KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

HX 8001-PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

REAL-TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

NALAIYA THIRAN PROJECT REPORT 2022

Submitted by

KAVIYA S	**621319106040**
GOBIKA S	**621319106019**
KAVIPRIYA R	**621319106039**
BRUNDHALAKSHMI	**621319106010**

Team ID: PNT2022TMID13407

NOVEMBER 2022

+

1. INTRODUCTION

1.1 PROJECT OVERVIEW

In this project, I created a mobile application that allows higher authorities to keep track of data like temperature, turbidity, PH, dangerous substances, and more. Based on these specifics, we can direct the authorities to inform the local communities not to consume the water if the water quality is poor.

1.2 PURPOSE

Water is a necessary component of life. Although most of the earth is covered with water, just a small portion of that is fresh water, and only a small portion of that is drinking water. In order to enhance the amount of fresh water and drinking water, it is necessary to analyse the water. The major objective is to provide a system for continuous river water quality monitoring in remote locations using wireless sensor networks with low power, low cost, and high detection accuracy pH, conductivity, turbitity level, etc. Because current laboratory-based systems are too slow to increase effective reaction and do not provide the level of real-time public protection, there is a demand for online water monitoring systems.

.

2. LITERATURE SURVEY:

TITLE: Water Quality Monitoring for Rural Areas-ASensor Cloud Based Economical Project

AUTHOR: Nikhil Kedi

YEAR OF PUBLICATION: 2020

Published in the Dehradun, India, edition of the International Conference on Next Generation Computing Technologies. The complete water quality monitoring process, including the sensors, embedded design, information dissipation process, and roles of the government, network operator, and villages in guaranteeing proper information dissipation, are highlighted in this study. Additionally, the Sensor Cloud domain is explored. At this time, it is not possible to automatically enhance water quality, but effective use of technology and cost-effective business strategies can help. Community-Based Environmental Monitoring (CBM) is a social practise that contributes significantly to the creation of active societies and the management of the environment for a sustainable future.

TITLE: Real Time Water Quality Monitoring System”

AUTHOR: Jayti Bhatt, Jignesh Patoliya

YEAR OF PUBLICATION: 2022

This paper explains how real-time monitoring of water quality is necessary to assure the supply of safe drinking water. An innovative method based on the Internet of Things (IoT) has been suggested for this purpose. In this research, we demonstrate the architecture of an IOT-based system for real-time water quality monitoring. This system includes a number of sensors that detect several aspects of water quality, including temperature, conductivity, pH, and levels of dissolved oxygen and turbidity. The microcontroller processes the sensor-measured values before transmitting them over the Zigbee protocol to the raspberry pi, which serves as the core controller. Finally, you may view sensor data online.

TITLE: An IOT based Smart Water Quality Monitoring System using cloud

AUTHOR:AjithB

YEAR OF PUBLICATION: 2022

The Internet of Things (IoT) is a network of physical objects, including furniture, cars, appliances, and other goods, that are connected to one another and share data. These objects are embedded with electronics, software, sensors, actuators, and connectivity. Therefore, it is crucial to build and create a low-cost system employing the Internet of Things (IoT) for real-time water quality monitoring. By using the Internet of Things (IoT) to monitor water quality in water bodies, we can fight environmental problems and raise the health and living standards of all living things.

TITLE: River Water Quality Robot Embedded with Real-Time Monitoring System: Design and Implementation.

AUTHOR: Mohd Amirul Aizad M. Shahrani

YEAR OF PUBLICATION: 2021

In order to improve water quality, this research suggested an autonomous robot equipped with real-time multimodal (pH, temperature, voltage, and rubbish level). To track the water quality, the data were collected using sensors, sent over Wi-Fi to a mobile application created by an MIT inventor, and then stored in the cloud. The river water robot is also connected to an autonomous power source that runs on wind and sun energy. Based on the results, it was determined that the river water under test had a pH between 2 and 4.6, which is regarded to be extremely acidic. The proposed robot has demonstrated functionality in the real-time receiving and transmitting of data, to sum up.

2.1 EXISTING PROBLEM

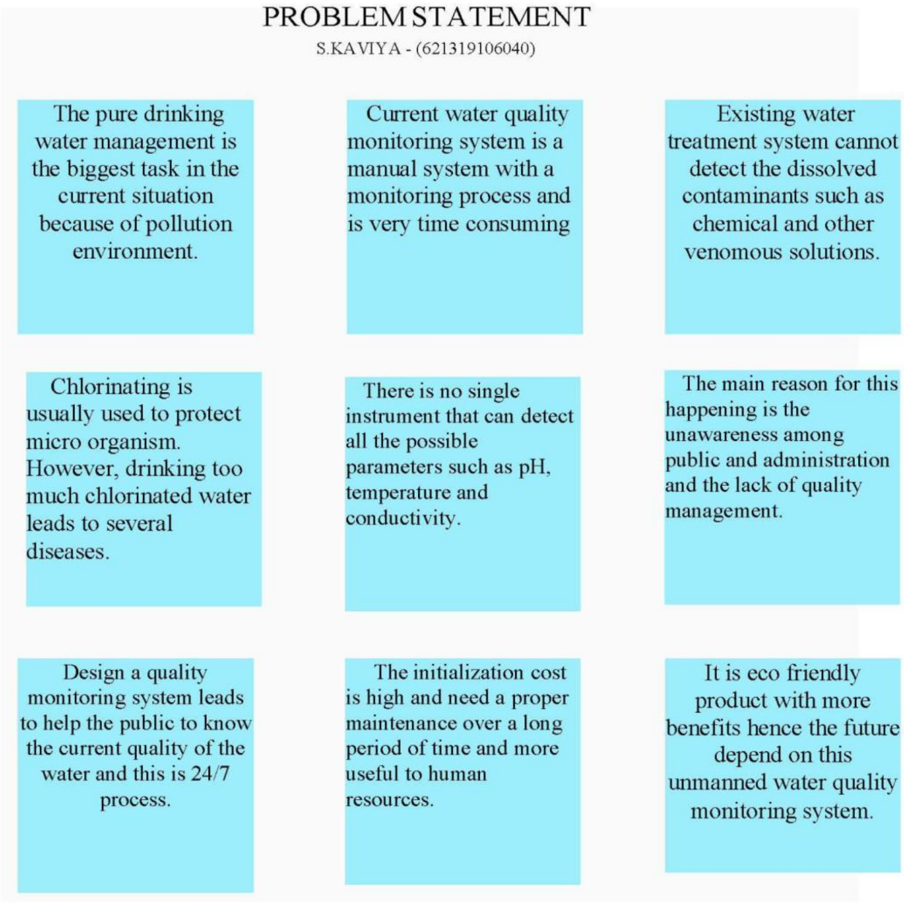
Water is an essential requirement for all living things, including humans. Alternative methods of acquiring water supply, like groundwater and surface water, will no longer be able to meet the increasing demand in water supply due to rapid development expansion. As a result of the anticipated water deficit in many emerging nations, this problem is predicted to worsen. Consequently, the current trend is to manage and monitor the available natural water supply in order to prevent water shortage issues during dry periods. Since it has the potential to replace manual monitoring, water monitoring technologies have made significant progress and are now widely used for the operation of water sources and water treatment facilities.

As a preliminary warning method, water monitoring is useful in disaster management in the eyes of consumers

2.2 REFERENCES

1. Zuraida Muhammad,Muhammad Azri Asyraf Mohd Hafez,Nor Adni Mat “Smart Agriculture Using Internet of Things with Raspberry Pi.” 2020.
2. Divya J., Divya M.,Janani V.”IoT based Smart Soil Monitoring System for Agricultural Production” 2017.
3. H.G.C.R.Laksiri, H.A.C.Dharmagunawardhana, J.V.Wijayakulasooriya “Design and Optimization of loT Based Smart Irrigation System in Sri Lanka”2019.
4. Anushree Math, Layak Ali, Pruthviraj U “Development of Smart Drip Irrigation System Using IoT”2018.
5. Shrihari M, “A Smart Wireless System to Automate Production of Crops and Stop Intrusion Using Deep Learning” 2020.
6. G. Sushanth1, and S. Sujatha, “IOT Based Smart Agriculture System”2018.
7. Dweepayan Mishra1, Arzeena Khan2 Rajeev Tiwari3 , Shuchi Upadhay, “Automated Irrigation System-IoT Based Approach”,2018.

2.3 PROBLEM STATEMENT DEFINITION



PROBLEM STATEMENT

S.KAVIYA - (621319106040)

The pure drinking water management is the biggest task in the current situation because of pollution environment.

Current water quality monitoring system is a manual system with a monitoring process and is very time consuming

Existing water treatment system cannot detect the dissolved contaminants such as chemical and other venomous solutions.

Chlorinating is usually used to protect micro organism. However, drinking too much chlorinated water leads to several diseases.

There is no single instrument that can detect all the possible parameters such as pH, temperature and conductivity.

The main reason for this happening is the unawareness among public and administration and the lack of quality management.

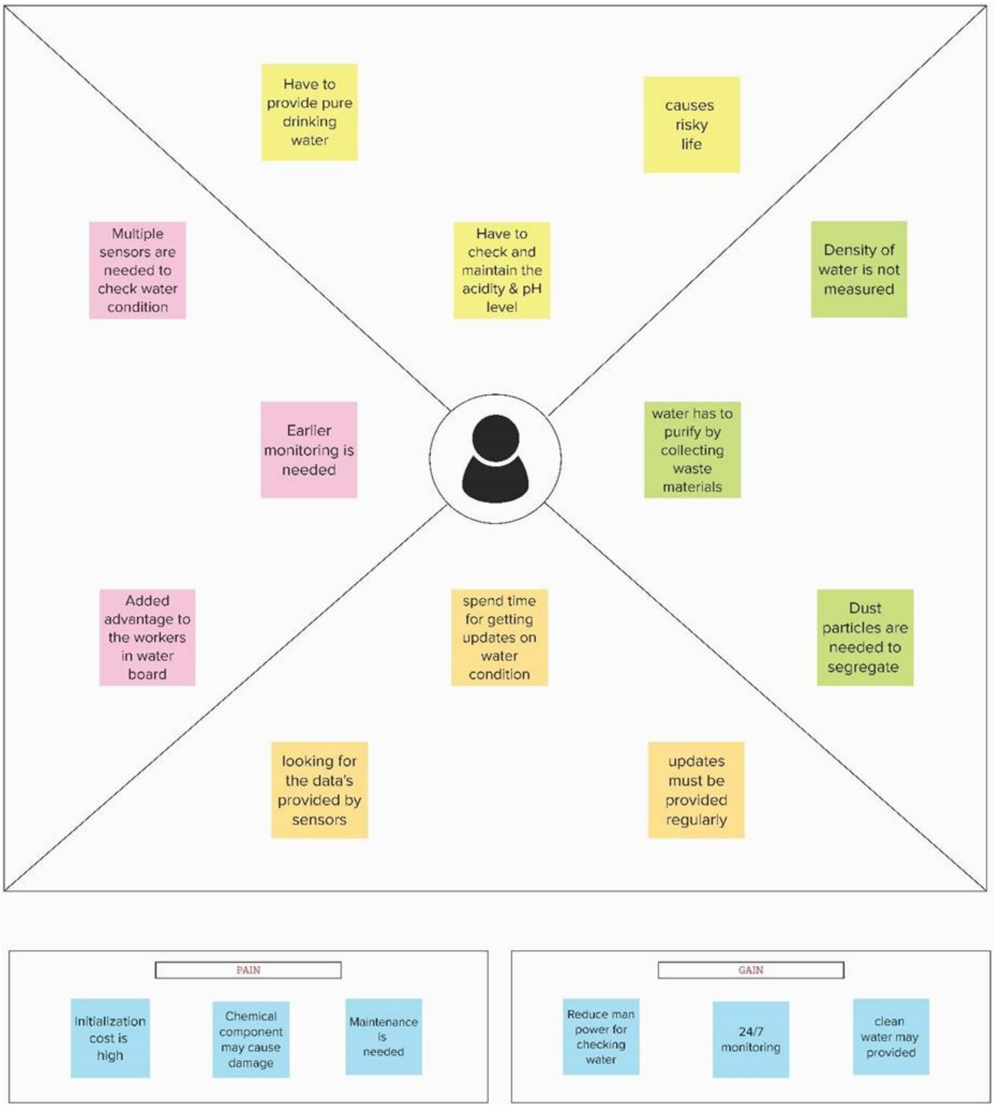
Design a quality monitoring system leads to help the public to know the current quality of the water and this is 24/7 process.

The initialization cost is high and need a proper maintenance over a long period of time and more useful to human resources.

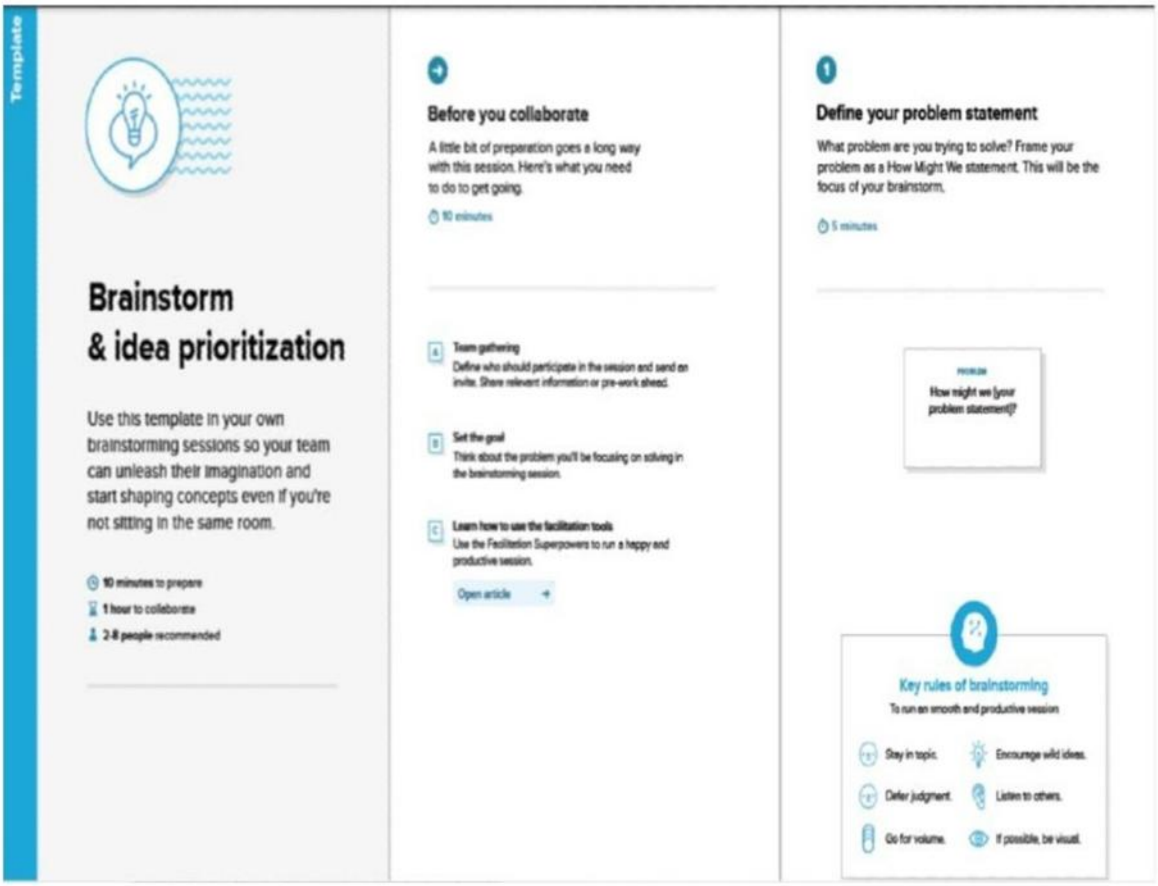
It is eco friendly product with more benefits hence the future depend on this unmanned water quality monitoring system.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS


Have to provide pure drinking water
causes risky life
Multiple sensors are needed to check water condition
Have to check and maintain the acidity & pH level
Density of water is not measured
Earlier monitoring is needed
water has to purify by collecting waste materials
Added advantage to the workers in water board
spend time for getting updates on water condition
Dust particles are needed to segregate
looking for the data's provided by sensors
updates must be provided regularly
PAIN
Initialization cost is high
Chemical component may cause damage
Maintenance is needed
GAIN
Reduce man power for checking water
24/7 monitoring
clean water may provided

IDEATION & BRAINSTORMING



Template

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-8 people recommended

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

C Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

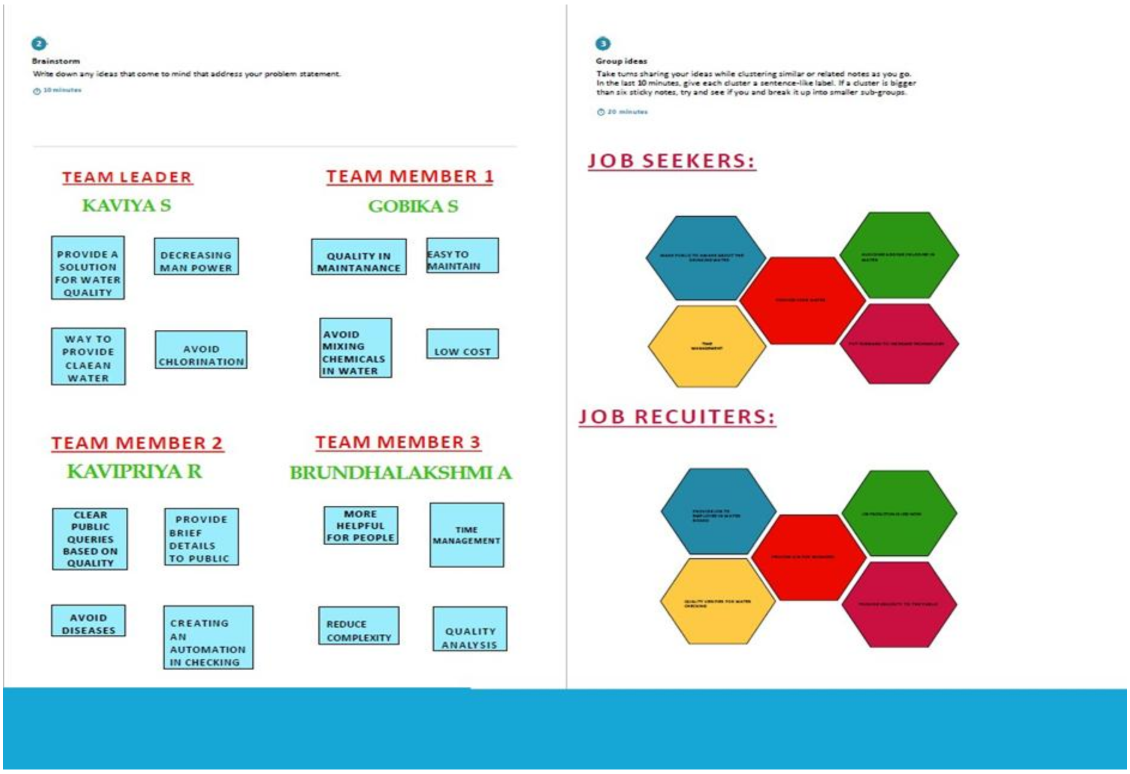
5 minutes

PROBLEM
How might we [your problem statement]?

Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TEAM LEADER

KAVIYA S

PROVIDE A SOLUTION FOR WATER QUALITY

DECREASING MAN POWER

WAY TO PROVIDE CLAEAN WATER

AVOID CHLORINATION

TEAM MEMBER 1

GOBIKA S

QUALITY IN MAINTANANCE

EASY TO MAINTAIN

AVOID MIXING CHEMICALS IN WATER

LOW COST

TEAM MEMBER 2

KAVIPRIYA R

CLEAR PUBLIC QUERIES BASED ON QUALITY

PROVIDE BRIEF DETAILS TO PUBLIC

AVOID DISEASES

CREATING AN AUTOMATION IN CHECKING

TEAM MEMBER 3

BRUNDHALAKSHMI A

MORE HELPFUL FOR PEOPLE

TIME MANAGEMENT

REDUCE COMPLEXITY

QUALITY ANALYSIS

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

JOB SEEKERS:

MAKE PUBLIC TO AWARE ABOUT THE DRINKING WATER

[illegible]

PROVIDE SAFE WATER

TIME MANAGEMENT

[illegible]

JOB RECUITERS:

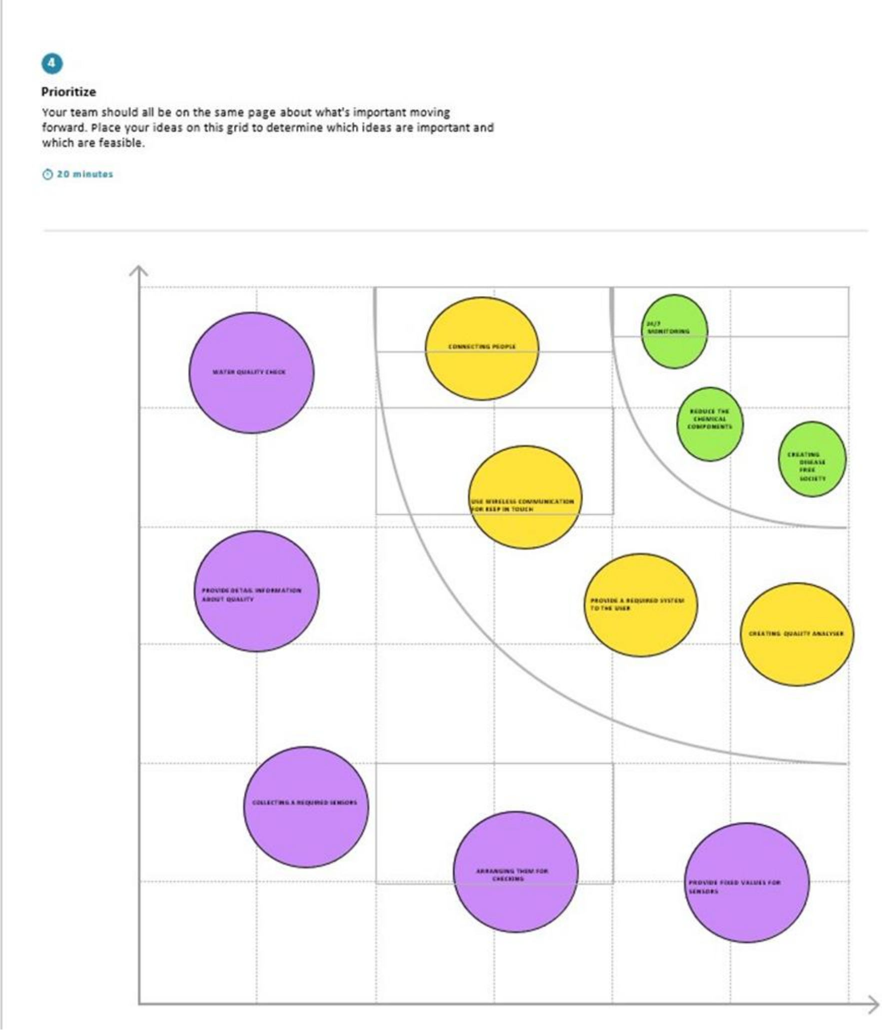
[illegible]

[illegible]

[illegible]

QUALITY CENTER FOR WATER CHECKING

[illegible]



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

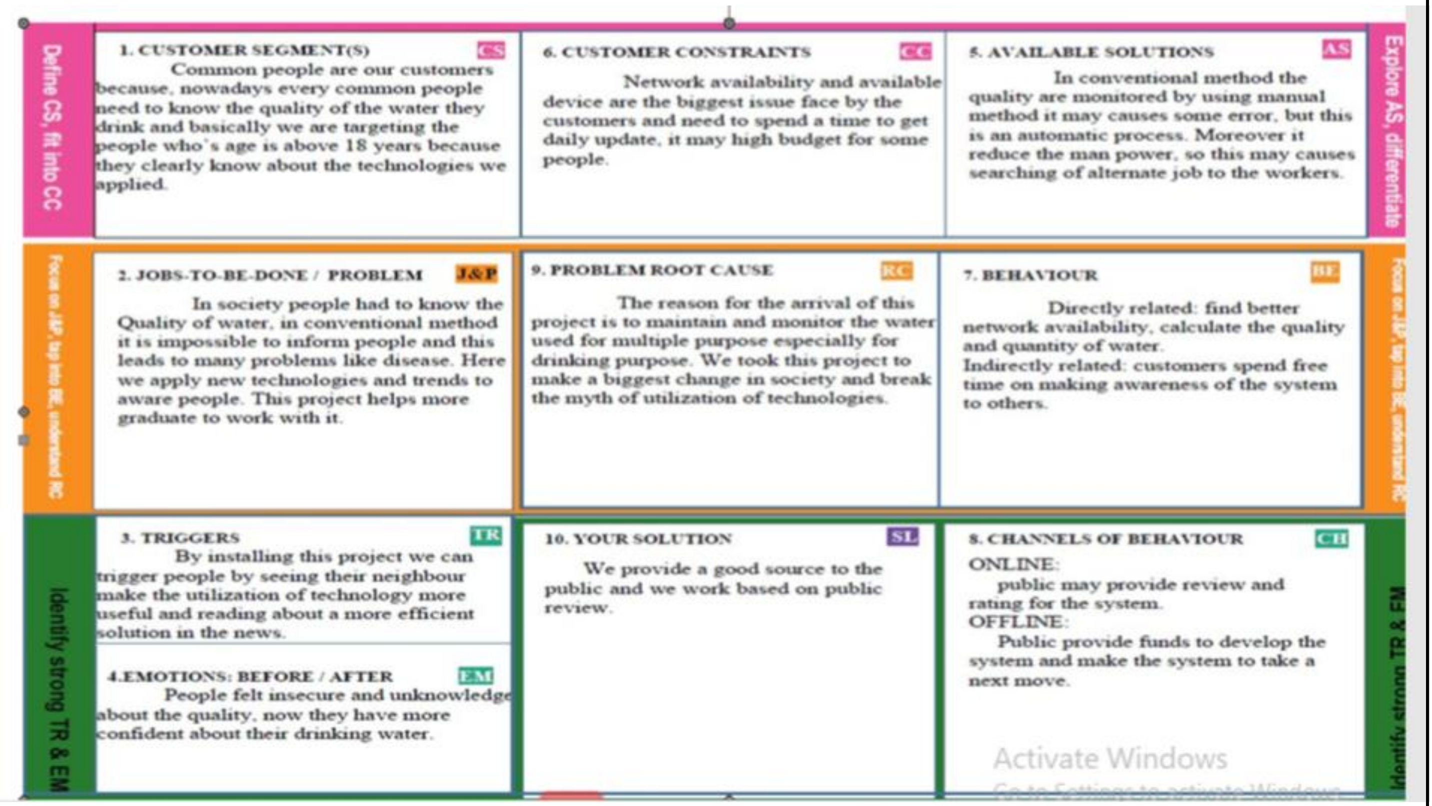
20 minutes

WATER QUALITY CHECK
CONNECTING PEOPLE
24/7 MONITORING
REDUCE THE CHEMICAL COMPONENTS
CREATING DISEASE FREE SOCIETY
USE WIRELESS COMMUNICATION FOR KEEP IN TOUCH
PROVIDE DETAIL INFORMATION ABOUT QUALITY
PROVIDE A REQUIRED SYSTEM TO THE USER
CREATING QUALITY ANALYSER
COLLECTING A REQUIRED SENSORS
ARRANGING THEM FOR CHECKING
PROVIDE FIXED VALUES FOR SENSORS

3.3 PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
1	Problem Statement (Problem tobe solved)	A water quality management system helps to check the quality of water which include temperature, humidity and pH in real time and more helpful for human resource.
2	Idea / Solution description	The idea for this project basically contain sensor for detection of water quality and provide pure water for the public in good condition.
3	. Novelty / Uniqueness	The uniqueness of the project is, it contain high quality sensor with high sensitivity and low cost with multiple use and it provide high quality water.
4	Social Impact / CustomerSatisfaction	Even-though, it reduces the manpower it help more graduates to work on this project and people can more aware about the latest trends and technologies.
5	Business Model (RevenueModel)	It is more profitable and simple model to manufacture. In business model it provide high revenue with low investment
6	Scalability of the Solution	It can withstand over a long period of time, easily usable product and utilization of more technologies.

3.4 PROBLEM SOLUTION FIT


Define CS, fit into CC
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM

1. CUSTOMER SEGMENT(S) CS

Common people are our customers because, nowadays every common people need to know the quality of the water they drink and basically we are targeting the people who's age is above 18 years because they clearly know about the technologies we applied.

6. CUSTOMER CONSTRAINTS CC

Network availability and available device are the biggest issue face by the customers and need to spend a time to get daily update, it may high budget for some people.

5. AVAILABLE SOLUTIONS AS

In conventional method the quality are monitored by using manual method it may causes some error, but this is an automatic process. Moreover it reduce the man power, so this may causes searching of alternate job to the workers.

2. JOBS-TO-BE-DONE / PROBLEM J&P

In society people had to know the Quality of water, in conventional method it is impossible to inform people and this leads to many problems like disease. Here we apply new technologies and trends to aware people. This project helps more graduate to work with it.

9. PROBLEM ROOT CAUSE RC

The reason for the arrival of this project is to maintain and monitor the water used for multiple purpose especially for drinking purpose. We took this project to make a biggest change in society and break the myth of utilization of technologies.

7. BEHAVIOUR BE

Directly related: find better network availability, calculate the quality and quantity of water.
Indirectly related: customers spend free time on making awareness of the system to others.

3. TRIGGERS TR

By installing this project we can trigger people by seeing their neighbour make the utilization of technology more useful and reading about a more efficient solution in the news.

4.EMOTIONS: BEFORE / AFTER EM

People felt insecure and unknowledge about the quality, now they have more confident about their drinking water.

10. YOUR SOLUTION SL

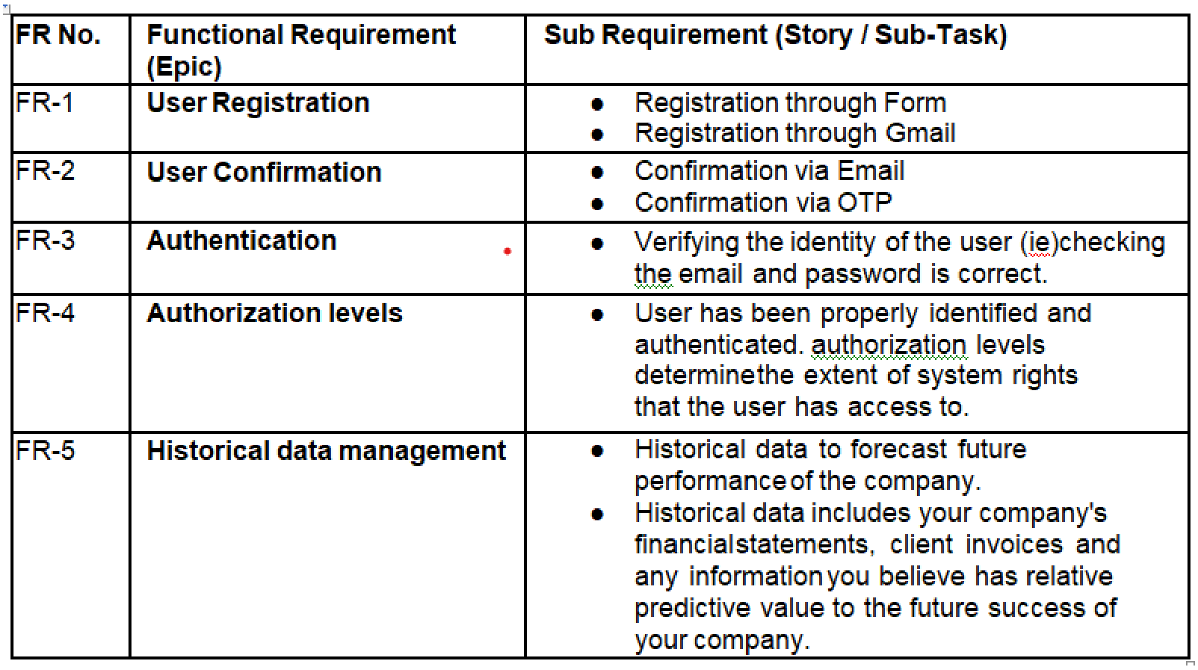
We provide a good source to the public and we work based on public review.

8. CHANNELS OF BEHAVIOUR CH

ONLINE:
public may provide review and rating for the system.
OFFLINE:
Public provide funds to develop the system and make the system to take a next move.

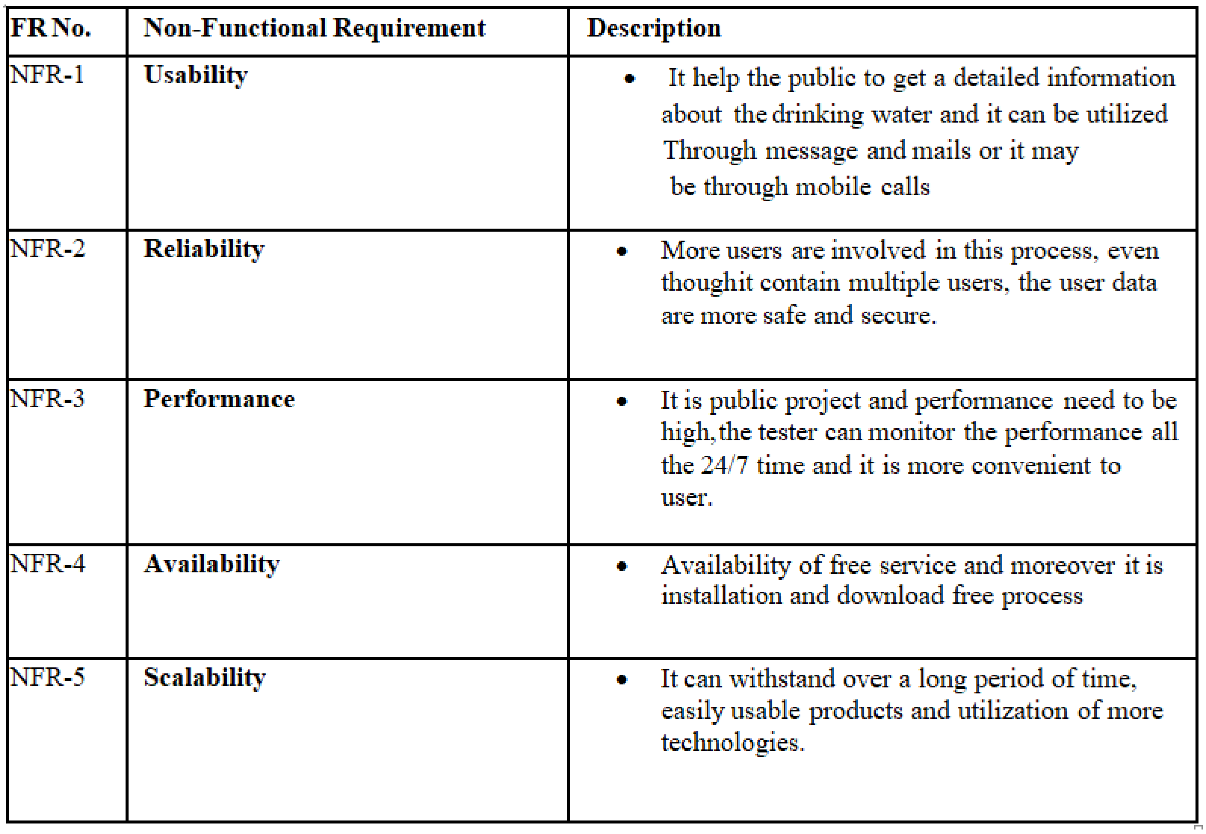
4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	● Registration through Form ● Registration through Gmail
FR-2	**User Confirmation**	● Confirmation via Email ● Confirmation via OTP
FR-3	**Authentication**	● Verifying the identity of the user (ie)checking the email and password is correct.
FR-4	**Authorization levels**	● User has been properly identified and authenticated. authorization levels determinethe extent of system rights that the user has access to.
FR-5	**Historical data management**	● Historical data to forecast future performance of the company. ● Historical data includes your company's financialstatements, client invoices and any informationyou believe has relative predictive value to the future success of your company.

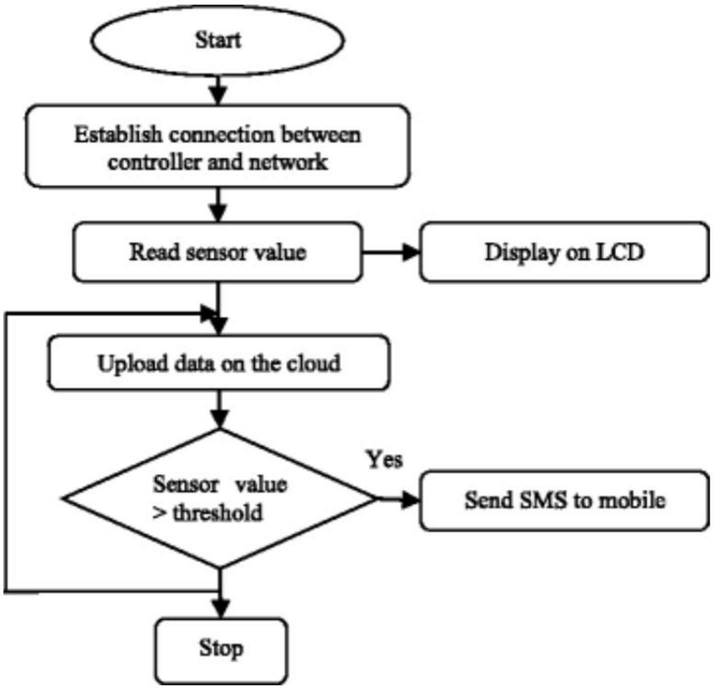
4.2 NON-FUNCTIONAL REQUIREMENTS



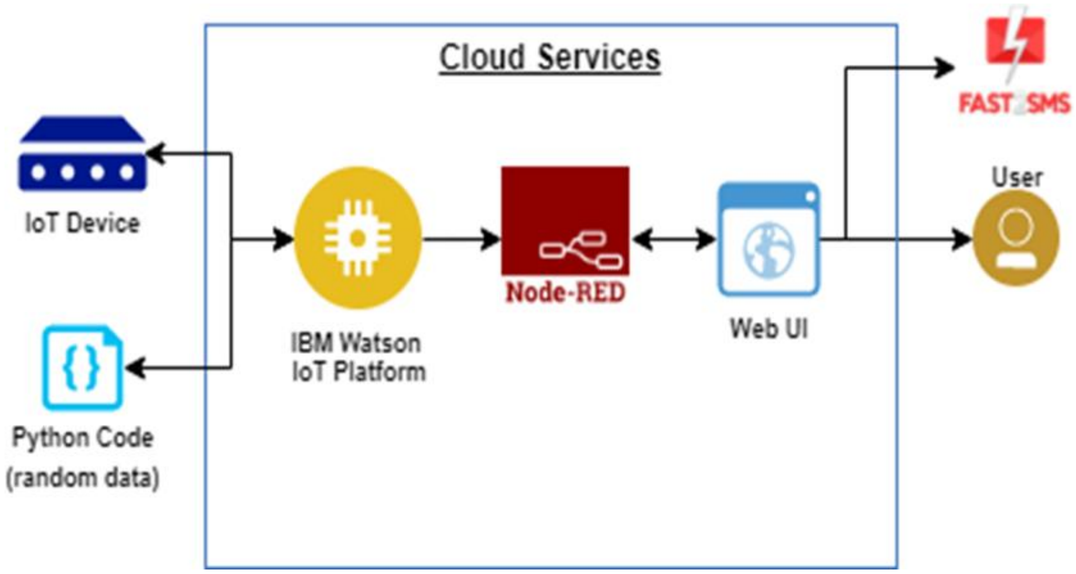
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	• It help the public to get a detailed information about the drinking water and it can be utilized Through message and mails or it may be through mobile calls
NFR-2	**Reliability**	• More users are involved in this process, even thoughit contain multiple users, the user data are more safe and secure.
NFR-3	**Performance**	• It is public project and performance need to be high, the tester can monitor the performance all the 24/7 time and it is more convenient to user.
NFR-4	**Availability**	• Availability of free service and moreover it is installation and download free process
NFR-5	**Scalability**	• It can withstand over a long period of time, easily usable products and utilization of more technologies.

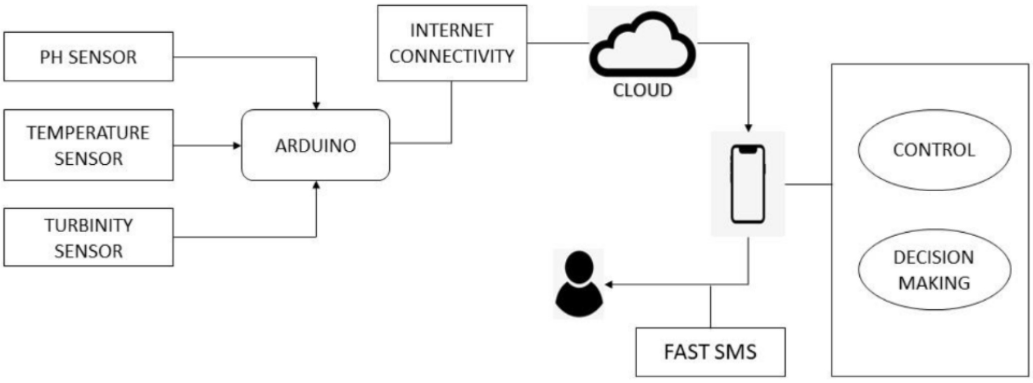
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

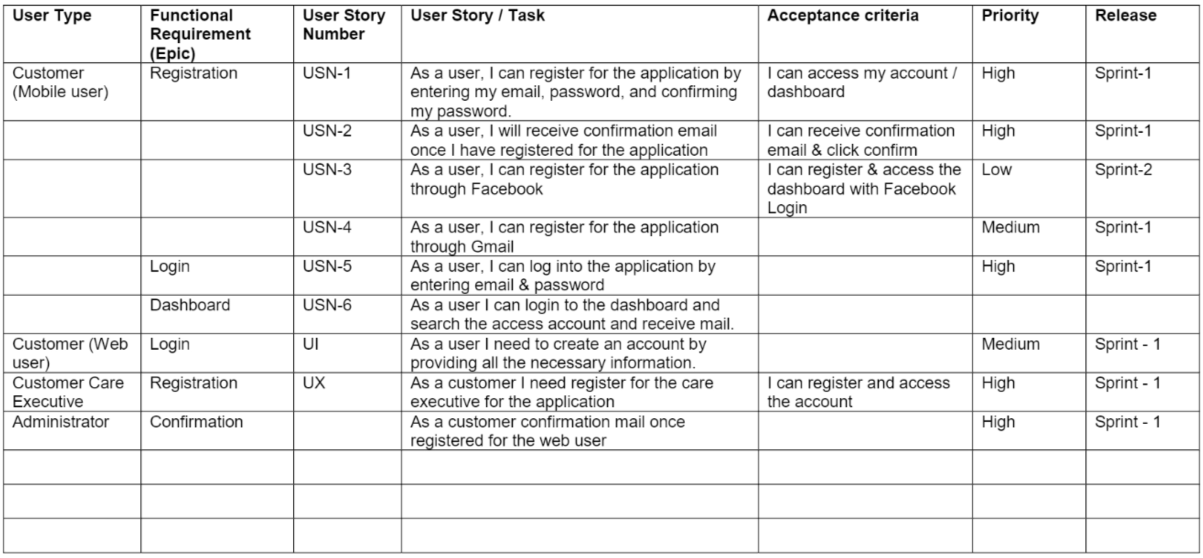


5.2 SOLUTION AND TECHNICAL ARCHITECTURE





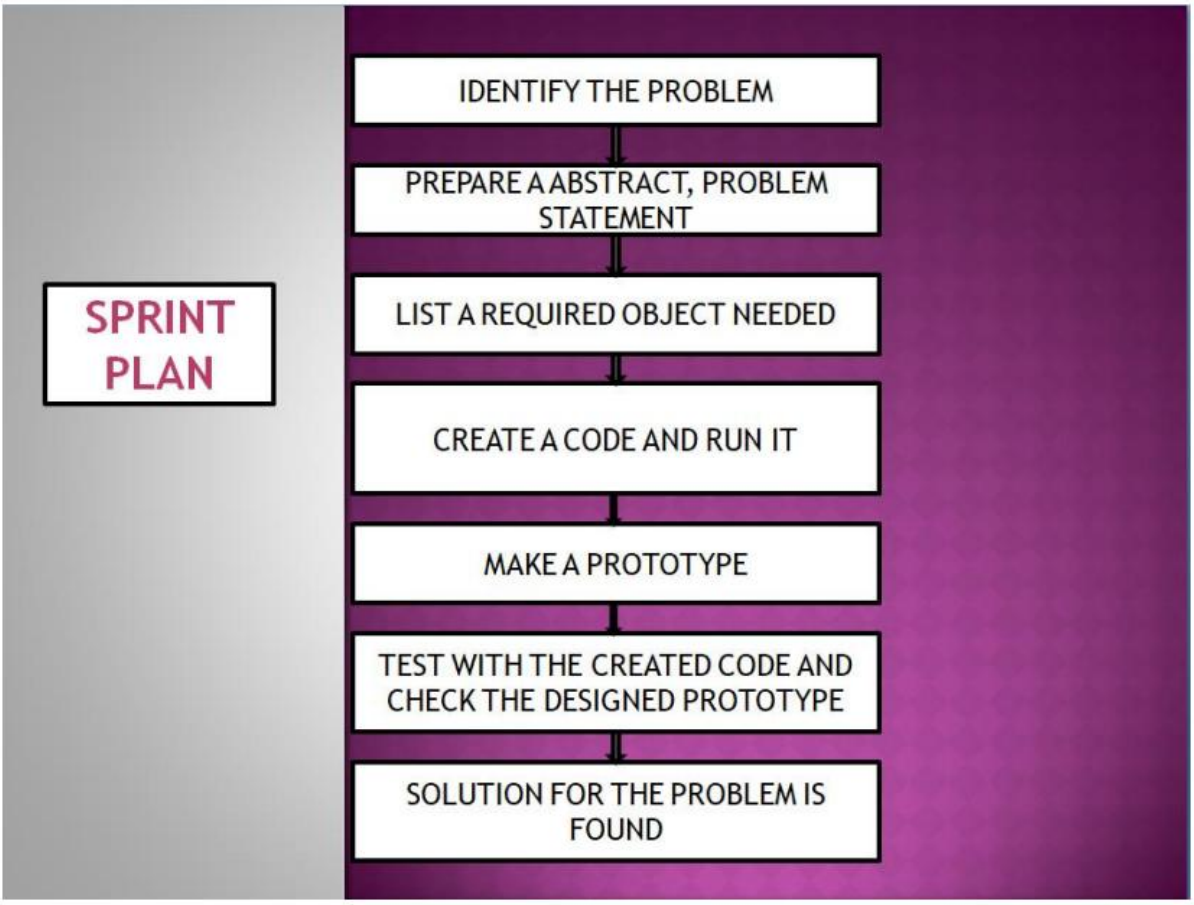
5.3 USER STORIES



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		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user I can login to the dashboard and search the access account and receive mail.			
Customer (Web user)	Login	UI	As a user I need to create an account by providing all the necessary information.		Medium	Sprint - 1
Customer Care Executive	Registration	UX	As a customer I need register for the care executive for the application	I can register and access the account	High	Sprint - 1
Administrator	Confirmation		As a customer confirmation mail once registered for the web user		High	Sprint - 1

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION


SPRINT PLAN
IDENTIFY THE PROBLEM
PREPARE A ABSTRACT, PROBLEM STATEMENT
LIST A REQUIRED OBJECT NEEDED
CREATE A CODE AND RUN IT
MAKE A PROTOTYPE
TEST WITH THE CREATED CODE AND CHECK THE DESIGNED PROTOTYPE
SOLUTION FOR THE PROBLEM IS FOUND

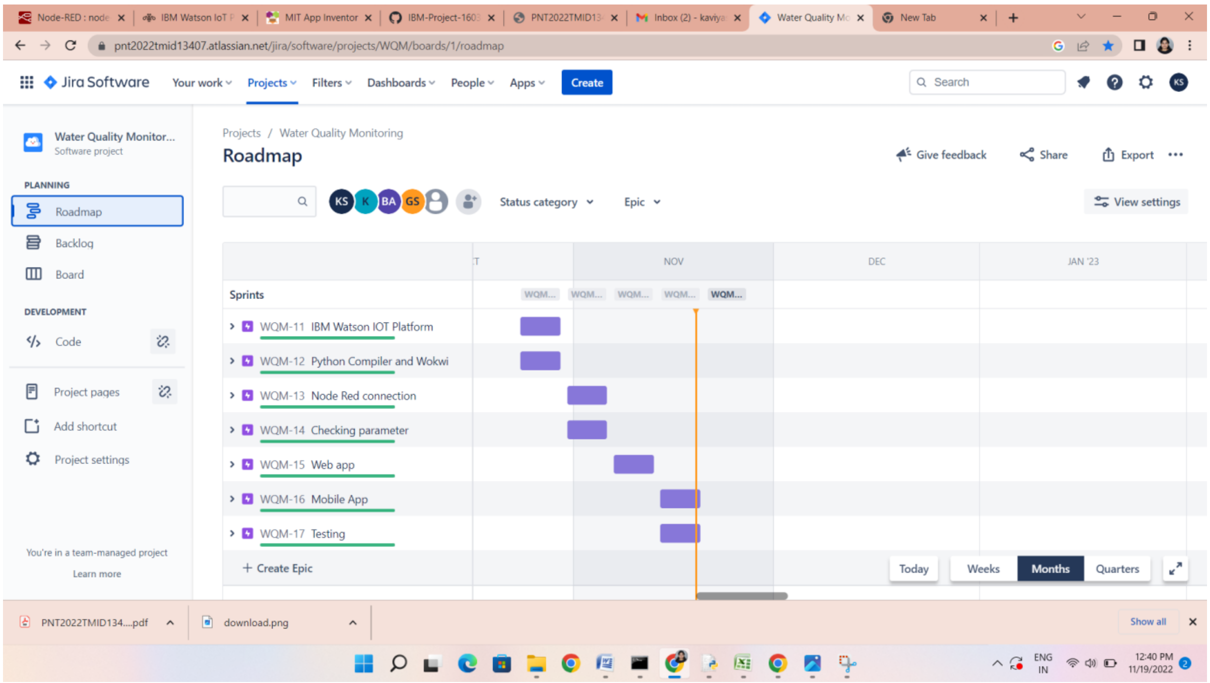
6.2 SPRINT DELIVERY SCHEDULE

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	IBM Watson IOT Platform	USN-1	Get into IBM cloud and login IBM Watson, there we create Device type, device Id along with organization ID, API tokens for performing cloud process.	1	High	Gobika S
Sprint-1	Python Compiler and Wokwi	USN-2	Using either python code or Wokwi the cloud get connected and output are shown in the IBM platform	1	Medium	Brundhalakshmi A
Sprint-2	Node Red connection	USN-3	The node red blocks are connected to provide an HTTP link there it provide dashboards for getting graphs for easy understanding	1	High	Kaviya S
Sprint-2	Checking parameter	USN-4	The parameters like pH, temperature, humidity has to be checked by using Python, Wokwi and Node red	1	Medium	Kavipriya R
Sprint-3	Web app	USN-5	The web app is created first by using node red http request and outputs are checked there	1	High	Brundhalakshmi A
Sprint-4	Mobile App	USN-6	The mobile app is developed by using MIT app the user get into the application and find the status at real time.	1	Medium	Kavipriya R
Sprint-4	Testing	USN-7	Every sprint provide output and they are tested and merged to get an final output	1	High	Kaviya S

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

Average Velocity (AV)=7/2=3.5

6.3 REPORTS FROM JIRA FILES



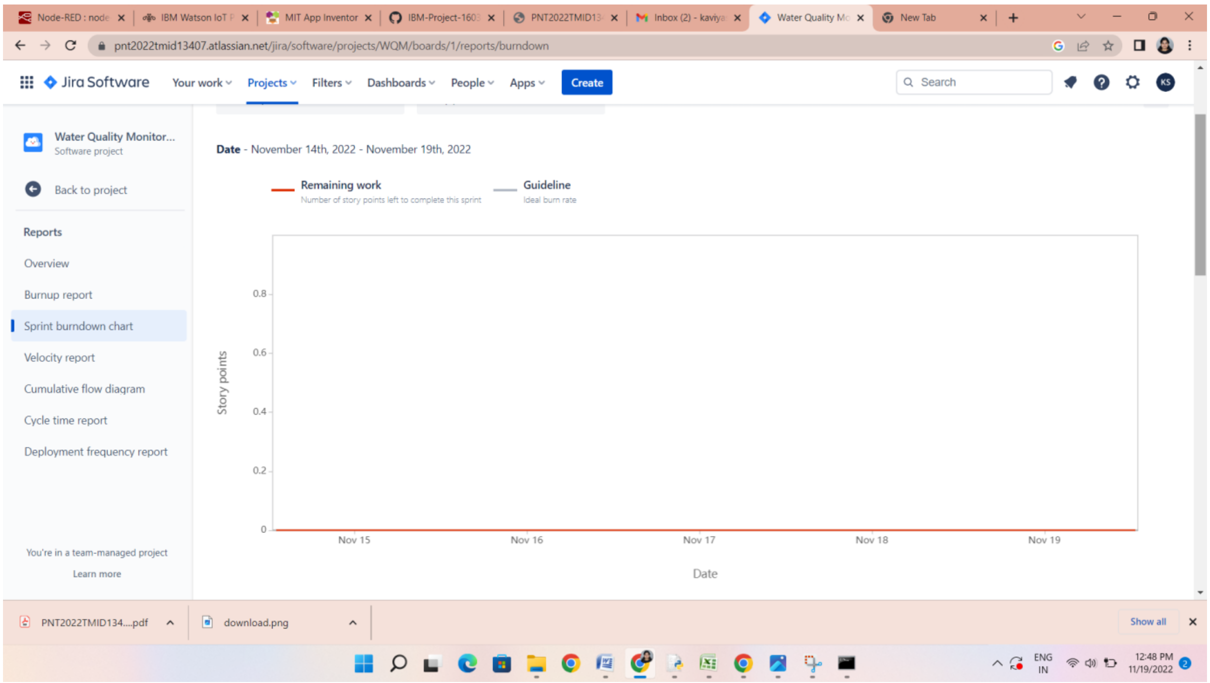
Projects / Water Quality Monitoring

Roadmap

Sprints

- WQM-11 IBM Watson IOT Platform
- WQM-12 Python Compiler and Wokwi
- WQM-13 Node Red connection
- WQM-14 Checking parameter
- WQM-15 Web app
- WQM-16 Mobile App
- WQM-17 Testing

+ Create Epic

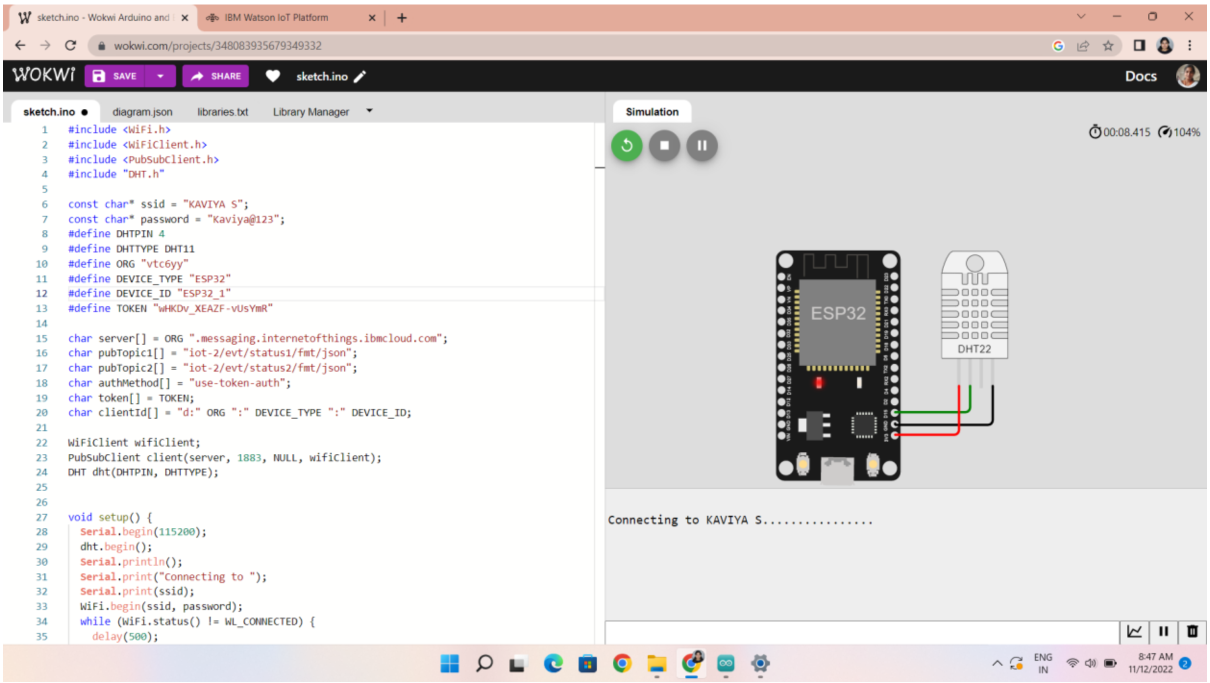


Date - November 14th, 2022 - November 19th, 2022

7. CODING & SOLUTIONING

7.1 Feature 1

The main feature of the project is measurement of pH, temperature and humidity by using graph, table and boards and here Wokwi is used for sample code.



```
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>
#include "DHT.h"

const char* ssid = "KAVIYA S";
const char* password = "Kaviya@123";
#define DHTPIN 4
#define DHTTYPE DHT11
#define ORG "vtc6yy"
#define DEVICE_TYPE "ESP32"
#define DEVICE_ID "ESP32_1"
#define TOKEN "wHKDv_XEAZF-vUsYmR"

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1[] = "iot-2/evt/status1/fmt/json";
char pubTopic2[] = "iot-2/evt/status2/fmt/json";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);
DHT dht(DHTPIN, DHTTYPE);


void setup() {
  Serial.begin(115200);
  dht.begin();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
```

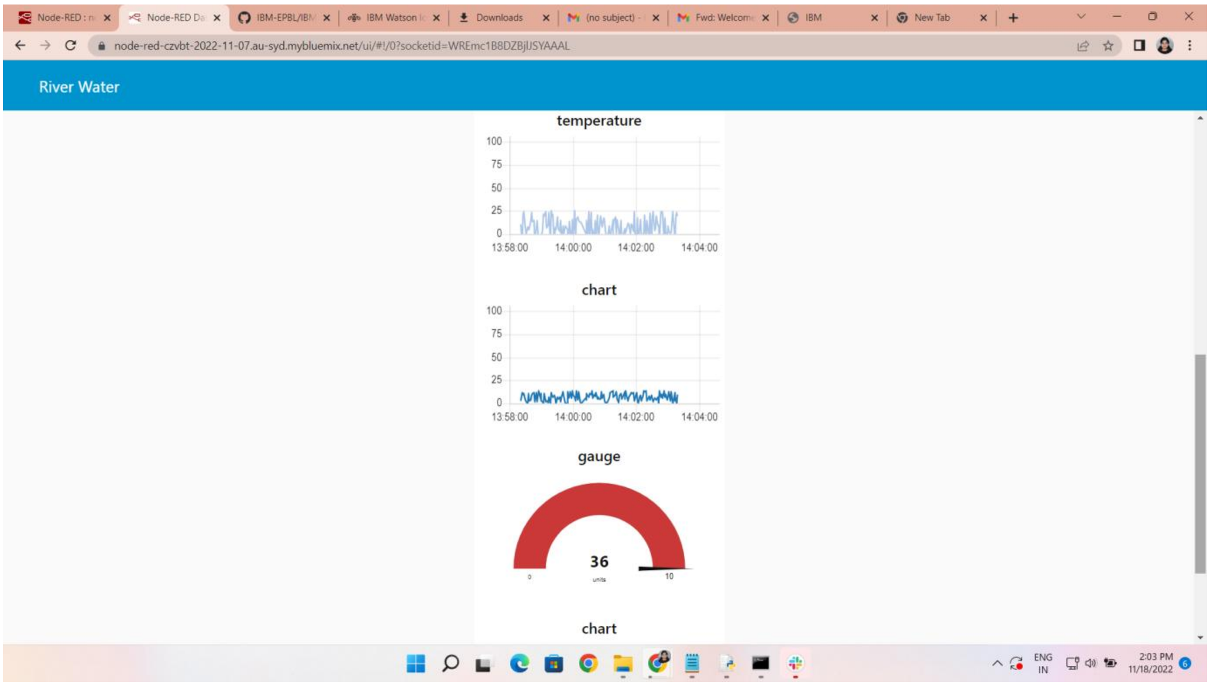
```
Connecting to KAVIYA S................
```

CODE:

```
#include <WiFi.h>

#include <WiFiClient.h>

#include <PubSubClient.h>

#include "DHT.h"


const char* ssid = "KAVIYA S";

const char* password = "Kaviya@123";

#define DHTPIN 4

#define DHTTYPE DHT11

#define ORG "vtc6yy"

#define DEVICE_TYPE "ESP32"

#define DEVICE_ID "1002"

#define TOKEN "wHKDv_XEAZF-vUsYmR"


char server[] = ORG ".messaging.internetofthings.ibmcloud.com";

char pubTopic1[] = "iot-2/evt/status1/fmt/json";

char pubTopic2[] = "iot-2/evt/status2/fmt/json";

char authMethod[] = "use-token-auth";

char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;


WiFiClient wifiClient;

PubSubClient client(server, 1883, NULL, wifiClient);

DHT dht(DHTPIN, DHTTYPE);


void setup() {

  Serial.begin(115200);

  dht.begin();

  Serial.println();
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");

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

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

long lastMsg = 0;
void loop() {
  client.loop();
  long now = millis();
  if (now - lastMsg > 3000) {
    lastMsg = now;
    float humidity = dht.readHumidity();
    float temperature = dht.readTemperature();
    String payload = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
    payload += ",\"temperature\":";
    payload += temperature;
    payload += "}}";

    Serial.print("Sending payload: ");
    Serial.println(payload);

    if (client.publish(pubTopic1, (char*) payload.c_str())) {
      Serial.println("Publish ok");
    } else {
      Serial.println("Publish failed");
    }
    String payload1 = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
    payload1 += ",\"humidity\":";
    payload1 += humidity;
    payload1 += "}}";

    if (client.publish(pubTopic2, (char*) payload1.c_str())) {
      Serial.println("Publish ok");
    } else {
      Serial.println("Publish failed");
    }
  }
}
```

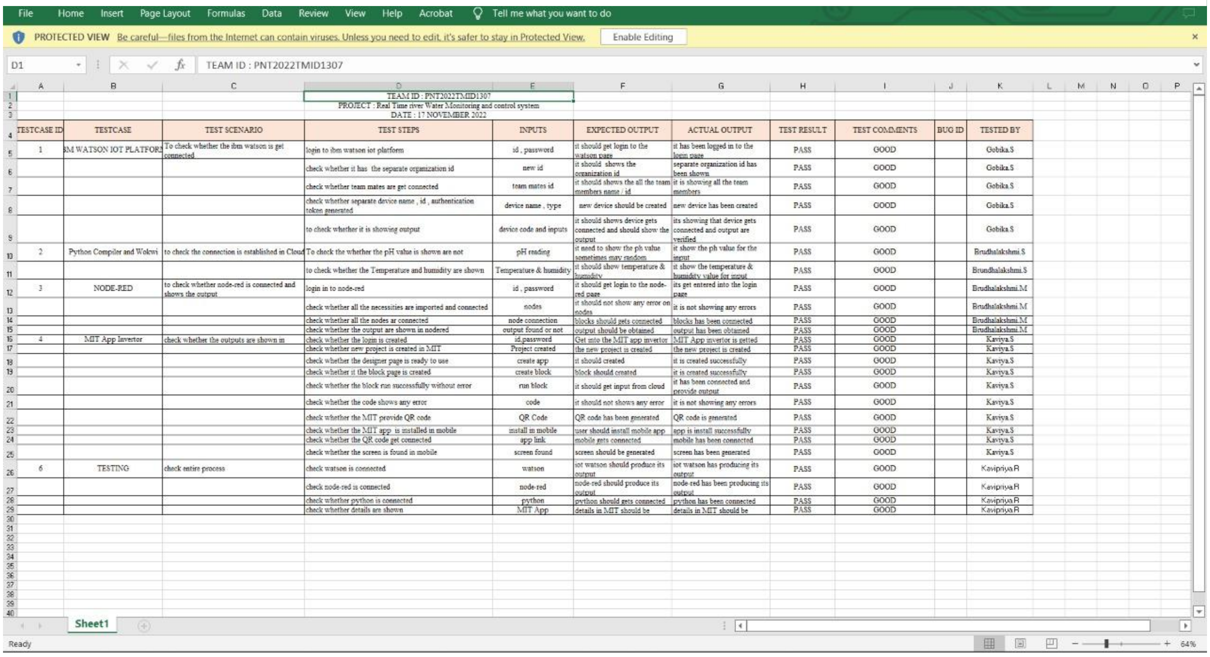
7.2 FEATURE 2

It contain the working of pH, Humidity and temperature in mobile application and python code is used for that implementation.



8. TESTING

8.1 TEST CASES


File
Home
Insert
Page Layout
Formulas
Data
Review
View
Help
Acrobat
Tell me what you want to do
PROTECTED VIEW
Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View.
Enable Editing
D1
Sheet1
Ready
64%

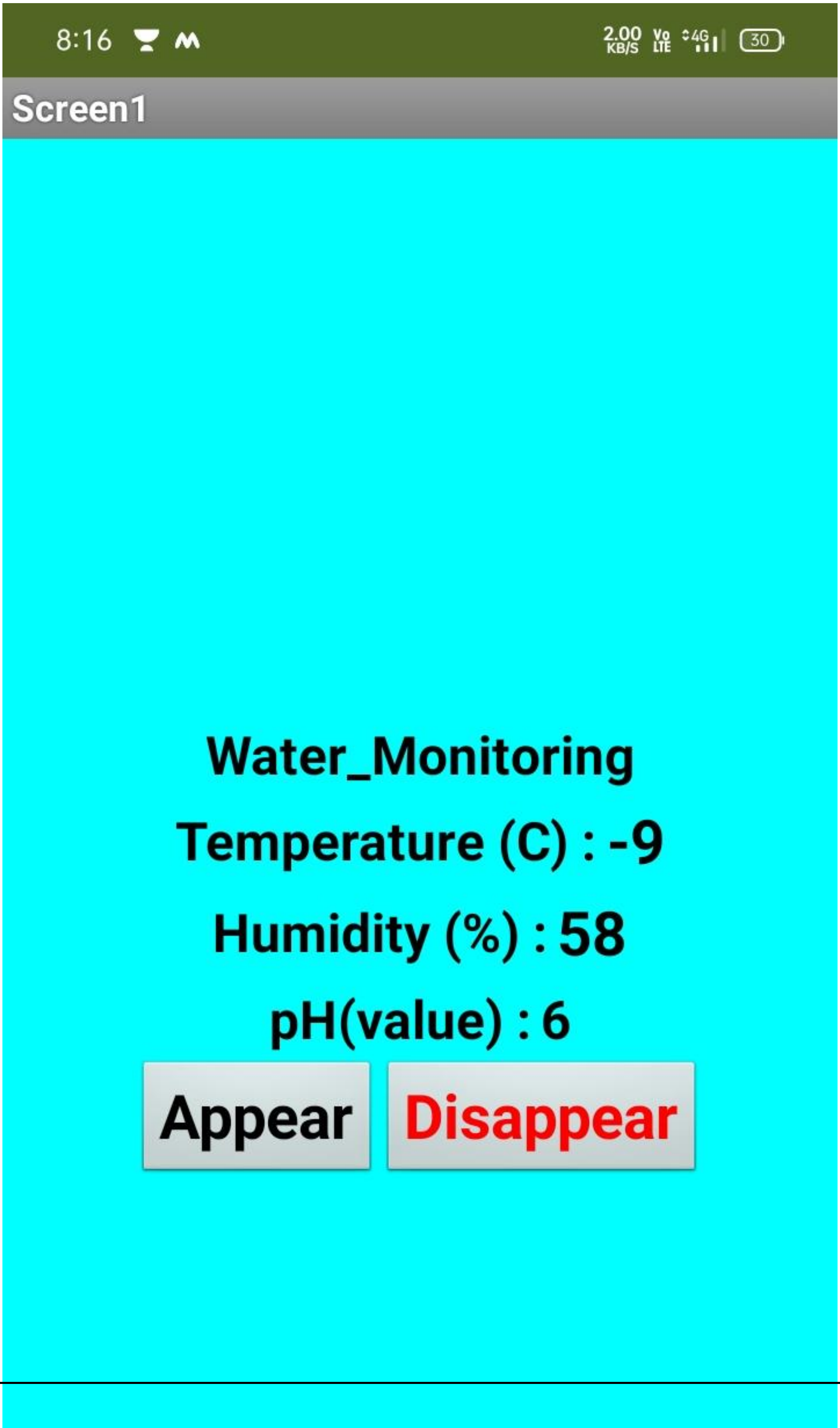
TEAM ID : PNT2022TMID1307

PROJECT : Real Time river Water Monitoring and control system

DATE : 17 NOVEMBER 2022

TESTCASE ID	TESTCASE	TEST SCENARIO	TEST STEPS	INPUTS	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT	TEST COMMENTS	BUG ID	TESTED BY
1	IBM WATSON IOT PLATFORM	To check whether the ibm watson is get connected	login to ibm watson iot platform	id , password	it should get login to the watson page	it has been logged in to the login page	PASS	GOOD		Gobika.S
			check whether it has the separate organization id	new id	it should shows the organization id	separate organization id has been shown	PASS	GOOD		Gobika.S
			check whether team mates are get connected	team mates id	it should shows the all the team members name / id	it is showing all the team members	PASS	GOOD		Gobika.S
			check whether separate device name , id , authentication token generated	device name , type	new device should be created	new device has been created	PASS	GOOD		Gobika.S
			to check whether it is showing output	device code and inputs	it should shows device gets connected and should show the output	its showing that device gets connected and output are verified	PASS	GOOD		Gobika.S
2	Python Compiler and Wokwi	to check the connection is established in Cloud	To check the whether the pH value is shown are not	pH reading	it need to show the ph value sometimes may random	it show the ph value for the input	PASS	GOOD		Brudhalakshmi.S
			to check whether the Temperature and humidity are shown	Temperature & humidity	it should show temperature & humidity	it show the temperature & humidity value for input	PASS	GOOD		Brundhalakshmi.S
3	NODE-RED	to check whether node-red is connected and shows the output	login in to node-red	id , password	it should get login to the node-red page	its get entered into the login page	PASS	GOOD		Brudhalakshmi.M
			check whether all the necessities are imported and connected	nodes	it should not show any error on nodes	it is not showing any errors	PASS	GOOD		Brudhalakshmi.M
			check whether all the nodes ar connected	node connection	blocks should gets connected	blocks has been connected	PASS	GOOD		Brudhalakshmi.M
			check whether the output are shown in nodered	output found or not	output should be obtained	output has been obtained	PASS	GOOD		Brudhalakshmi.M
4	MIT App Invertor	check whether the outputs are shown in	check whether the login is created	id,password	Get into the MIT app invertor	MIT App invertor is getted	PASS	GOOD		Kaviya.S
			check whether new project is created in MIT	Project created	the new project is created	the new project is created	PASS	GOOD		Kaviya.S
			check whether the designer page is ready to use	create app	it should created	it is created successfully	PASS	GOOD		Kaviya.S
			check whether it the block page is created	create block	block should created	it is created successfully	PASS	GOOD		Kaviya.S
			check whether the block run successfully without error	run block	it should get input from cloud	it has been connected and provide output	PASS	GOOD		Kaviya.S
			check whether the code shows any error	code	it should not shows any error	it is not showing any errors	PASS	GOOD		Kaviya.S
			check whether the MIT provide QR code	QR Code	QR code has been generated	QR code is generated	PASS	GOOD		Kaviya.S
			check whether the MIT app is installed in mobile	install in mobile	user should install mobile app	app is install successfully	PASS	GOOD		Kaviya.S
			check whether the QR code get connected	app link	mobile gets connected	mobile has been connected	PASS	GOOD		Kaviya.S
			check whether the screen is found in mobile	screen found	screen should be generated	screen has been generated	PASS	GOOD		Kaviya.S
6	TESTING	check entire process	check watson is connected	watson	iot watson should produce its output	iot watson has producing its output	PASS	GOOD		Kavipriya.R
			check node-red is connected	node-red	node-red should produce its output	node-red has been producing its output	PASS	GOOD		Kavipriya.R
			check whether python is connected	python	python should gets connected	python has been connected	PASS	GOOD		Kavipriya.R
			check whether details are shown	MIT App	details in MIT should be	details in MIT should be	PASS	GOOD		Kavipriya.R

8.2 User Acceptance testing



Water_Monitoring

Temperature (C) : -9

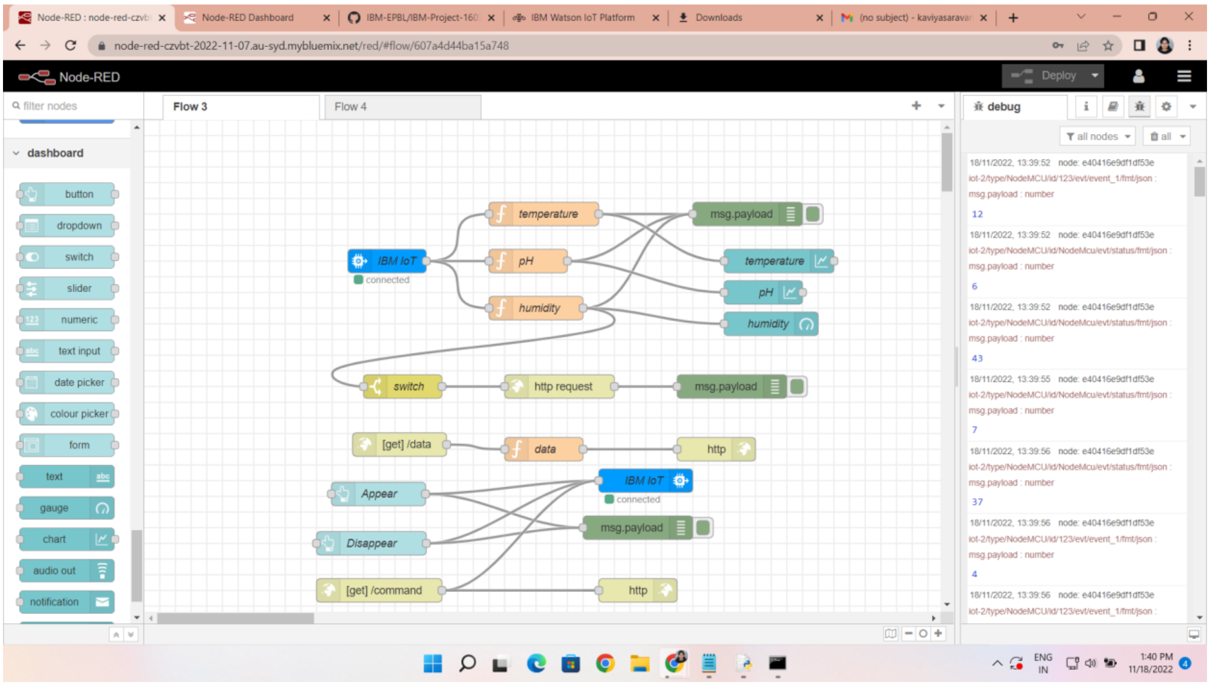
Humidity (%) : 58

pH(value) : 6

9. RESULTS

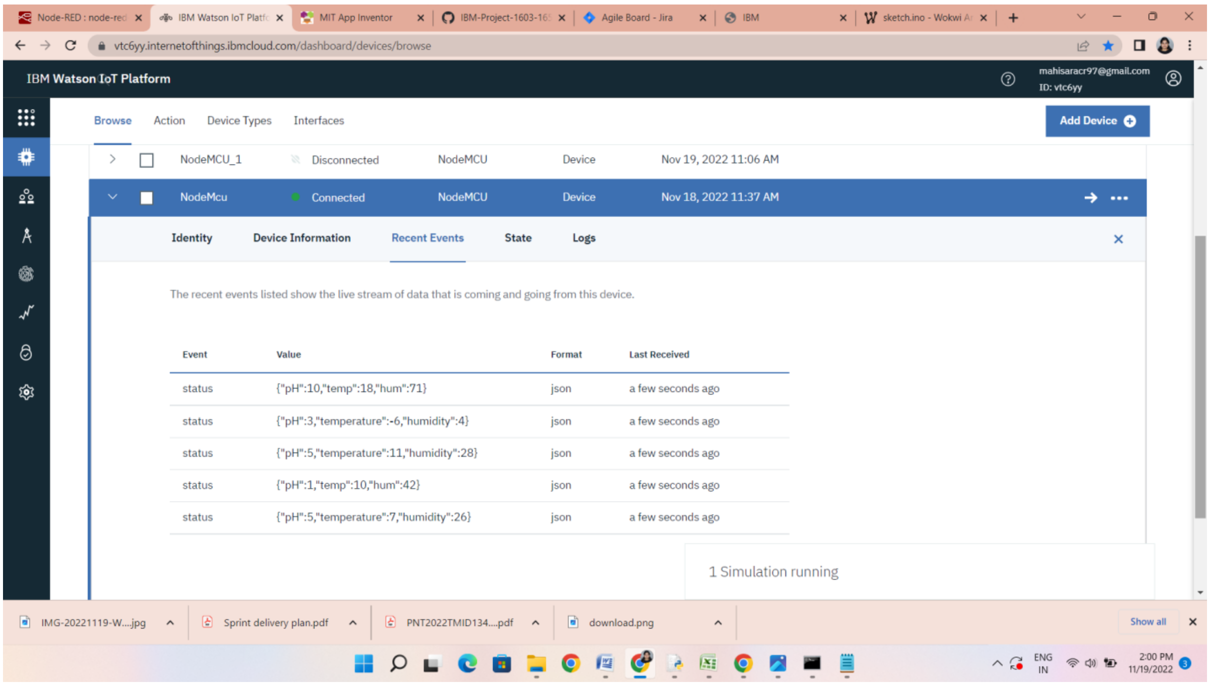
9.1 Performance Metrics

Node Red Output



```
18/11/2022, 13:39:52   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/123/evt/event_1/fmt/json :
msg.payload : number
12
18/11/2022, 13:39:52   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/NodeMcu/evt/status/fmt/json :
msg.payload : number
6
18/11/2022, 13:39:52   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/NodeMcu/evt/status/fmt/json :
msg.payload : number
43
18/11/2022, 13:39:55   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/NodeMcu/evt/status/fmt/json :
msg.payload : number
7
18/11/2022, 13:39:56   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/NodeMcu/evt/status/fmt/json :
msg.payload : number
37
18/11/2022, 13:39:56   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/123/evt/event_1/fmt/json :
msg.payload : number
4
18/11/2022, 13:39:56   node: e40416e9df1df53e
iot-2/type/NodeMCU/id/123/evt/event_1/fmt/json :
```

Watson Output



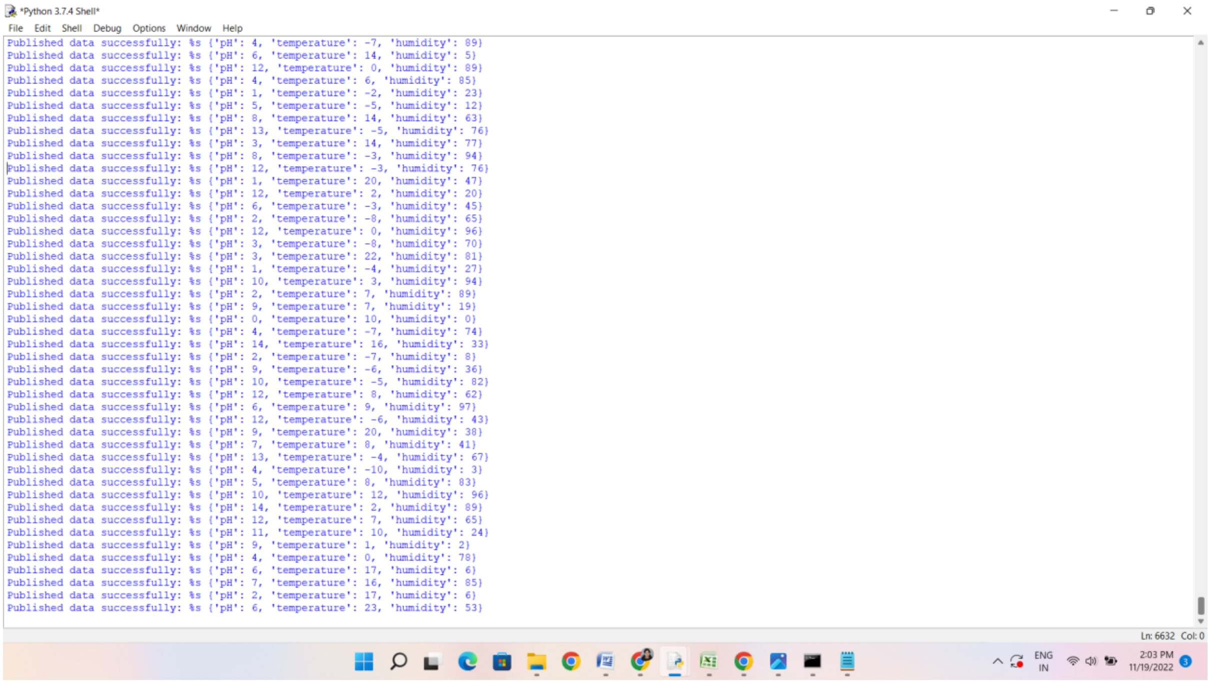
NodeMCU_1	Disconnected	NodeMCU	Device	Nov 19, 2022 11:06 AM
NodeMcu	Connected	NodeMCU	Device	Nov 18, 2022 11:37 AM

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"pH":10,"temp":18,"hum":71}	json	a few seconds ago
status	{"pH":3,"temperature":-6,"humidity":4}	json	a few seconds ago
status	{"pH":5,"temperature":11,"humidity":28}	json	a few seconds ago
status	{"pH":1,"temp":10,"hum":42}	json	a few seconds ago
status	{"pH":5,"temperature":7,"humidity":26}	json	a few seconds ago

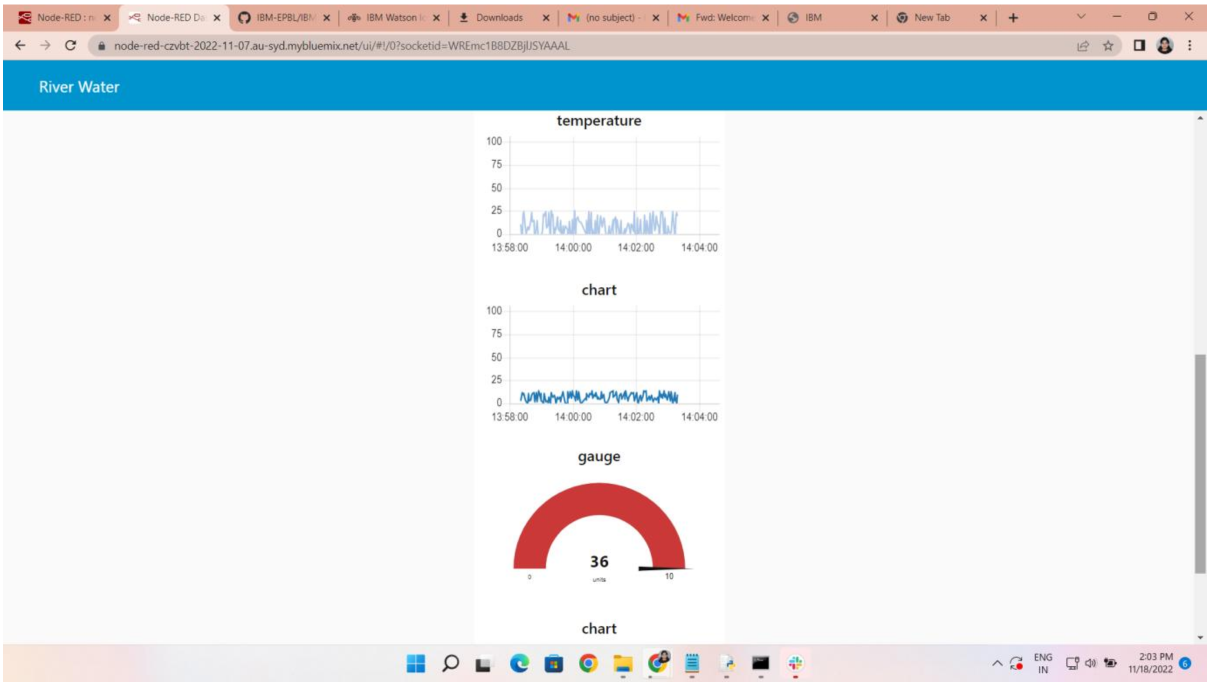
1 Simulation running

Python Output

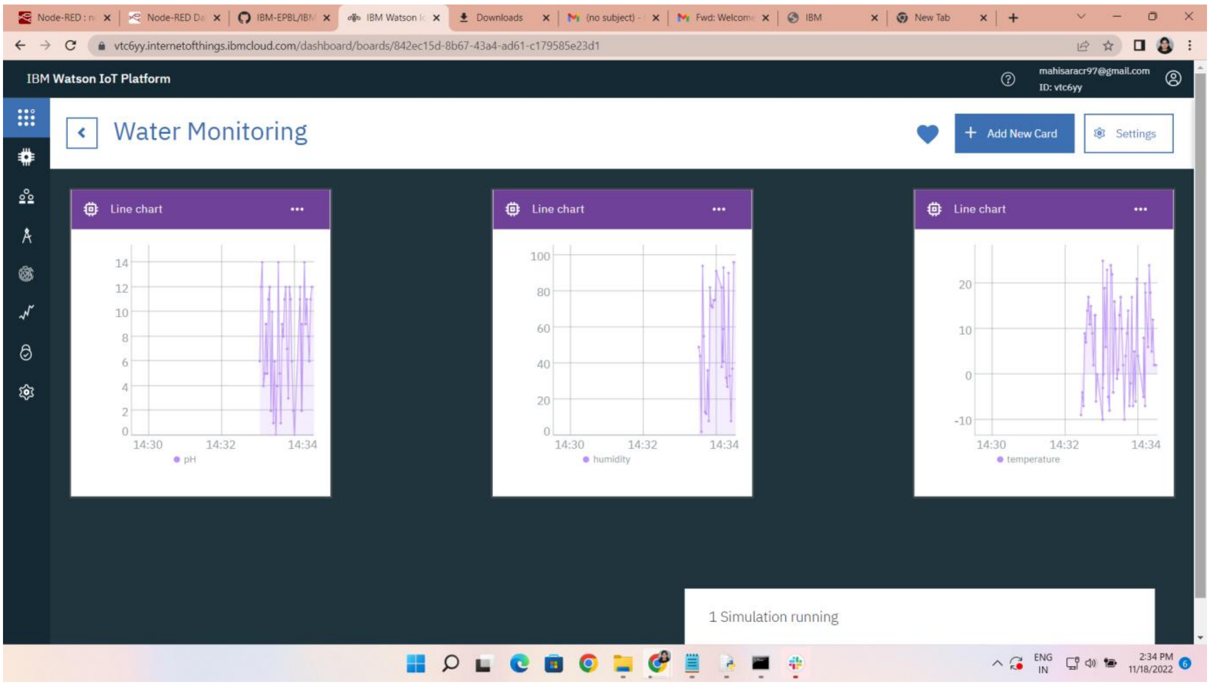


```
Published data successfully: %s {'pH': 4, 'temperature': -7, 'humidity': 89}
Published data successfully: %s {'pH': 6, 'temperature': 14, 'humidity': 5}
Published data successfully: %s {'pH': 12, 'temperature': 0, 'humidity': 89}
Published data successfully: %s {'pH': 4, 'temperature': 6, 'humidity': 85}
Published data successfully: %s {'pH': 1, 'temperature': -2, 'humidity': 23}
Published data successfully: %s {'pH': 5, 'temperature': -5, 'humidity': 12}
Published data successfully: %s {'pH': 8, 'temperature': 14, 'humidity': 63}
Published data successfully: %s {'pH': 13, 'temperature': -5, 'humidity': 76}
Published data successfully: %s {'pH': 3, 'temperature': 14, 'humidity': 77}
Published data successfully: %s {'pH': 8, 'temperature': -3, 'humidity': 94}
Published data successfully: %s {'pH': 12, 'temperature': -3, 'humidity': 76}
Published data successfully: %s {'pH': 1, 'temperature': 20, 'humidity': 47}
Published data successfully: %s {'pH': 12, 'temperature': 2, 'humidity': 20}
Published data successfully: %s {'pH': 6, 'temperature': -3, 'humidity': 45}
Published data successfully: %s {'pH': 2, 'temperature': -8, 'humidity': 65}
Published data successfully: %s {'pH': 12, 'temperature': 0, 'humidity': 96}
Published data successfully: %s {'pH': 3, 'temperature': -8, 'humidity': 70}
Published data successfully: %s {'pH': 3, 'temperature': 22, 'humidity': 81}
Published data successfully: %s {'pH': 1, 'temperature': -4, 'humidity': 27}
Published data successfully: %s {'pH': 10, 'temperature': 3, 'humidity': 94}
Published data successfully: %s {'pH': 2, 'temperature': 7, 'humidity': 89}
Published data successfully: %s {'pH': 9, 'temperature': 7, 'humidity': 19}
Published data successfully: %s {'pH': 0, 'temperature': 10, 'humidity': 0}
Published data successfully: %s {'pH': 4, 'temperature': -7, 'humidity': 74}
Published data successfully: %s {'pH': 14, 'temperature': 16, 'humidity': 33}
Published data successfully: %s {'pH': 2, 'temperature': -7, 'humidity': 8}
Published data successfully: %s {'pH': 9, 'temperature': -6, 'humidity': 36}
Published data successfully: %s {'pH': 10, 'temperature': -5, 'humidity': 82}
Published data successfully: %s {'pH': 12, 'temperature': 8, 'humidity': 62}
Published data successfully: %s {'pH': 6, 'temperature': 9, 'humidity': 97}
Published data successfully: %s {'pH': 12, 'temperature': -6, 'humidity': 43}
Published data successfully: %s {'pH': 9, 'temperature': 20, 'humidity': 38}
Published data successfully: %s {'pH': 7, 'temperature': 8, 'humidity': 41}
Published data successfully: %s {'pH': 13, 'temperature': -4, 'humidity': 67}
Published data successfully: %s {'pH': 4, 'temperature': -10, 'humidity': 3}
Published data successfully: %s {'pH': 5, 'temperature': 8, 'humidity': 83}
Published data successfully: %s {'pH': 10, 'temperature': 12, 'humidity': 96}
Published data successfully: %s {'pH': 14, 'temperature': 2, 'humidity': 89}
Published data successfully: %s {'pH': 12, 'temperature': 7, 'humidity': 65}
Published data successfully: %s {'pH': 11, 'temperature': 10, 'humidity': 24}
Published data successfully: %s {'pH': 9, 'temperature': 1, 'humidity': 2}
Published data successfully: %s {'pH': 4, 'temperature': 0, 'humidity': 78}
Published data successfully: %s {'pH': 6, 'temperature': 17, 'humidity': 6}
Published data successfully: %s {'pH': 7, 'temperature': 16, 'humidity': 85}
Published data successfully: %s {'pH': 2, 'temperature': 17, 'humidity': 6}
Published data successfully: %s {'pH': 6, 'temperature': 23, 'humidity': 53}
```

DashBoard Output



Graph



10. ADVANTAGES & DISADVANTAGES

Advantages:

- This project provide the information for all 24/7 and more helpful for human resource
- The Quality of water is determined in real manner
- No requirement of man power
- High technology are involved

Disadvantages:

- It need high internet connection
- The implementation cost is bit high

11. CONCLUSION

This project concluded that it provided a great source for IOT in real time and more helpful for human resources. Water resource maintenance is an emergency thing, with the help of IOT, the real time monitoring are detected easily. It mainly focus on saving human from various diseases and chlorination may reduced. It introduce a new technology to the people and help people to learn new technology and great example for upcoming technology.

12. FUTURE SCOPE

While developing this project, it provide a great way for technology and communication and more useful for public. It enhances the technology grooming in IOT. Through this project we can increase fresh water supply and decrease the availability of impure water. The unavailability of drought may happen and more flexible for future uses. There is a possibility of reducing salt water content when we develop this project.

13. APPENDIX

13.1 SOURCE CODE

Python Code:

```
        import wiotp.sdk.device
import time
import os
import datetime
import random

myConfig = {
    "identity":{
        "orgId":"vtc6yy",
        "typeId":"NodeMCU",
        "deviceId":"NodeMcu"
        },
    "auth":{
        "token":"12345678"
        }
    }
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()


def myCommandCallback(cmd):
    print("Message received from IBM IoT platform: %s" % cmd.data['command'])
    m=cmd.data['command']
    if(m=="Appear"):
        print("Appear")
    elif(m=="Disappear"):
        print("Disappear")
    print(" ")
while True:
    pH=random.randint(0,14)
    temp=random.randint(-10,25)
    hum=random.randint(0,100)
    myData={'pH':pH,'temperature':temp,'humidity':hum}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data successfully: %s", myData)
    time.sleep(2)
    client.commandCallback = myCommandCallback
client.disconnect()
```

WOKWI CODE:

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>
#include "DHT.h"

const char* ssid = "KAVIYA S";
const char* password = "Kaviya@123";
#define DHTPIN 4
#define DHTTYPE DHT11
#define ORG "vtc6yy"
#define DEVICE_TYPE "ESP32"
#define DEVICE_ID "1002"
#define TOKEN "wHKDv_XEAZF-vUsYmR"

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1[] = "iot-2/evt/status1/fmt/json";
char pubTopic2[] = "iot-2/evt/status2/fmt/json";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```
WiFiClient wifiClient;
PubSubClient client(server, 1883, NULL, wifiClient);
DHT dht(DHTPIN, DHTTYPE);


void setup() {
  Serial.begin(115200);
  dht.begin();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");

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

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

long lastMsg = 0;
void loop() {
  client.loop();
  long now = millis();
  if (now - lastMsg > 3000) {
    lastMsg = now;
    float humidity = dht.readHumidity();
    float temperature = dht.readTemperature();
    String payload = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
    payload += ",\"temperature\":";
    payload += temperature;
    payload += "}}";

    Serial.print("Sending payload: ");
    Serial.println(payload);

    if (client.publish(pubTopic1, (char*) payload.c_str())) {
      Serial.println("Publish ok");
    } else {
      Serial.println("Publish failed");
    }
    String payload1 = "{\"d\":{\"Name\":\"" DEVICE_ID "\"";
    payload1 += ",\"humidity\":";
    payload1 += humidity;
    payload1 += "}}";

    if (client.publish(pubTopic2, (char*) payload1.c_str())) {
      Serial.println("Publish ok");
    } else {
      Serial.println("Publish failed");
    }
  }
}
```

GITHUB ID:

https://github.com/IBM-EPBL/IBM-Project-1603-1658402144

VIDEO DEMO LINK:

https://youtu.be/Ksv3qSinjJU