SMART FARMER -
IOT ENABLED SMART FARMING APPLICATION

A IBM PROJECT REPORT

Submitted by

ALAMEEN. AA (311019205005)
DINESH KUMAR. A (311019205011)
SURENDHARAN. EB (311019205042)
VIGNESHWAR. J (311019205044)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

This Smart Irrigation System is used to help farmers in the irrigation process.The System provides data on the parameters which can be used to monitor the condition of the field to maintain and protect the crops.The parameters like temperature, soil moisture, the water level in the field, etc., can be accessed through the system.The sensors in the system monitor the parameters and provide them to the farmer through the Wi-Fi module to the IBM cloud to take the necessary measures.

1.2 PURPOSE

By making farming more connected and intelligent, precision agriculture helps reduce overall costs and improve the quality and quantity of products, the sustainability of agriculture and the experience for the consumer. Increasing control over production leads to better cost management and waste reduction.

The ability to trace anomalies in crop growth or livestock health, for instance, helps eliminate the risk of losing yields. Additionally, automation boosts efficiency. With smart devices, multiple processes can be activated at the same time, and automated services enhance product quality and volume by better controlling production processes.

Smart farming systems also enable careful management of the demand forecast and delivery of goods to market just in time to reduce waste. Precision agriculture is focused on managing the supply of land and, based on its condition, concentrating on the right growing parameters – for example, moisture, fertilizer or material content – to provide production for the right crop that is in demand.

The types of precision farming systems implemented depend on the use of software for the management of the business. Control systems manage sensor input, delivering remote information for supply and decision support, in addition to the automation of machines and equipment for responding to emerging issues and production support.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

A. SMART AGRICULTURE USING IOT

Author: Jayakumar R, Karthikeyan S N, Naveen Perumal M, Methini M

Published Year: June, 2019

The main objective of this project is to improve the crop yield and thereby meet the demand. This project remotely measures and monitor water moisture levels in the soil to ensure that crops are getting optimal water resources and automatically trigger sprinkler systems to address low moisture levels in the soil to prevent crop damage or loss. This idea will improve the crop yield and manage them.

B. RESEARCH PAPER ON SMART AGRICULTURE USING IOT

Author: Ritika Srivastava, Vandana Sharma, Vishal Jaiswal, Sumit Raj

Published Year: July, 2020

This paper proposes a system which can monitor temperature, level of water, moisture and even the movement, if any, happens in the field which may destroy the crops in agricultural field through sensors using Arduino UNO board. Smart agriculture is an emerging concept, because IOT sensors are capable of providing information about agriculture fields and then act upon based on the user input. The project aims at making use of evolving technology i.e., IOT and smart agriculture using automation. Once hardware has been developed depending on the change in requirements and technology the software needs the updating.

C. SMART AGRICULTURE: IOT BASED SMART SENSORS AGRICULTURE

Author: Anand Nayyar and Er. Vikram Puri

Published Year: November, 2016

This paper describes Internet of Things (IOT) technology has brought revolution to each and every field of common man's life by making everything smart and intelligent. IOT refers to a network of things which make a self-

configuring network. The development of Intelligent Smart Farming IOT based devices is day by day turning the face of agriculture production by not only enhancing it but also making it cost-effective and reducing wastage. The aim / objective of this paper is to propose a Novel Smart IOT based Agriculture assisting farmers in getting Live Data (Temperature, Soil Moisture) for efficient environment monitoring which will enable them to do smart farming and increase their overall yield and quality of products.

D. SMART FARMING SYSTEM USING DATA MINING

Author: Priyanka P.Chandak, Dr. A. J. Agrawal

Published Year: 2017

This paper proposes a smart farming system which provides better solution to farmers for high yield. In this system all 3 main modules i.e., Irrigation, Fertilizer and pesticide modules are integrated. Smart farming system is a web application with huge amount of dataset available in backend. The data mining is used in the process of finding correlations or patterns among the dozens of fields in relational databases. Clustering algorithm is used. Clustering is the process which partitions given data set into homogeneous group based on similarities and dissimilarity. Initially farmer have to send soil for testing and feed the soil testing report details (which include nitrogen, potassium, phosphorus, calcium, magnesium, etc.) in application. These details are necessary for prediction of water required, fertilizer and pesticides. Also, for the first time, it has to save the exact location of farm so that the longitude and latitude of farm is identified which is useful to get the exact temperature of farm location. Temperature of farm location is identified from satellite and online whether forecasting sources. We need to insert initial crop information such as crop name, crop stage, soil condition, etc.

2.2 REFERENCES

1. Meera.S, Sharmikha Sree.R, Kalpana R.A, S.R.Manasvinii, Haritha.V and Dr. K.Valarmathi, “IOT Based SMART FARMING System Using Arduino and Node MCU”, Smart Intelligent Computing and Communication Technology, October-2021.

2.M.Prasanna, M.Iyapparaja, M.Vinothkumar, B Ramamurthy, S.S.Manivannan, “An Intelligent SMART FARMING using Internet of Things”, International Journal of Recent Technology and Engineering (IJRTE), November-2019.

2.3 PROBLEM STATEMENT DEFINITION

Indian agriculture is being plagued by various problems. These problems, directly and indirectly, affect the life of a farmer. Farming practices and other activities of agriculture consume time as well as the efforts of a farmer. The drastic climatic change also affect the productivity and yield of farmers. These problems faced by farmers go unnoticed in the entire process of harvesting crops.

The major problem among these is the requirement of water for irrigation which needs to be supplied in precise amount for proper growth of crops. The weather conditions like temperature, humidity, soil moisture, etc,. also plays a vital role in the cultivation of crops. These kind of problems need creative technological solution for healthier and large crops.

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

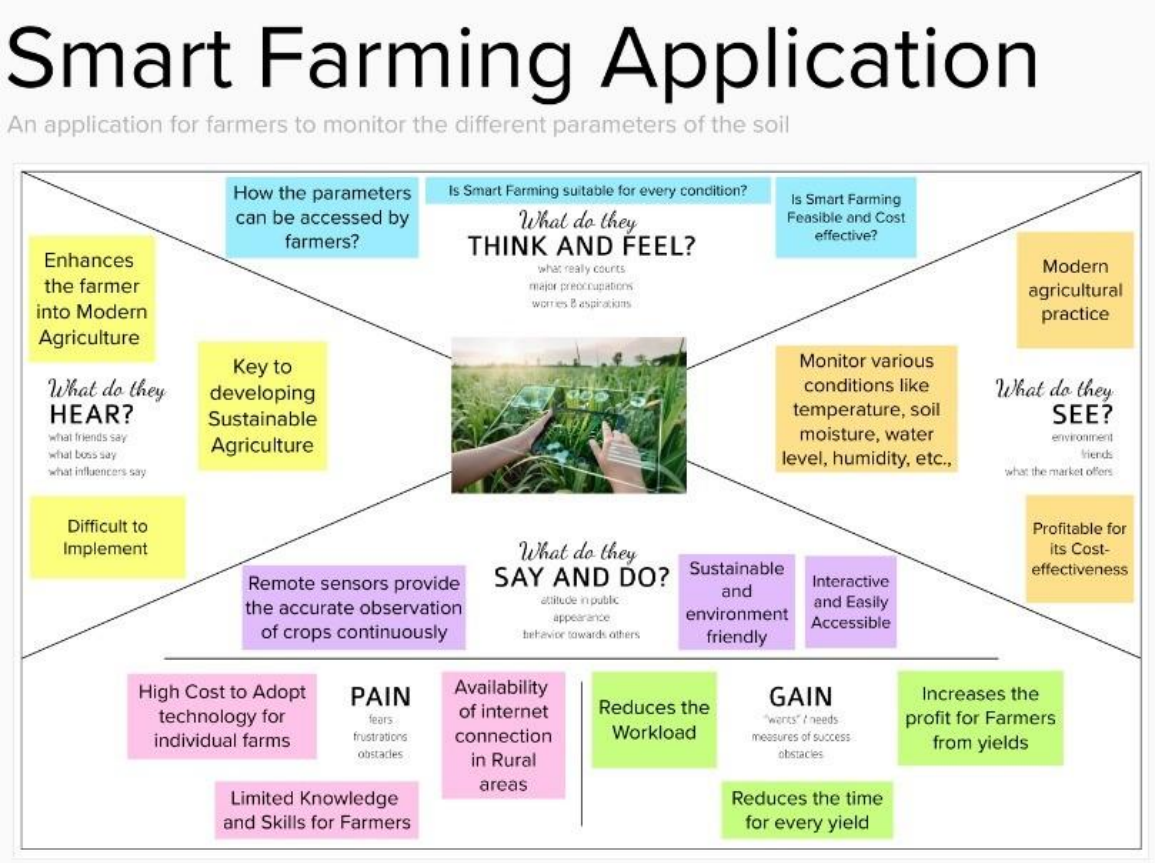
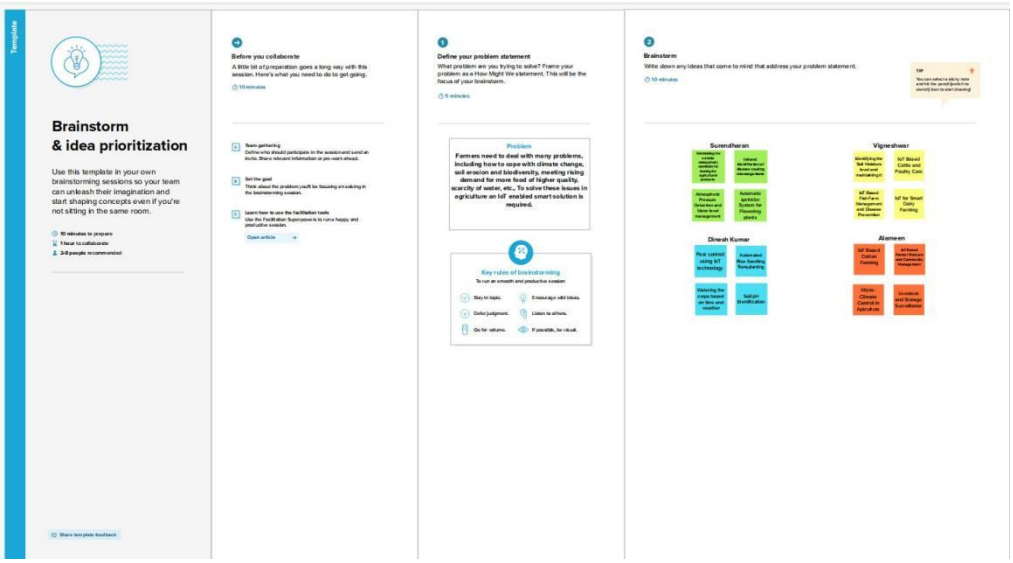


Figure 3.1: Empathy Map

3.2 IDEATION & BRAINSTROMING



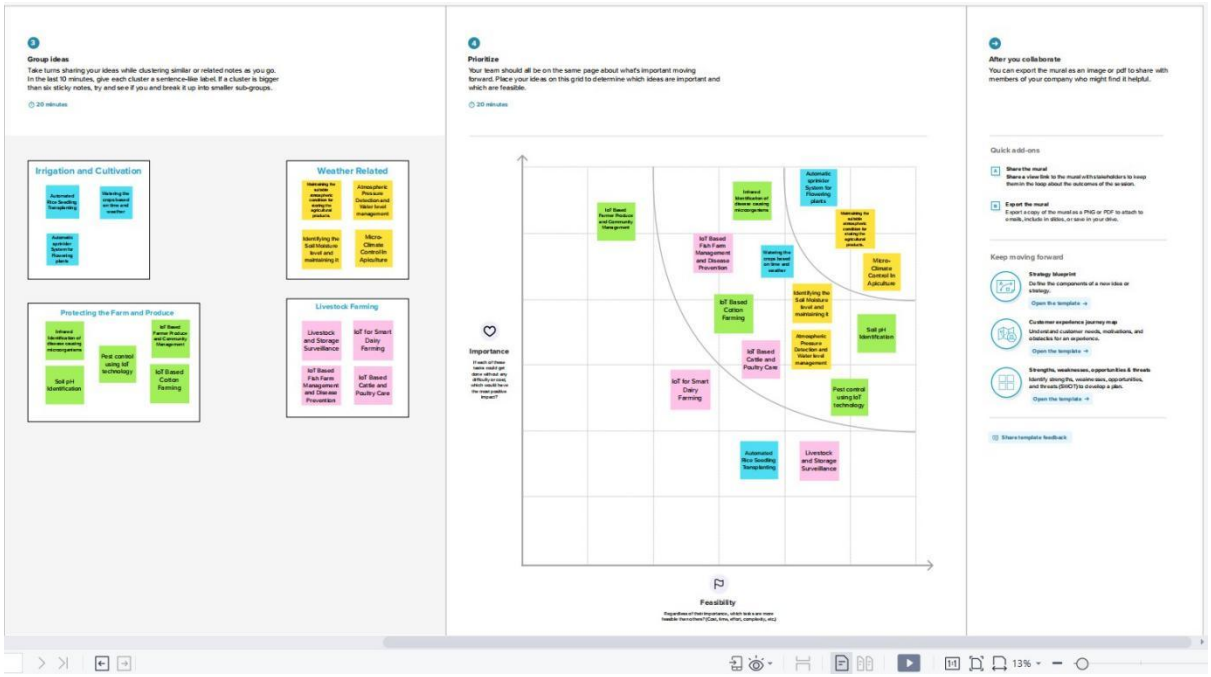


Figure 3.2: Ideation And Brainstroming

3.3 PROPOSED SOLUTION

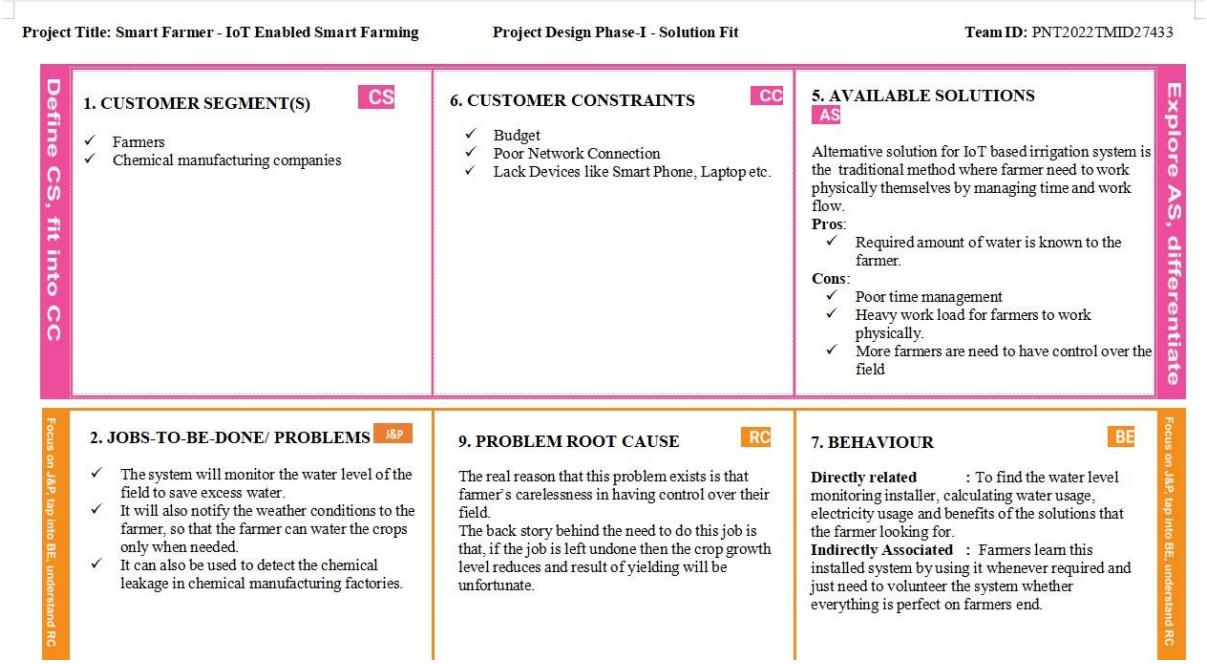
S.No.	Parameter	Description
1.	Problem Statement	The requirement of water for irrigation needs to be supplied in precise amount for proper growth of crops. Excess or less water can lead to poor growth and overall health.
2.	Idea/Solution description	These problems can be overthrown by using the advanced IoT system for irrigation.
3.	Novelty/Uniqueness	The system will monitor the water level of the field and also the weather parameters of the particular location.
4.	Social Impact/Customer Satisfaction	The farmer/customer can be able to assess the water level of the field through a mobile application and the weather parameters to know whether it will rain or not.

5.	Business Model	**Key Partners:** • Farmers • Chemical Factories **Key Activities:** • The system will monitor the water level of the field. • It will also report the weather conditions of the location which is efficient for plant growth. **Value Proposition:** • The Smart Irrigation System will reduce the workload of the farmer. • It will be useful to preserve the excess water which can be used for growing other crops. • The system can also be used in some chemical manufacturing factories in which the leakage of chemicals can be detected using the level indicator. **Cost Structure:** • Cost estimation: ➢ The estimates will vary depending upon the cost of the sensors and the software used. • Cost budget: ➢ The budget will be set based on the requirements of the Smart Irrigation system.

		• Cost Control: ➢ The cost can be reduced by using efficient and low-cost sensors.
6.	Scalability of the Solution	The Smart Irrigation System can send the information to the cloud, so that it can be viewed from anywhere which makes it an ideal system for agricultural needs.

Table 3.1: Proposed Solution

3.4 PROBLEM SOLUTION FIT



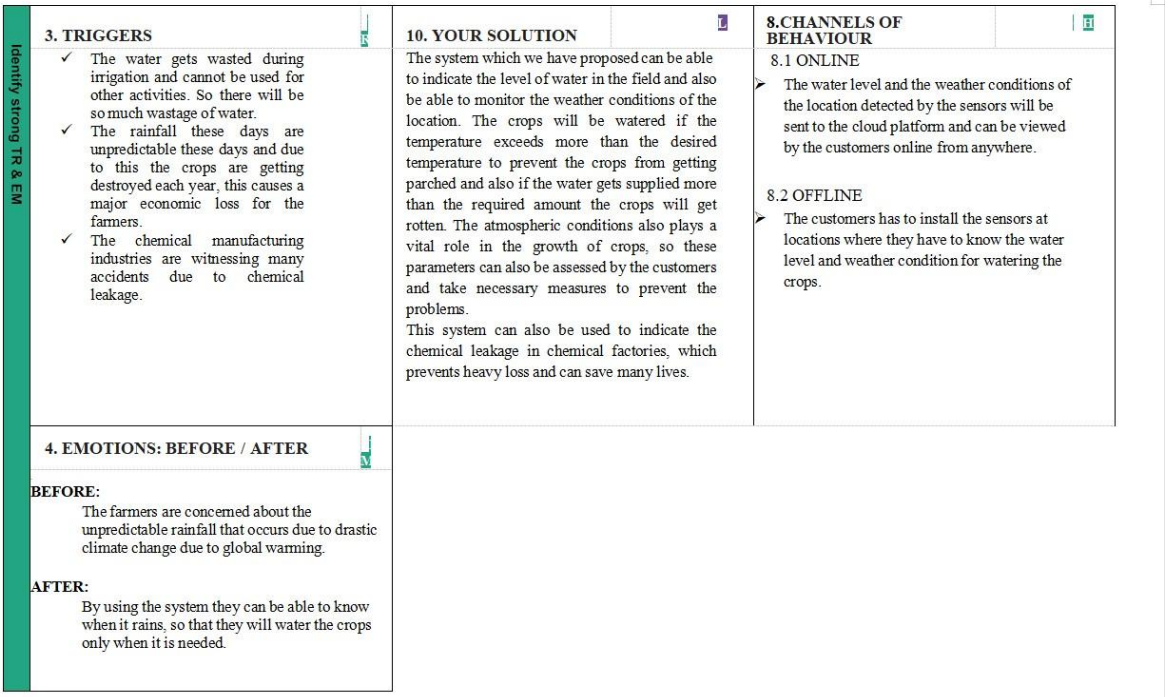


Figure 3.3: Problem Solution Fit

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

FR NO.	Functional Requirement(Epic)	Sub Requirement(Story/Sub-Task)
FR - 1	User Registration	Registration through GMAIL Fill the details like Phone No. Etc. Create Username and Password
FR - 2	User Confirmation	Confirmation via EMAIL Confirmation via OTP
FR - 3	User Log in	User credentials like username and password are validated.
FR - 4	Sensors	Required sensors are connected.
FR - 5	Sensor value	Values that are obtained from sensor are verified and noted.
FR - 6	Irrigation system	Verify the amount of water for the crops and maintain the level.
FR - 7	Log out	Exit

Table 4.1:Functional Requirements

4.2 NON - FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR - 1	**Usability**	Smart farming is safe to use. User with low level of understanding can easily grasp the concept of the system. The quality of the output will be as expected by the user.

NFR - 2	**Security**	Smart farming system will have protection against malware as each user will be provided with a specific credentials to access data in IBM cloud. In case if the data is lost, using restore option the user can regain the data.
NFR - 3	**Reliability**	The Smart Farming System will give highly precise sensor data to the cloud, so that the user
NFR - 4	**Performance**	This system perform at topmost level, where user just need to monitor the system. The remaining job that are to be done by the system will be as fixed in the system.
NFR - 5	**Availability**	The Smart Farming System is flexible with any type of devices like PC, mobile phone etc,. as the customer can view it in the cloud platform and also through a mobile app.
NFR - 6	**Scalability**	This system will be very useful to the maximum extent that the user can use. Apart from irrigation system, some sensors to display weather parameters are also connected to the irrigation system by which user know the temperature, pressure and humidity values based on that also watering the crops is managed.

Table 4.2: Non Functional Requirements

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

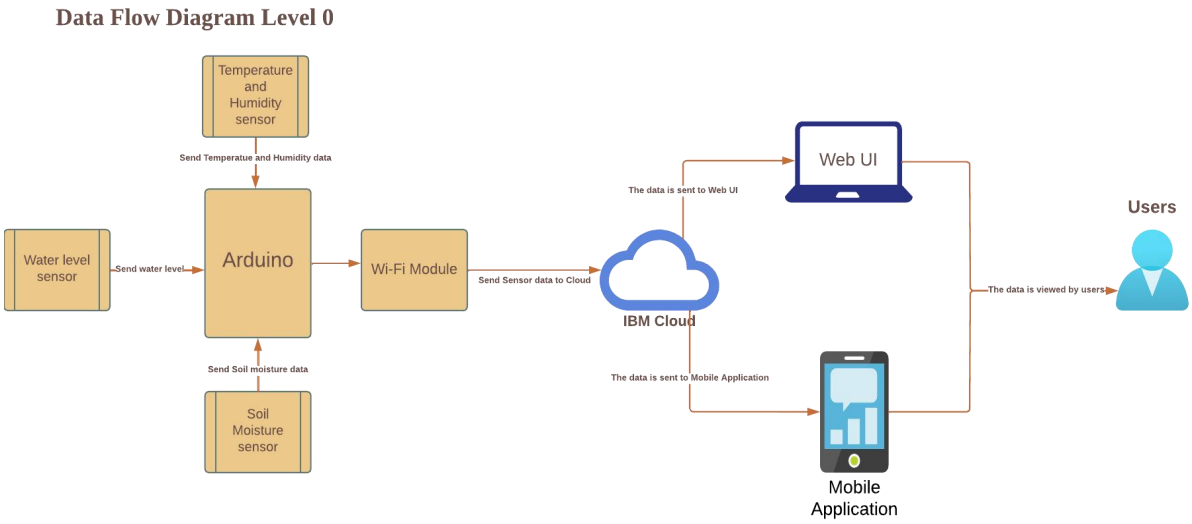


Figure 5.1: Data Flow Diagram

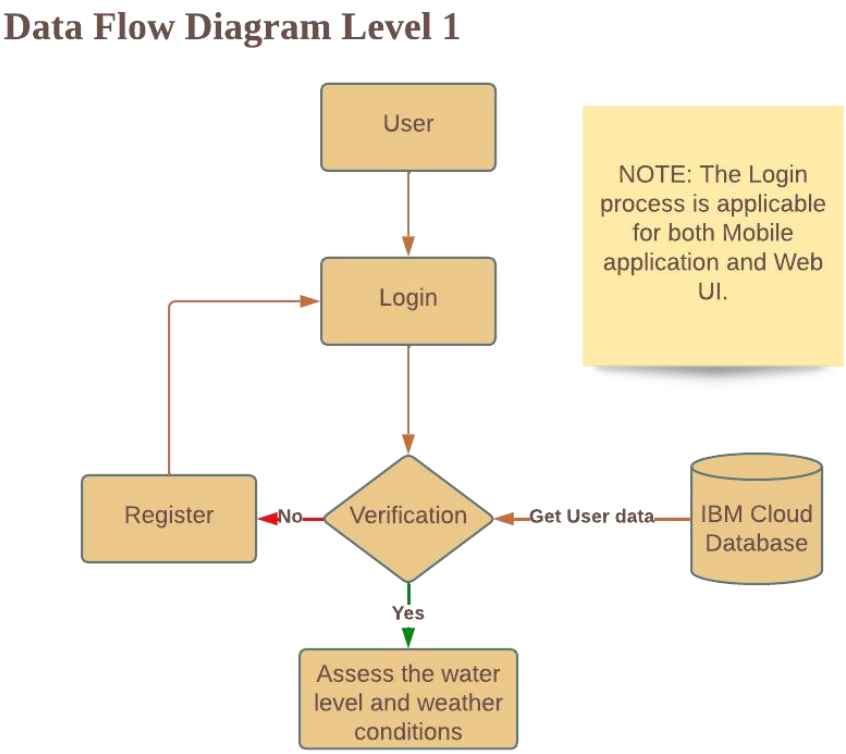


Figure 5.2: Data Flow Level 1

5.2 SOLUTION AND TECHNICAL ARCHITECTURE

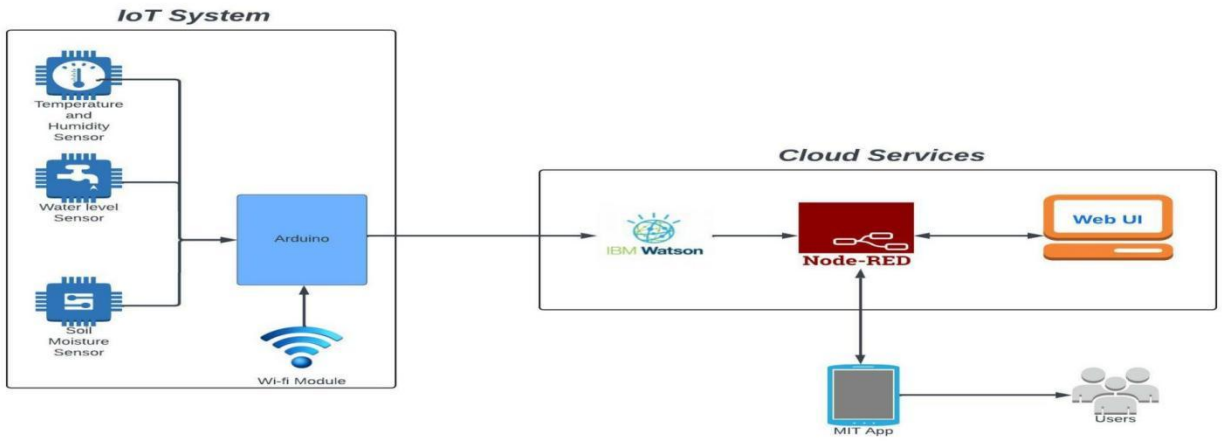


Figure 5.3: Solution And Technical Architecture

S.No	Component	Description	Technology
1.	User Interface	It is to make the user to interact with the application.	MIT App Inventor
2.	Application Logic-1	Controlling the water pump remotely using mobile phone through the application.	Python
3.	Application Logic-2	Based on the rainfall, the water level in the field can be adjusted and also the atmospheric conditions can also be accessed.	Python
4.	Application Logic-3	The chemical storage facilities can be monitored for leakage and also the pressure conditions can be accessed.	Python
5.	Database	Data that are to generally gets stored.	IBM Watson IoT, IBM Watson Node-Red
6.	Cloud Database	Cloud database is a database built to run in a public or hybrid cloud environment to help organize, store and manage data within an organization.	IBM DB2, IBM Cloudant

7.	File Storage	File storage is used to store data in the system that are of high volume.	IBM File Storage
8.	External API-1	To obtain various weather parameters.	OpenWeather
9.	Infrastructure (Server / Cloud)	Deployment of the application happens on local system/ cloud server configuration.	IBM Cloudant, IBM IoT platform

Table 5.1: Solution And Technical Architecture

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story/Task	Acceptance criteria	Priority	Release
Farmer (Mobile User)	Registration	USN - 1	As a farmer, I want to first register using my email and create a password for the account.	I am given access to my account and can use it after installing the application.	High	Sprint - 1
	Login	USN - 2	As a registered user, I need to easily log in using the registered account.	I should be able to access the dashboard and can view the needed stats.	High	Sprint - 1
	User Interface	USN - 3	As a user, I want an attractive user interface to view the parameters remotely through	I can access the dashboard after logging in using the navigation panel and view the needed parameters in my field.	Medium	Sprint - 2

			mobile.			
	Data Visualization	USN - 4	The data from the sensors transmitted through Arduino needs to be viewed in the mobile application remotely from anywhere.	I can view the visualization of data in the mobile application.	High	Sprint - 3
Farmer (Web User)	Registration	USN - 1	As a farmer, I want to first register using my email and create a password for the account.	I am given access to my account and can use it.	High	Sprint - 1
	Login	USN - 2	As a registered user, I need to easily login log into my registered account via the webpage in minimum time.	I can access the webpage from the cloud platform and can be able to access the necessary data.	High	Sprint - 1
	Web UI	USN - 3	As a user, I need to have a friendly user	I can easily access and monitor the parameters in	Medium	Sprint - 4

			interface to easily view and access the resources.	the field at ease.		
Chemical Manufacturer (Web user)	Registration	USN - 1	As a new user, I want to first register using my organizatio n email and create a password for the account.	I am now given access to the account.	High	Sprint - 1
	Login	USN - 2	As a registered user, I need to easily log in using the registered account via the webpage.	I can access the webpage and can be able to access the necessary data needed for monitoring chemical storage facilities.	High	Sprint - 1
	Web UI	USN - 3	As a user, I need to have a user-friendly interface to easily view and access the resources.	I can easily access and monitor the parameters in the chemical storage at the factory from anywhere through the webpage.	Medium	Sprint - 3
Chemical Manufacturer (Mobile User)	Registration	USN - 1	As a user, I want to first register using my email and create a password	I am given access to use the application and can use it on my mobile.	High	Sprint - 1

			for the account.			
	Login	USN - 2	As a registered user, I need to easily log in to the application.	I should be able to access the dashboard and can view the needed stats.	High	Sprint - 1
	Data Visualization	USN - 3	The data from the sensors transmitted through Arduino needs to be viewed in the mobile application remotely from anywhere.	I can view the visualization of data from the chemical storage facilities in the mobile application.	High	Sprint - 2
	User Interface	USN - 4	As a user, I want an attractive user interface to view the parameters remotely through mobile.	I can access the dashboard after logging into the application and viewing the conditions of the chemical storage facility.	Medium	Sprint - 3
Administrator	Login	USN - 1	As an admin, I need to log in to the account and should view the status of all the installed devices.	I can log in to the webpage and access the basic information of the customer and their device status and their type of work/domain.	High	Sprint - 1

	Upgrades	USN - 2	As an admin, I need to update new upgrades and services up to the date.	I have upgraded the webpage or application with advanced services according to the kind of task they need to accomplish.	Medium	Sprint - 4
	Support	USN - 3	As an admin, I need to address customer queries and take necessary measures.	I have addressed the customer's queries and provided the necessary services.	Low	Sprint - 4

Table 5.2: User Stories

CHAPTER 6

PROJECT PLANNING &SCHEDULE

6.1 SPRINT PLANNING AND ESTIMITION

Sprint	Functional Requirement (Epic)	User Story Number	User Story /Task	Story Points	Priority	Team Member
Sprint-1	Registration (Farmer - Mobile User)	UNS-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Dinesh Kumar A (Member 2)
Sprint-1	Login	UNS-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Dinesh Kumar A (Member 2)
Sprint-2	User Interface	UNS-3	As a user, I can register for the application through Facebook	2	Low	Dinesh Kumar A (Member 2)
Sprint-1	Data Visualization	UNS-4	As a user, I can register for the application through GMAIL	2	Medium	Dinesh Kumar A (Member 2)
Sprint-3	Registration (Farmer - Web User)	USN - 1	As a user, I can log into the application by entering	1	High	Dinesh Kumar A (Member 2)

			email and password			
Sprint - 1	Login	USN - 2	As a registered user, I need to easily login log into my registered account via the web page in minimum time	1	High	Al Ameen A A (Member 3)
Sprint - 4	Web UI	USN - 3	As a user, I need to have a friendly user interface to easily view and access the resources	1	Mediu m	Al Ameen A A (Member 3)
Sprint - 1	Registratio n (Chemical Manufactur er - Web user)	USN - 1	As a new user, I want to first register using my organization email and create a password for the account.	2	High	Al Ameen A A (Member 3)
Sprint - 1	Login	USN - 2	As a registered user, I need to easily log in using the registered account via the web page.	1	High	Al Ameen A A (Member 3)
Sprint - 3	Web UI	USN - 3	As a user, I need to have a userfriendly interface to easily view and access the resources.	2	Mediu m	Surendharan E B (Leader)

Sprint - 1	Registration (Chemical Manufacturer - Mobile User)	USN - 1	As a user, I want to first register using my email and create a password for the account.	2	High	Surendharan E B (Leader)
Sprint - 1	Login	USN - 2	As a registered user, I need to easily log in to the application.	2	Low	Surendharan E B (Leader)
Sprint - 2	Data Visualization	USN - 3	The data from the sensors transmitted through Arduino needs to be viewed in the mobile application remotely from anywhere	2	Medium	Surendharan E B (Leader)
Sprint - 3	User Interface	USN - 4	As a user, I want an attractive user interface to view the parameters remotely through mobile	1	Medium	Vigneshwar J (Member 1)
Sprint - 1	Login (Administrator)	USN - 1	As an admin, I need to log in to the account and should view the status of all the installed devices.	2	Medium	Vigneshwar J (Member 1)
Sprint - 4	Upgrade	USN - 2	As an admin, I need to update	2	High	Vigneshwar J (Member 1)

			new upgrades and services up to the date.			
Sprint - 4	Support	USN - 3	As an admin, I need to address customer queries and take necessary measures.	1	High	Vigneshwar J (Member 1)

Table 6.1: Sprint Planning And Estimation

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Point Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	07 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	14 Nov 2022

Table 6.2: Sprint Delivery Schedule

CHAPTER 7

CODING AND SOLUTIONING

7.1 FEATURE 1

The main feature of the Smart Farmer project is that it will be able to monitor the atmospheric parameters of the field using the sensors and send these data to the IBM cloud. From the IBM cloud, the data will be transferred through the ESP to the mobile application through which the farmer can be able to access the parameters remotely from anywhere. The system provides data of the parameters like temperature, humidity, soil moisture and water level in the field. The farmer by checking these parameters, can be able to make decisions whether to water the crops or not by controlling the motor from the mobile application itself.

The data provided by the system cannot only be viewed in mobile application but can also be visualized using Web UI provided by the Node-RED in its dashboard. The data can be visualized in any visual representation of the farmer's choice.

7.2 FEATURE 2

Another distinct feature of the Smart Farmer system is that the parameters of the field can also be viewed through a LCD which is provided with the system. The farmer through this LCD can view the atmospheric conditions of the field directly from the system when there is no internet connection. This feature helps the farmers to assess the field even if there is any disaster situation during when there will be no internet connectivity.

CHAPTER 8

TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
HomePage_01	Functional	Home Page	Verify user can see the Login page when user clicked on Go to Login button	App should be installed on the mobile.	1.Click on Go to Login button. 2. The app will enter the login page.	The login page should be displayed.	Working as expected.	Pass	Steps are clear and simple.	N	Nil	Vigneshwar J
LoginPage_02	Functional	Login Page	Verify the user can view and login in the login page.	Go to login button should be clicked on the previous page.	1.Enter Username and Password. 2. Click on Login button below to get into the Info page. 3. Verify login with below UI elements: a. Username text box b. Password text box c. Login button	1. Application should go to the Info page after successfully Logging in. 2. If the credentials are wrong, an alert notifier should pop up.	Working as expected.	Pass	Simple design.	N	Nil	Alameen A A

Test Cases

LoginPage_03	UI	Login page	Verify user can log into application with Valid credentials	The user should be navigated from the home page.	1.Enter Valid username in text box. 2. Enter valid password in password text box. 3. Click on login button.	User should navigate to app information page.	Working as expected.	Pass	Easy and simple user-friendly UI.	N	Nil	Vigneshwar J
LoginPage_03	UI	Login page	Verify user can log into application with Valid credentials	The user should be navigated from the home page.	1.Enter Valid username in text box. 2. Enter valid password in password text box. 3. Click on login button.	User should navigate to app information page.	Working as expected.	Pass	Easy and simple user-friendly UI.	N	Nil	Vigneshwar J
InfoPage_04	Functional	Info page	Verify user can view the information about the Smart Farmer System.	The user should successfully login from Login page.	1. Enter the Info page. 2. The user can know about the Smart Farmer Application. 3. The user can also logout from this page to the login page.	1. Application should show Monitor Parameters and Logout button. 2. When clicked monitor parameters button, the user should navigate to the Main page. 3. When clicked Logout button, the user should navigate back to the Login page.	Working as expected.	Pass	Clear information was given.	N	Nil	Alameen A A
InfoPage_05	UI	Info page	1.Verify user can navigate into the main page by clicking the Monitor parameters button. 2. When clicked Logout button, verify the user should navigate back to the Login page.	The user should successfully login from Login page.	1. Enter the Info page. 2. The user can know about the Smart Farmer Application. 3. The user can get into the main page from here through the monitor parameters button. 3. The user can also logout from this page to the login page.	1. The user should be navigated to the Main page. 2. The user can also logout to the Login page.	Working as expected.	Pass	Clear information was given.	N	Nil	Dinesh Kumar A
MainPage_06	Functional	Main page	1.Verify user can monitor the parameters from the main page. 2. Verify user can control the motor function through the main page.	The user needs to successfully navigated from the Info Page.	1.Enter the Main page. 2. Click on Motor control buttons to control the motor remotely from the Mobile. 3. Click on the Back button to navigate back to Info Page. 4. Click on the Exit button to completely exit from the application. 5.The user can remotely	1. The user can able to view the parameters of the field remotely from anywhere. 2. The user can also control the motor remotely from anywhere to regulate water in fields.	Working as expected.	Pass	1.The parameters can be viewed remotely. 2. Motor controlling is easy.	N	Nil	Surendharan E B

Figure 8.1:Test Cases

8.2 USER ACCEPTANCE TESTING

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Sub Total
By Design	5	1	0	0	**6**
Duplicate	0	0	0	0	**0**
External	8	0	4	1	**12**
Fixed	13	2	4	1	**18**
Not reproduced	7	2	0	0	**9**
Skipped	1	0	0	0	**1**
Won't Fix	0	0	0	0	**0**
Totals	**34**	**5**	**8**	**2**	**46**

Table 8.1:Defect Analysis

Section	Total Cases	Not Tested	Fail	Pass
Temperature and Humidity Sensor	35	0	0	35
Ultrasonic Sensor	40	0	0	40
Soil Moisture	25	0	0	25
Wi-Fi Module	2	0	0	2
Transmission of data to IBM Cloud	3	0	1	2
Data Transmission from Cloud to Mobile App	5	0	2	3

User login in Mobile Application	10	0	0	10
Accessing the Parameters in Mobile App	15	0	2	13
Controlling the Motor from the Mobile App	5	0	0	5
Viewing the parameters in the Node RED	5	0	0	5
Controlling the Motor from Node RED	3	0	0	3

Table 8.2 Test Case Analysis

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

			NFT - Risk Assessment						
S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Volume Changes	Risk Score	Justification
1.	Smart Farmer	Existing- Simulating the project through the Tinkercad with Temperature and humidity sensors, soil moisture, ultrasonic distance sensors, and DC and servo motors.	Moderate	High	High	No data transmission to Cloud	>80 to 90%	ORANGE	There is no Wi-Fi module in the Tinkercad simulator so data can't be sent to IBM Cloud.
2.	Smart Farmer	New- Simulating the project through the Wokwi simulator with Temperature and humidity sensor, ultrasonic distance sensors, servo motor, and LCD.	High	High	Moderate	The non-availability of certain sensors in Wokwi.	>30 to 40%	YELLOW	The random function is used for the Soil Moisture sensor to generate some random value.
3.	Smart Farmer	Existing – Visualizing the weather parameters in the Watson IoT platform.	Moderate	No Changes	Low	Delayed Visualization of Data.	>50 to 60%	GREEN	The stable internet connection is enough for a constant data transmission.
4.	Smart Farmer	Existing- Visualizing the weather parameters in the Watson IoT platform.	No Changes	No Changes	Moderate	Delayed Visualization of Data.	>40 to 50%	GREEN	The data can be easily transferred to other applications and also can be visualized in the dashboard.
5.	Smart Farmer	New- Login to the Smart Farmer mobile application and viewing the parameters.	Moderate	No Changes	High	Latency of data will be high.	>20 to 10%	GREEN	The parameter send by the module will be stored in the cloud and then sent to the mobile app, so there will be less latency.
6.	Smart Farmer	New – Controlling the motor from the mobile application and its indication in the simulator.	Low	Low	Low	Motor control will be delayed.	>30 to 20%	YELLOW	The motor control can be controlled by sending a response from the mobile app to the module.

Figure 9.1:Performance Metrics

NFT - Detailed Test Plan				
S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/Sign Off
1.	Smart Farmer	Spike Testing – For the sensors in the module.	1. For the temperature and humidity sensor, the values should be tested at extreme high, moderate, and extreme low levels to know that the indication is going on correctly. 2. For the Ultrasonic distance sensor, the distance will be increased and decreased to simulate the water level in the field. 3. For soil moisture, the random function should generate the values within the limit. 4. The ESP32 module should process and transmit data to IBM cloud.	Approved
2.	Smart Farmer	Endurance Testing – For Watson IoT visualization boards.	1. The parameter data should be accessed through the IBM Watson IoT Platform. 2. The visualization data should be continuously stored for a specified long duration.	Approved
3.	Smart Farmer	Resilience Testing – For Node-Red Dashboard Visualization.	1. The Node-Red should be able to perform well with different datasets or payloads coming from the module. 2. The Node-Red should display the correct parameter data and both the IBM and Node-Red data should match.	Approved
4.	Smart Farmer	Load Testing – For accessing the parameter data and controlling the motor from the mobile application.	1. The parameter data can be viewed and the motor should be controlled from the mobile application itself. 2. The data should be precise even if multiple user data for visualization.	Approved

Figure 9.2 :NFT - DETAILED TEST PLAN

End Of Test Report							
S. No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Identified Defects (Detected/Closed/Open)	Approvals /Sign Off
1	Smart Farmer	Performance Testing	No delay in logging in to the application.Controlling motor like ON or OFF should not take more than 5 seconds. Live update of parameters through IBM Watson IoT platform to mobile application should not take more than 10 to 15 seconds.	POSITIVE	GO	Closed	Approved
2	Smart Farmer	Stress Testing	Unexpected load given to the application does not cause any error to the system.	POSITIVE	GO	Closed	Approved
3	Smart Farmer	Load Testing	Expected load given to the system to make sure that system works fine. Like large number of user installing application to view the parameters.	POSITIVE	GO	Closed	Approved
4	Smart Farmer	Compatibility Testing	Application developed can be installed in all versions of android smart phone.	POSITIVE	GO	Closed	Approved
5	Smart Farmer	Recovery Testing	If the application crashes, it can be uninstalled and can reinstall. Data that are passed to the mobile application are stored in IBM Watson IoT platform for future use.	POSITIVE	GO	Closed	Approved

Figure 9.3: END OF TEST REPORT

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

1. The Smart Farmer system helps the farmer to monitor the atmospheric conditions of the field remotely through the mobile application and also through the LCD.

2. The system enables the farmer to take necessary actions by assessing the conditions of the crops in the field.

3. The watering of the field can be done remotely by controlling the motor from the mobile application or through the Web UI.

4. With the help of the Smart Farmer system, the health of the crops will be improved leading to a better yield.

DISADVANTAGES

1. The Smart Farmer system can be integrated with the weather forecast system, which will help the farmers to prevent the damage to the crops, if any disaster occurs.

2. The systems in various locations can be linked to form a network, so that the weather condition of each area can be accessed and will be useful for predicting the disasters.

CHAPTER 11

CONCLUSION

IoT is considered the backbone of smart agricultural technology, as it connects all components of smart systems, not only in the agricultural field but also the other applications. Concerning the use of IoT in agriculture, it can be used in many practices such as farm monitoring, irrigation, pest control, harvesting, etc. IoT connects several sensors with processing units, then analyses data, then makes appropriate decisions in real-time.Finally, these smart technologies should be supported by governments in third world countries at the level of small farms, they aim to increase production and improve the efficient use of land and water resource.

CHAPTER 12

FUTURE SCOPE

1. The Smart Farmer system can be integrated with the weather forecast system, which will help the farmers to prevent the damage to the crops, if any disaster occurs.

2. The systems in various locations can be linked to form a network, so that the weather condition of each area can be accessed and will be useful for predicting the disasters.

APPENDIX

SOURCE CODE

```
#include <Wire.h>              //Includes the library for connections
#include <ESP32Servo.h>        //Includes the library for Servo motor
#include <LiquidCrystal_I2C.h> //Includes the library for LED
#include <DHTesp.h>            //Includes the library for DHT22 sensor

// WiFi libraries:
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>

#define ORG "oqy2ad" // Organization ID of IBM Cloud
#define DEVICE_TYPE "ESP32"
#define DEVICE_ID "NodeMCU"
#define TOKEN "123456789"

// Publishing Event in Watson IOT platform:
char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; //
oqy2ad.messaging.internetofthings.ibmcloud.com
char pubTopic[] = "iot-2/evt/status1/fmt/json";
char subTopic[] = "iot-2/cmd/command/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

const char *ssid = "Wokwi-GUEST";
const char *password = "";

const int led = 4;
const int servoPin = 2;
const int echo = 12;
const int trig = 14;
const int r = 27;
const int g = 26;
const int b = 25;
const int y = 33;
const int sec = 0;
const int dht = 15;
long lastMsg = 0;
```

```
Servo s;
String data3;

void callback(char *subTopic, byte *payload, unsigned int payloadLength);

#define I2C_ADDR 0x27
#define LCD_COLUMNS 20
#define LCD_LINES 4

LiquidCrystal_I2C lcd(I2C_ADDR, LCD_COLUMNS, LCD_LINES);
DHTesp dhtSensor;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback, wifiClient);

void setup()
{
  Serial.begin(115200);
  Wire.begin();
  pinMode(A0, INPUT);          // Temperature Sensor
  pinMode(trig, OUTPUT);       // Ultra sonic Trigger
  pinMode(echo, INPUT);        // Ultra sonic Echo
  pinMode(b, OUTPUT);          // BLUE light for LED
  pinMode(g, OUTPUT);          // GREEN light for LED
  pinMode(r, OUTPUT);          // RED light for LED
  pinMode(y, OUTPUT);          // YELLOW light for LED
  pinMode(led, OUTPUT);        // LED for Motor Indication
  s.attach(servoPin, 500, 2400); // Servo Motor
  lcd.init();                  // LCD Display
  lcd.setBacklight(0);
  dhtSensor.setup(dht, DHTesp::DHT22);

  Serial.println();
  // Connecting the ESP32 with WiFi:
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password, 6);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
```

```
  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());

  // Connecting to IBM Cloud:
  if (!client.connected())
  {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token))
    {
      Serial.print(".");
      delay(500);
    }

    client.setCallback(callback);
    if (client.subscribe(subTopic))
    {
      Serial.println("Subscription to cmd OK");
    }
    else
    {
      Serial.println("Subscription to cmd FAILED");
    }

    Serial.println("Bluemix connected");
    Serial.println("");
  }
}

float readDistanceCM()
{
  digitalWrite(trig, LOW);
  delayMicroseconds(2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  int duration = pulseIn(echo, HIGH);
  return duration * 0.034 / 2;
}

void loop()
{
```

```
    client.loop();
    long now = millis();

    // Temperature:
    TempAndHumidity data = dhtSensor.getTempAndHumidity();
    float t = data.temperature;
    float h = data.humidity;
    Serial.println("Temperature: " + String(t) + " degrees");
    Serial.println("Moisture: " + String(h) + " %");

    // Ultrasonic sensor:
    float distance = readDistanceCM();
    Serial.print("Measured distance: ");
    Serial.println(readDistanceCM());

    // Soil Moisture:
    int soil = random(0, 100); // As there is no soil moisture sensor, random
function is used for it.
    Serial.println("Soil Moisture: " + String(soil) + "%");

    // LCD Display:
    lcd.setBacklight(1);
    lcd.clear();

    digitalWrite(b, 0);
    digitalWrite(g, 0);
    digitalWrite(r, 0);
    digitalWrite(y, 0);

    // Conditions:
    /*If the temperature is Greater than 30 and less than 40 and also humidity or
soil moisture is greater than 30 and
    less than 70 then the GREEN light will be turned ON indicating the Normal
condition */
    if (t > 30 & t < 40 && h > 30 & h<70 | soil> 30 & soil < 70)
    {
      digitalWrite(g, 1);
      s.write(90);
      Serial.println("Normal Condition");
      Serial.println("Water Partially Flows");
      lcd.setCursor(3, 1);
      lcd.println("ON Motor");
```

```
    delay(1000);
    lcd.clear();
  }

  /*If the temperature is greater than 40 OR the humidity or soil moisture is less than 30, then the RED light will
  be turned ON indicating the Hot or Low humid condition */
  else if (t > 40 | h < 30 | soil < 30)
  {
    digitalWrite(r, 1);
    s.write(180);
    Serial.println("High Temperature or Low humid condition");
    Serial.println("Water Fully Flows");
    lcd.setCursor(3, 1);
    lcd.println("ON Motor");
    delay(1000);
    lcd.clear();
  }

  /*If the level of water is MORE in the field it will be indicated by distance sensor for less than
  10cm and soil moisture is greater than 70, then the YELLOW light will be turned ON indicating the high water level */
  else if (distance<10 & soil> 70)
  {
    digitalWrite(y, 1);
    s.write(0);
    Serial.println("Water Does Not Flow");
    Serial.println("Water is Full in the field");
    lcd.setCursor(2, 1);
    lcd.println("Drain the water");
    delay(1000);
    lcd.clear();
  }

  /*If the temperature is less than 30 OR the humidity or soil moisture is greater than 70, then the BLUE light will
  be turned ON indicating the Cool or High humid condition */
  else if (t<30 | h> 70 | soil > 70)
  {
    digitalWrite(b, 1);
    s.write(0);
    Serial.println("Cool Temperature or High Humid Condition");
```

```
    Serial.println("Water Does Not Flow");
    lcd.setCursor(3, 1);
    lcd.println("OFF Motor");
    delay(1000);
    lcd.clear();
}

else
{
    digitalWrite(b, 1);
    s.write(0);
    Serial.println("Water Does Not Flow");
}

// Sending payload:
Serial.println("");
if (now - lastMsg > 1000)
{
    lastMsg = now;

    // Payload for Parameters:
    String payload = "{\"Name\":\"" DEVICE_ID "\"";
    payload += ",\"Temperature\":";
    payload += t;
    payload += ",\"Humidity\":";
    payload += h;
    payload += ",\"Distance\":";
    payload += distance;
    payload += ",\"SoilMoisture\":";
    payload += soil;
    payload += "}";
    Serial.print("Sending payload: ");
    Serial.println(payload);
    Serial.println("");
    if (client.publish(pubTopic, (char *)payload.c_str()))
    {
        Serial.println("Publish ok for payload");
    }
    else
    {
        Serial.println("Publish failed");
    }
}
```

```
  Serial.println("--------------------------------------------------");
  lcd.setCursor(1, 0);
  lcd.print("Temp: ");
  lcd.print(t);
  lcd.print(" degree");
  lcd.setCursor(1, 1);
  lcd.print("Humidity: ");
  lcd.print(h);
  lcd.print(" %");
  lcd.setCursor(1, 2);
  lcd.print("Distance: ");
  lcd.print(distance);
  lcd.print(" cm");
  lcd.setCursor(1, 3);
  lcd.print("Soil Moisture: ");
  lcd.print(soil);
  lcd.print(" %");
  delay(5000);
  lcd.clear();
}
void callback(char *subTopic, byte *payload, unsigned int payloadLength)
{
  Serial.println("----------------------Callback!!----------------------");
  Serial.print("Callback invoked for topic:");
  Serial.println(subTopic);
  for (int i = 0; i < payloadLength; i++)
  {
    data3 += (char)payload[i];
  }
  Serial.println("Data:" + data3);
  if (data3 == "motoron")
  {
    Serial.println("Motor is ON");
    digitalWrite(led, 1);
  }
  else
  {
    Serial.println("Motor is Off");
    digitalWrite(led, 0);
  }
  data3 = "";
  Serial.println("--------------------------------------------------");
}
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-20302-1659716643

PROJECT DEMO

https://drive.google.com/drive/folders/13N0OUUuUgd7yWVaOIM4LNSF8xFTne8t0?usp=sharing