SMART FARMING APPLICATION

TEAM ID : PNT2022TMID51347

Project Report

1.Introduction:

1.1 Project overview:

Internet of Things Smart technology enables new digital agriculture. Today technology has become a necessity to meet current challenges and several sectors are using the latest technologies to automate their tasks. Advanced agriculture, based on Internet of Thingstechnologies, is envisioned to enable producers and farmers to reduce waste and improve productivity by optimizing the usage of fertilizers to boost the efficiency of plants. It gives better control to the farmers for their livestock, growing crops, cutting costs,and resources.

1.2 Purpose:

We have tried to focus on different scientific applications which could be put together in the agricultural field for better accuracy with better productivity using less manpower. Moreover, we include a method for monitoring the agricultural fields from any remote location and assessing the basic condition of the field. This is the project from the motivation of the farmers working in the farmlands who are solely dependent on the rains and bore wells for irrigation of their land. In recent times, the farmers have been using irrigation techniques through manual control in which the farmers irrigate the land at regular intervals by turning the water-pumpON/OFF

2.literature survey:

2.1 Existing Problem:

Agriculture is the foundation of our Nation. In the past, agriculturists used to figure out the ripeness of soil and influenced presumptions to develop which kind of product. They didn't think about the dampness, level of water and especially climate conditions which are more horrifying to an agriculturist. They utilize pesticides in view of a few suspicions which lead a genuine impact to the yield if the supposition isn't right. Profitability relies upon the last phase of the harvest on which agriculturists depend

2.2 References:

[1] S. S. Gill, I. Chana, C. Science, and R. Buyya, “IoT Based Agriculture as aCloud and Big Data Service: The Beginning of Digital India,” vol. 29, no. 4, pp. 1–23, 2017.
[2] R. K. Kodali and A. Sahu, “An IoT based Weather Information Prototype Using WeMos,” no. 1, pp. 612–616, 2016.
[3] I. Mat, M. Rawidean, M. Kassim, A. N. Harun, I. M. Yusoff, and M. Berhad, “Smart Agriculture Using Internet of Things,” 2018 IEEE Conf. Open Syst., pp. 54–59, 2018.
[4] S. R. Prathibha, A. Hongal, and M. P. Jyothi, “IOT BASED MONITORING SYSTEM IN SMART AGRICULTURE,” pp. 5–8, 2017.
[5] Y. Bo and H. Wang, “The Application of Cloud Computing and The Internet of Things in Agriculture and Forestry,” pp. 168–172, 2011.
[6] Tien Cao Huang and Can Nguyen Duy, “Environment Monitoring for Agricultural Application Based on Wireless Sensor Network.”, April 16-17.
[7] M. K. Gayatri, “Providing Smart Agricultural Solutions to Farmers for better yielding using IoT,” no. Tiar, pp. 40–43, 2015.
[8] S. V. Mukherji, R. Sinha, S. Basak, and S. P. Kar, “Smart Agriculture using Internet of Things and MQTT Protocol,” 2019 Int. Co

2.3 Problem Statement Definition:

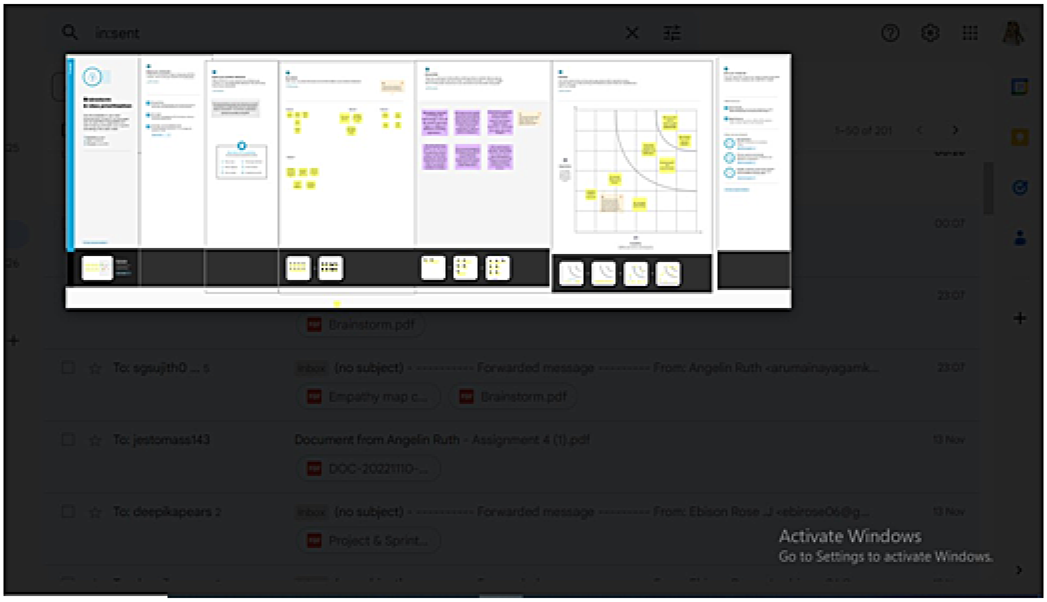
The researcher is supposed to implement an "IoT Based Smart Farming System" with various sensors, which will help to collect the data and analyse it. The proposed system collects information about different agricultural parameters (temperature, humidity, moisture) using an IoT sensor. These values collected are then sent over the mobile via SMS. Farmers can view all the parameters required for a smart farming system through thewebpage.

3. IDEATION AND PROPOSED SOLUTION:

3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

3.2 Ideation & Brainstorming:

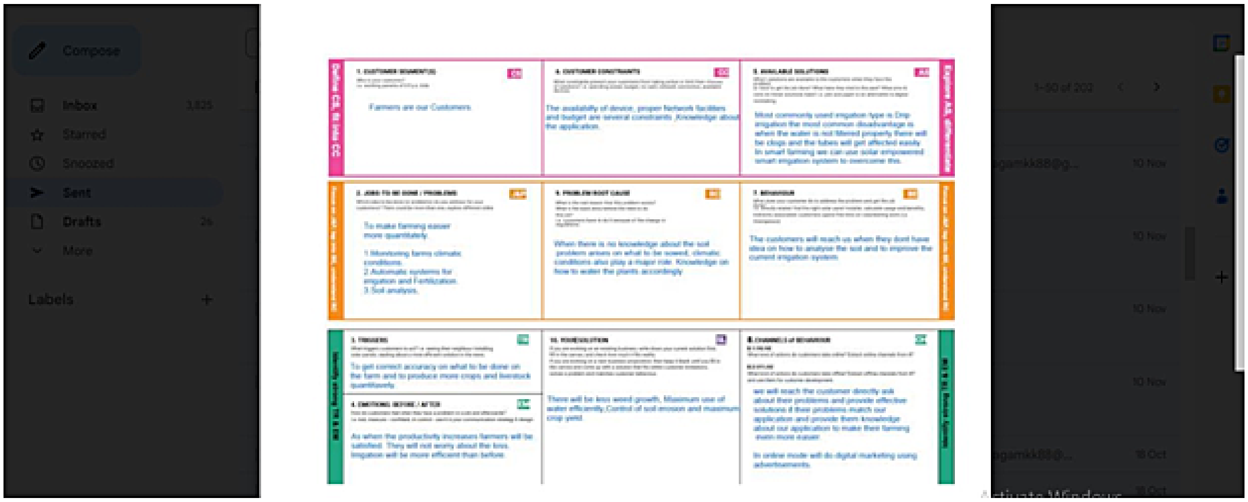


3.3 Proposed Solution:

To improve the efficiency of the product
thereby supporting both rancher and country we need to utilize the innovation which appraises the nature of harvest and gives recommendations. The Internet of things (IOT) is revamping agribusiness, engaging farmers by a broad assortment of techniques, for instance, accuracy and conservative cultivation to go up against challenges in the field.

In this project, on a farm, management can
monitor different environmental parameters effectively using sensor devices such as temperature sensor, relative humidity sensor and soil moisture sensor. Periodically (30 seconds) the sensors are collecting information of the agriculture field area and are being logged and stored online using cloud computing and Internet of Things. By using wireless transmission, the sensed data is forwarded towards the web server database. If irrigation is automated, then that means if the moisture and temperature fields fall below the potential range. The user can monitor and control the system remotely with the help of an application which provides a web interface to the user.

3.4 Problem Solution fit:



4. REQUIREMENT ANALYSIS

4.1 Functional requirement:

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	IoT devices	Sensors and Wifi module.
FR-2	Software	Web UI, Node-red, IBM Watson, MIT app

4.2 Non-Functional requirements:

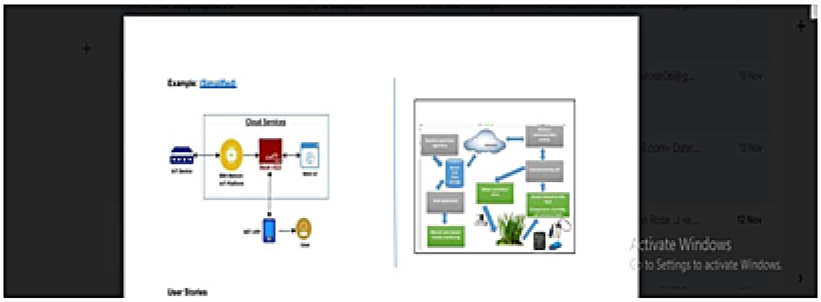
Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Time consumability is less, Productivity is high.
NFR-2	**Security**	It has low level of security features due to integration of sensor data.
NFR-3	**Reliability**	Accuracy of data and hence it is Reliable.
NFR-4	**Performance**	Performance is high and highly productive.
NFR-5	**Availability**	With permitted network connectivity the application is accessible
NFR-6	**Scalability**	It is perfectly scalable many new constraints can be added

5. PROJECT DESIGN

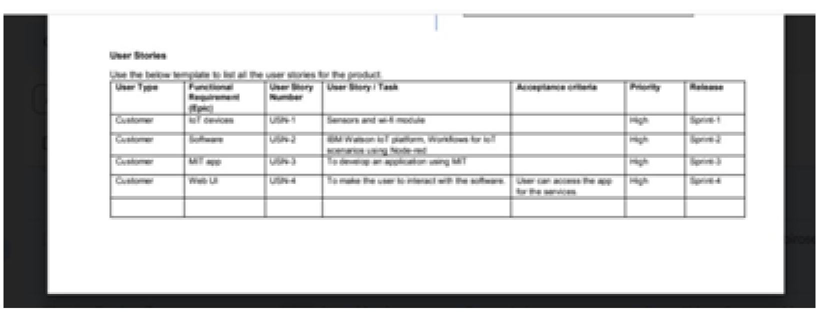
5.1 Data Flow Diagrams:



5.2 Solution & Technical Architecture:

● IoT-based agriculture systems help the farmer in monitoring different parameters of his field like soil moisture, temperature, and humidity using some sensors.
● Farmers can monitor all the sensor parameters by using a web or mobile application even if the farmer is not near his field. Watering the crop is one of the important tasks for farmers.
● They can make the decision whether to water the crop or postpone it by monitoring the sensor parameters and controlling the motor pumps from the mobile application itself.

5.3 User Stories:



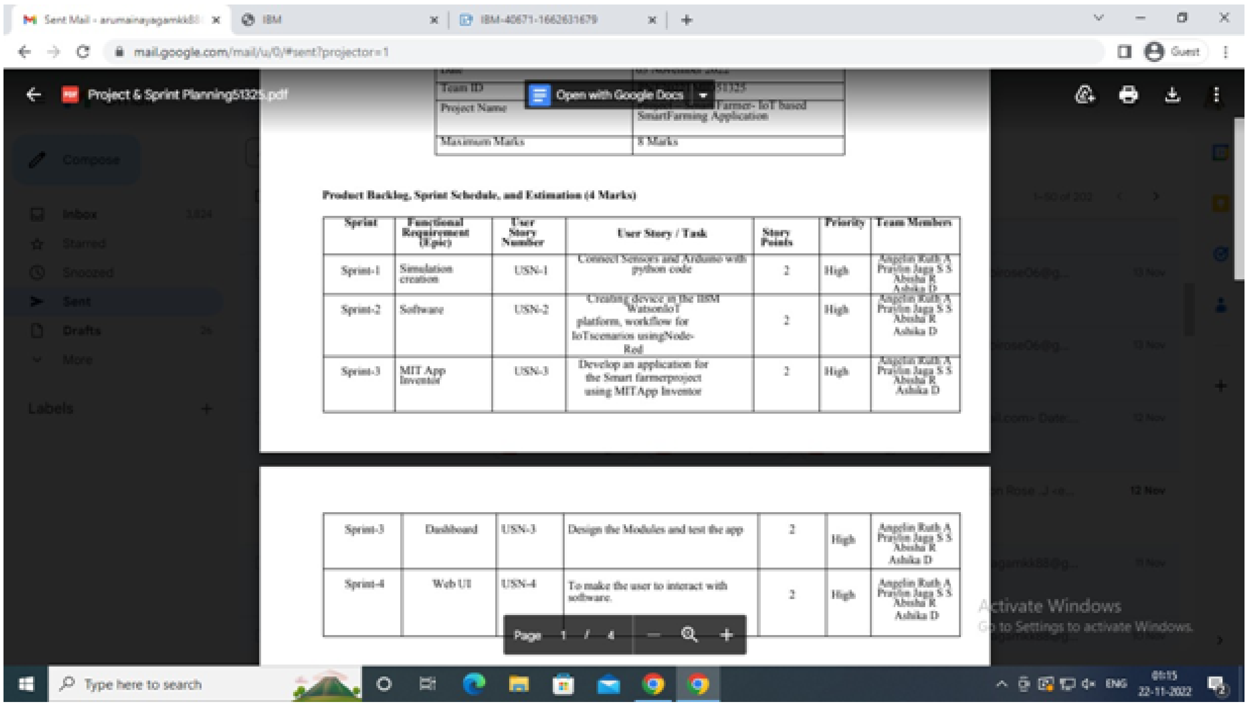
User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	IoT devices	USN-1	Sensors and wi-fi module		High	Sprint-1
Customer	Software	USN-2	IBM Watson IoT platform, Workflows for IoT scenarios using Node-red		High	Sprint-2
Customer	MIT app	USN-3	To develop an application using MIT		High	Sprint-3
Customer	Web UI	USN-4	To make the user to interact with the software.	User can access the app for the services.	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint
Planning & Estimation:



Team ID	51325
Project Name	Smart Farmer- IoT based SmartFarming Application
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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	Angelin Ruth A Praylin Jaga S S Absha R Ashika D
Sprint-2	Software	USN-2	Creating device in the IBM WatsonIoT platform, workflow for IoTscenarios usingNode-Red	2	High	Angelin Ruth A Praylin Jaga S S Absha R Ashika D
Sprint-3	MIT App Inventor	USN-3	Develop an application for the Smart farmerproject using MITApp Inventor	2	High	Angelin Ruth A Praylin Jaga S S Absha R Ashika D
Sprint-3	Dashboard	USN-3	Design the Modules and test the app	2	High	Angelin Ruth A Praylin Jaga S S Absha R Ashika D
Sprint-4	Web UI	USN-4	To make the user to interact with software.	2	High	Angelin Ruth A Praylin Jaga S S Absha R Ashika D

6.2 Sprint Delivery Schedule:

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

PYTHON CODE:

To develop the python code to publish and subscribe to the commands
from the IBM cloud.

PROGRAM:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device
Credentials organization = "olwipo"
deviceType = "abcd" deviceId = "12345"
authMethod = "token" authToken =
"12345678"
# Initialize GPIO
```

```
def myCommandCallback(cmd): print("Command
received:
%s" % cmd.data['command']) m=cmd.data['command']
if (m=="motoron"):
print ("motor is switched on") elif
(m=="motoroff"):
print ("motor is switched 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 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
moist=random.randint(0,100)
temp=random.randint(-20,125)
Humid=random.randint(0,100)
data = { 'moist' : moist , 'temp' : temp ,'Humid': Humid}
#print data def
myOnPublishCallback():
print ("Published moist = %s C" % moist, "temp= %s %%" %
temp, "Humid = %s %%" % Humid, "to IBM Watson")
success = deviceCli.publishEvent("IoTSensor", "json", data,
qos=0, on_publish=myOnPublishCallback)
if not success: print("Not connected
to IoTSensor") time.sleep(10)
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

8. TESTING

8.1 Test Cases:

1. IBM Watson IOT service:
To create an IBM Watson IOT service Steps to create an IBM Watson
IOT service: and create a device using it.
● Click on catalog in IBM cloud account.
● Click on services.
● Enter as an Internet of things platform.
● Enter region and pricing plan.
● Enter service name and click create.
● Click on launch.
● Then IBM Watson OT platform opens.
● Click on sign in.
● Enter IBM Id.
● Enter Password.
● Then you can access IBM Watson IOT platform.
Steps to create a device:
● Click on devices in IBM Watson IOT platform.
● Choose to create a device● Enter the device type as abcd.
● Enter the device ID as 12345.
● Click next.
● Enter device credentials (optional).
● Click next.
● Enter the authentication token as 12345678.
● Click on continue.
● Click on next.
● Click finish Device is created successfully, and we can see
device credential
2. Creation of Node – Red Service:
To create a Node Red service.
Steps to create a Node-Red service:
● Click on catalog in IBM cloud account.
● Click on services.
● Enter the Node red service.
● Node red app opens click on get started.

● Enter app name as default.
● Enter the region as London.
● Choose a pricing plan as lite.
● Click create.
● You will be redirected to a new page.
● Click on deploy your app.
● Choose cloud foundry.
● Enter IBM API key (by clicking new+).
● Choose memory size as default.
● Enter the region as London.
● Click next.
● Click create.
● Status will be updated after creation.
● Click on App URL.
● Click next.
● Choose not recommended.
● Click next.
● You will see the Node red page.
● Go to your node red flow editor.
● In the left panel choose nodes

9. RESULTS:

The yield appearing beneath signifies the temperature, soil moisture and humidity data received from the IoT simulator sensor and open weather API. The web app displays all this data for the past one hour. There are a set of buttons on the web application that can be used to control the motor and light on the farm to turn them ON/OFF remotely.

10. ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

1. Communicating the device at a larger distance through web application. It will play an important role in reducing the manpower and traveling expenses of a farmer.
2. Monitoring parameters like temperature, humidity etc. will play an important role in improving the growth of the plant.
3. Integrating the weather station to the web browser will

provide the details of status of the cloud, wind speed etc. It will allow thefarmer to protect their plants from naturalcalamities.

DISADVANTAGE:

1. Since the real time sensor will be connected to the controller, the controller requires continuous supply of the internet to transfer the data.
2. Non availability of weather prediction for a long period of time.Since the long weather prediction requires additional payment to open weather.

11. CONCLUSION:

IoT based SMART AGRICULTURE SYSTEM for Live Monitoring of Temperature and Soil Moisture and to control motor and light remotely have 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 a 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 systems to current strategies in this way making simple profitable and temperate trimming.

12. FUTURE SCOPE:

The project has vast scope in developing the system and making it more user friendly and the additional features of the system like:

- By installing a webcam in the system, photos of the crops can be captured and the data can be sent to a database.
- Speech based options can be implemented in the

system for the people who are less literate.

● GPS (Global Positioning System) can be integrated to provide specific location of the farmer and more accurate weather reports of agriculture fields and gardens.

● Regional language features can be implemented to make it easy for the farmers who are aware of only their regional language.

13. APPENDIX:

```
Import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device
Credentials organization = " olwipo "
deviceType = "abcd" deviceId = "12345"
authMethod = "token" authToken =
"12345678"
# Initialize GPIO
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")
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 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(90,110)
Humid=random.randint(60,100)
moist=random.randint(10,80)
data = { 'temp' : temp, 'Humid': Humid, 'moist' : moist}
#print data def
myOnPublishCallback():
print ("Published Temperature = %s C" % temp, "Humidity =
%s
%%" % Humid, "moisture = %s %%" % moist, "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()

1. IBM LINK :

https://careereducation.smartinternz.com/Student/guided
_project_workspace/40671

2.PROJECT DEMO LINK

https://drive.google.com/file/d/16zuYgYej4W0SQZX_bHXzXVt5EhGM9nSQ/view?usp=drivesdk