REAL-TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

Domain: INTERNET OF THINGS

PROJECT REPORT

Submitted by

S.NO	***NAME***	***ROLL NO.***
1.	Smitha Rajini T	9517201904151
2.	Siva Sangeetha R	9517201904149
3.	Sheron S	9517201904146
4.	Gobika RM	9517201904041

From

MEPCO SCHLENK ENGINEERING COLLEGE

In fulfillment of project in ***IBM-NALAYATHIRAN 2022***

TEAM ID: **PNT2022TMID18013**

PROJECT GUIDES

Industry mentor: Mr. Bharadwaj **Faculty mentor**: Mrs. G. Priyanka

INDEX

1. INTRODUCTION:

1.1 PROJECT OVERVIEW:

A real-time river water quality management system is built since nowadays, the river water is mostly contaminated without the knowledge of contamination people drink this contaminated river water and they suffer from various diseases. To avoid this condition the various parameters of water are needed to be tested so that they should satisfy certain condition according to their parameters. The parameters that should be examined are pH (range 6.5-8.5), temperature (10-23 degree Celsius) and turbidity (0 to 1). If the river water range exceeds these values, then we can conclude that the water is contaminated and not suitable for drinking, whenever these parameters are not satisfied the user will get a notification saying that the water is contaminated. This is achieved using node-red, MIT app inventor and IBM iot Watson platform.

1.2 PURPOSE:

Drinking of contaminated water may result in various health issues or even death. A statistical report says that in India almost 37.7 million Indians being affected by waterborne diseases every year and close to 3 lakh children dying due to diarrhoea. To avoid these conditions, we need to measure various parameters of water, analyse them and report it to the local authorities regarding the same to avoid unnecessary health issues.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM:

Water is uniquely vulnerable to pollution. Known as a "universal solvent," water is able to dissolve more substances than any other liquid on earth. It's the reason we have Kool-Aid and brilliant blue waterfalls. It's also why water is so easily polluted. Toxic substances from farms, towns, and factories readily dissolve into and mix with it, causing water pollution.

According to the most recent surveys on national water quality from the U.S. Environmental Protection Agency, nearly half of our rivers are polluted and unfit for human consumption and survival of aquatic organisms. Nutrient pollution, which includes nitrates and phosphates, is the leading type of contamination in these freshwater sources. While plants and animals need these nutrients to grow, they have become a major pollutant due to farm waste and fertilizer runoff. Municipal and industrial waste discharges contribute their fair share of toxins as well. There is also all the random junk that industry and individuals dump directly into river water.

2.2 REFERENCES:

- ✓ **Water quality monitoring system based on Internet of Things**
  Author: Chengcheng Zhang, Jian Wu, Jiancheng Liu
  Publication: IEEE 2020

  Chengcheng et al presents a solution that integrates the design of STM32 single- chip microcomputer, sensors, WiFi wireless transmission and remote water quality management system. It monitors water quality turbidity, pH value, temperature and uploads the data to the management center through wireless communication.

- ✓ **IoT Based Real-time River Water Quality Monitoring System**
  Author: Mohammad Salah Uddin Chowdurya, Talha Bin Emran b, Subhasish Ghosha , Abhijit Pathak a, Mohd. Manjur Alama, Nurul Absar a, Karl Andersson c, Mohammad Shahadat Hossain d
  Publication: Science Direct 2019

  Mohammad et al proposed a manual method for sensor- based water quality monitoring system with high frequency, high mobility, and low power. Here the data collected at the site can be displayed in a visual format on a server PC with the help of Spark streaming analysis through Spark MLlib, Deep learning neural network models, Belief Rule Based (BRB) system and is also compared with standard value.

- ✓ **Efficient Cloud Based Real Time Water Quality Monitoring System Using Internet Of Things**
  Author: M.Usha Rani, Dr.R.Alageswaran, Sathish Kumar A
  Publication: JASC: Journal of Applied Science and Computations(2018)

  M.Usha Rani et al proposes water sampling system with required sensor. Whenever the water level in the lakes or ponds reaches the lower/upper level it is identified and notification is sent to the administrator. It can also predict overflow and water scarcity in future from the past results. The parameters like PH, calcium, sulphate and nitrate ions that is present in the water is also identified.

- ✓ **Water Quality Monitoring System Using IOT**
  Author: Dr. Nageswara Rao Moparthi, Ch. Mukesh, Dr. P. Vidya Sagar
  Publication: IEEE 2018

  Dr. Nageswara Rao Moparthi et al implements Water Quality Monitoring System for municipal water tanks and drinking water reservoirs using an Arduino board and GSM module. This module can be easily implemented when a wireless oxygen sensor is used.

- ✓ **Real-Time Water Quality Monitoring System**
  Author: Jyotirmaya Ijaradar1, Subhasish- Chatterjee
  Publication: International Research Journal of Engineering and Technology (IRJET) (2018)

  Jyotirmaya et al proposed real-time water quality monitoring system for water health at residential places. It measures various chemical and physical properties of water like pH, temperature and particle density of water using sensors and send the data to cloud and trigger an alarm when discrepancies are found in water quality.

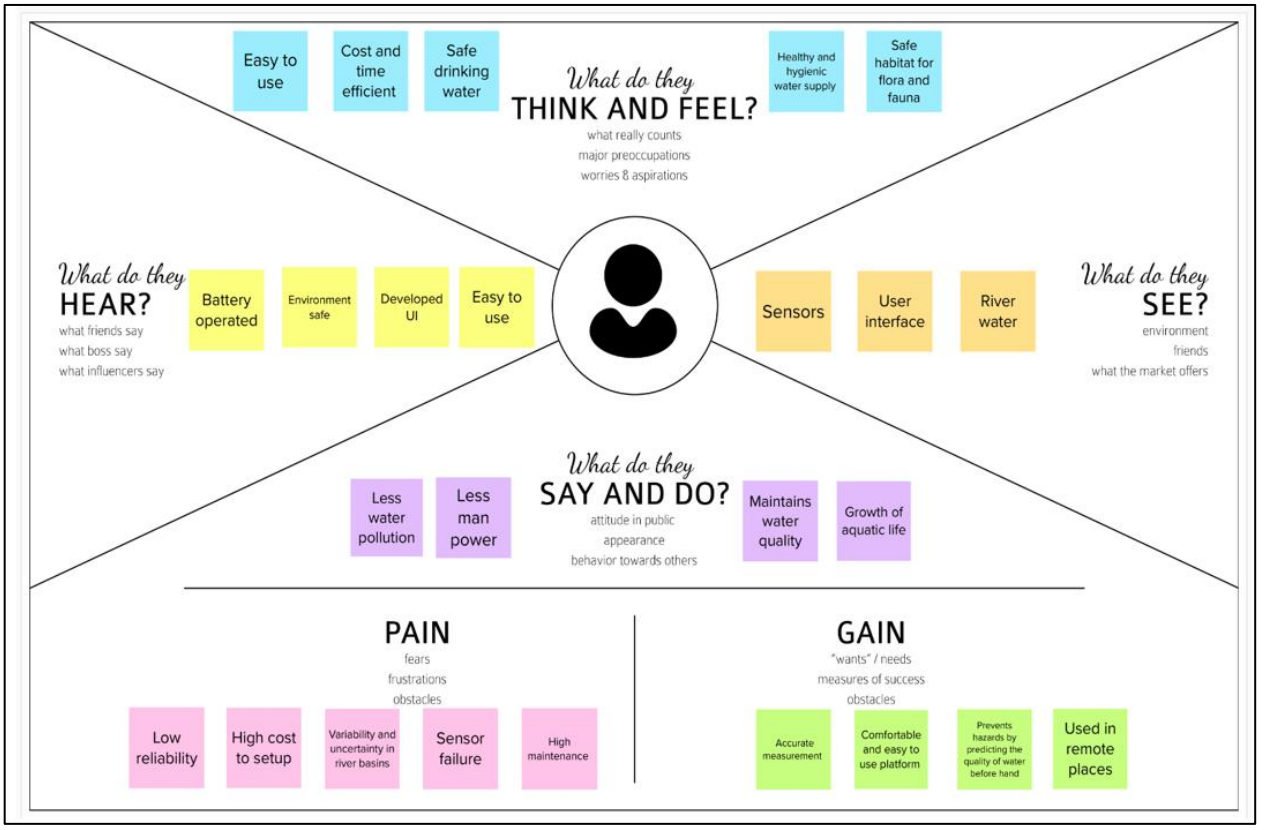
2.3 PROBLEM STATEMENT:

To detect and monitor the quality of river water since river water pollution is a global environmental threat and to determine the parameters such as pH, temperature and turbidity.

3. IDEATION & PROPOSED SOLUTION

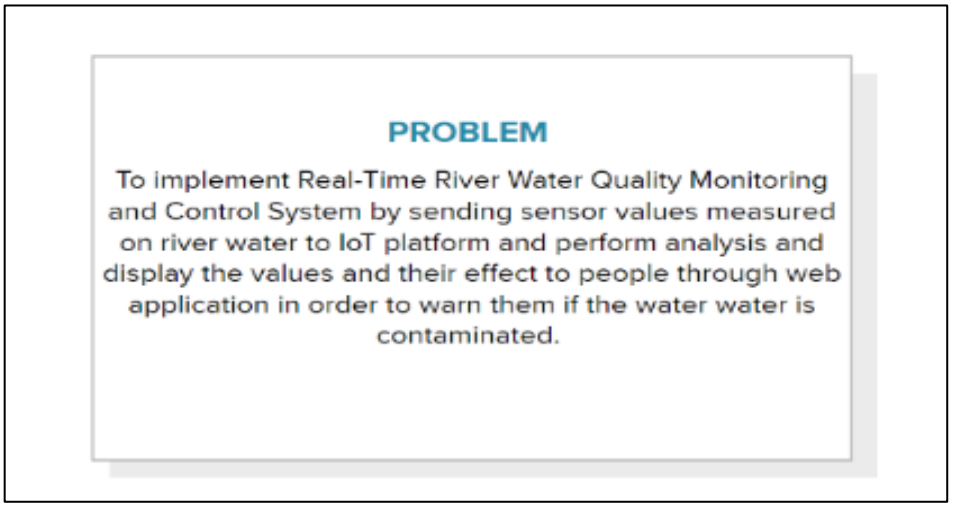
3.1 EMPATHY MAP CANVAS:

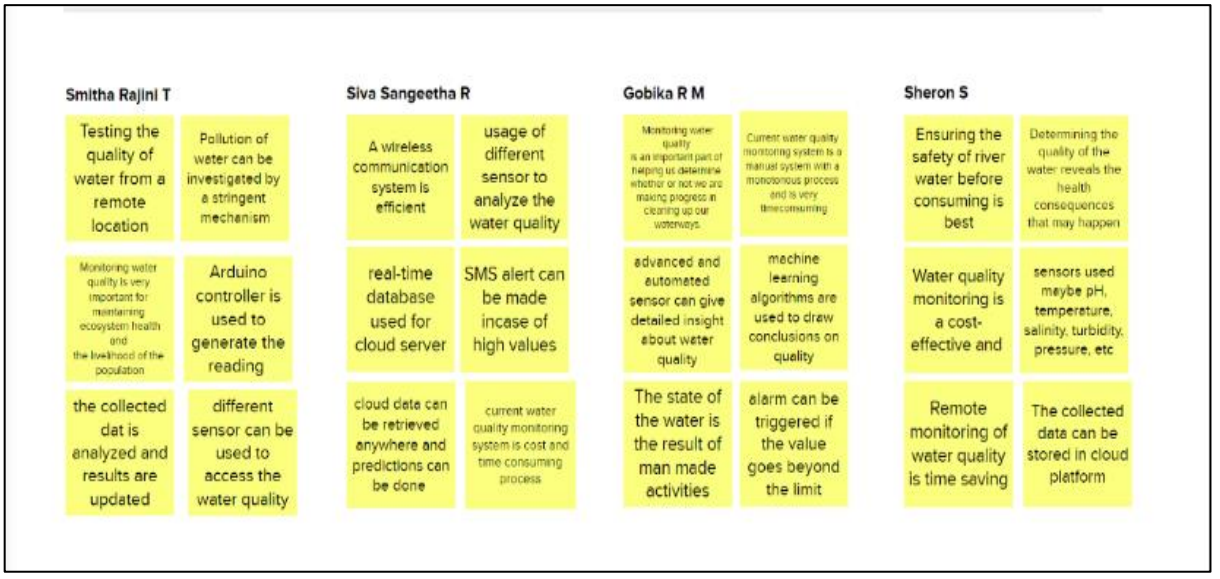
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

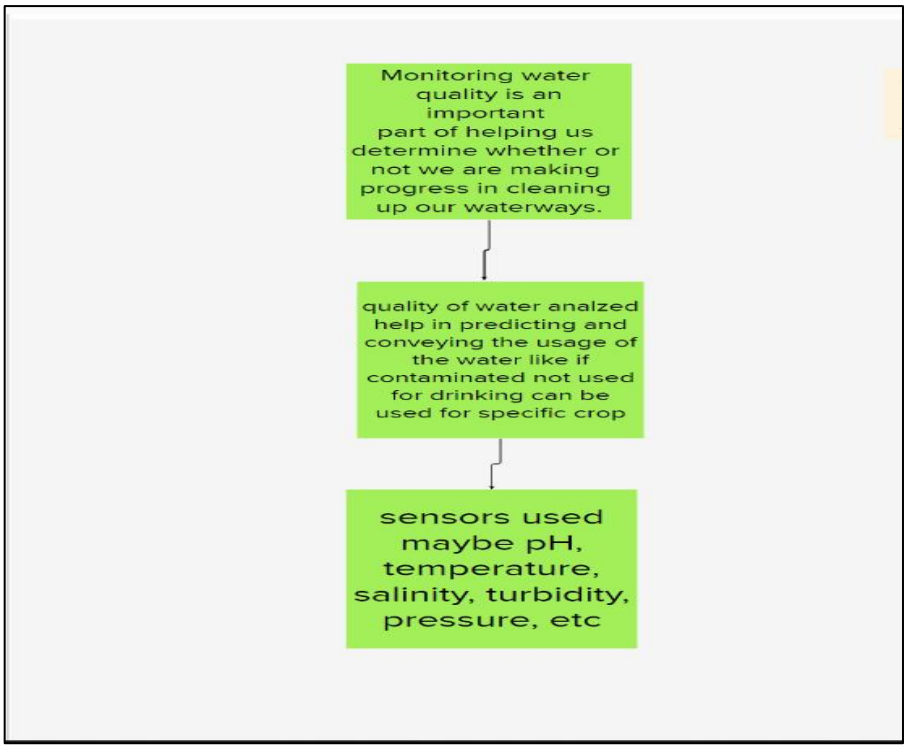


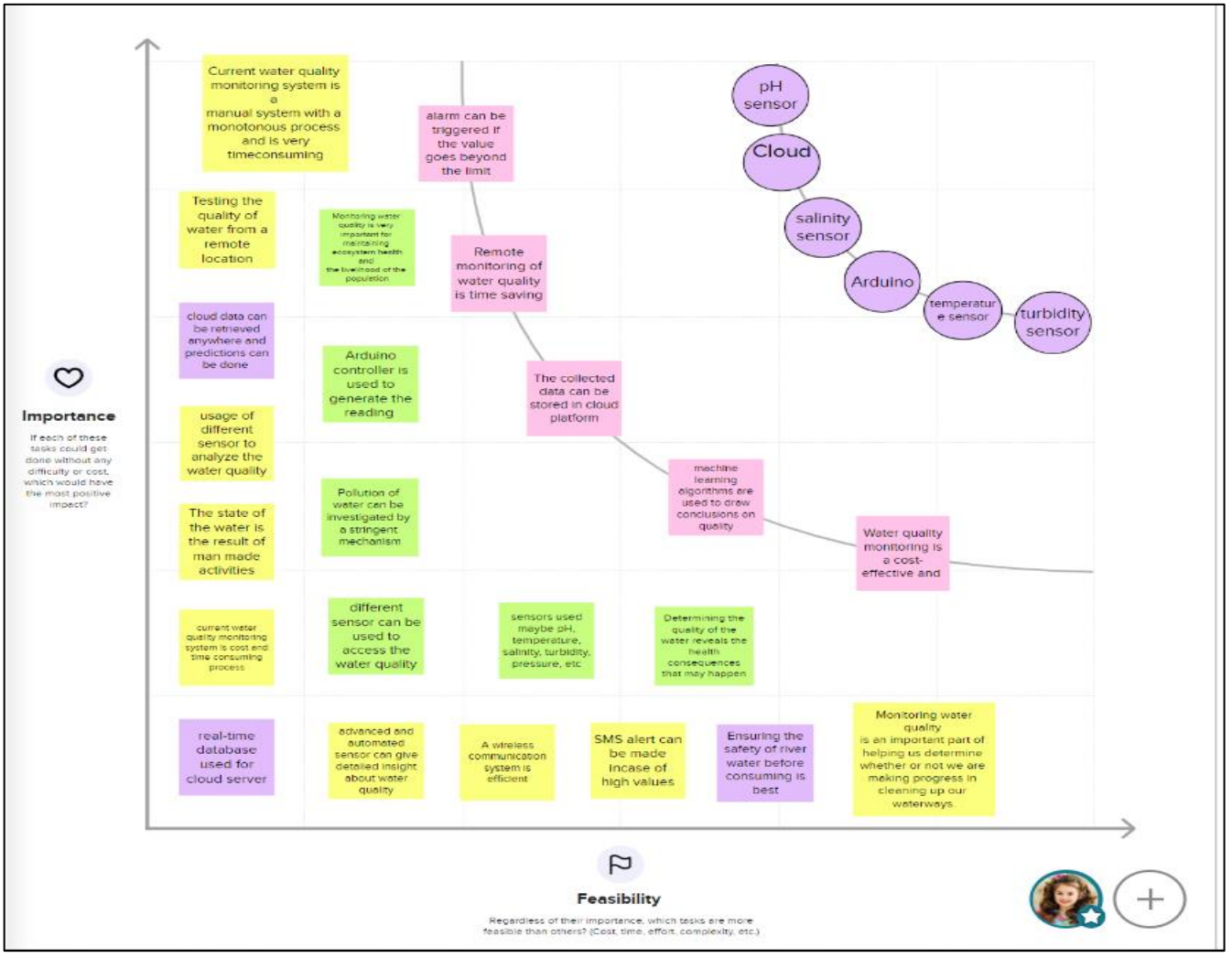
3.2 IDEATION & BRAINSTORMING:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out of the box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.





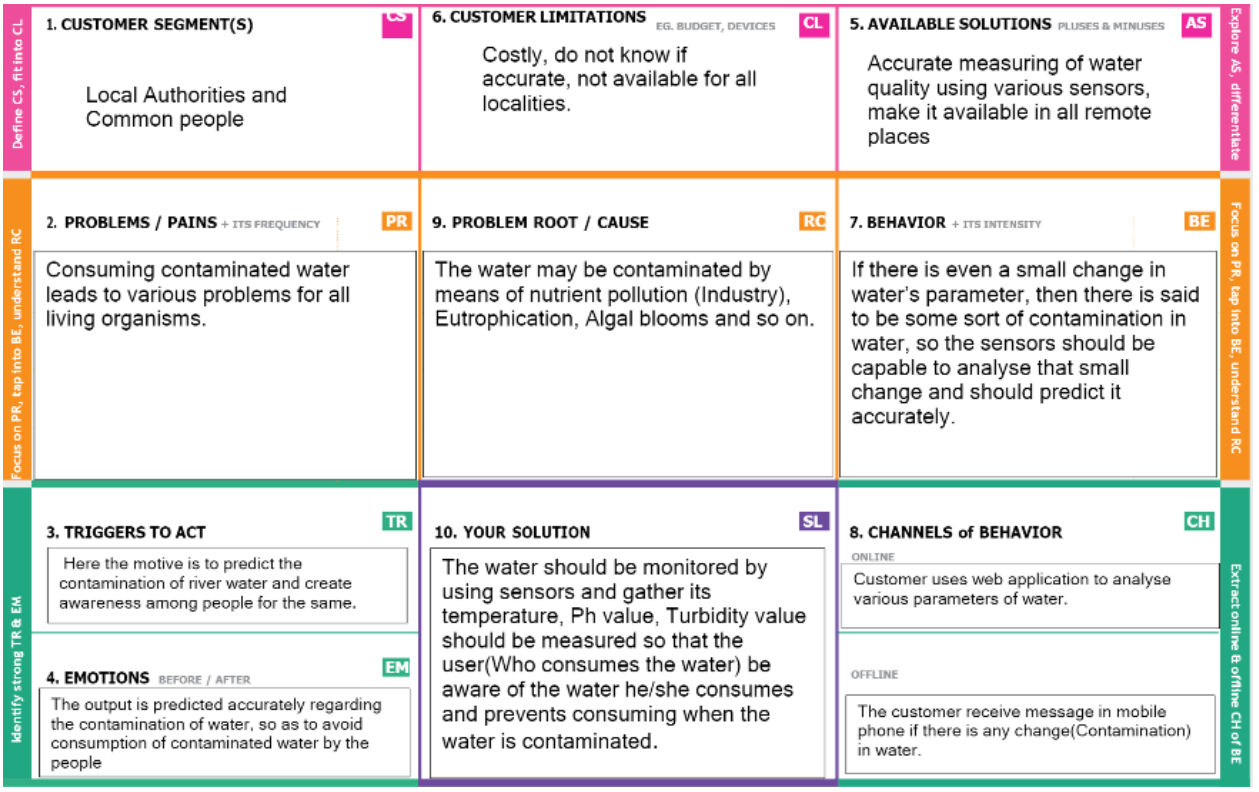

Monitoring water quality is an important part of helping us determine whether or not we are making progress in cleaning up our waterways.
quality of water analzed help in predicting and conveying the usage of the water like if contaminated not used for drinking can be used for specific crop
sensors used maybe pH, temperature, salinity, turbidity, pressure, etc


Current water quality monitoring system is a manual system with a monotonous process and is very timeconsuming
alarm can be triggered if the value goes beyond the limit
pH sensor
Cloud
salinity sensor
Arduino
temperatur e sensor
turbidity sensor
Testing the quality of water from a remote location
Monitoring water quality is very important for maintaining ecosystem health and the livelihood of the population
Remote monitoring of water quality is time saving
cloud data can be retrieved anywhere and predictions can be done
Arduino controller is used to generate the reading
The collected data can be stored in cloud platform
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
usage of different sensor to analyze the water quality
machine learning algorithms are used to draw conclusions on quality
Pollution of water can be investigated by a stringent mechanism
The state of the water is the result of man made activities
Water quality monitoring is a cost-effective and
current water quality monitoring system is cost and time consuming process
different sensor can be used to access the water quality
sensors used maybe pH, temperature, salinity, turbidity, pressure, etc
Determining the quality of the water reveals the health consequences that may happen
real-time database used for cloud server
advanced and automated sensor can give detailed insight about water quality
A wireless communication system is efficient
SMS alert can be made incase of high values
Ensuring the safety of river water before consuming is best
Monitoring water quality is an important part of helping us determine whether or not we are making progress in cleaning up our waterways.
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

3.3 PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	IOT Based Real Time River Water Quality Monitoring and Control System
2.	Idea / Solution description	1. To monitor the quality of water using sensors like temperature, potentiometer(pH), turbidity, salinity and so on. 2. Collecting those data and storing it in cloud and perform analyze to check if the water is contaminated or not for drinking. 3. If the water is contaminated an alert is made to the user/ local authority through SMS or can be viewed through web application anytime.
3.	Novelty / Uniqueness	4. 1. Based on the collected data prediction is made whether the water can be used for cultivation of specific crops and suitable for the aquatic animals.
4.	Social Impact / Customer Satisfaction	Algal growth, fertilizers, pesticides cause river pollution which can impact all living beings. Better monitoring and control measures can impact health and vegetation massively.
5.	Business Model (Revenue Model)	Service based product is developed to serve the local people to know the quality of water before consuming it or using it for any purpose. This prevents health issues or at most loss of living being.
6.	Scalability of the Solution	Developing the product as both web and mobile application it is portable, and data can be accessed from anywhere anytime. provide a real-time monitoring and a feasible solution for remote or distant places where water quality laboratory is not present.

3.4 PROBLEM SOLUTION FIT:



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT:

Following are the functional requirements of the proposed solution

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Login	Confirmation through verified password
FR-2	View Water Details	View current water details in website View traditional water eligibility in website
FR-3	Logout	Logs out the user successfully

4.2 NON-FUNCTIONAL REQUIREMENT:

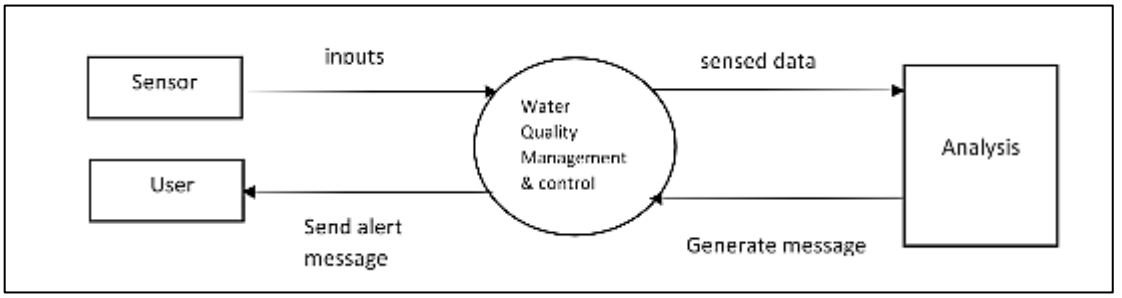
Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Load time for user interface screens shall not be more than 2 seconds.
NFR-2	**Security**	User account is password protected Account creation done only after email verification
NFR-3	**Reliability**	Users can access their account 98% of the time without failure
NFR-4	**Performance**	Load time for user interface screens shall not be more than 2 seconds. Login info verified within 10 seconds.
NFR-5	**Availability**	Maximum down time will be about 4 hours
NFR-6	**Scalability**	System can handle about 1000 users at any given time

5. PROJECT DESIGN

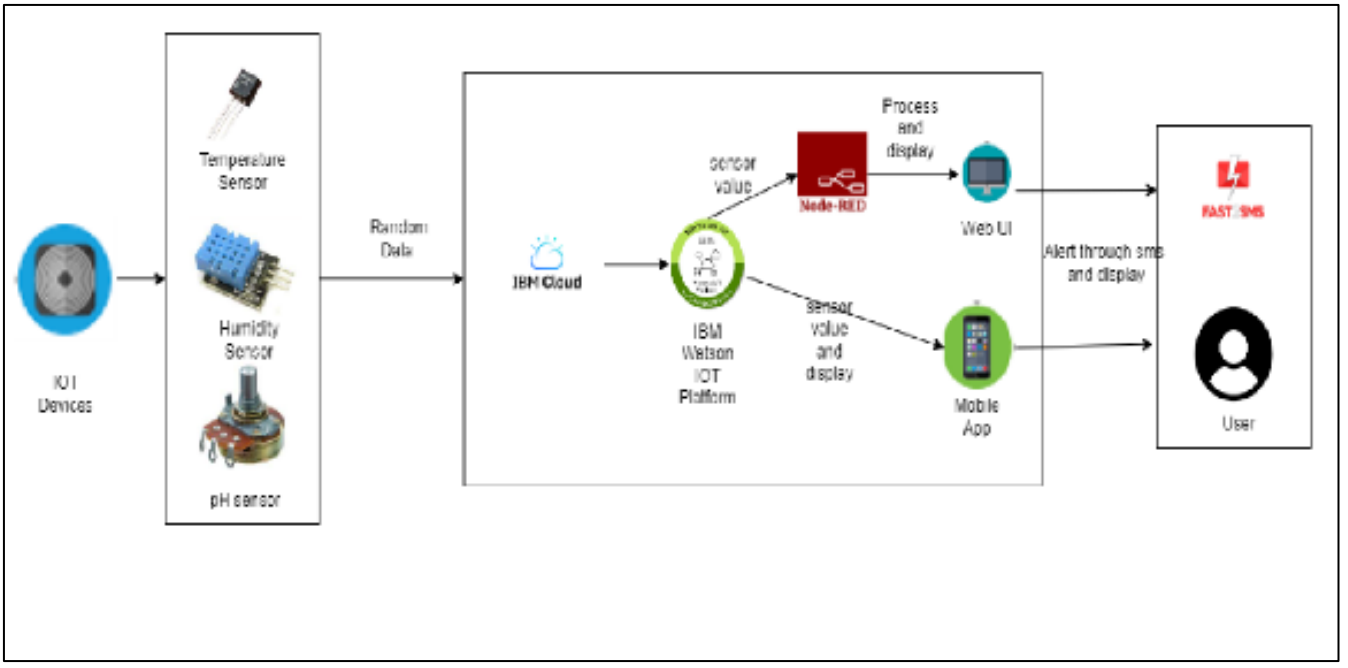
5.1 DATA FLOW DIAGRAM:

A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement.



5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

Technology architecture, it mainly focuses on the design and documentation of software applications. Thus, technical architects create blueprint schematics of technical solutions making sure that new products or systems meet specified requirements.



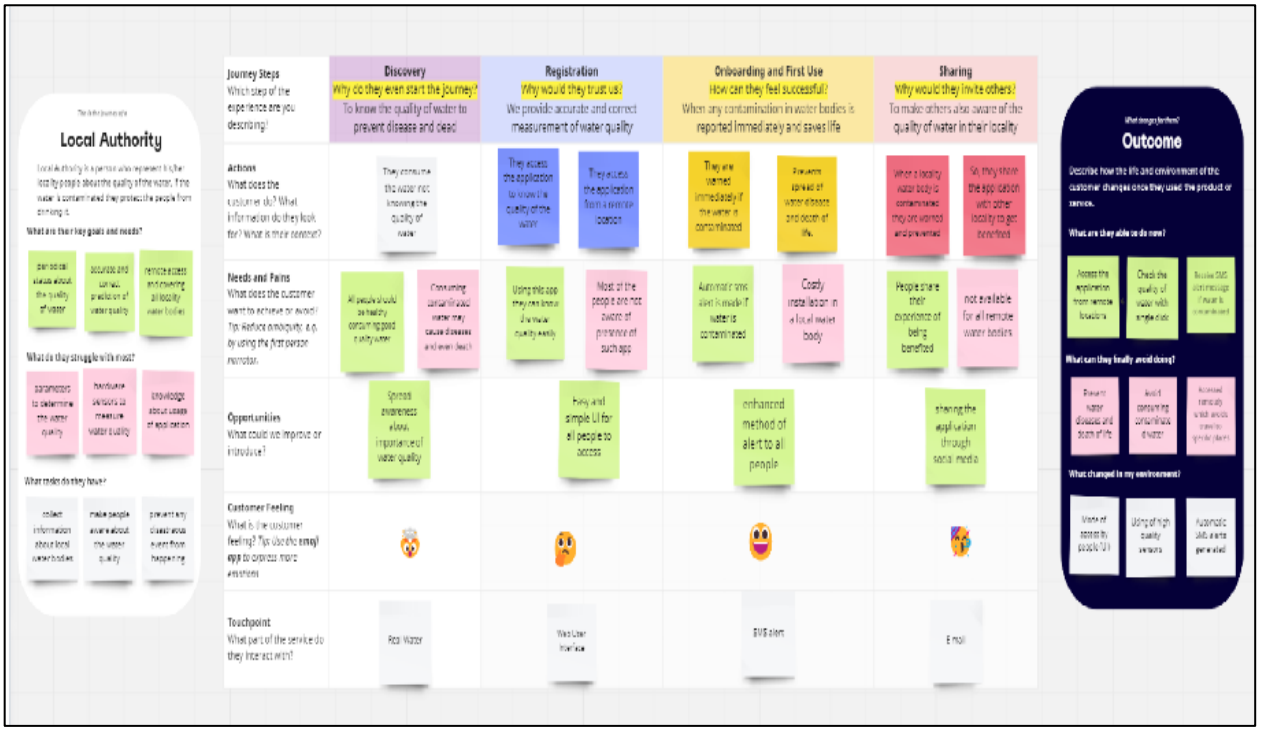
Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Mobile App	Node – Red, Kubernetes, MIT mobile app inventor
2.	Application Logic-1	Generate random data	Python
3.	Application Logic-2	Generate random sensor data	IBM Watson IOT Platform
4.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant,
5.	External API-1	Send SMS to customer	Fast SMS API
6.	Infrastructure (Server / Cloud)	Application Deployment on Cloud	Cloud Foundry, Kubernetes

Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	open-source frameworks used to develop our project	Node – Red, IBM Cloudant, IBM Watson IOT Platform
2.	Security Implementations	Use of Login facility with username and password for individual user	Password protection in MIT App
3.	Scalable Architecture	Web Ui designed for use in Mobile and computer with adaptive screen size	Node – Red (Web UI)
4.	Availability	Available for the user in both web UI and Mobile App	Node – Red(Web UI), MIT App(Mobile App)
5.	Performance	Give accurate results and immediate alert in case of contamination of water	Node – Red(Web UI), MIT App(Mobile App)

5.3 USER STORIES:



6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

TITLE	DISCRIPTION	DATE
Literature Survey& Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	03 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	07 SEPTEMBER 2022
Problem Statement	Prepare Problem statement of Industry-specific intelligent fire management system	10 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	16 SEPTEMBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 SEPTEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	26 SEPTEMBER 2022
Solution Architecture	Prepare solution architecture document.	30 SEPTEMBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions and experiences with the application (entry to exit).	8 OCTOBER 2022

Functional Requirement	Prepare the functional requirement document.	11 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	14 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	16 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	24 OCTOBER 2022
Sprint Schedules	Prepare the sprint plan and divided tasks according to agile method.	24 OCTOBER 2022
Project Development-Delivery Sprint - 1	Develop & submit the developed code by testing it.	29 OCTOBER 2022
Sprint - 2	Develop & submit the developed code by testing it.	05 NOVEMBER 2022
Sprint - 3	Develop & submit the developed code by testing it.	12 NOVEMBER 2022
Sprint - 4	Develop & submit the developed code by testing it.	19NOVEMBER 2022

6.2 Sprint delivery schedule:

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Check Notification	USN-1	As a user, I can check the notification of the alert message.	20	High	Gobika RM
Sprint-2	Check water parameters	USN-2	As a user, I can check the level of water parameters like temperature, humidity, PH level etc.	20	High	Siva Sangeetha R
Sprint-3	Registration Page	USN-3	As a user, I can register into the application	20	High	Sheron S
Sprint-4	Login Page	USN-4	As a user, I can login into the application	20	High	Smitha Rajini T

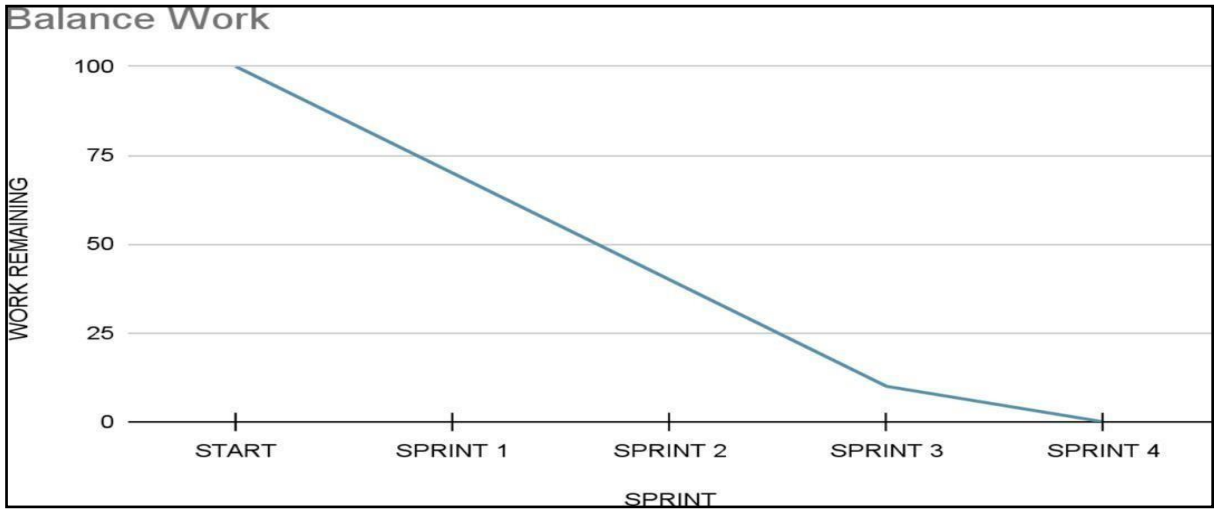
Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date	Story Points	Sprint Release Date
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	09 Nov 2022	20	09 Nov 2022
Sprint-3	20	6 Days	10 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	12 Nov 2022	13 Nov 2022	20	13 Nov 2022

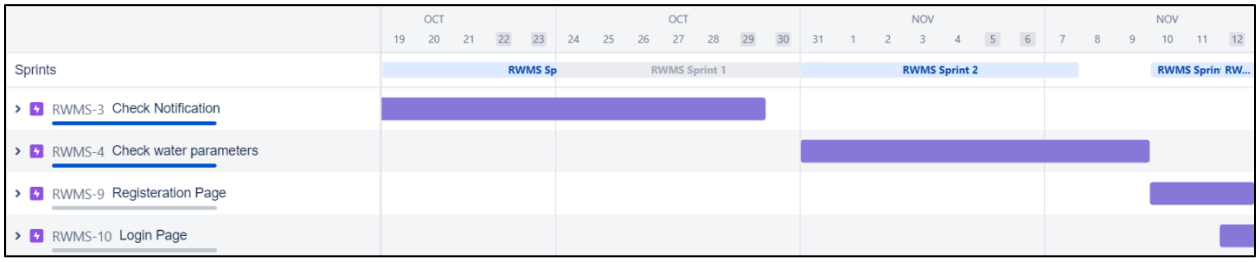
Velocity:

Average Sprint Velocity = 80 / 4 = 20

Burndown Chart:

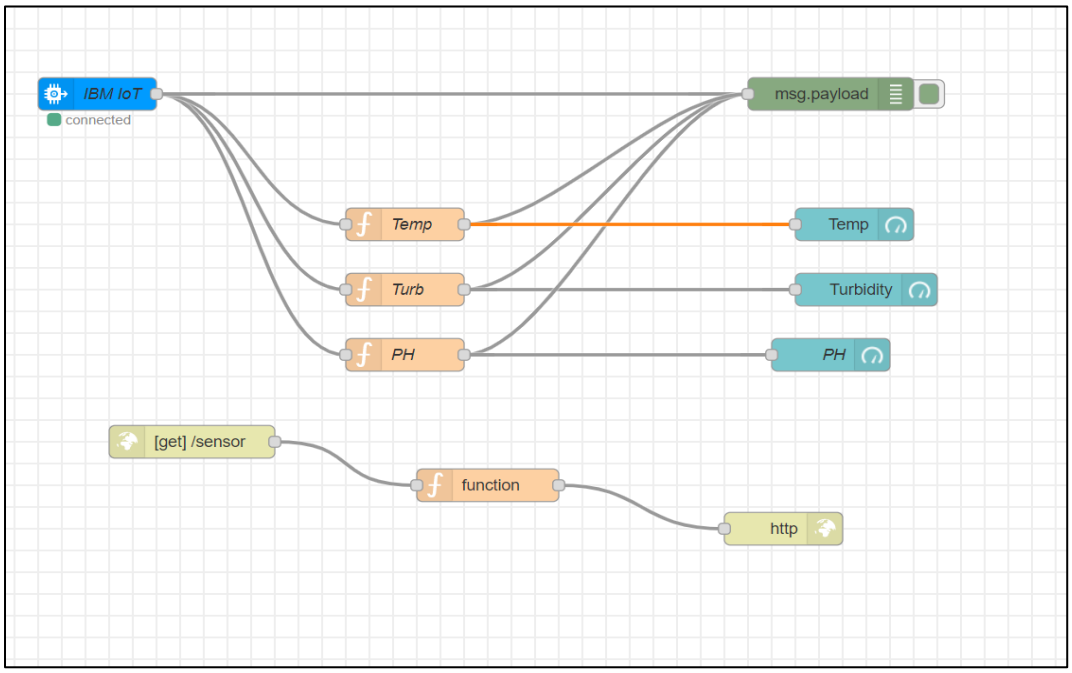


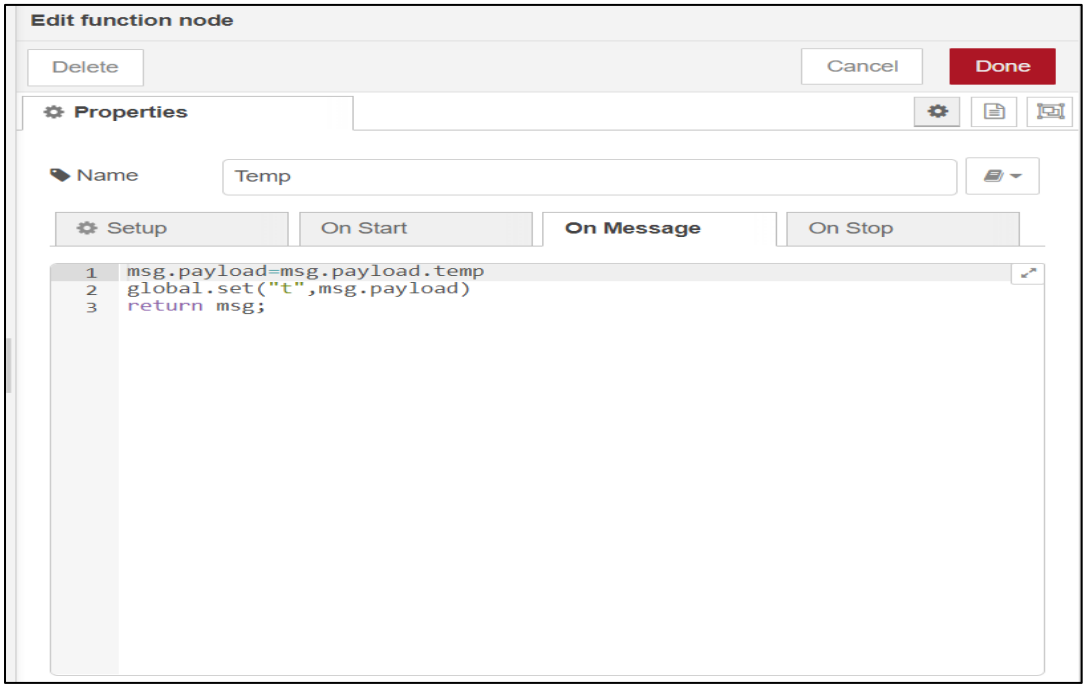
6.3 Report from Jira:

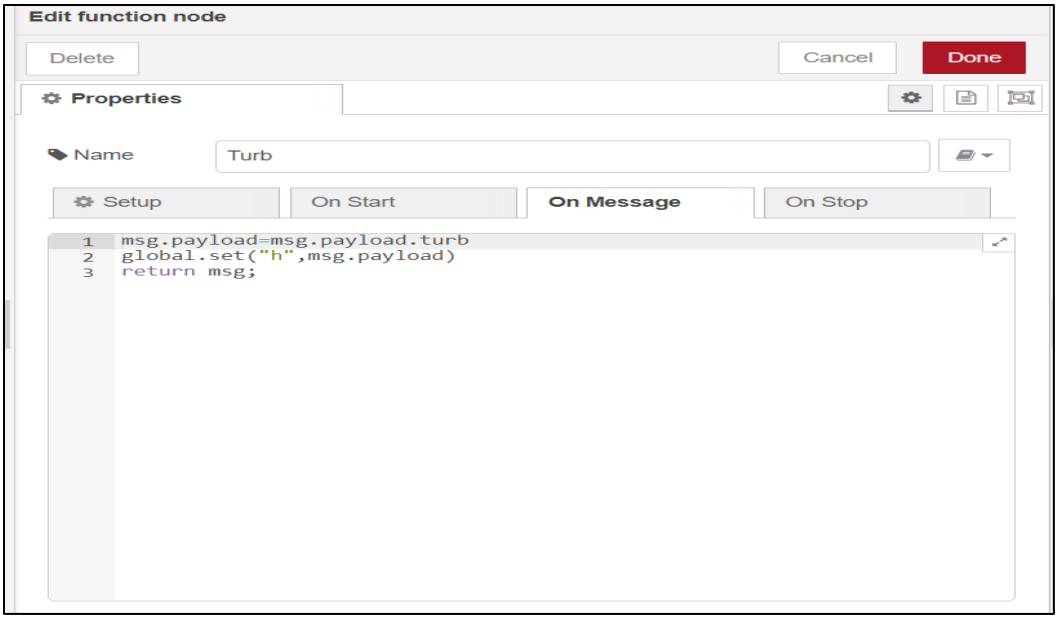


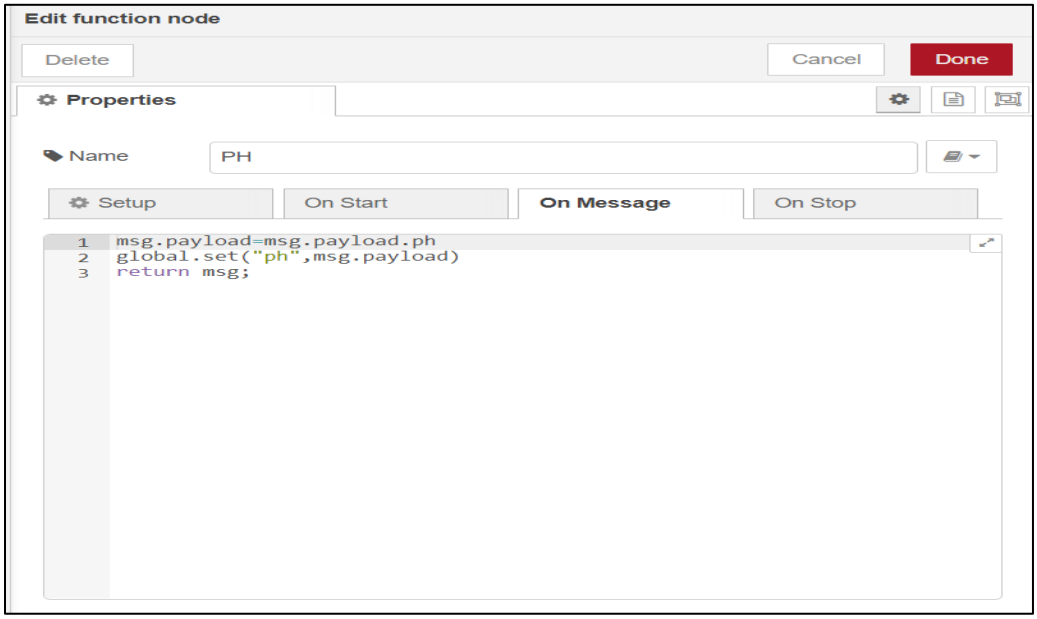
7. CODING AND SOLUTION

7.1 Node Red service associated with IBM Cloud

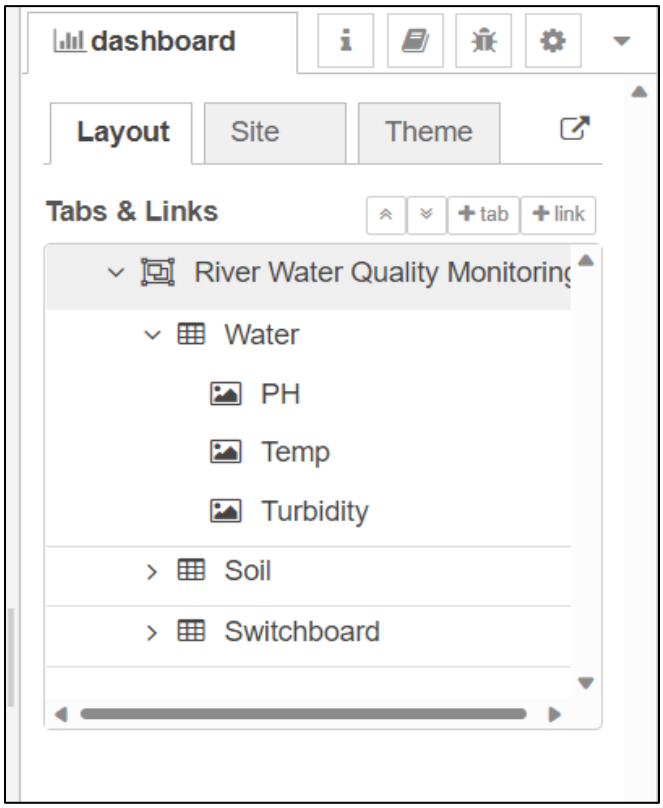


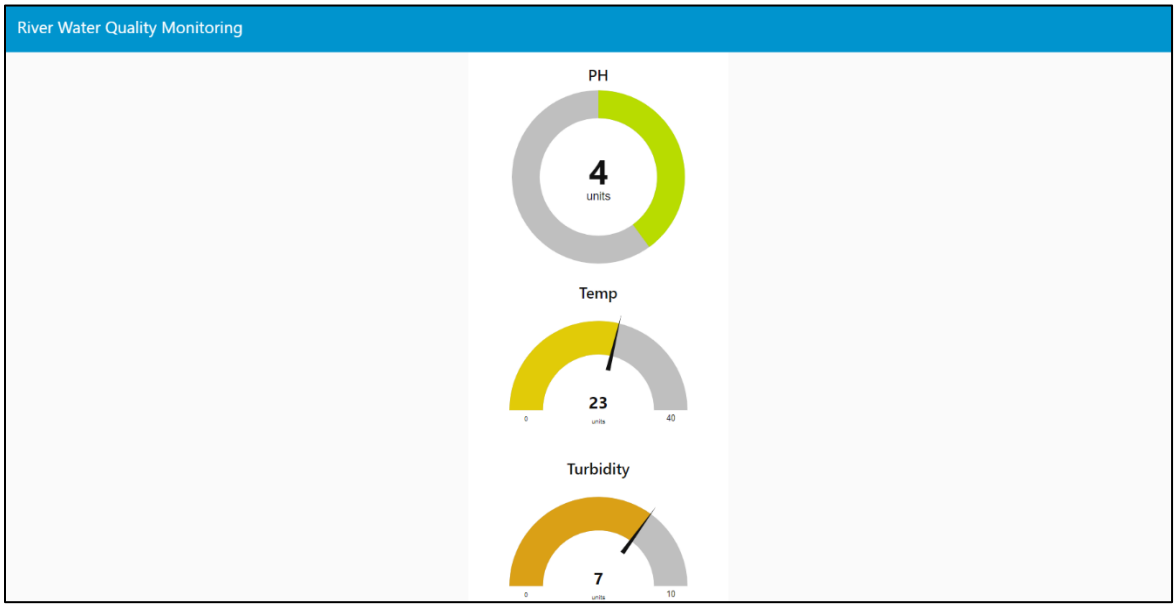



Edit function node
Delete
Cancel
Done
Properties
Name
Turb
Setup
On Start
On Message
On Stop
1 msg.payload=msg.payload.turb
2 global.set("h",msg.payload)
3 return msg;

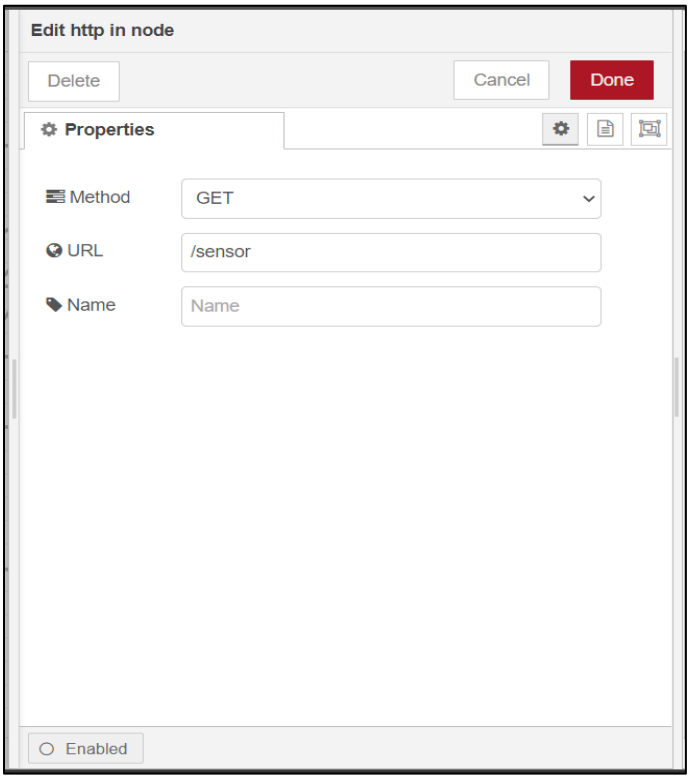

Edit function node
Delete
Cancel
Done
Properties
Name
PH
Setup
On Start
On Message
On Stop
1 msg.payload=msg.payload.ph
2 global.set("ph",msg.payload)
3 return msg;

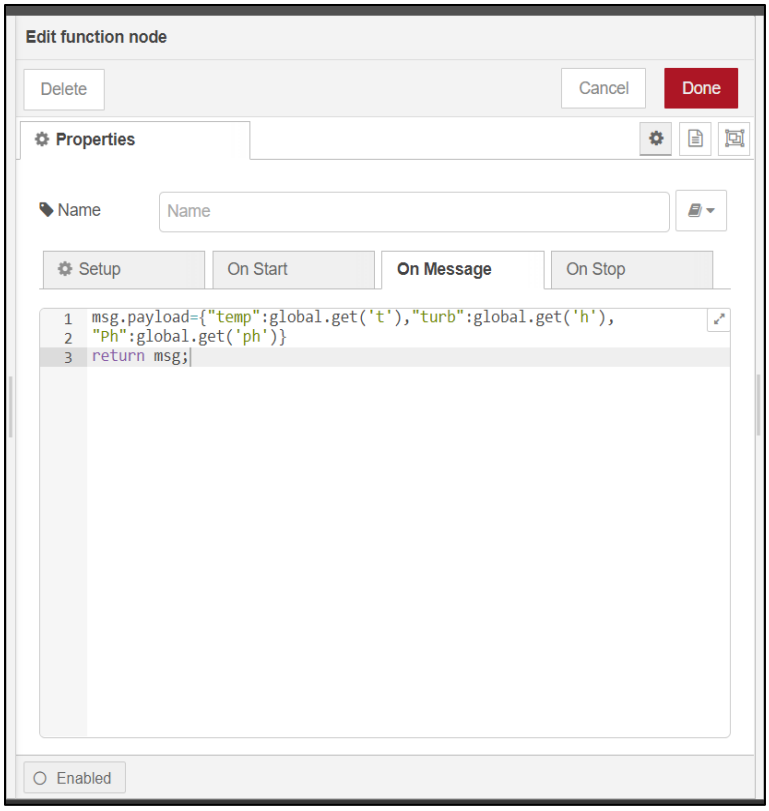
Node Red Dashboard

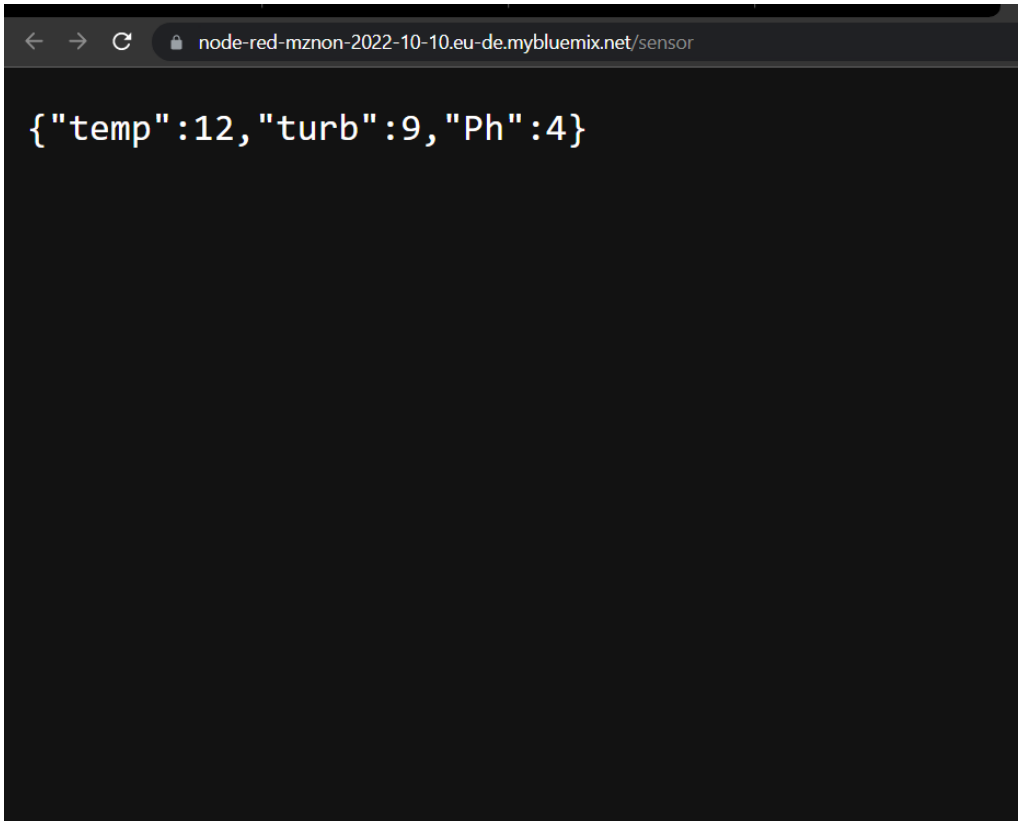




Http request to connect with MIT App




Edit function node
Delete
Cancel
Done
Properties
Name
Name
Setup
On Start
On Message
On Stop
1 msg.payload={"temp":global.get('t'),"turb":global.get('h'),
2 "Ph":global.get('ph')}
3 return msg;
Enabled


node-red-mznon-2022-10-10.eu-de.mybluemix.net/sensor
{"temp":12,"turb":9,"Ph":4}

8. TESTING

8.1 Test Case Analysis:

Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status
HomePage_TC_OO1	UI	Home Page	Verify user is able to see the Login popup when user clicks on the app	Click the app icon		Home page popup should display	Working as expected	Pass
LoginPage_TC_OO2	UI	Home Page	Verify the UI elements in Login popup	Enter the username and password	Username: 1234 Password: 1234	Login successfully	Working as expected	Pass
WaterLevel_TC_003	UI	Content Page	User is able to view the Temperature, Turbidity and PH values	Automatic		Water parameters are shown	Working as expected	Pass
AlertMessage_TC_004	UI	SMS Page	The user selects the contact for sending message	Click the select contact button and select the contact to whom you want to send the SMS	XYZ 1234567890	SMS is sent successfully	Working as expected	Pass

8.2 User Acceptance Testing

Resolution	**Severity 1**	**Severity 2**	**Severity 3**	**Severity 4**	**Subtotal**
By Design	9	5	4	3	21
Duplicate	2	0	2	0	4
External	3	1	4	2	10
Fixed	14	1	6	10	31
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	0	1
Won't Fix	0	0	2	1	3
Totals	22	11	14	25	72

9. RESULT

9.1 Performance table:

PARAMETER	PERFORMANCE	DESCRIPTION
ADMIN TESTING	95%-100%	The testing is done before it is deployed as an app
CUSTOMER SATISFACTION	75-85%	The customer needs to be satisfied with the mobile application
USER INTERFACE	65-85%	The app can used by anyone. (Ease of access)
SEVER RESPONSE	50-75%	URL - response
DATA VALIDATION WITH NO. OF TEST CASE	60-80% (15-30 TESTCASE)	Valid data from the app
ERROR	3-5%	Real-time delay may occur

10. ADVANTAGES AND DISADVANTAGES

Advantages:

- The prototype developed for river water quality monitoring helps to safeguard human health and ecological balance.
- The use of sensors, microcontrollers and software makes the prototype more efficient.
- The web application on developed is very useful in updating the pollution level at regular intervals.

Disadvantages:

- It is difficult to implement the automation system in various water bodies.
- High design and maintenance cost.

11. CONCLUSION

River Water Monitoring System uses a Raspberry Pi and an existing Cloud system to measure temperature, turbidity, and ph. This system is low-cost, does not require someone to be on duty, and can automatically check water quality. It also promptly activates alerts to avert any health dangers. As a result, the method is probably more efficient, practical, and quick. The method is very adaptable. Other water quality criteria can only be monitored with this system by updating the required sensors and software applications. The process is easy. It is possible to expand the system to track hydrologic conditions, air pollution, industrial and agricultural output, and other variables. It has a wide range of uses and extension value.

12. FUTURE SCOPE

The test outcomes have always been successful. We come to the conclusion that the suggested system has succeeded in achieving all of its goals. Other sensors can be added to the system to test more aspects of the water quality for some applications. The system may be used by all types of users and is accessible and economical.

It is possible to expand the system to track hydrologic conditions, air pollution, industrial and agricultural output, and other variables. It has a wide range of uses and extension value.

13. APPENDIX

13.1 Source Code

Python Code to Publish Data

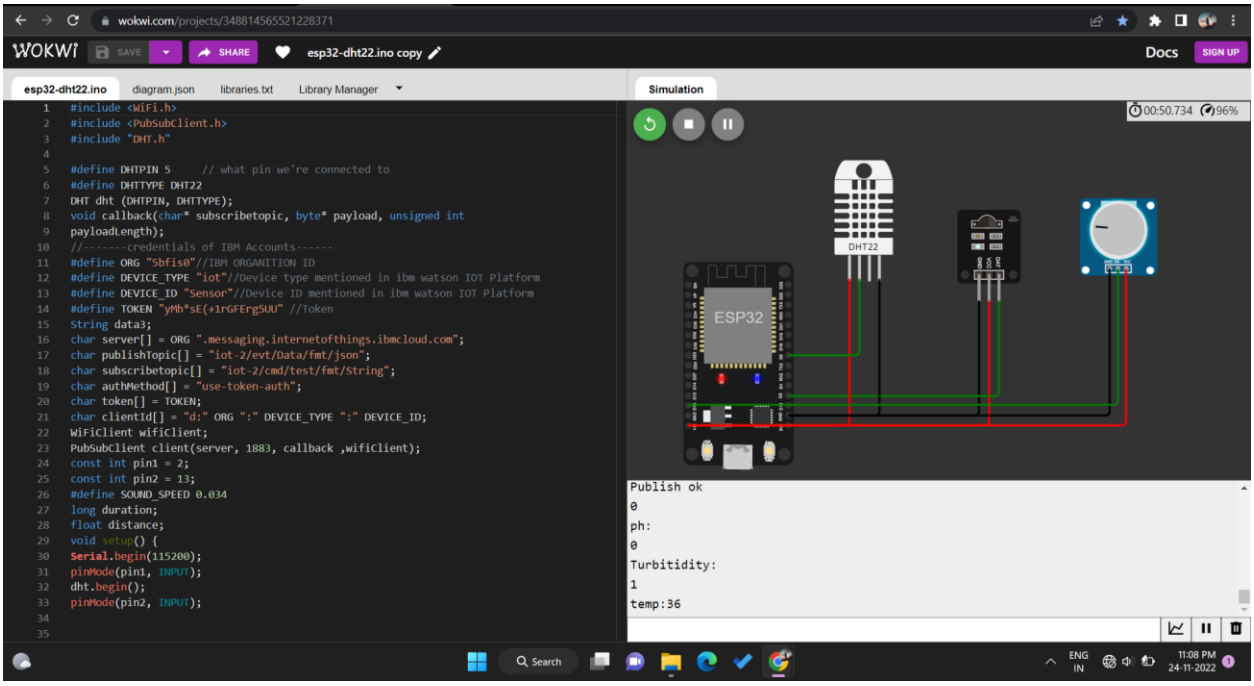
```
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"

#define DHTPIN 5     // what pin we're connected to
#define DHTTYPE DHT22
DHT dht (DHTPIN, DHTTYPE);
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
//-------credentials of IBM Accounts------
#define ORG "5bfis0"//IBM ORGANITION ID
#define DEVICE_TYPE "iot"//Device type mentioned in ibm watson IOT
Platform
#define DEVICE_ID "Sensor"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "yMh*sE(+1rGFErg5UU" //Token
String data3;
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
const int pin1 = 2;
const int pin2 = 13;
#define SOUND_SPEED 0.034
long duration;
float distance;
```

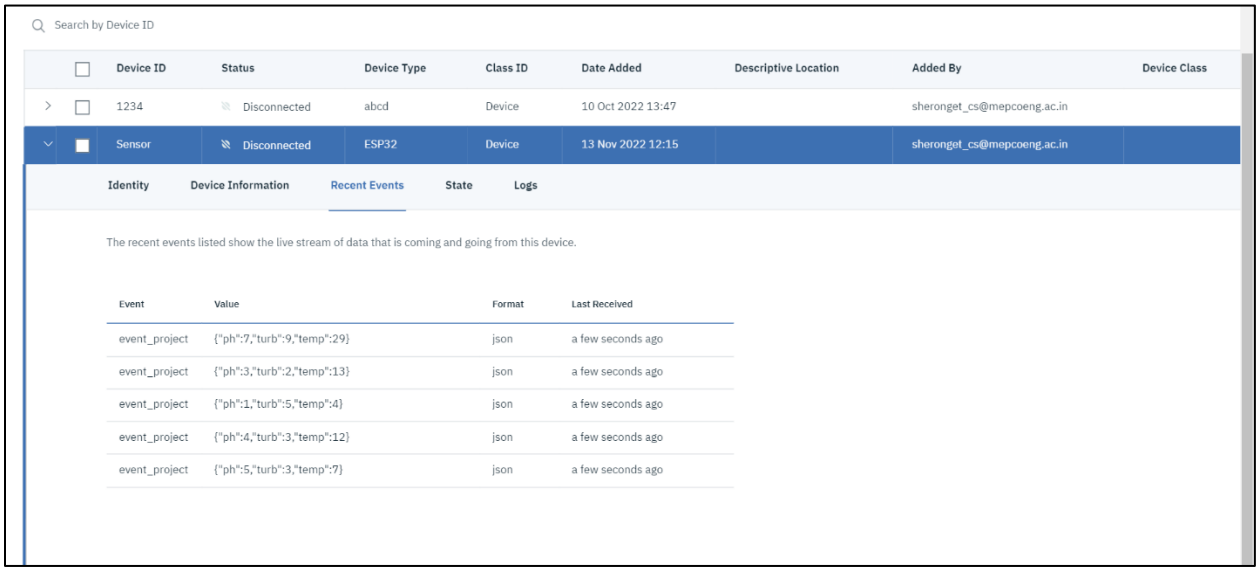
```
void setup() {
Serial.begin(115200);
pinMode(pin1, INPUT);
dht.begin();
pinMode(pin2, INPUT);


wificonnect();
mqttconnect();
}
void loop()
{
 int turb = digitalRead(pin1);
  int val = analogRead(pin2);
  Serial.println(val);         // reads the value of the potentiometer (value between 0
and 1023)
 int ph = map(val, 0, 1023, 0, 180);      // scale it to use it with the servo (value
between 0 and 180)
   Serial.println("ph:");
  Serial.println(ph);                // sets the servo position according to the scaled value
  Serial.println("Turbitidity:");
  Serial.println(turb);
  int temp = dht.readTemperature();
  Serial.print("temp:");
  Serial.println(temp);

  delay(1000);
 if (turb == 1 || ph>8.5 || ph<6.5 || temp<10|| temp>23) // Check if the pin high or
not
  {
    Serial.println("Water is contaminated!");
    Serial.println("ALERT!!");
  }

PublishData(turb,ph,temp);
```

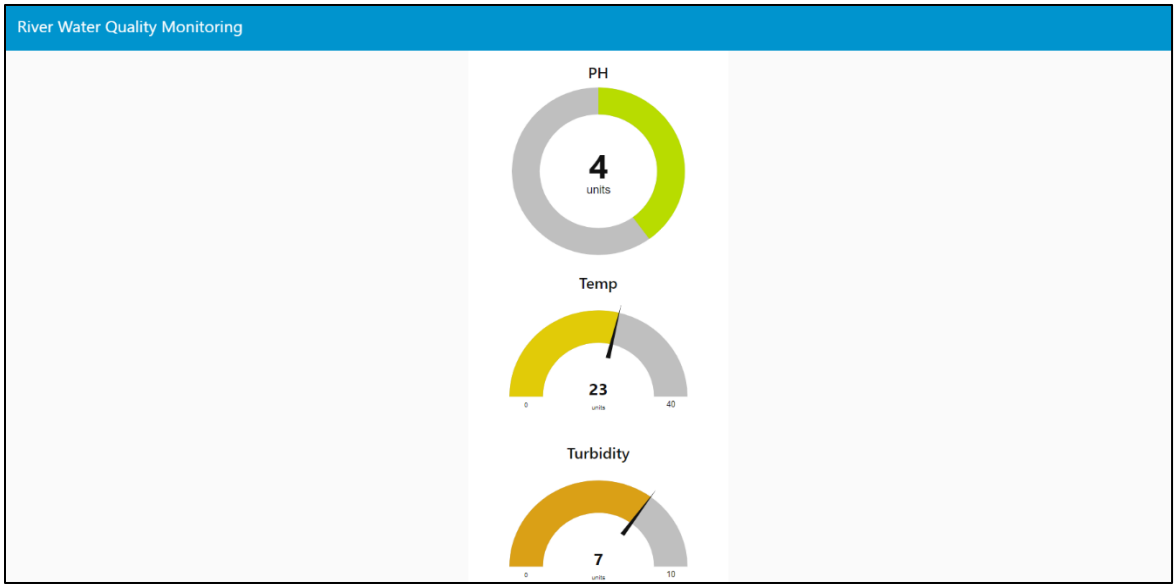
```
delay(1000);
if (!client.loop()) {
mqttconnect();
}
delay(1000);
}
void PublishData(int turb,int temp,int ph) {
mqttconnect();
String payload = "{\"ph\":";
payload += temp;
payload += "," "\"turb\":";
payload += turb;
payload += "," "\"temp\":";
payload += ph;
payload += "}";
Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("Publish ok");
} else {
Serial.println("Publish failed");
}
}
void mqttconnect() {
if (!client.connected()) {
Serial.print("Reconnecting client to ");
Serial.println(server);
while (!!!client.connect(clientId, authMethod, token)) {
Serial.print(".");
delay(500);
}
initManagedDevice();
Serial.println();
}
```

```
}
void wificonnect()
{
Serial.println();
Serial.print("Connecting to ");
WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}
void initManagedDevice() {
if (client.subscribe(subscribetopic)) {
Serial.println((subscribetopic));
Serial.println("subscribe to cmd OK");
} else {
Serial.println("subscribe to cmd FAILED");
}
}
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
Serial.print("callback invoked for topic: ");
Serial.println(subscribetopic);
for (int i = 0; i < payloadLength; i++) {
//Serial.print((char)payload[i]);
data3 += (char)payload[i];
}
Serial.println("data: "+ data3);
data3="";
}
```

13.2 OUTPUT

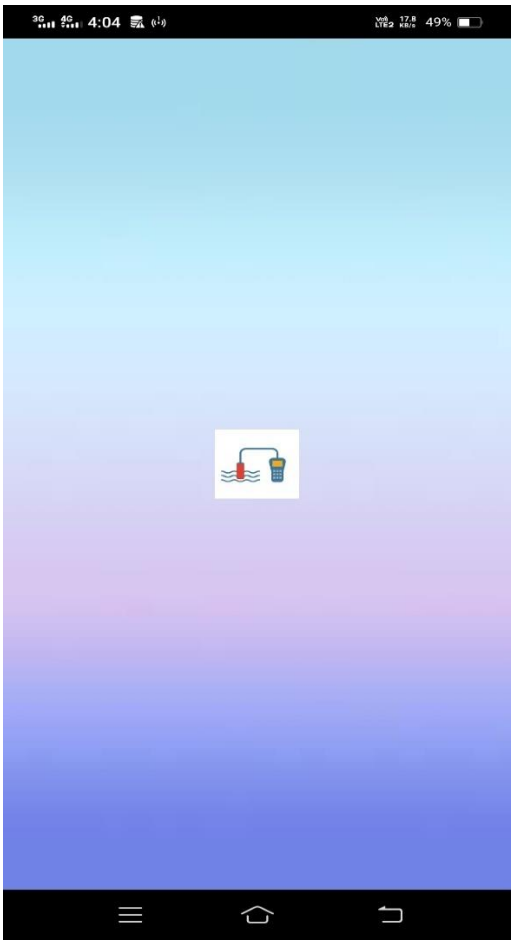

wokwi.com/projects/348814565521228371
WOKWI
SAVE
SHARE
esp32-dht22.ino copy
Docs
SIGN UP
esp32-dht22.ino
diagram.json
libraries.txt
Library Manager
Simulation
00:50.734
96%
DHT22
ESP32
Publish ok
0
ph:
0
Turbitidity:
1
temp:36
Search
ENG
IN
11:08 PM
24-11-2022


Search by Device ID
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
Device Class
1234
Disconnected
abcd
Device
10 Oct 2022 13:47
sheronget_cs@mepcoeng.ac.in
Sensor
Disconnected
ESP32
Device
13 Nov 2022 12:15
sheronget_cs@mepcoeng.ac.in
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
event_project
{"ph":7,"turb":9,"temp":29}
json
a few seconds ago
event_project
{"ph":3,"turb":2,"temp":13}
json
a few seconds ago
event_project
{"ph":1,"turb":5,"temp":4}
json
a few seconds ago
event_project
{"ph":4,"turb":3,"temp":12}
json
a few seconds ago
event_project
{"ph":5,"turb":3,"temp":7}
json
a few seconds ago



MOBILE APP

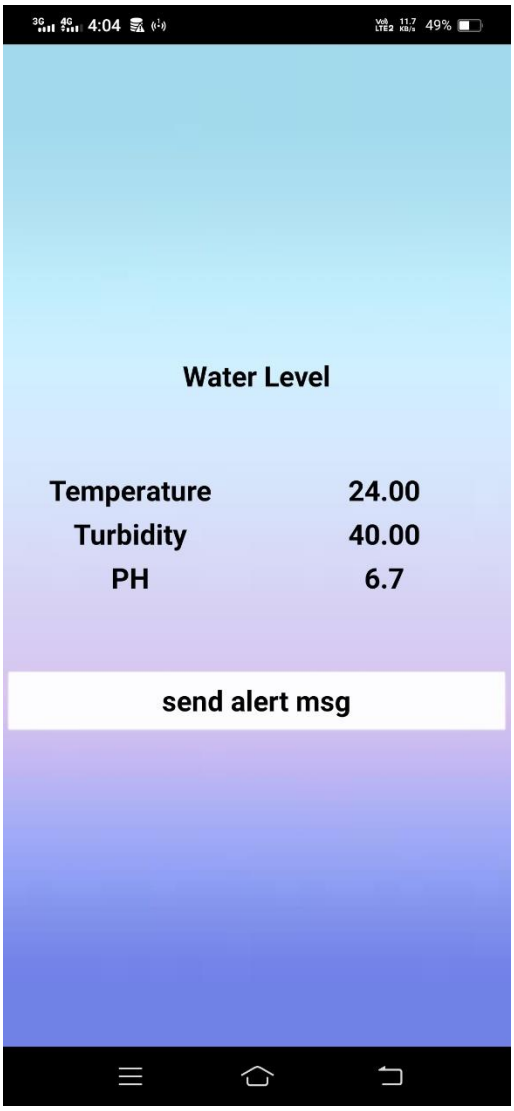
Home Page



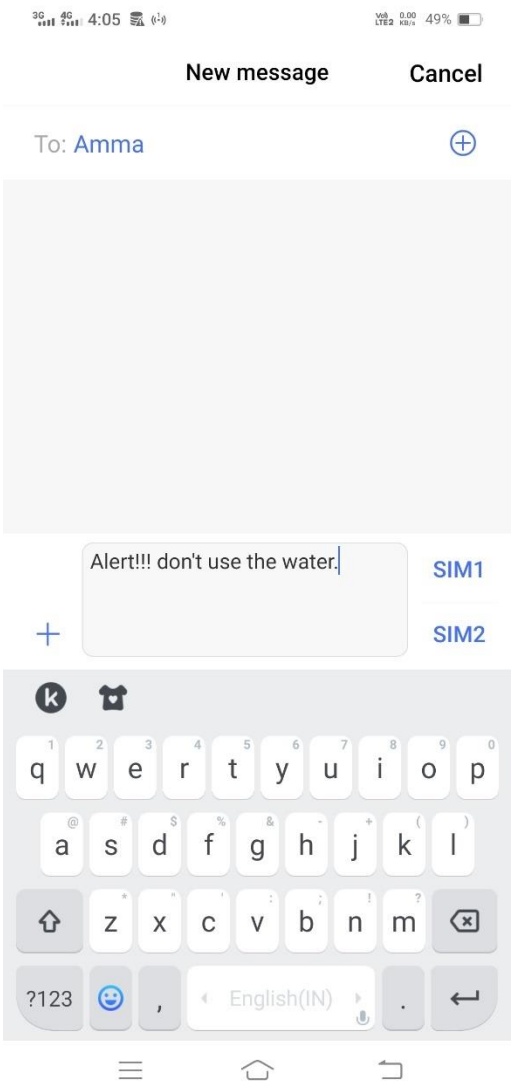
Login Page

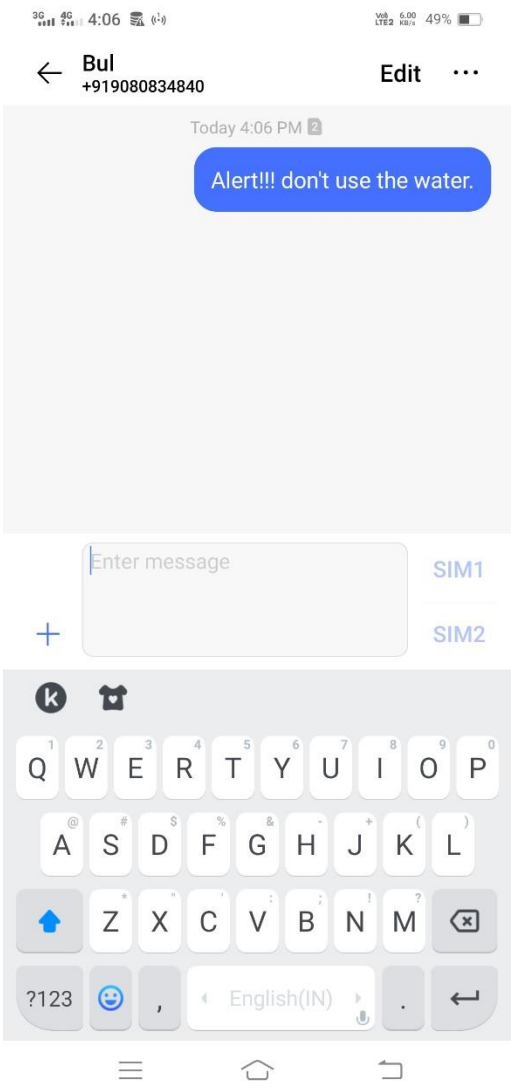
Login using username and password

Water Parameter display page



SMS Page





13.3 Git-Hub Link:

IBM-EPBL/IBM-Project-2078-1658426430: Real-Time River Water Quality Monitoring and Control System (github.com)

Project Demo Link:

https://drive.google.com/file/d/1xSxO3v0VLb04IKWKBXqir_xZEAg_kh1k/view?usp=share_link