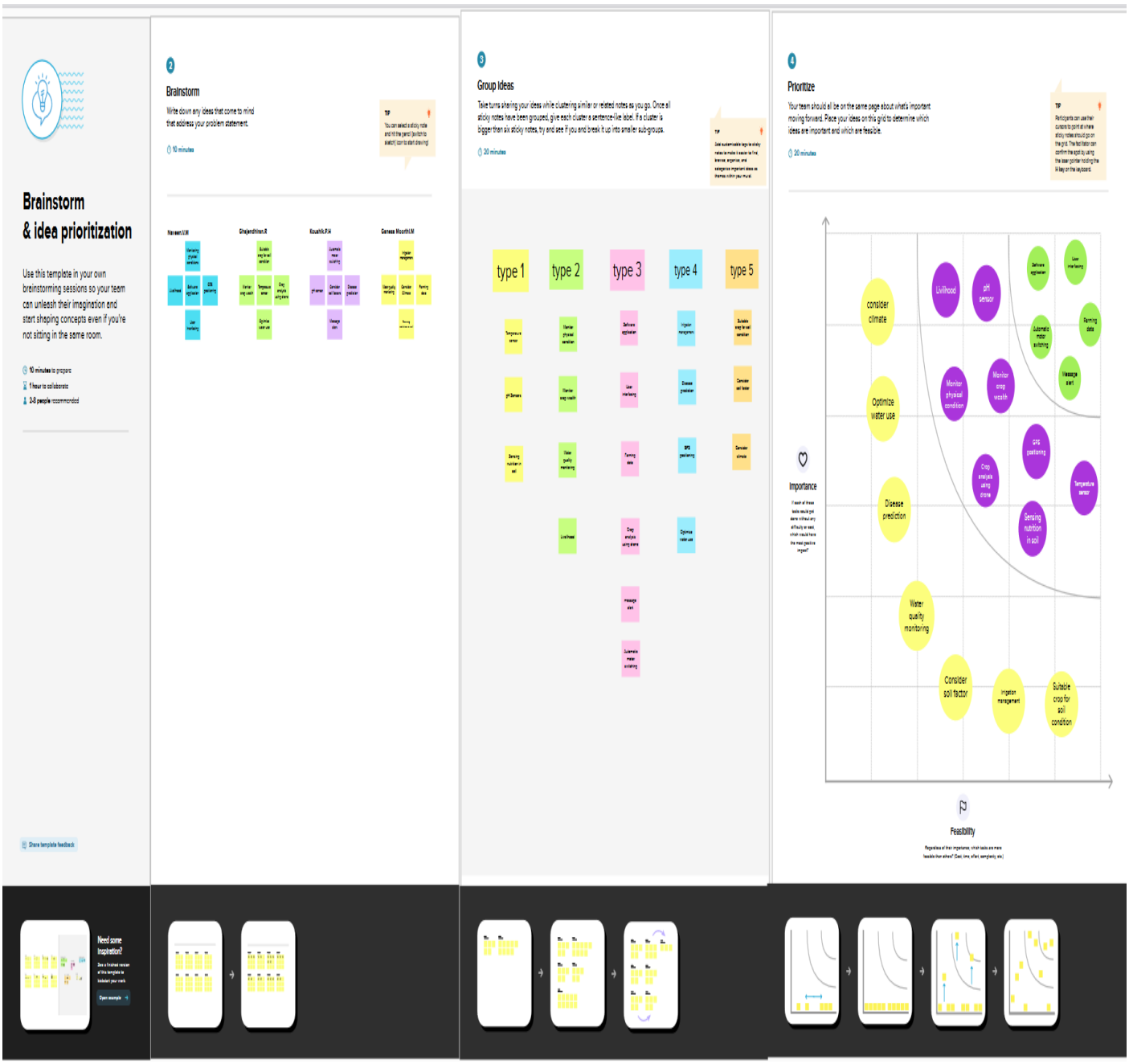
1

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



Share your feedback

b. IDEATION & BRAINSTORMING:



c. Proposed Solution Template:

Project team shall fill the following information in proposed solutiontemplate.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To make farming easier by choosing severalconstraints in agriculture and to overcomethose constraints, to increase production quality and quantity using IOT.
2.	Idea / Solution description	Using smarttechniques like monitoring farms climate, smart irrigation and soil analysis.
3.	Novelty / Uniqueness	Solar power smart irrigation systemwhich helps you tomonitor temperature, moisture ,humidity using smartsensors.
4.	Social Impact / CustomerSatisfaction	It is better than the present modernirrigation system by usingthis method we can controlsoil erosion. There will be betterproduction yield.
5.	Business Model (RevenueModel)	As the productivity increases customer satisfaction also increasesand hence need for the application also increases, whichincreases the revenue of the business.
6.	Scalability of the Solution	It is definetly scalable we ca increase the constraints when the problemarises.

d. PROBLEM SOLUTIONS FIT :

1.Customer Segments : Here farmers are the customers. There are different categories of farmers. Among them large scale farmers prefer smart farming.	5.Available Solutions : Smart farming can be achieved by IoT which includes user interfacing, sensor and software application.	8.Channels of Behaviour : The channels of behavior include user interfacing with the help of IoT, precision farming, Cloud services and so on.
2.Jobs to be done : The rapid changes in climate, soil erosion improper usage of pesticides should be solved by Internet of Things via Smart Farming.	6.Customer Constrains : The customer constraints include climate changes, farm policies, lack of knowledge about the current technologies to handle farm data, access to markets and need for off-farm income.	9.Problem Route Cause : Mostly Indian farming are dependent on rains, soil, dampness and environment challenges which forces them to take a modern decisions on farming.
3.Triggers : Some of the triggers in smart farming are advertising in television and create awareness about smart farming. 4.Emotions : The customers feel happy and comfortable since the project yields high crops with less investments.	7.Behaviour : Smart farming sustainably increases the agricultural productivity and incomes. It offers high-precision crop control and automated farming techniques.	10.Solution : The solution for our project is the initiate Smart Farming using IoT which includes the involvement of sensors, data gathering techniques, and mainly the software application for monitoring and watering the crops.

5. **REQUIREMENT ANALYSIS:**

a. FUNCTIONAL ANALYSIS:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	UserRegistration	Registration through Form Registration through Gmail Registration throughLinkedIn
FR-2	UserConfirmation	Confirmation via Email Confirmation via OTP
FR-3	System login	Check authorization Check access
FR-4	Manage schedule	Manage system admins Manageuser consent Manageuser
FR-5	Check details	Moist details Temperature details
FR-6	Log out	Exit

b. NON FUNCTIONAL REQUIREMENTS:

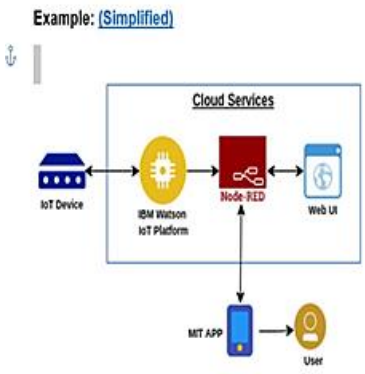
Following arethe non-functional requirements of the proposedsolution.

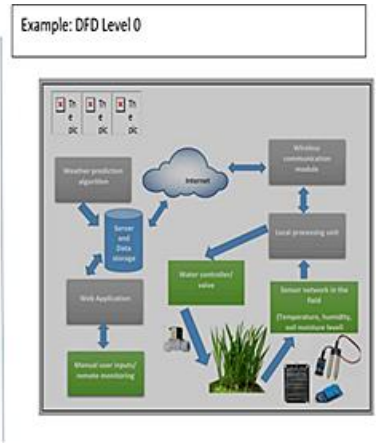
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The quality attributes of a system are specified by its usability. This requirement can be the rate at which a systemmust perform in order to meet userexpectations.
NFR-2	**Security**	Individual and sensitive data must be safeguarded by their provision until the decision-making storing stages.
NFR-3	**Reliability**	The promise dataset is used in thiscase. To avoidfarm neglect, the model employs diligence and shared protection. service
NFR-4	**Performance**	Low power consumption and data transmission rates are required. This implementation concept integrated sensors with soiland environmental parameters.
NFR-5	**Availability**	Cost, sensitivity, design complexity, storage capacity, development process, response criteria, and environmental impact and farming equipment made possible by crops, weather, humidity, and so on are among the quality characteristics takeninto account.
NFR-6	**Scalability**	It is the primary concern for the IoT platform. It has various IOT platform options that affect the system and in real time. Accountability in the workplace

6. PROJECT DESIGN:

a. DATA FLOW DAIGRAMS AND USER STORIES:

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





b. SOLUTIONS AND TECHNICALARCHITECTURAL:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

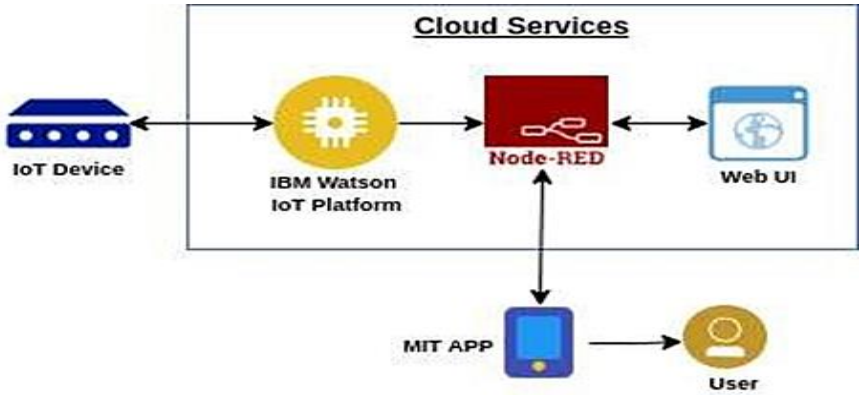


Table-1 : Components& Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, MobileApp, Chatbot etc.	MIT app
2.	ApplicationLogic-1	Logic for a process in the application	Node red/IBM Watson/MIT app
3.	ApplicationLogic-2	Logic for a process in the application	Node red/IBM Watson/MIT app
4.	ApplicationLogic-3	Logic for a process in the application	Node red/IBM Watson/MIT app
5.	Database	Data Type, Configurationsetc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM cloud.
7.	Temperature sensor	Monitors the temperature of the crop	
8.	Humidity sensor	Monitors the humidity	
9.	Soil moisture sensor(Tensiometers)	Monitors the soiltemperature	
10.	Weather sensor	Monitors the weather	.
11.	Solar panel		.
12.	RTC module	Date and time configuration	
13.	Relay	To get the soil moisturedata	

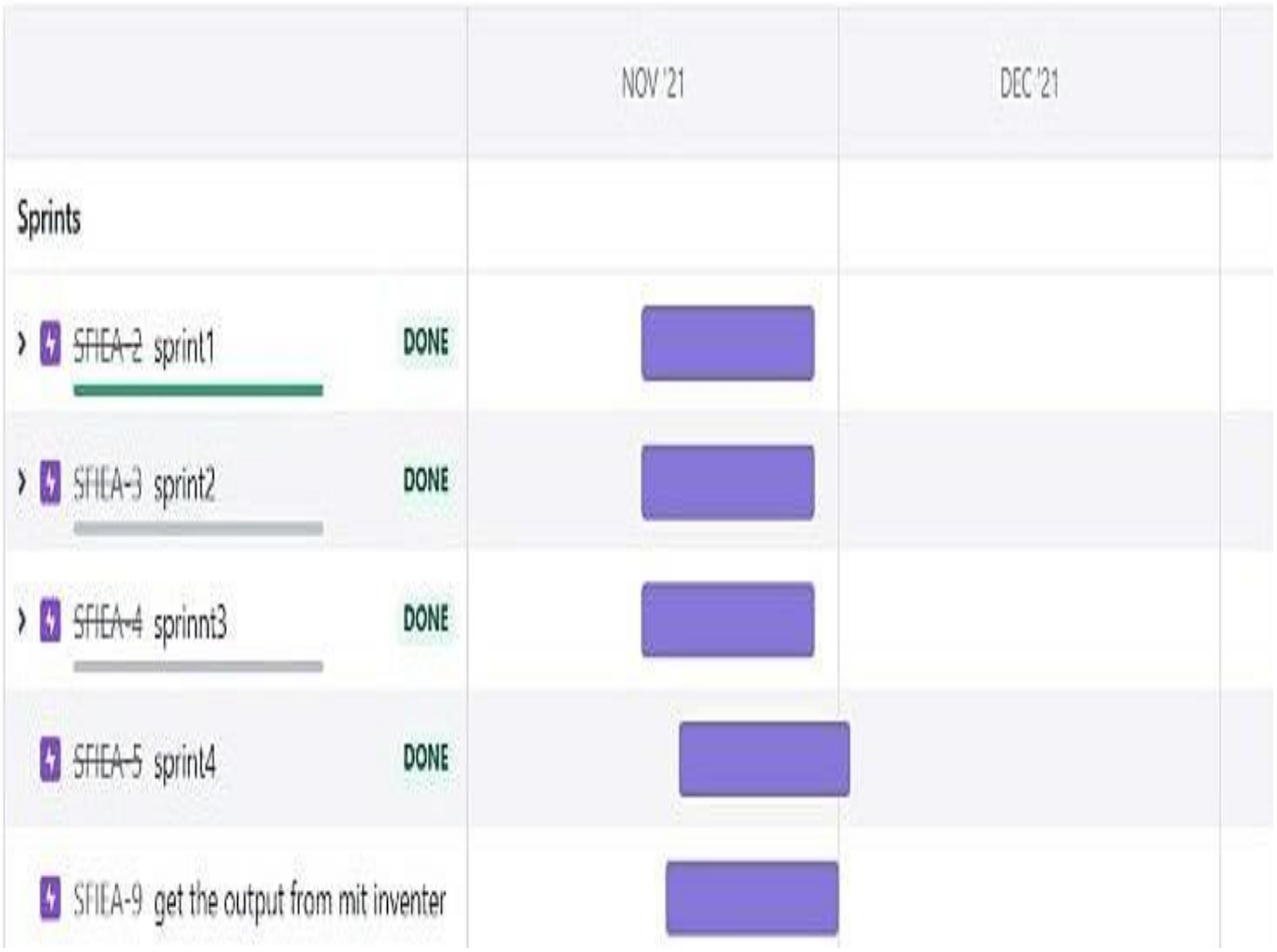
Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	MIT app,Node-Red	Software
2.	Scalable Architecture	Drone technology, pesticide monitoring ,Mineral identification insoil	Hardware

PROJECT PLANNING AND SCHEDULING:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Simulation creation	USN-1	Connect Sensors and Arduino with python code	2	High	Naveen.v.m Ghaja.R
Sprint-2	Software	USN-2	Creating devicein the IBM Watson IoT platform, workflow for IoT scenarios using Node-Red	2	High	Ganesa moorthi .M Koushik.P.H
Sprint-3	MIT App Inventor	USN-3	Develop an application for the Smart farmerproject using MIT App Inventor	2	High	Naveen.V.M Koushik.P.H
Sprint-3	Dashboard	USN-3	Design the Modulesand test the app	2	High	Naveen.V.M
Sprint-4	Web UI	USN-4	Tomake the userto interact with software.	2	High	Koushik.P.H Ghaja.R

REPORT FROM JIRA:



7. CODING & SOLUTIONS:

FEATURE :

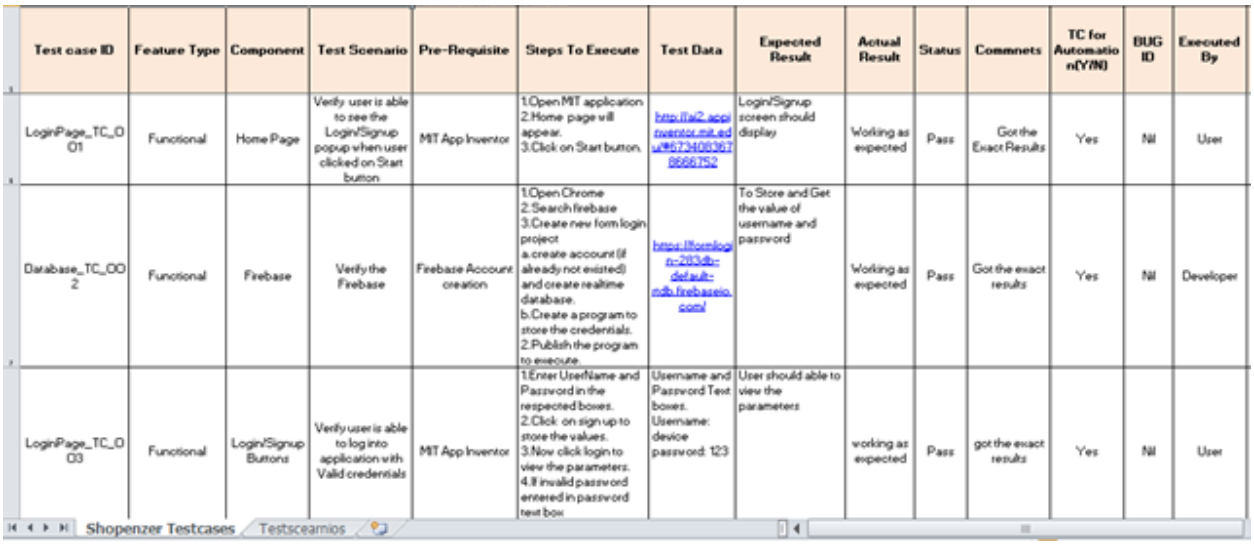
RED NODE.py - F:/RED NODE.py (3.11.0)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "382h93" #replace the ORG ID
deviceType = "OUT"#replace the Device type wi
deviceId = "1234"#replace Device ID
authMethod = "token"
authToken = "q?Sr)VFQ&QUm_q79wi" #Replace the authtoken
# Initialize GPIO

#Receives Command from Node-red
def myCommandCallback(cmd):
    print ("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("motor is on")
    elif status == "motoroff" :
        print ("motor is off")
    elif status == "motor30" :
        print ("motor is on for 30 minutes")

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:
    #Get Sensor Data from DHT11

    temp=random.randint(0,100)
    Humid=random.randint(0,100)
    soilmoisture=random.randint(0,100)

    data = { 'temp' : temp, 'Humid': Humid, 'soilmoisture': soilmoisture }
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "soilmoisture = %s %%"
%soilmoisture, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
            time.sleep(5)

deviceCli.commandCallback = myCommandCallback

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

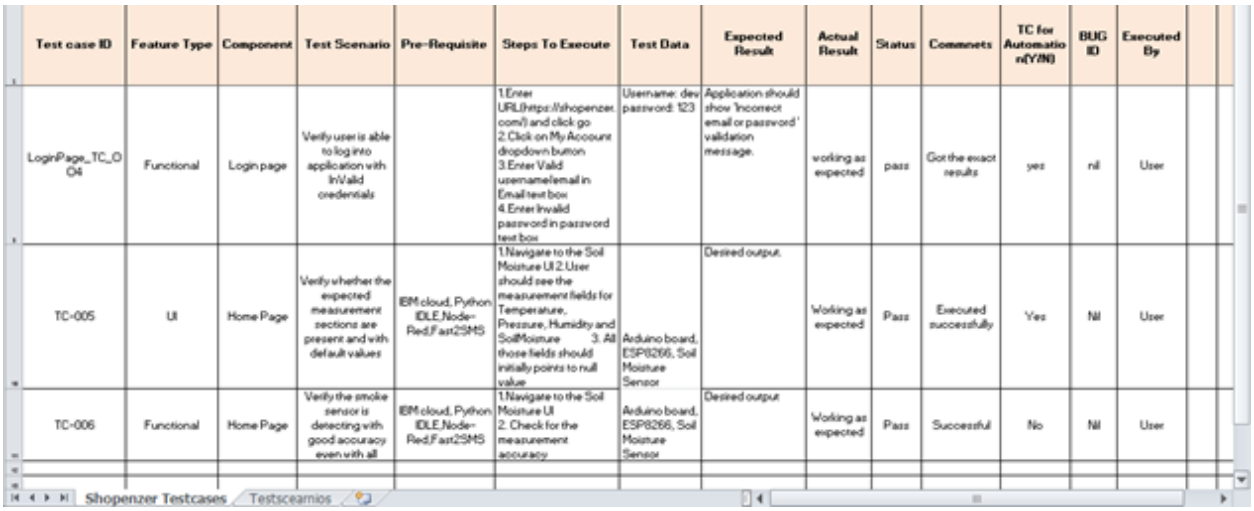
8. TESTING:

a. TEST CASE:



Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on Start button	MIT App Inventor	1.Open MIT application 2.Home page will appear. 3.Click on Start button.	http://ai2.appinventor.mit.edu/#6734083678666752	Login/Signup screen should display	Working as expected	Pass	Got the Exact Results	Yes	Nil	User
Database_TC_OO2	Functional	Firebase	Verify the Firebase	Firebase Account creation	1.Open Chrome 2.Search firebase 3.Create new form login project a.create account (if already not existed) and create realtime database. b.Create a program to store the credentials. 2.Publish the program to execute.	https://loveblogin-283db-default-rtdb.firebaseio.com/	To Store and Get the value of username and password	Working as expected	Pass	Got the exact results	Yes	Nil	Developer
LoginPage_TC_OO3	Functional	Login/Signup Buttons	Verify user is able to log into application with Valid credentials	MIT App Inventor	1.Enter UserName and Password in the respected boxes. 2.Click on sign up to store the values. 3.Now click login to view the parameters. 4.If invalid password entered in password text box	Username and Password Text boxes. Username: device password: 123	User should able to view the parameters	working as expected	Pass	got the exact results	Yes	Nil	User

Shopenzer Testcases | Testsceamios



Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL(https://shopenzer.com/) and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter Invalid password in password text box	Username: dev password: 123	Application should show 'Incorrect email or password' validation message.	working as expected	pass	Got the exact results	yes	nil	User
TC-005	UI	Home Page	Verify whether the expected measurement sections are present and with default values	IBM cloud, Python IDLE,Node-Red,Fast2SMS	1.Navigate to the Soil Moisture UI 2.User should see the measurement fields for Temperature, Pressure, Humidity and SoilMoisture 3. All those fields should initially points to null value	Arduino board, ESP8266, Soil Moisture Sensor	Desired output.	Working as expected	Pass	Executed successfully	Yes	Nil	User
TC-006	Functional	Home Page	Verify the smoke sensor is detecting with good accuracy even with all	IBM cloud, Python IDLE,Node-Red,Fast2SMS	1.Navigate to the Soil Moisture UI 2. Check for the measurement accuracy	Arduino board, ESP8266, Soil Moisture Sensor	Desired output	Working as expected	Pass	Successful	No	Nil	User

Shopenzer Testcases | Testsceamios

User Acceptance Testing

1. Purpose of Document

The purpose of this documentis to briefly explain the test coverage and open issues of the Smart Farmer IoT enabled smart farming application

project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolvedor closed bugs at eachseverity level, and how they were resolved.

Resolution	Severity1	Severity 2	Severity3	Severity 4	Subtotal
Improper network connectivity	10	6	4	2	22
Humidity alone is detected.	12	10	6	4	32
ContinuousBattery Consumption	20	9	5	2	36
Detection Coverage Area	14	6	2	2	24
Altering the Calibration Curve	20	9	7	6	42
Maintenance	11	3	2	1	17
Accuracy detection of parameters	17	9	6	3	35

Totals	104	52	32	20	208

3. Test Case Analysis

Thisreport shows the number of test cases that have passed, failed,and untested

Section	Total Cases	NotTested	Fail	Pass
Improper network connectivity	6	2	1	1
Humidity aloneis detected.	15	0	0	15
Continuous Battery Consumption	12	0	0	12
Detection Coverage Area	5	0	1	4

Altering the Calibration Curve	4	0	0	4
Maintenance	5	0	0	5
Accuracy detection of parameters	1	0	0	1

9. RESULT:

9.1 Performance Metrics

S.No	Project Name	Scope/feature
1	IOT-ENABLED-SMART-FARMING-APPLICATION	Agriculture
1	IOT-ENABLED-SMART-FARMING-APPLICATION	smart grid
1	IOT-ENABLED-SMART-FARMING-APPLICATION	water supply
1	IOT-ENABLED-SMART-FARMING-APPLICATION	Temperature
1	IOT-ENABLED-SMART-FARMING-APPLICATION	Humidity

NIT-Detailed test plan

S.No	project overview	NFT test approach
1	IOT weather reporting system	Tests the weather condition
2	Weather monitoring using temperature sensor	Temperature is monitored
3	Weather monitoring using humidity sensor	Humidity is monitored

S.No	project overview	NFT test approach
1	IOT weather reporting system	Tests the weather condition
2	Weather monitoring using temperature sensor	Temperature is monitored
3	Weather monitoring using humidity sensor	Humidity is monitored

S.No	project overview	NFT test approach	Assumption/dependencies/ Risks	NFR.Met
1	IOT weather reporting system	Tests the weather condition	Weather	Privacy interoperability
2	Weather monitoring using temperature sensor	Temperature is monitored	Temperature	Performance

3	Weather monitoring usinghumidity sensor	Humidity is monitored	Humidity	Maintainability

Test outcome	Recommendation	Result
Access weather conditions	Designing techniques	It provides information on weather monitoring
specify the temperature	Developind devices	Temperature is monitored and logged automatically
Trace the humidity level	Developing sensor	Provides the actualhumidity condition

10. ADVANTAGES AND DISADVANTAGES:

a. ADVANTAGES:

- All the data like climatic conditions and changes in them, soil orcrop conditions everything can be easily monitored.
- Risk of crop damage can be loweredto a greater extent.

- Many difficult challenges can be avoided making the processautomated and the quality of crops can be maintained.
- The process included in farming can be controlled using the webapplications from anywhere, anytime.

b. **DISADVANTAGES:**

- Smart Agriculture requires internet connectivity continuously, butrural parts cannot fulfil this requirement.
- Any faults in the sensors can cause great loss in the agriculture,due to wrong records and the actions of automated processes.
- IOT devices need much money to implement.

11. CONCLUSION:

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

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 resources like electricity and water IOT can be implementedin most of the places.

13. APPENDIX:

SOURCE CODE:

```
import
wiotp.sdk.d
eviceimport
time
import sys
import
ibmiotf.applic
ationimport
ibmiotf.devic
e import
random

#Provide your IBM
Watson
Deviceorganization =
"382h93" deviceType =
"sensor"
deviceId = "123"
authMethod = "token"
authToken="llwprk4R
```

```
Wz0I2Sqice"# Initialize
GPIO
def  myCommandCallback(cmd):
    print("Commandreceived: %s" %
    cmd.data['command'])
    status=cmd.data['command']
    if
        status=
        ="moto
        ron":
        print
        ("motor
        is on")
    elif status == "motoroff":




        print("
    motor is
    off")else :
        print ("please send proper command")
```

```
try:
        deviceOptions = {"org": organization, "type":
                deviceType, "id":
deviceId, "auth-method": authMethod, "auth-token":
    authToken}deviceCli =
    ibmiotf.device.Client(deviceOptions)
#.....
except Exceptionas e:
  print("Caught exception connecting device: %s"
  %str(e))sys.exit()
# Connect and send a datapoint "hello" with value "world" into
the cloudas aneventoftype "greeting" 10 times
device
Cli.con
nect()
while
True:
#Get Sensor Data
  fromDHT11
  temp=random.randin
  t(90,110)
  Humid=random.rand
  int(60,100)
  Mois=random.
  randint(20,120)
```

```
   data = { 'temp' : temp, 'Humid':Humid
,'Mois': Mois}#printdata
def myOnPublishCallback():
  print ("Published Temperature = %s C" %temp, "Humidity = %s
  %%"
%Humid, "Moisture =%s deg c" % Mois, "to IBM Watson")


  success =
deviceCli.publishEvent("IoTSensor", "json",
data,qos=0,on_publish=myOnPublishCallback)
if not success:
  print("Not
connected to IoTF")
time.sleep(10)
deviceCli.commandCallback =
myCommandCallback #Disconnect the device
and application from the cloud
deviceCli.disconnect()
```

Github link :https://github.com/IBM-EPBL/IBM-Project-2597-1658476499

Project Demo link :
https://drive.google.com/file/d/1k90k4N47t3rE1km655AE3CYqzxDJfph2/view?usp=sharing

THANK YOU…..