FINAL DELIVETABLES - PROJECT REPORT

Real-Time River Water Quality Monitoring And Control System

DOMAIN	IoT
TOPIC	Real-Time River Water Quality Monitoring And Control System
TEAM MEMBER	1. Anto Shawn Roche A 2. Aswinth J 3. John Nikkith J 4. Kabin Bose Y
SUBMISSION DATE	19/11/2022

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

A real-time river monitoring and control system that is self-reliable. Monitoring different levels of water contamination with a probe. The web application allows for the monitoring of river water quality. Can determine whether the water contains any dust particles. It is possible to check the water's PH level. And population growth. Ensuring safe water supply of drinking water is big challenge for modern civilization. Traditional methods that rely on collecting water samples, testing and analyses in water laboratories are not only costly but also lack capability for real time. You may check the temperature of the water. If the water quality is poor, letting the authorities know so they may go and tell the community not to consume that water.

1.2 Purpose

The need for effective and efficient monitoring, evaluation and control of water quality in residential area has become more demanding in this era of urbanization, pollution data capture, analyses and fast dissemination of information to relevant stakeholders for making timely and informed decisions. In this time water quality monitoring system prototype developed for water quality monitoring in Residential home is presented. The development was preceded by evaluation of prevailing environment including availability of cellular network coverage paper, a real at the site of operation. It detects water temperature, dissolved oxygen, pH, and electrical conductivity in real-time and disseminates the information in graphical and tabular formats to relevant stakeholders through a web-based portal and mobile phone platforms. The experimental results show that the system has great prospect and can be used to operate in real world environment for optimum control and protection of water resources by providing key actors with relevant and timely information to facilitate quick action taking.

2. LITERATURE SURVEY

2.1 Existing problem

Water has become the world's most important resource due to the scarcity of available supplies and the dangers posed by contamination.

The development of the modern way of life is also one of the factors contributing to the growing threat of water-borne diseases and water scarcity.

Therefore, in order to eliminate diseases and increase the availability of water, we need to monitor the quality as well as implement a control system. A water quality monitoring system is to be proposed using components of wessirel sensors network with the assistance of IOT that checks all the quality parameters and provides better performance rate with perfect accuracy.

2.2 References

1. IoT Based Real-time River Water Quality Monitoring System by Mohammad

Salah Uddin Chowdurya, Talha Bin Emranb, Subhasish Ghosha, Abhijit Pathaka, Mohd. Manjur Alama, Nurul Absara, Karl Anderssonc, Mohammad Shahadat Hossaind published in 2019, uses different sensors to measure water parameters such as pH, dissolved oxygen, turbidity, conductivity and etc. assembles data from these sensor nodes and send it to the base station by the wireless channel

2. Development and Implementation of Water Quality Assessment Monitoring (WQAM) System using the Internet of Things (IoT) in Water Environment by Muhammad Farhan Johan, Samihah Abdullah , Nor Shahanim published in 2021,A cloud storagebased system that uses two devices to monitor water at the center of the lake and by the bank of the lake. It also uses sensors to measure pH level, turbity, conductivity. Uses Thinkspeak platform

3. IoT-Enabled Water Quality Monitoring System by G.kanagaraj and T.Primiya published in 2020 uses controller with inbuilt Internet connectivity module to monitor parameters such as temperature and turbidity using low cost and less complex smart water quality monitoring system. The system contains an appropriate webpage for enhancing the user convenience on the deviation of water quality parameters.

2.3 Problem Statement Definition

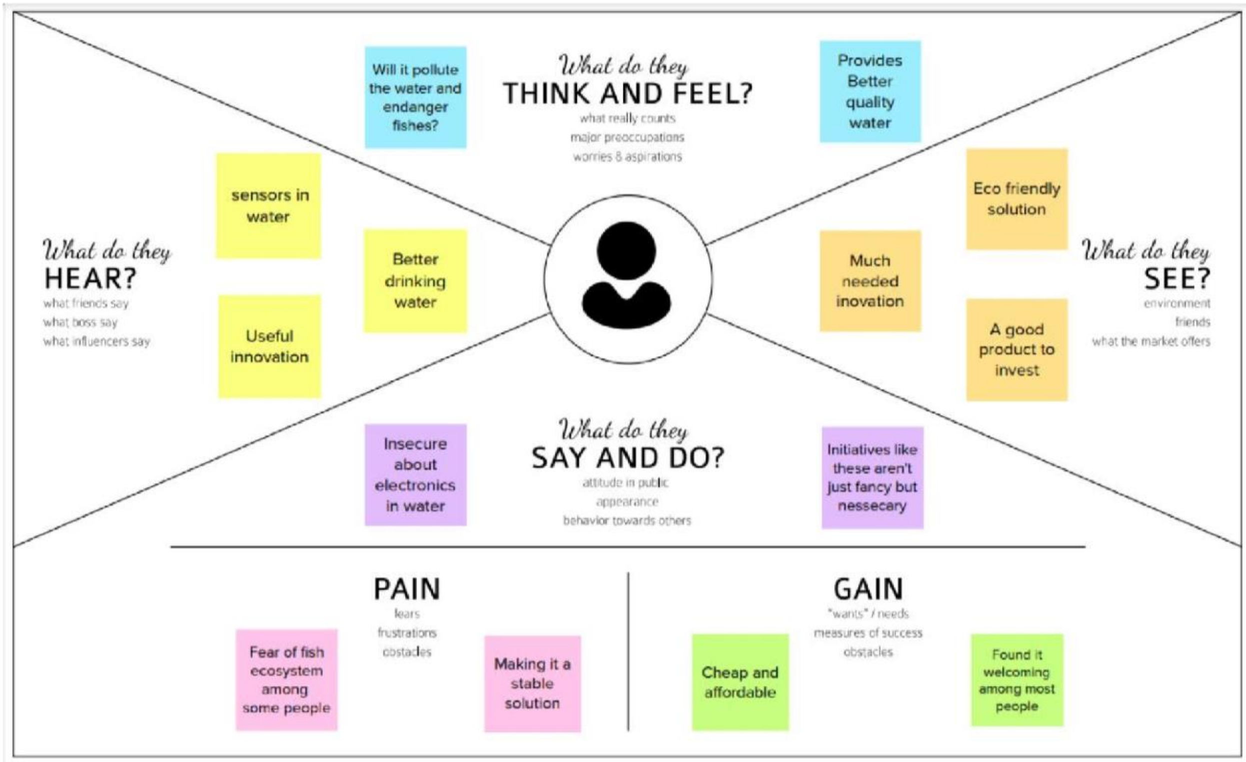
The current technique for monitoring water quality is manual, tedious, and time consuming.

IOT technology and remote monitoring allow for real-time data access.

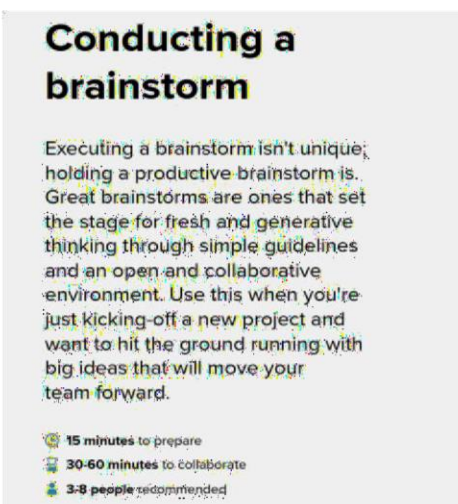
The goal of the proposed article is to develop a low-powered, highly mobile, and frequent water monitoring device.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

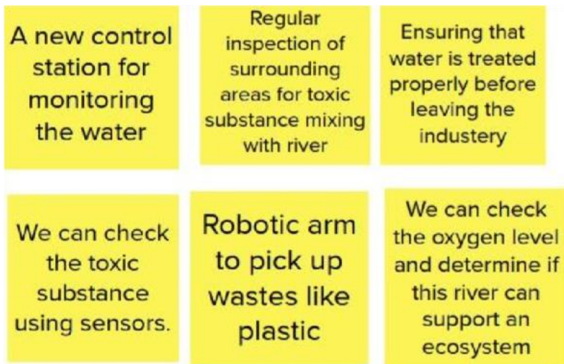


3.2 Ideation & Brainstorming

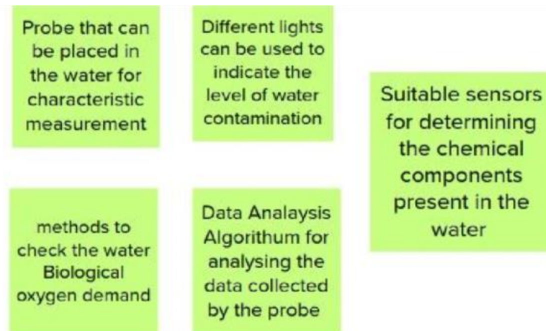


Stickey Notes

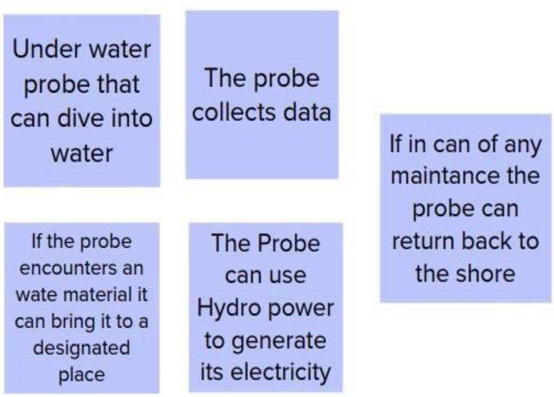
Anto Shawn Roche A



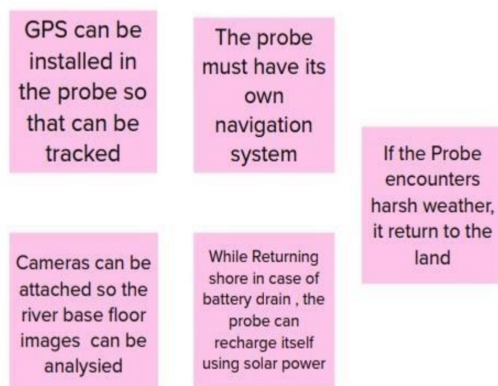
Aswinth J

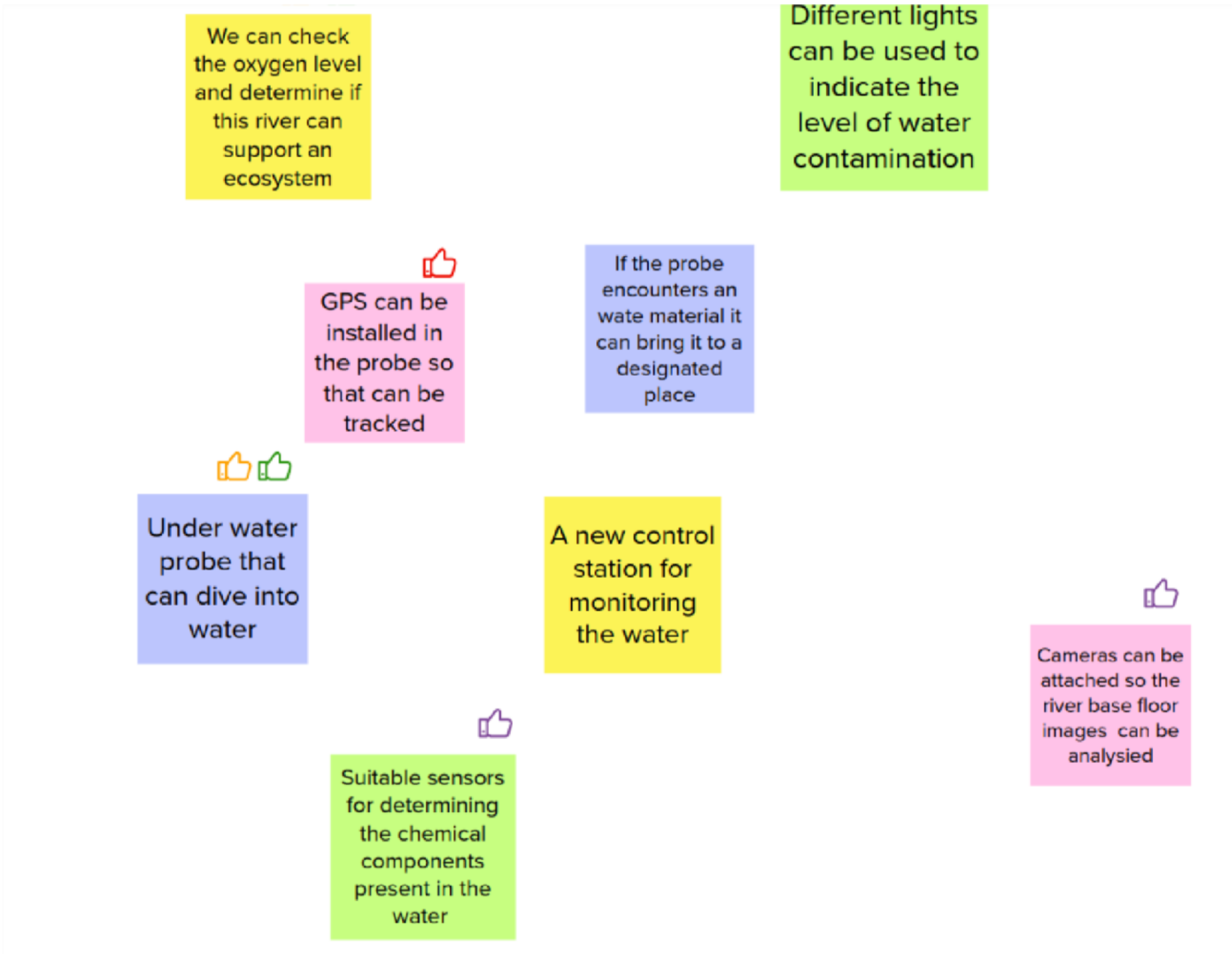


John Nikkith J



Kabin Bose Y




We can check the oxygen level and determine if this river can support an ecosystem
Different lights can be used to indicate the level of water contamination
GPS can be installed in the probe so that can be tracked
If the probe encounters an wate material it can bring it to a designated place
Under water probe that can dive into water
A new control station for monitoring the water
Cameras can be attached so the river base floor images can be analysied
Suitable sensors for determining the chemical components present in the water

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To develop a self-reliable real time river monitoring and control system.
2.	Idea / Solution description	Using probe to monitor various level of contamination in the water. Using the same probe to collect the waste.
3.	Novelty / Uniqueness	The solution does not exist, and existing other solutions does not include warning systems which includes flood warning.
4.	Social Impact / Customer Satisfaction	Helps to reduce crop losses due to water contamination, spread of water borne diseases, maintain good water ecosystem and faster evacuation during flood times. Its design makes it suitable for almost all environmental conditions.
5.	Business Model (Revenue Model)	Using a self-reliable probe and sensors to achieve this.
6.	Scalability of the Solution	Sustainability. This can be mass produced

3.4 Problem Solution fit

Project Title: Real time water quality monitoring and control

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID14819

1. CUSTOMER SEGMENT(S) CS

The target is Government bodies which monitor river quality and private NGOs or private environmental enthusiastic organizations.

6. CUSTOMER CONSTRAINTS CC

The target is Government bodies which monitor river quality and private NGOs or private environmental enthusiastic organizations.

5. AVAILABLE SOLUTIONS AS

In India we mostly rely on government water treatment plants for solving this issue. However, this cannot keep track of all the wastewater. This is only applicable in cities.

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

Detect water quality and alert the government officials for any anomalities.

9. PROBLEM ROOT CAUSE RC

The main cause for this problem is industries not treating their water before releasing it into rivers. The next cause is that people release their household wastes in rivers. During rainy seasons water from agricultural land washes the fertilizers and mix it with river.

7. BEHAVIOUR BE

With the available solutions the government checks the industries regularly, but they are not able to find the exact region where the water stars to get polluted.

3. TRIGGERS TR

By participating in hackathons or by proposing this to NGOs. Demonstrating its efficiency. Social media.

10. YOUR SOLUTION SL

We propose an probe which can monitor the river water for pollutants or harmful materials. This will also have the capacity for detecting floods and give warning.

8. CHANNELS of BEHAVIOUR CH

Online: -The government has online forum for people to complaint about illegal disposing of water into river without treatment.

Offline – The government checks the industries regularly and NGOs take some social activities to clean rivers.

4. EMOTIONS: BEFORE / AFTER EM

There are lot of family that depend on river for their day today needs. The pollution has made their life a life of suffering this can bring Joy to people. This also can be said true for agriculture lands near contaminated rivers as they can directly get water from river for irrigation.

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 Registration	Registration through Gmail Registration through website Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Historical Data	The date is stored in the cloud from the beginning stage till the update
FR-4	pH detection	To monitor the water quality pH sensor is used
FR-5	O_2 Detection	The level of O_2 in the water is constantly measured

4.2 Non-Functional requirements

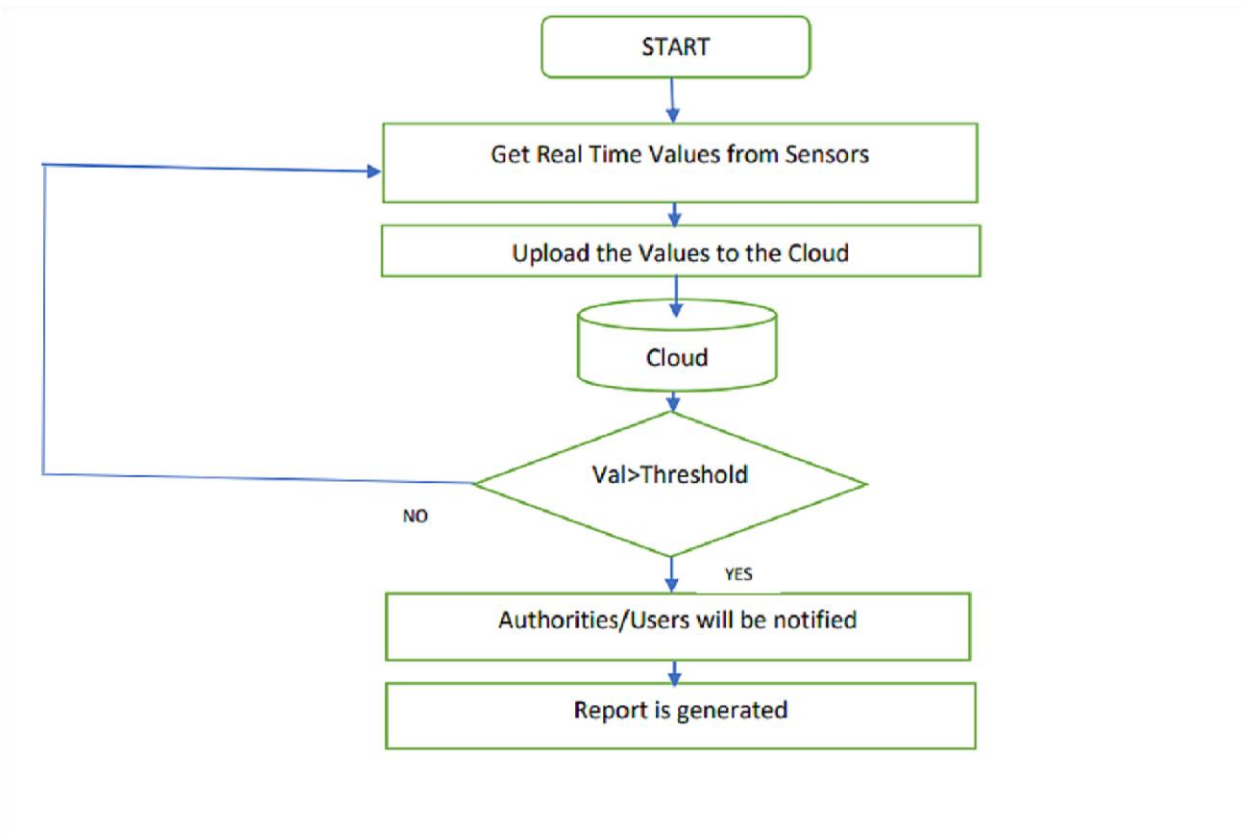
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It should be easy to monitor the data
NFR-2	**Security**	Unauthorized users should not be allowed to access the data
NFR-3	**Reliability**	If there is a sensor fault the message must be informed to the authority
NFR-4	**Performance**	High quality sensor are used to maximize the performance
NFR-5	**Availability**	The Data should be accessible 24/7
NFR-6	**Scalability**	The system must be compact and easily transported
NFR-7	**Efficiency**	It should consume low power and provides highly accurate output

1

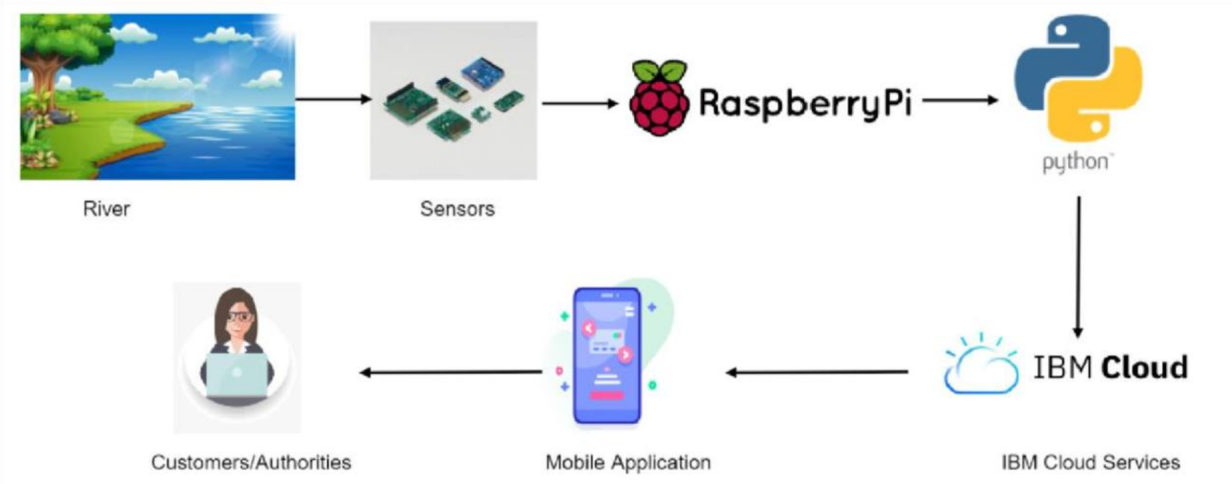
5. PROJECT DESIGN

5. Data Flow Diagrams



5.2 Solution & Technical Architecture

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.



1

5.3 User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register through mail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can receive log in details	High	Sprint-1
Customer (Web user)	Dashboard	USN-6	As a user I should be able to access specific parameter	I can able to know the quality of water	High	Sprint-2
Administrator	Risk tolerant	USN-7	An admin should update and take care of the application	Admin should monitor the records properly	Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

Sprint	Functional Requirement (Epic)	User Story Number	User Story/task	Story Points	Priority	Team Members

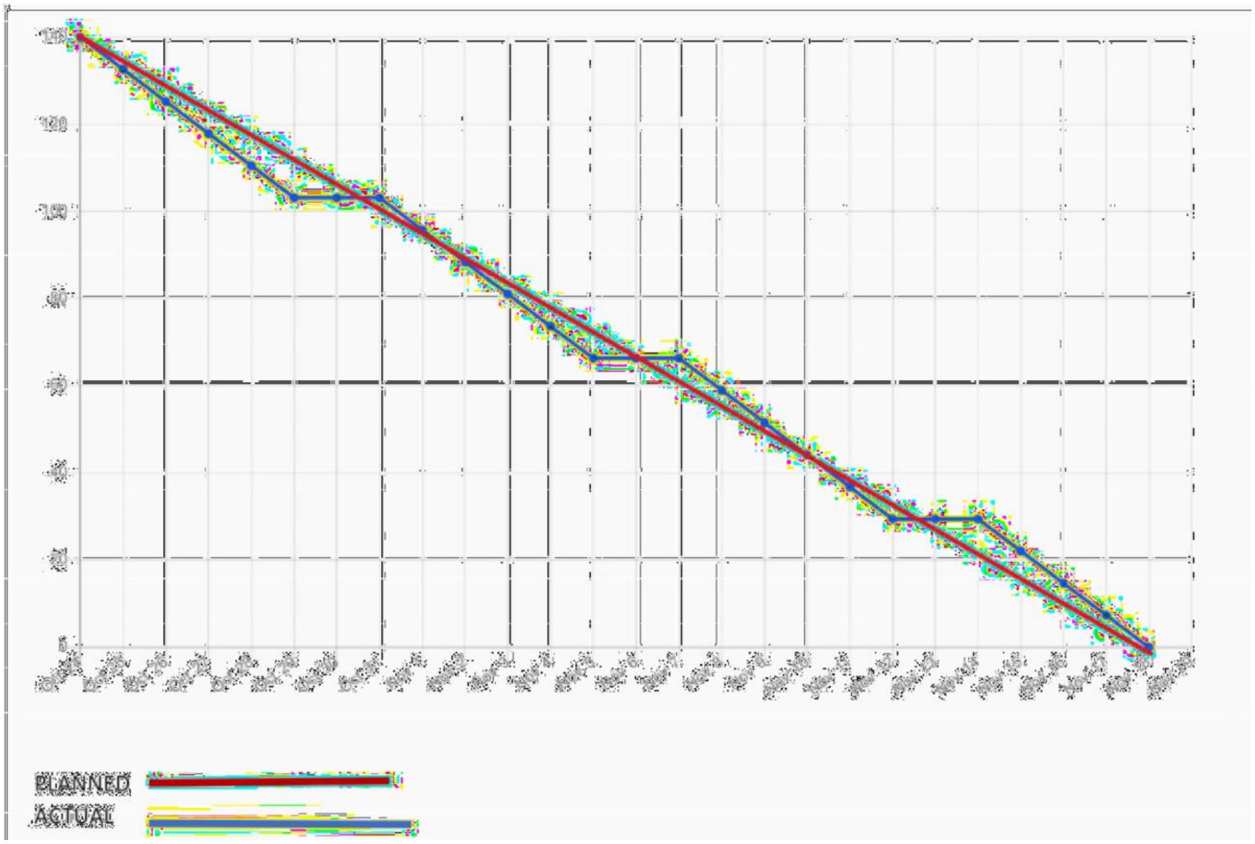
Sprint1	Simulation creation	USN-1	Connect Sensors and arduino with python code	2	High	Anto Shawn Roche A Aswinth J Kabin Bose Y John Nikkith J
Sprint2	Software	USN-2	Creating device in the IBM Watson IoT platform, workflow for IoT scenarios using Node-Red	2	High	Anto Shawn Roche A Aswinth J Kabin Bose Y John Nikkith J
Sprint3	MIT App Inventor	USN-3	Develop an application for the real time river water quality management project using MIT App Inventor	2	High	Anto Shawn Roche A Aswinth J Kabin Bose Y John Nikkith J
Sprint4	Dashboard	USN-4	Design the modules and test the app	2	High	Anto Shawn Roche A Aswinth J Kabin Bose Y John Nikkith J
Sprint5	Web UI	USN-5	To make the user to interact with software.	2	High	Anto Shawn Roche A Aswinth J Kabin Bose Y John Nikkith J

6.2 Sprint Delivery Schedule

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	20	6Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint 2	20	6Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint 3	20	6Days	07 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint 4	20	6Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

1

6.3 Reports from JIRA



7. CODING & SOLUTIONING

7. pH Sensor

Given that low and graduate pH values can have significant effects on the author, the pH of something is a helpful constant to display. A statement's pH ranges from 1 to 14. A pH sensor is a piece of equipment that gauges the hydrogen-ion density in bleach to determine how acidic or alkaline it is. Its composition ranges from pH 0 to pH 14. pH values also process the solubility of elements and compounds making them cyanogenetic. Mathematically pH is referred as, pH = -log [H+].

7.2 Temperature Sensor

The water temperature sensor has a resolution of 0.1 degree and can measure temperatures in the range of -5 degrees Celsius to +50 degrees Celsius (or 23 degrees Fahrenheit to 122 degrees Fahrenheit). Our testing tools for water quality are designed to use in the field.

7.3 Turbidity sensor

The purity of the element or muddiness of the water is measured using a turbidity train sensor. The open sliced food typically has muddiness between 255 NTU. Irrigation is evident at levels higher than 80 NTU. 130 NTU to 250 NTU is the range for intemperance liquid standards. The transmitter must transmit an unsubtle bright signal in order for the turbidity device, which consists of a soft sender and acquirer, to be considered operational. Turbidity reduces water clarity, is unsightly, slows photosynthesis, and increase water temperature.

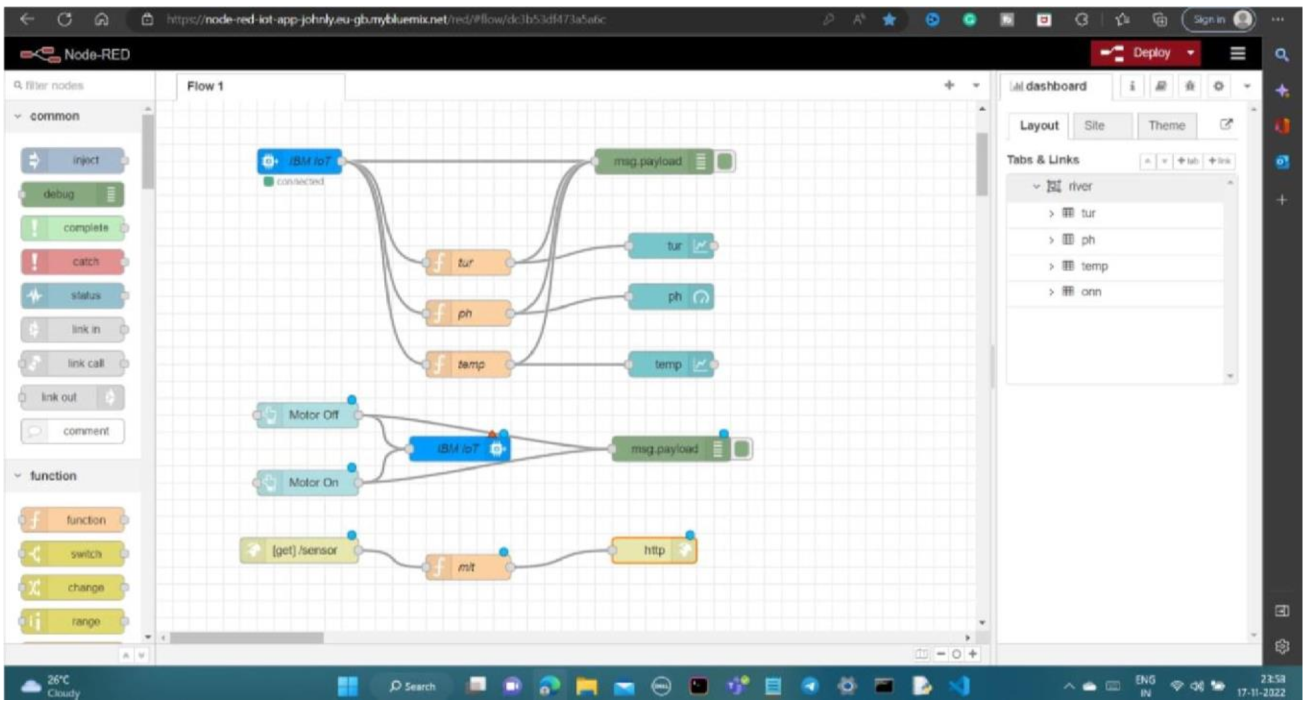
Code

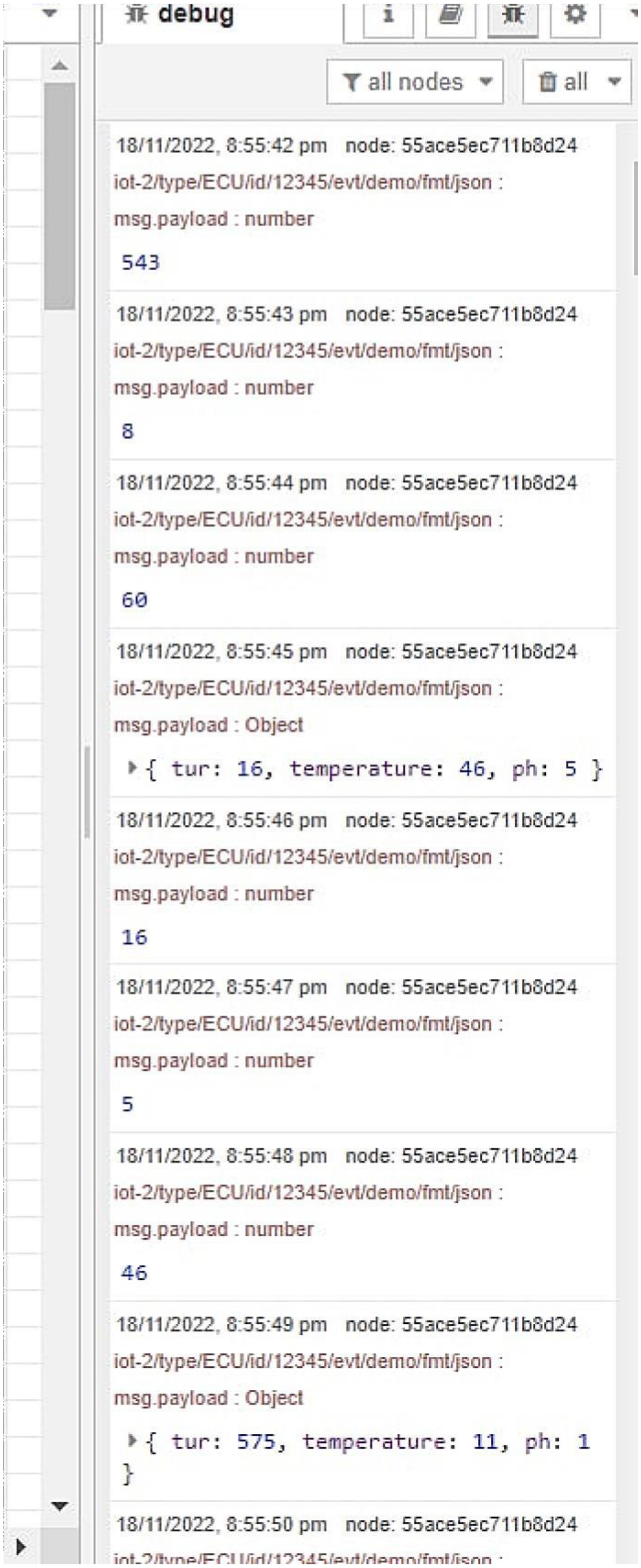
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#IBM WATSON Credentials
organization = "0mfbus"
deviceType = "ECU"
deviceId = "12345"
authMethord ="token"
authToken = "12345678"


def myCommandCallback(cmd):
    print("Command Received: %s" % cmd.data['command'])
    status = cmd.data['command']
    if status == "lighton":
        print("Led is on")
    elif status == "lightoff":
        print("Led is off")
    else:
        print("Proper command required")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method":authMethord,"auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)

except Exception as e:
    print("caught exception connecting device: %s" % str(e))
    sys.exit()

deviceCli.connect()
```

```
while True:

    turbidity = random.randint(1, 1000)
    temp=random.randint(-20,125)
    pH=random.randint(0,14)
    myData={'tur':turbidity,'temperature':temp, 'ph':pH}


    def myOnPublishCallback():
        print("Published pH= %s" % pH, "Turbidity:%s" % turbidity, "Temperature:%s" %temp)

    success = deviceCli.publishEvent("demo", "json", myData, qos=0,on_publish=myOnPublishCallback)

    if not success:
        print("Not Connected to ibmiot")
    time.sleep(5)

    deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
```

8. TESTING

8.1 Test Cases

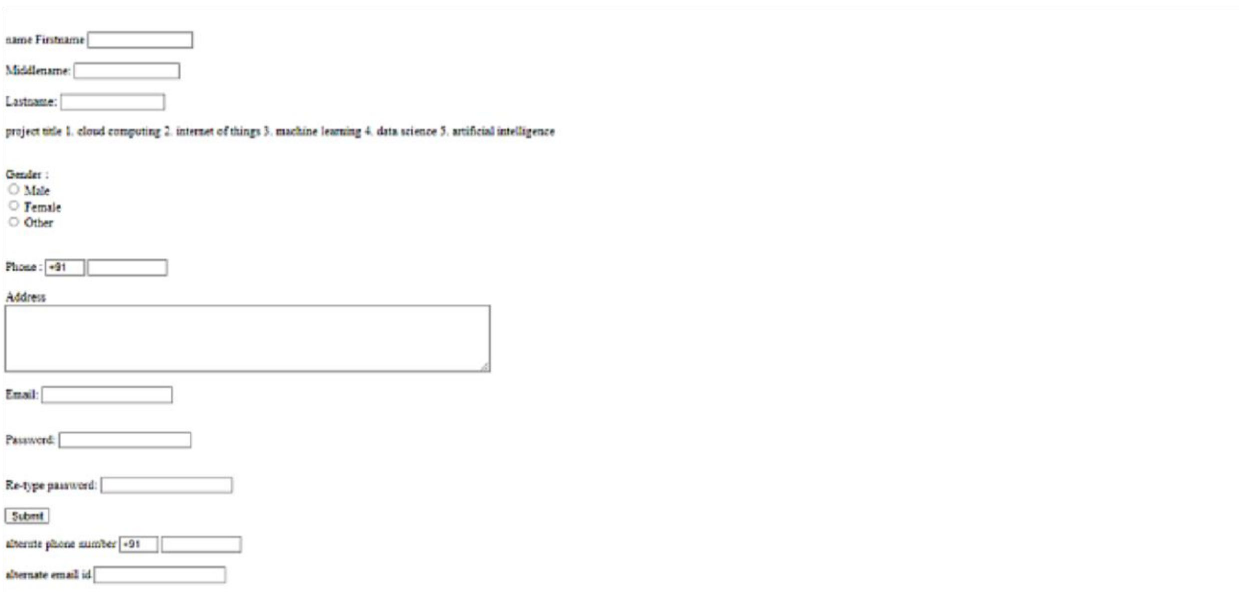



debug
all nodes
all
18/11/2022, 8:55:42 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
543
18/11/2022, 8:55:43 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
8
18/11/2022, 8:55:44 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
60
18/11/2022, 8:55:45 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : Object
{ tur: 16, temperature: 46, ph: 5 }
18/11/2022, 8:55:46 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
16
18/11/2022, 8:55:47 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
5
18/11/2022, 8:55:48 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : number
46
18/11/2022, 8:55:49 pm node: 55ace5ec711b8d24
iot-2/type/ECU/id/12345/evt/demo/fmt/json :
msg.payload : Object
{ tur: 575, temperature: 11, ph: 1 }
18/11/2022, 8:55:50 pm node: 55ace5ec711b8d24

```
*Python 3.7.0 Shell*
File  Edit  Shell  Debug  Options  Window  Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: C:\Users\Johnly7\Desktop\ibmiot.py ================
2022-11-18 20:30:42,833   ibmiotf.device.Client      INFO    Connected successfu
lly: d:0mfbus:ECU:12345
Published pH= 1 Turbidity:221 Temperature:-5
Published pH= 9 Turbidity:949 Temperature:75
```

```
Published pH= 78 Turbidity:471 Temperature:-11
Published pH= 12 Turbidity:69 Temperature:107
Published pH= 24 Turbidity:802 Temperature:4
Published pH= 99 Turbidity:96 Temperature:47
Published pH= 73 Turbidity:167 Temperature:1
Published pH= 93 Turbidity:821 Temperature:-18
Published pH= 3 Turbidity:598 Temperature:58
Published pH= 15 Turbidity:786 Temperature:-8
Published pH= 48 Turbidity:927 Temperature:56
Published pH= 46 Turbidity:293 Temperature:1
Published pH= 23 Turbidity:438 Temperature:49
Published pH= 9 Turbidity:437 Temperature:107
Published pH= 25 Turbidity:816 Temperature:19
```

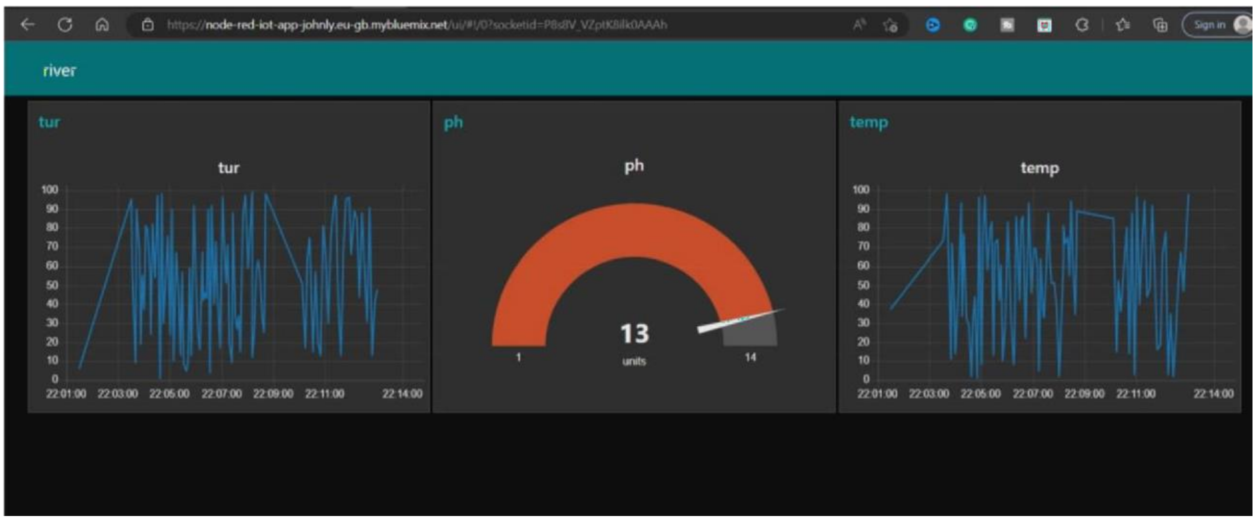
8.2 User Acceptance Testing

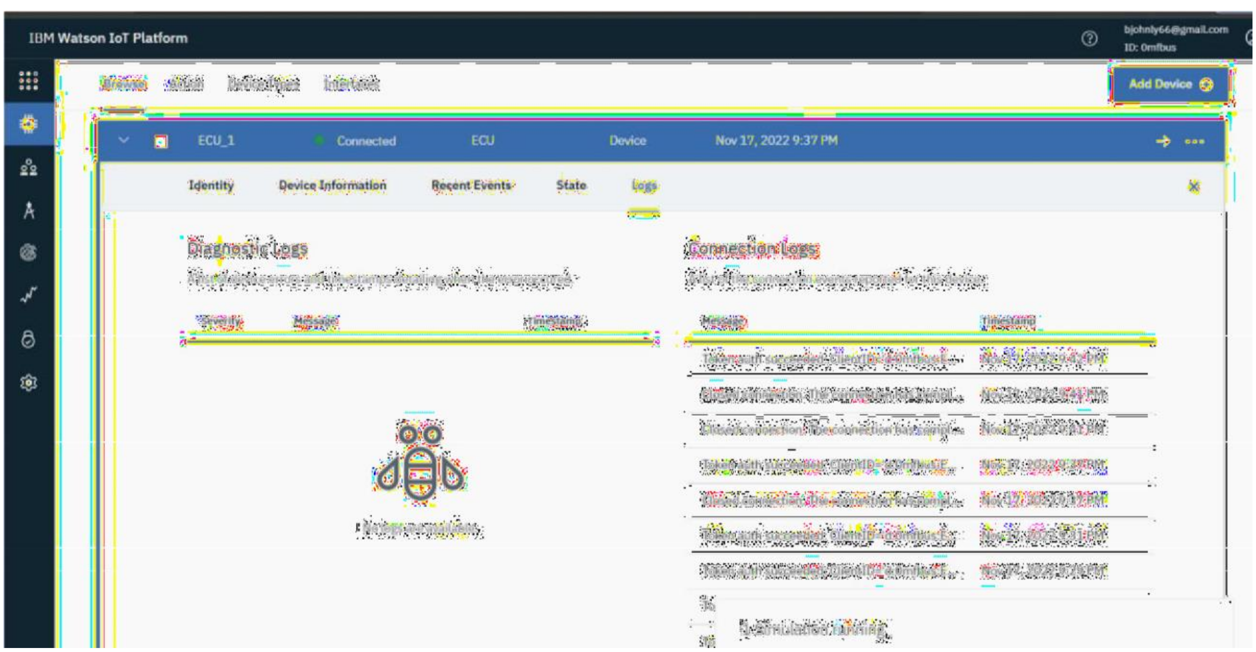


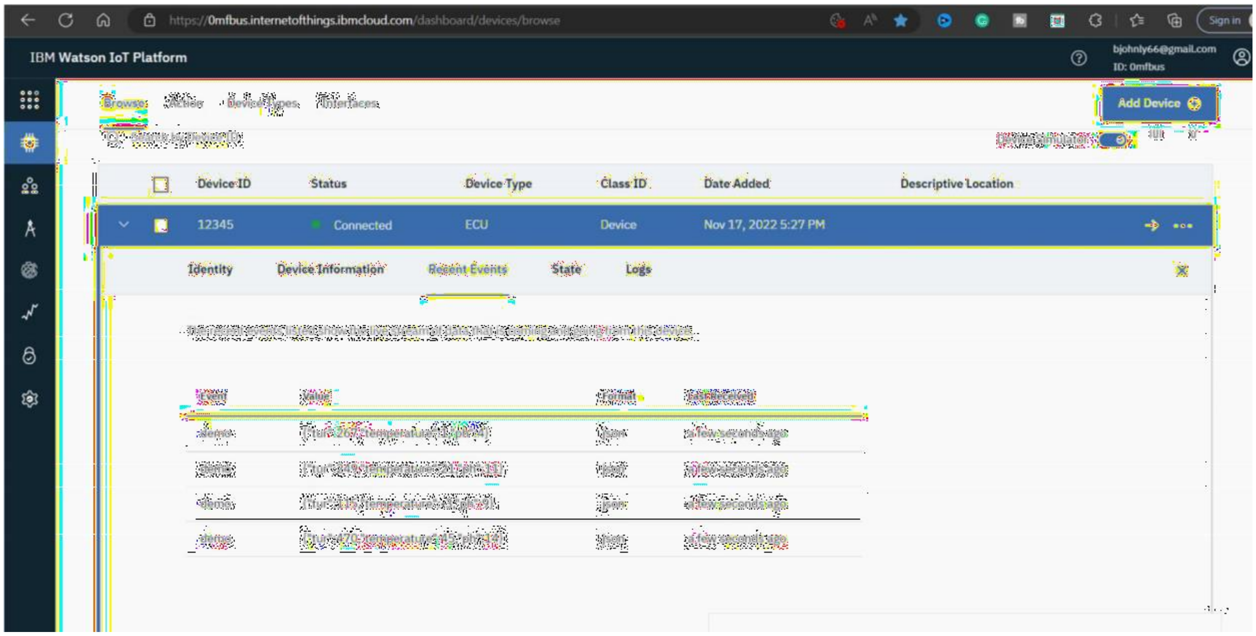
9. RESULTS

9.1 Performance Metrics

It continuously senses the values of pH, temp, turbidity and the resulting values are displayed to the LCD, PC or mobile in real-time.






https://0mfbus.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
bjohnly66@gmail.com
ID: 0mfbus
Add Device
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
12345
Connected
ECU
Device
Nov 17, 2022 5:27 PM
Identity
Device Information
Recent Events
State
Logs

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES	DISADVANTAGES
Remote river water quality monitoring is possible.	Control System is not much efficient
Easily access information about river water.	Human intervention is required if any anomalies occur.
Know real time information continuously.	Not fully automated
Multiple device data access is possible.	There is a need for internet in devices.
Login Credentials are required which enable data secuirity.	Only authorities will recieve messages first.
Directly communicates with authorities.	Cloud data may be hacked.
Increased water quality	User infos may be hacked.
pH, Temperature and turbidity monitroring is possible	SMS service may not work all the time.

10.CONCLUSION

Real-time water quality monitoring utilising IoT integrated data analytics would greatly assist individuals in becoming aware of the dangers of using contaminated water as well as in stopping water pollution. Real-time river water quality monitoring is the main focus of the research. Because IoT integrated data analytics can offer dependability, scalability, speed, and permanence, it appears to be a better solution.

11. FUTURE SCOPE

We only measure the characteristics that affect the quality of river water due to budgetary constraints. This project could be expanded into a productive local water management system.

Additionally, other characteristics that were outside the purview of this experiment can also be quantified, including total dissolved solids, chemical oxygen demand, and dissolved oxygen. Therefore, greater funding is needed to continue to enhance the system as a whole.

12 .APPENDIX

Source Code

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#IBM WATSON Credentials
organization = "0mfbus"
deviceType = "ECU"
deviceId = "12345"
authMethord ="token"
authToken = "12345678"


def myCommandCallback(cmd):
    print("Command Received: %s" % cmd.data['command'])
    status = cmd.data['command']
    if status == "lighton":
        print("Led is on")
    elif status == "lightoff":
        print("Led is off")
    else:
        print("Proper command required")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method":authMethord,"auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)

except Exception as e:
    print("caught exception connecting device: %s" % str(e))
    sys.exit()

deviceCli.connect()
```

```
while True:

    turbidity = random.randint(1, 1000)
    temp=random.randint(-20,125)
    pH=random.randint(0,14)
    myData={'tur':turbidity,'temperature':temp, 'ph':pH}


    def myOnPublishCallback():
        print("Published pH= %s" % pH, "Turbidity:%s" % turbidity, "Temperature:%s" %temp)

    success = deviceCli.publishEvent("demo", "json", myData, qos=0,on_publish=myOnPublishCallback)

    if not success:
        print("Not Connected to ibmiot")
    time.sleep(5)

    deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
```

GitHub &Project Demo Link

GitHub Link : https://github.com/IBM-EPBL/IBM-Project-44493-1660724899

Demo Video Link : https://youtu.be/NlNIz6jsFsk