IBM NALAIYATHIRAN

PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

SMARTFARMER – IoT ENABLED SMART FARMING APPLICATION

A PROJECT REPORT

Submitted by

TEAM ID : PNT2022TMID38523

TEAM LEADER	**P. SURYA**	**420119104042**
TEAM MEMBER 1	**L. JARISH**	**420119104013**
TEAM MEMBER 2	**A. RAJA**	**420119104030**
TEAM MEMBER 3	**P. VISHNU RAM**	**420119104045**

AKT MEMORIAL COLLEGE OF ENGINEERING & TECHNOLOGY
KALLAKURICHI-606202

ANNA UNIVERSITY::CHENNAI 600025

NOVEMBER-2022

1. INTRODUCTION

1.1 PROJECT OVERVIEW :

Agriculture is done in every country from ages. Agriculture is the science and art of cultivating plants. Agriculture was the key development in the rise of sedentary human civilization. Agriculture is done manually from ages. As the world is trending into new technologies and implementations it is a necessary goal to trend up with agriculture also. IOT plays a very important role in smart agriculture. IOT sensors are capable of providing information about agriculture fields. we have proposed an IOT and smart agriculture system using automation. This IOT based Agriculture monitoring system makes use of wireless sensor networks that collects data from different sensors deployed at various nodes and sends it through the wireless protocol. This smart agriculture using IOT system is powered by Arduino, it consists of Temperature sensor, Moisture sensor, water level sensor, DC motor. When the IOT based agriculture monitoring system starts it checks the water level, humidity and moisture level. It sends SMS alert on the phone about the levels. Sensors sense the level of water if it goes down, it automatically starts the water pump. If the temperature goes above the level, fan starts. This all is displayed on the LCD display module. This all is also seen in IOT where it shows information of Humidity, Moisture and water level with date and time, based on per minute. Temperature can be set on a particular level, it is based on the type crops cultivated. If we want to close the water forcefully on IOT there is button given from where water pump can be forcefully stopped.

1.2 PURPOSE :

Smart farming is a management concept focused on providing the agricultural industry with the infrastructure to leverage advanced technology – including big data, the cloud and the internet of things (IoT) – for tracking, monitoring, automating and analyzing operations.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM :

Farms are located in remote areas and are far from access to the internet. A farmer needs to have access to crop data reliably at any time from any location, so connection issues would cause an advanced monitoring system to be useless.

2.2 REFERENCES :

[1] A Study On Smart Irrigation Systems For Agriculture Using Iot (Dr. J. JegatheshAmalraj, S. Banumathi, J. JereenaJohn) International Journal Of Scientific & Technology Research Volume 8, Issue 12, December 2019

[2] IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture Laura García ,Lorena Parra , Jose M. Jimenez , Jaime Lloret and Pascal Lorenz , Sesnors 2020

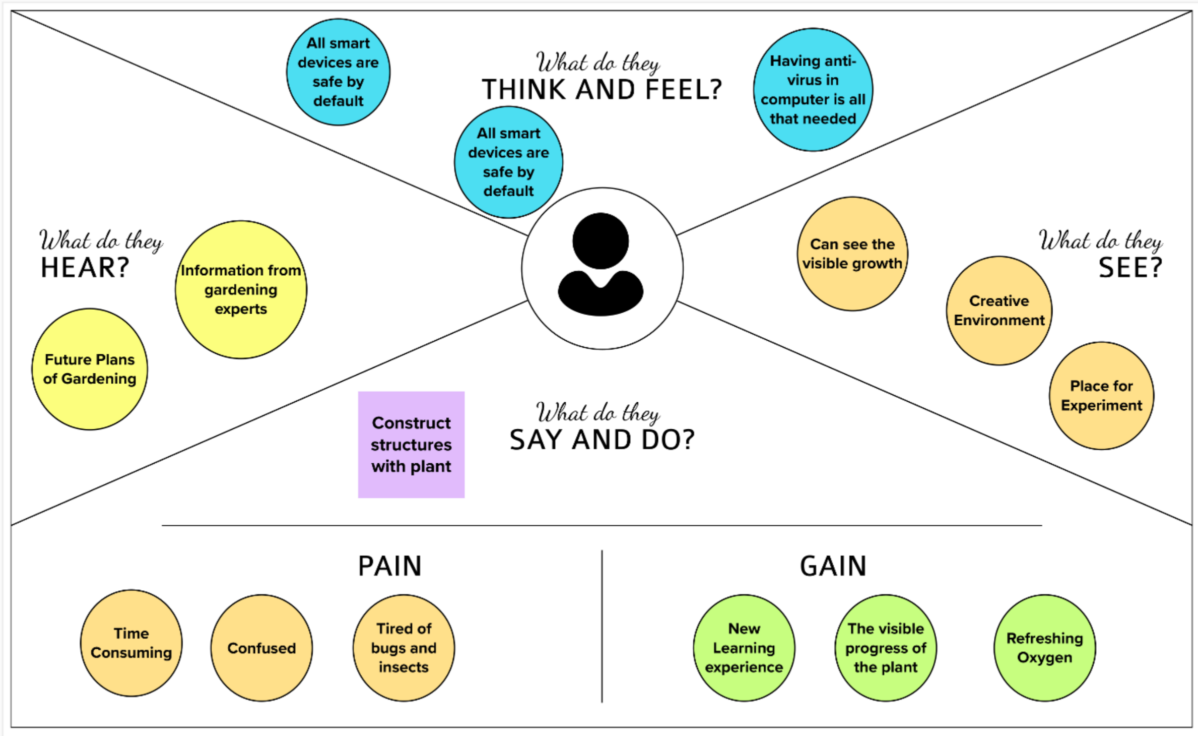
[3] Sungheetha, Akey, and Rajesh Sharma. "Real Time Monitoring and Fire Detection using Internet of Things and Cloud based Drones." Journal of Soft Computing Paradigm (JSCP) 2, no. 03 (2020): 168-174

2.3 PROBLEM STATEMENT DEFINITION :

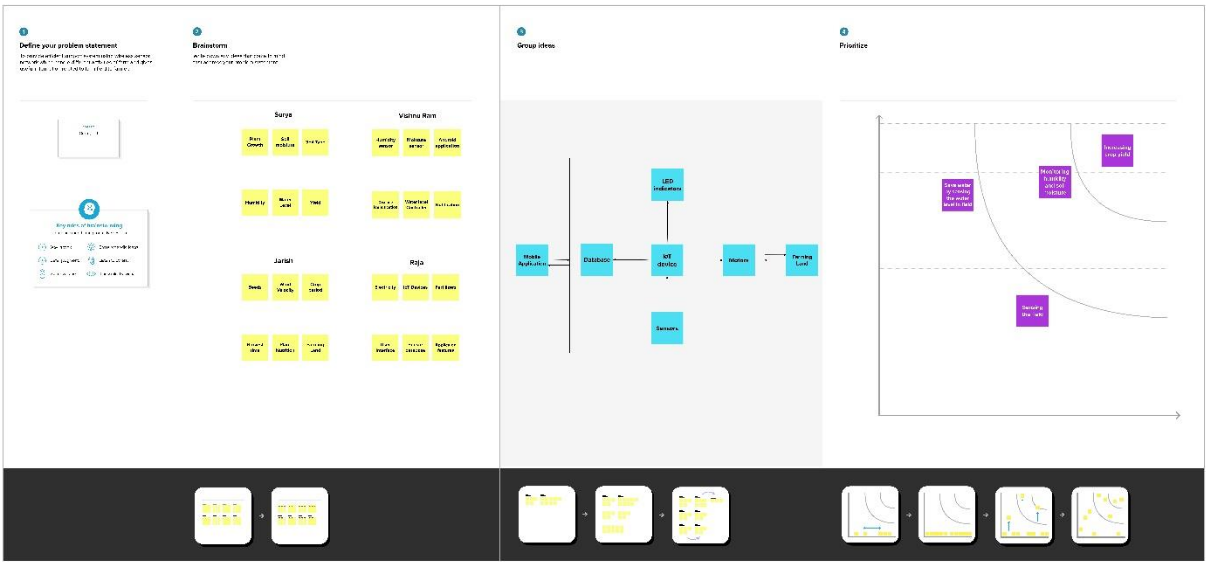
This is the project from the motivation of the farmers working in the farm lands are solely dependent on the rains and bore wells for irrigation of their land. In recent times, the farmers have been using irrigation technique through the manual control in which the farmers irrigate the land at regular intervals by turning the water-pump ON/OFF when required. Moreover, for the power indication they are glowing a single bulb between any one of phase and neutral, meanwhile when there is any phase deduction occurs in other phases, the farmer cannot know their supply is low. If they Switch ON any of the motor, there will be the sudden defuse in motor circuit. They may have to travel so far for SWITCHING ON/OFF the motor. They may be suffering from hot Sun, rain and night time too. After reaching their farm, they found that there is no power, so they quietly disappointed to it.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS :



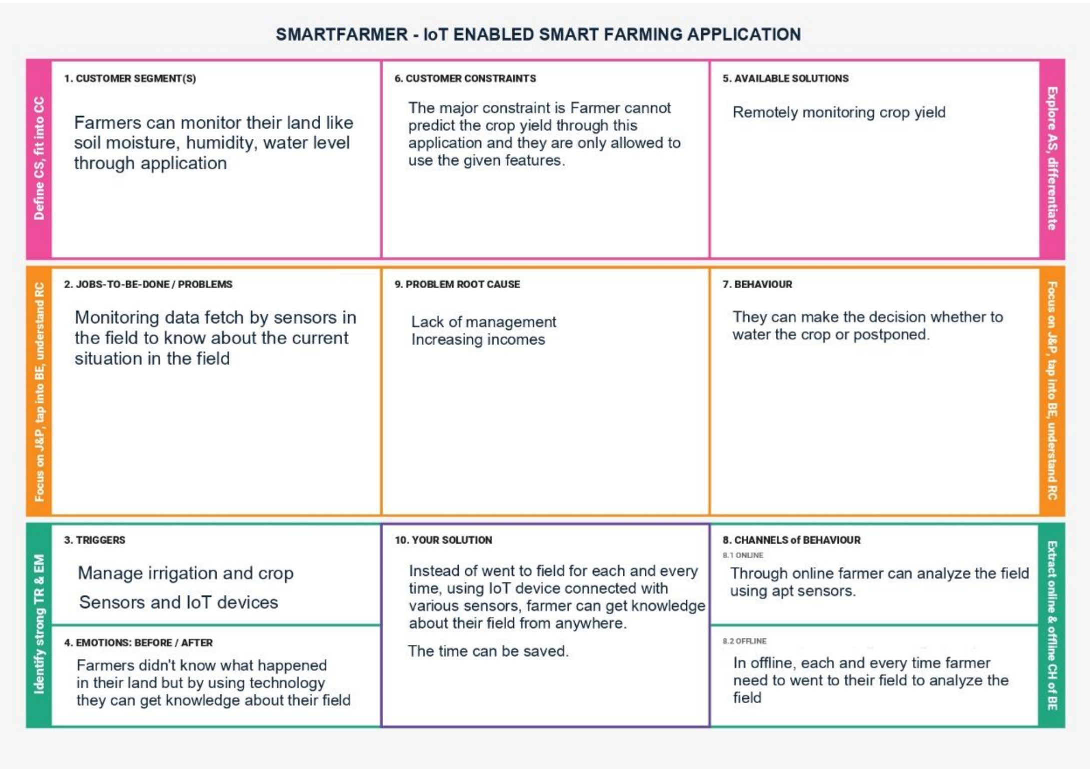
3.2 IDEATION & BRAINSTORMING :



3.3 PROPOSED SOLUTION :

Our proposed system concentrates on monitoring the farming conditions through sensors like Humidity, Temperature, and soil moisture; LDR is used to sense the light intensity for the farm, and also IR sensor is used to detect the pest, birds, and humans by their body temperature and alerts the user through the message format to their mobile These sensors are the interface to process module Arduino-UNO. The LCD is used to display the status of different sensors. When there is a change in temperature condition, the sensor detects and turns ON the DC and cools down the condition. After the temperature comes to a normal state, the DC fan will turn OFF. LDR (Light Dependent Resistor) is used to detect the light intensity in the farm. When the light intensity is less on the farm, the LDR senses the condition and turns ON the bulb. When the required light intensity is back, the bulb will turn OFF. The soil moisture sensor is used to sense the moisture level in soil (water level) when the water levels are reached low in the ground. The ground gets dry, and the sensor detects it, then turn ON the DC water pump. When floor gets moisturized, the DC water pump will turn OFF. The user can monitor these conditions in mobile phone with the help of Wifi module through IOT mobile app.

3.4 PROBLEM SOLUTION FIT :



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS :

FR No.	Functional Requirement(Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Login	Login via Username and Password Login via Google
FR-4	Password reset	Reset password via Email Reset password via Phone Number
FR-5	Password Change	Change password via Email Change password via Phone Number
FR-6	Settings	Change settings for the convenience

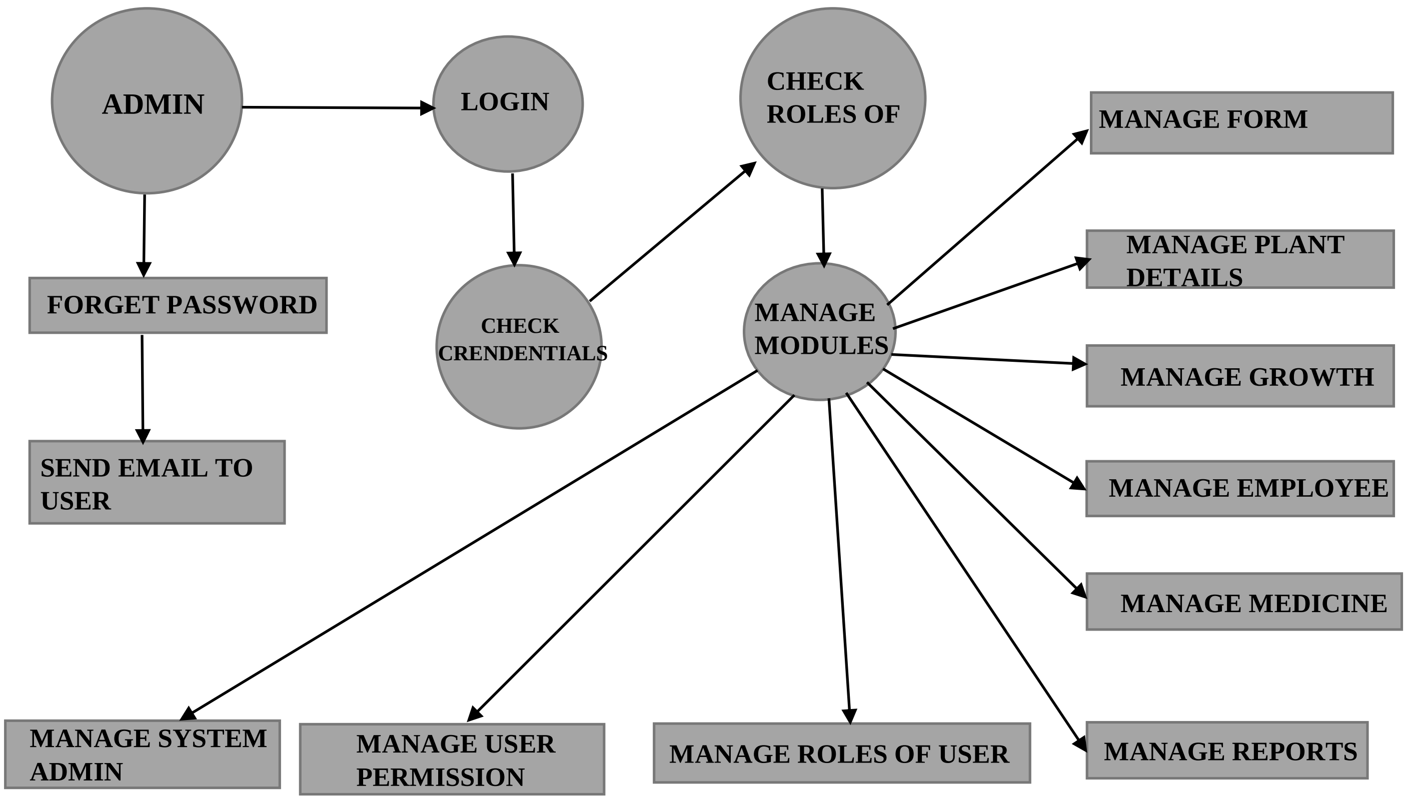
4.2 NON-FUNCTIONAL REQUIREMENTS :

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Application is easy to use with better user experience and the controls given with that application.
NFR-2	Security	The user can register or login through their mailid and password. The security attacks could notbe done until the user share his/her login credentials to someone.
NFR-3	Reliability	The data are stored in the trusted cloud storageand it can be kept confidential. The user and the developer are able to access the data stored in cloud storage.
NFR-4	Performance	The user can control and analyse the data about their field or farm through application given with many features.
NFR-5	Availability	The user can easily access the analysed data from the sensors connected with IoT devices which placed in the farming land and the sensor analysed data are stored in a cloud storage for future references.
NFR-6	Scalability	The application features are upgraded randomly for easy access and better user experience.

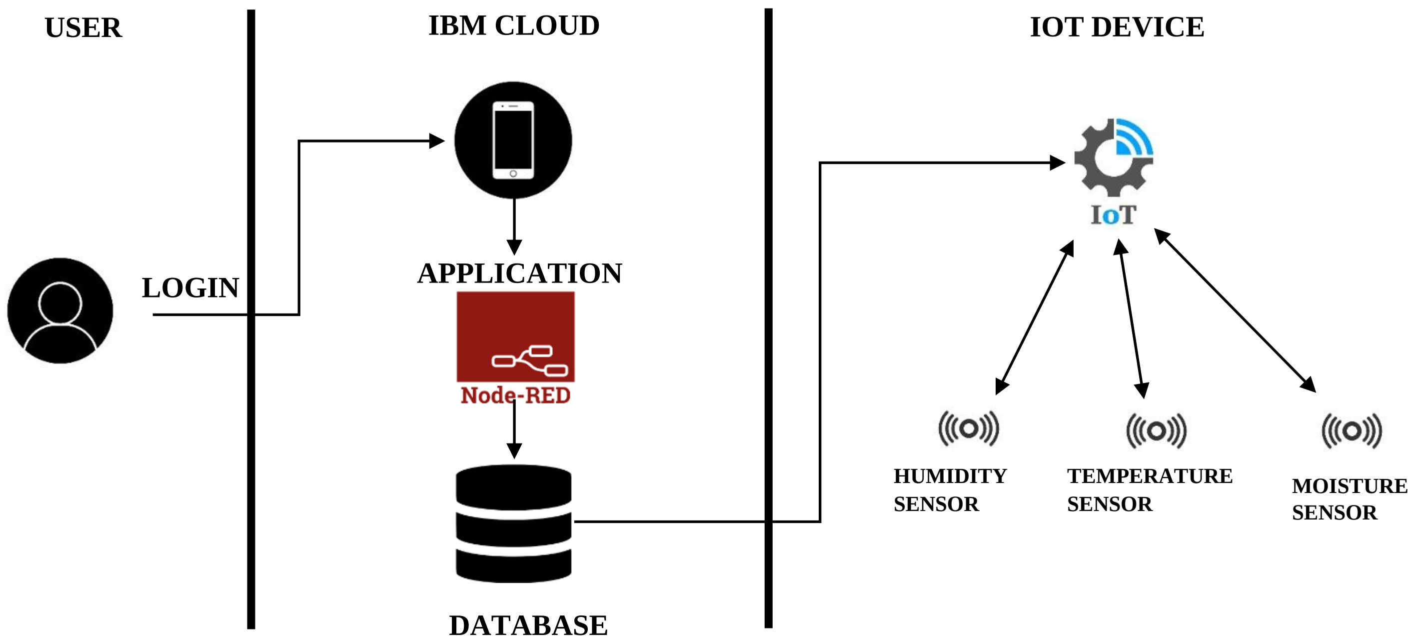
5. PROJECT DESIGN

5.1 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 :



S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database to store the data fetch by sensors connected with IoT device.	IBM DB2, IBM Cloudant etc.
7.	File Storage	The user can store the data in any format	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Because of farming land it will be need to monitoring weather, so the weather API are used.	IBM Weather API.
9.	Machine Learning Model	It is necessary to monitor and identify the disease infection.	Object Recognition Model.
10.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

5.3 USER STORIES :

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register the application through gmail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can log into the application by entering email and password	High	Sprint-3
	Dashboard	USN-6	As a user, I can access the features of the application through dashboard	I can access the features of the application available in dashboard	Medium	Sprint-2
Customer (Web user)	Registration	USN-7	As a user, I can register for the web application by entering my email, password.	I can register for the web application.	Medium	Sprint-3

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION :

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.
Sprint-1	Login	USN-2	As a user, I will receive confirmation email once I have registered for the application
Sprint-2	User Interface	USN-3	As a user, I can register for the application through Facebook
Sprint-1	Data Visualization	USN-4	As a user, I can register for the application through Gmail
Sprint-3	Registration (Web User)	USN-5	As a user, I can log into the application by entering email & password
Sprint-2	Dashboard	USN-6	As a user, I can access the features of the application in dashboard.
Sprint-4	Cloud Registration	USN-7	As a user, I can store the data in cloud storage for future reference.
Sprint-4	Controls	USN-8	As a user, I can control the IoT devices via Mobile and also monitor the field with the help of this IoT deivices.

6.2 SPRINT DELIVERY SCHEDULE :

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	P. Surya
Sprint-1	Login	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	A. Raja
Sprint-2	User Interface	USN-3	As a user, I can register for the application through Facebook	3	Low	L. Jarish
Sprint-1	Data Visualization	USN-4	As a user, I can register for the application through Gmail	2	Medium	P. Vishnu Ram
Sprint-3	Registration (Web User)	USN-5	As a user, I can log into the application by entering email & password	3	High	L. Jarish
Sprint-2	Dashboard	USN-6	As a user, I can access the features of the application in dashboard.	3	Medium	P. Surya
Sprint-4	Cloud Registration	USN-7	As a user, I can store the data in cloud storage for future reference.	2	Medium	A. Raja
Sprint-4	Controls	USN-8	As a user, I can control the IoT devices via Mobile.	3	High	P. Vishnu Ram

7. CODING & SOLUTIONING

7.1 FEATURE 1

```
# MULTIPLE LINEAR REGRESSION
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, Y_train)

Y_pred = regressor.predict(X_test)
np.set_printoptions(precision = 2)
print(np.concatenate((Y_pred.reshape(len(Y_pred), 1),
                      Y_test.reshape(len(Y_pred), 1)), axis = 1))
"""
"""
# BAYESIAN RIDGE REGRESSION
from sklearn.linear_model import BayesianRidge
bay_ridge = BayesianRidge()
bay_ridge.fit(X_train, Y_train)
Y_pred = bay_ridge.predict(X_test)

np.set_printoptions(precision = 2)
print(np.concatenate((Y_pred.reshape(len(Y_pred), 1),
                      Y_test.reshape(len(Y_pred), 1)), axis = 1))
"""
"""
# PLOYNOMIAL REGRESSION
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
```

```
# Go for a polynomial of degree 4 but 20 is good as well
poly_reg = PolynomialFeatures(degree = 4)
X_train_poly = poly_reg.fit_transform(X_train)
poly_model = LinearRegression()
poly_model.fit(X_train_poly, Y_train)

Y_pred = poly_model.predict(poly_reg.fit_transform(X_test))
for i in range(0, len(Y_pred)):
    Y_pred[i] = round(Y_pred[i])

np.set_printoptions(precision = 2)
print(np.concatenate((Y_pred.reshape(len(Y_pred), 1),
                      Y_test.reshape(len(Y_pred), 1)), axis = 1))
 """

 """
 import Fert_Dataset as fd
import os

loc_Fert = os.getcwd() + r'/Datasets/FertPredictDataset.csv'

dataset = fd.get_fert_dataset(loc_Fert)

X = dataset.iloc[:, :3].values
Y = dataset.iloc[:, 3].values
```

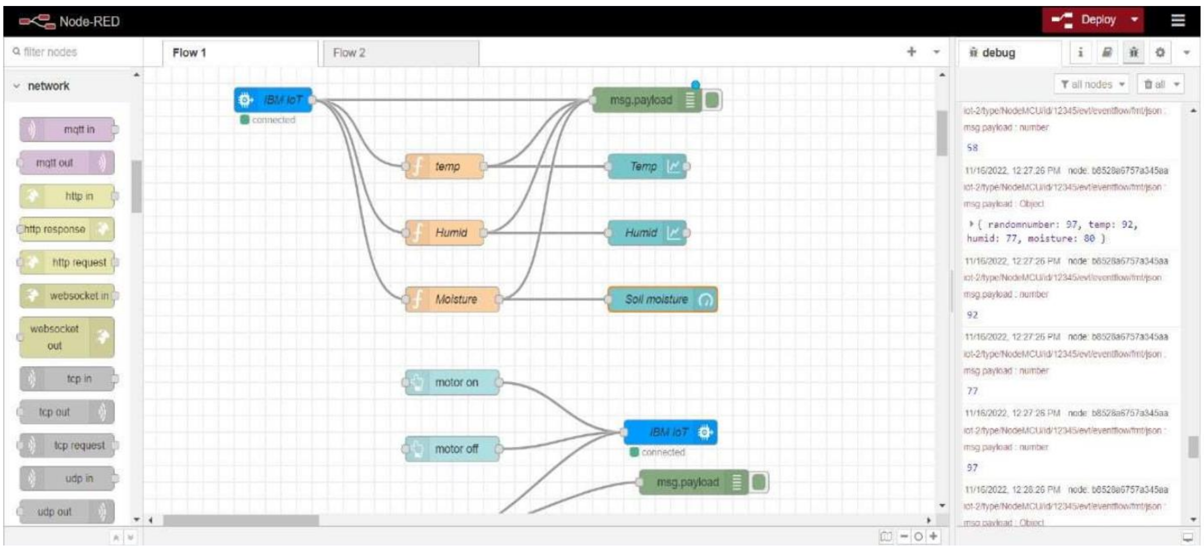
```
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split

X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
                                                    test_size = 0.2,
                                                    random_state = 0)

from sklearn.tree import DecisionTreeClassifier
dtree_model = DecisionTreeClassifier(max_depth = 2).fit(X_train, Y_train)

dtree_pred = dtree_model.predict(X_test)
"""
```

7.2 FEATURE 2

```
import Fert_Dataset as fd
import os


def Predict_Fertiliser(sensor_value):
    loc_Fert = os.getcwd() + r'/Datasets/FertPredictDataset.csv'

    dataset = fd.get_fert_dataset(loc_Fert)

    X = dataset.iloc[:, :3].values
    Y = dataset.iloc[:, 3].values

    from sklearn.model_selection import train_test_split

    X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
                                                        test_size = 0.2,
                                                        random_state = 0)

    from sklearn.tree import DecisionTreeClassifier
    dtree_model = DecisionTreeClassifier(max_depth = 2).fit(X_train,
                                                           Y_train)

    dtree_pred = dtree_model.predict(sensor_value)

    return dtree_pred
```

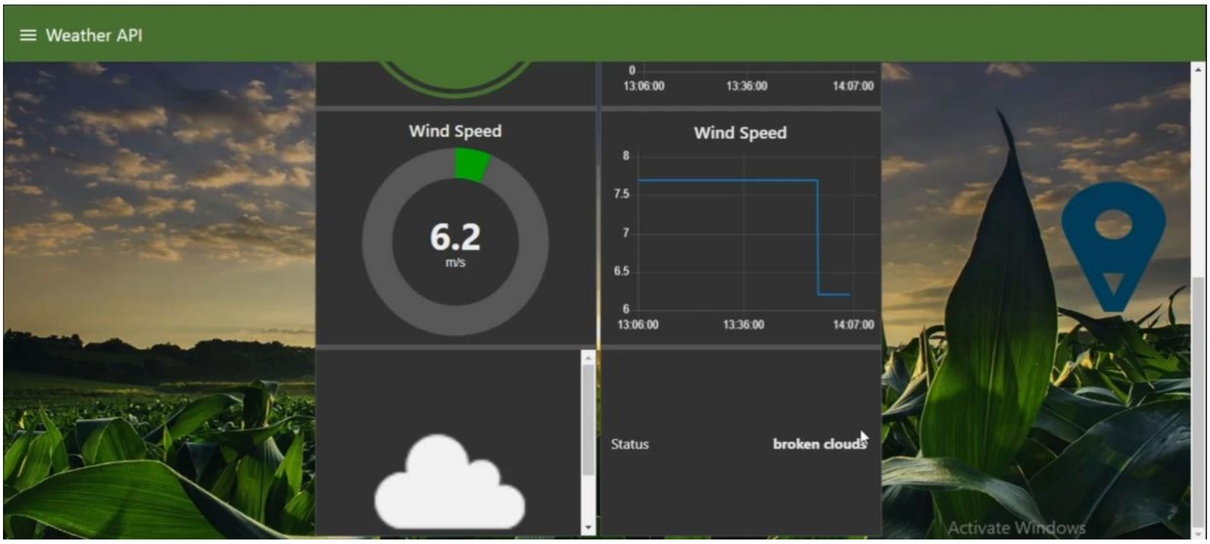
8. TESTING

8.1 TEST CASES :



9. RESULTS

9.1 PERFORMANCE METRICS :



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES :

1. Sensors installed on IoT devices are able to collect a large volume of useful information for farmers. As we mentioned below, some examples are climatic conditions, soil quality and plantation progress.

2. Such data can be used to monitor the status of the farm, as well as the performance of workers and the efficiency of the appliances.

3. With greater production control, IoT in agriculture facilitates cost-efficient management. From smart devices, producers can more accurately identify any anomaly in the crop.

4. With IoT, farmers can monitor the health of farm animals closely, even if they are physically distant. Thus, one can reduce the search time of cows and sheep in the pasture, for example, if they are part of the herd.

5. One more benefit is increased harvest —as we mentioned in the above topics—that yields a competitive advantage in business. To exemplify, we can mention preventive maintenance.

6. Once sensors are installed on a tractor, for example, the collected data can quickly notify whenever any technical failure arises.

7. In addition, one can also save in the process of irrigation and fertilization. After all, there are sensors installed in the agricultural machinery, which can generate a lot of information about the soil.

8. Another advantage is the possibility of programming the sensors to notify about the ideal harvest time. In this way, waste is avoided in the crop.

9. Such data can be used to monitor the status of the farm, as well as the performance of workers and the efficiency of the appliances.

10. Intelligent data collection. Sensors installed on IoT devices are able to collect a large volume of useful information for farmers.

DISADVANTAGES :

1. One huge disadvantage of smart farming is that it requires an unlimited or continuous internet connection to be successful.

2. This means that in rural communities, especially in the developing countries where we have mass crop production, it is completely impossible to operate this farming method.

3. In places where internet connections are frustratingly slow, smart farming will be an impossibility.

4. As pointed out earlier, smart farming makes use of high techs that require technical skill and precision to make it a success.

5. It requires an understanding of robotics and ICT. However, many farmers do not have these skills.

6. Even finding someone with this technical ability is difficult or even expensive to come by, at most.

7. And Advantages and Disadvantages of Smart Farming, this can be a discouraging factor hindering a lot of promising farmers from adopting it.

11. CONCLUSION

Smart farming is a modern farming management concept with IoT technology to increase the productivity in agriculture. With the use of smart farming, users can effectively monitor the crop field the quality, quantity of their crops and to irrigate the crops by using mobile application . Various parameters can be analyzed from the mobile application such as temperature, humidity and ph.

12. FUTURE SCOPE

Smart farming refers to managing farms using modern Information and communication technologies to increase the quantity and quality of products while optimizing the human labor required. Among the technologies available for present-day farmers are: Sensors: soil, water, light, humidity, temperature management.

13. APPENDIX

SOURCE CODE :

```
# -*- coding: utf-8 -*-
"""

"""

# PREPARE STRINGS FOR STATEMENT

#===========================================================================================

low_potassium = "\nThe amount of potassium in your soil is low! We recommend using a class 1 fertiliser to improve your soil condition to grow the best crops for the season!"


low_nitrogen = "\nThe nitrogen content of your soil is low! We recommend using a class 2 fertiliser to improve your soil condition to make the most of your field!"


low_phosphorous = "\nThe phosphorous content in your soil is low! We recommend using a class 3 fertiliser to improve your soil quality to get the best out of your field!"

#===========================================================================================


import Sensor_values as sv

import Crop_Pred as cp

import Fertiliser_Prediction as fp

import numpy as np


# GET THE SENSOR VALUES INTO THE CODE

sensor_values = sv.get_readings()

user_location = sensor_values[0]
```

```
sensor_values = sensor_values[1]

# CROP AND FERTILISER PREDICTION
#=============================================================
========
# KNP FOR FERTILISER PREDICTION
Fertiliser_Input = np.array(sensor_values[1 : 4])

# NPK & pH FOR CROP PREDICTION
crop_input = sensor_values[0:4]
temp = crop_input[0]
for i in range(0, len(crop_input)-1):
    crop_input[i] = crop_input[i+1]

crop_input[len(crop_input)-1] = temp

# FINAL CROP PREDICTION
crop = cp.Predict_Crop(crop_input)
keys = list(crop[1].keys())
values = list(crop[1].values())

for i in range(0, len(values)):
    if(int(crop[0]) == i):
        crop_name = keys[i]

print("\nThe crop you should grow to get the most out of your field is ",
      crop_name)
```

```
fertiliser = int(fp.Predict_Fertiliser([Fertiliser_Input]))

if(fertiliser == 1):
    print(low_potassium)
elif(fertiliser == 2):
    print(low_nitrogen)
else:
    print(low_phosphorous)
#==================================================================
========



# WEATHER STATION (HARDCODED)
#==================================================================
========
temp = sensor_values[4]
humidity = sensor_values[5]
pressure = sensor_values[6]

if(humidity > 70):
    print("\nIt's likely to rain today!")
elif(pressure < 100 and humidity > 70):
    print("\nHigh chances of a thunderstorm! Stay safe!")
elif(pressure < 99):
    print("\nStrong winds headed your way!")
else:
    print("\nThe weather is clear today!")
```

GITHUB LINK - https://github.com/IBM-EPBL/IBM-Project-47627-1660800585

PROJECT DEMO LINK -
https://drive.google.com/file/d/1vY7twAdZtYhbtlUvjN-pS7mdPEFf0ZZR/view?usp=share_link