

IBM - NALAIYA THIRAN

PROJECT DOCUMENTATION

on

**REAL-TIME RIVER WATER QUALITY MONITORING AND
CONTROL SYSTEM**

Submitted by

VIMALKARTHIK J	AC19UCS137
VINAYAGA MOORTHY R	AC19UCS138
VISHAL K	AC19UCS139
CHARAN ROHITH D	AC19UCS146

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ADHIYAMAAN COLLEGE OF ENGINEERING

DR. M.G.R NAGAR, HOSUR-635130

REPO ID: [IBM-Project-12889-1659498368](#)

TEAM ID: PNT2022TMID08054

FACULTY MEMBER: V Kalaivani

INDUSTRIAL MEMBER: Bharadwaj

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE.
1	INTRODUCTION	
	1.1 Project Overview	5
	1.2 Purpose	5
2	LITERATURE SURVEY	
	2.1 Existing problem	6
	2.2 References	6
	2.3 Problem Statement Definition	7
3	IDEATION & PROPOSED SOLUTION	
	3.1 Empathy Map Canvas	8
	3.2 Ideation & Brainstorming	9
	3.3 Proposed Solution	12
	3.4 Problem Solution fit	13
4	REQUIREMENT ANALYSIS	
	4.1 Functional requirement	14
	4.2 Non-Functional requirements	15

5	PROJECT DESIGN	
	5.1 Data Flow Diagrams	17
	5.2 Solution & Technical Architecture	17
	5.3 User Stories	18
6	PROJECT PLANNING & SCHEDULING	
	6.1 Sprint Planning & Estimation	20
	6.2 Sprint Delivery Schedule	22
	6.3 Reports from JIRA	23
7	CODING & SOLUTIONING (Explain the features added in the project along with code)	
	7.1 Feature 1	25
	7.2 Feature 2	29
	7.3 Database Schema (if Applicable)	-
8	TESTING	
	8.1 Test Cases	30
	8.2 User Acceptance Testing	30
9	RESULTS	
	9.1 Performance Metrics	32
10	ADVANTAGES & DISADVANTAGES	34

11	CONCLUSION	35
12	FUTURE SCOPE	36
13	APPENDIX	
	Source Code	38
	GitHub & Project Demo Link	40

CHAPTER 1

INTRODUCTION

1.1 Overview

Water is the primary need of all living beings and living without water is impossible. With the advancement of technology and industrialization, environmental pollution has become a major concern. Water pollution is one of the most serious types of this environmental pollution. Pollution will degrade the quality and purity of water. Any imbalance in the quality of water would severely affect the humans' health and at the same time it would affect the ecological balance among all species.

1.2 Purpose

This project aims to obtain a water monitoring system with high frequency, high mobility, and low power. Therefore, our system will immensely help populations to become conscious against contaminated water as well as to stop polluting the water.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing problem

- Existing system has mechanisms which are semi-automated or manually controlled devices which are handled by a person responsible for monitoring the water quality.
- There is a need to have human intervention in taking various reading parameters.

2.2 References

Water quality monitoring system:

https://www.ripublication.com/awmc17/awmcv10n5_24.pdf

IOT Based Water Quality Monitoring System:

<https://ijsrcseit.com/CSEIT1831361>

Water Quality Monitoring System Implemented With IoT:

<http://www.warse.org/IJETER/static/pdf/file/ijeter29972021.pdf>

IoT Based Real-time River Water Quality Monitoring System:

<https://www.sciencedirect.com/science/article/pii/S1877050919309391>

2.3 Problem Statement Definition

Achieving and maintaining suitable water quality is one of the important parameters to ensure health and well-being of the human as well as ecosystems. Among the various aquatic ecosystems, riverine ecosystems are more prone to pollution and therefore need to be monitored frequently and on regular time intervals. In this context, a real-time water quality monitoring system offers an excellent opportunity to keep track of the water quality on a continuous basis; which not only helps to identify the affected location and pollution source, but also creates alerts enabling the authorities to take immediate action. One such real-time water quality monitoring system was installed in the River Ganga (India), considering the fragility and significance of the Gangetic ecosystem. In this paper, we have presented the details of the real-time water quality monitoring system installed in River Ganga and results obtained through it for various parameters. The results have also been compared with the standard values. Additionally, based on this preliminary investigation, limitations and recommendations have also been presented to further enhance the utility of the system.

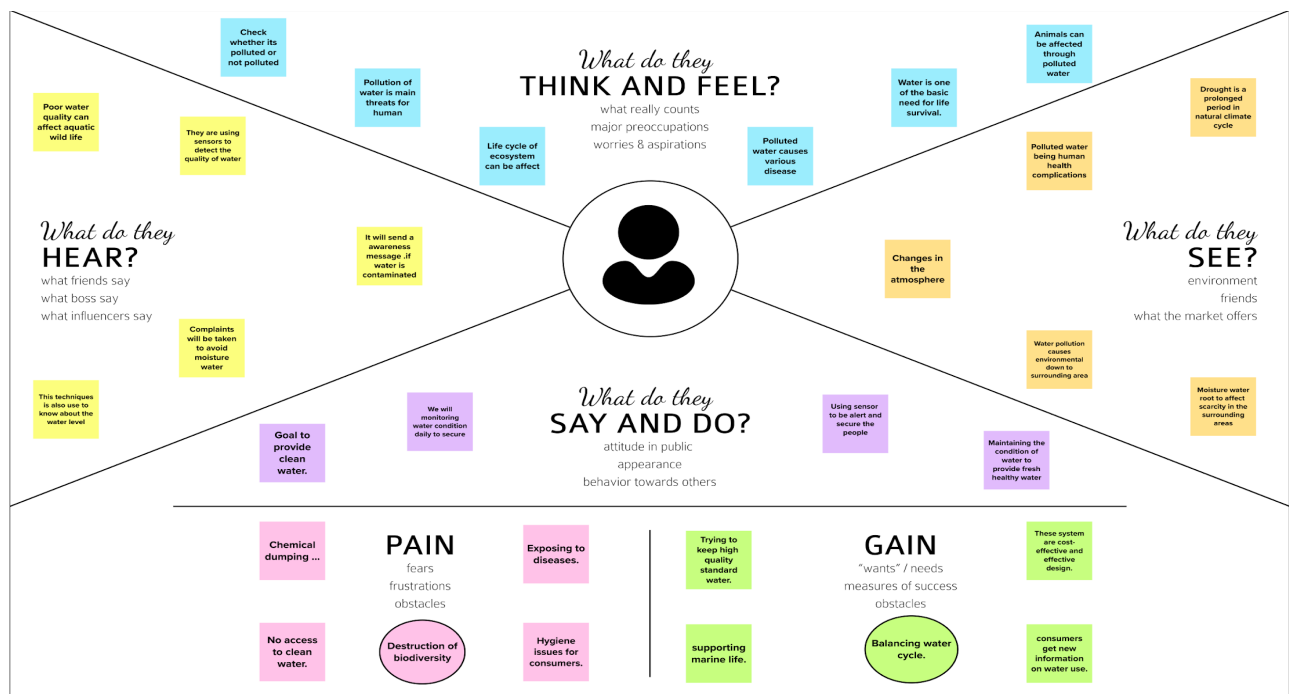
CHAPTER 3

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 behaviors and attitudes. It is a useful tool to help 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 amount of creative solutions. 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.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

In this step team members gather and provide their ideas and collaborate those ideas and select their problem statement. The ideas should be relevant to their problem statement.


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


The main purposes of Real-time water quality monitoring is to help people avoiding polluted water and immediately resolve the problem by informing to the control team.





Key rules of brainstorming


To run a smooth and productive session


 Stay in topic.

 Encourage wild ideas.

 Defer judgment.

 Listen to others.

 Go for volume.

 If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

In this step they put their ideas and views which are prioritized based on their importance and the ideas are grouped. These ideas are categorized according to their relevant classifications.

2

Brainstorm

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

10 minutes

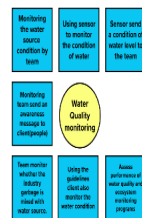
Vinayaga Moorthy R



Charan Rohith D



Vimalkarthik J



Vishal K



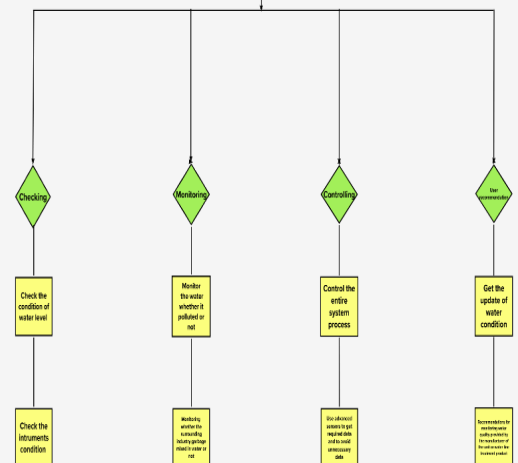
3

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 can break it up into smaller sub-groups.

20 minutes

Real- Time River Water Quality Monitoring And Controlling System



Step-3: Idea Prioritization

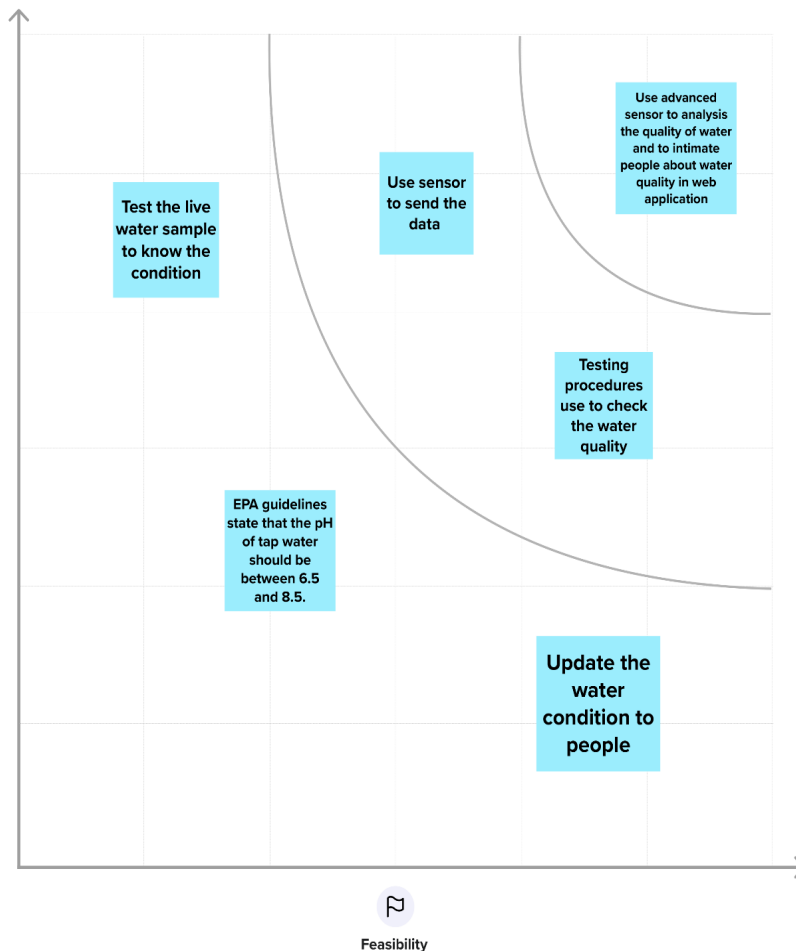
As mentioned, idea prioritization is just a part of the idea management process. Having a structured idea management process and a systematic way of gathering, evaluating and prioritizing new ideas takes time. To make it work, the entire idea management process should be integrated into everyday ways of working.

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



→

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

- A Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.
- B Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

- Strategy blueprint**
Define the components of a new idea or strategy.
[Open the template →](#)
- Customer experience journey map**
Understand customer needs, motivations, and obstacles for an experience.
[Open the template →](#)
- Strengths, weaknesses, opportunities & threats**
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
[Open the template →](#)

[Share template feedback](#)

3.3 Proposed Solution:

- **Problem Statement:**

To monitor the real time river water quality with the help of a control system.

- **Idea / Solution description:**

By using an Arduino with a water based sensor it detects the water quality and alerts the authority to announce whether the water is good or bad for localities.

- **Novelty / Uniqueness:**

We use the web application to present the water condition and announce the awareness message.

- **Social Impact / Customer Satisfaction:**

Save water and river resources from the polluted water and impure water.

- **Business Model (Revenue Model):**

→ By without using: The water resource will be spoiled without maintenance.

→ By using: Get the healthy water and good environment

- **Scalability of the Solution:**

It is well organized to provide pure water for the people and used for farming.

3.4 Problem Solution fit:

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS <ul style="list-style-type: none">Localities.Dam safety organizer.Water consumers.	6. CUSTOMER CONSTRAINTS CC <ul style="list-style-type: none">Wifi modulus are used.Sensors are used.Low power consumptionClouds for storage purpose.	5. AVAILABLE SOLUTIONS AS <ul style="list-style-type: none">In this technique several sensors is employed to measuring physical and chemical parameters of the water.Manual system with a monotonous process and is very time consuming	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS JP <p>To identify any dust particles present in the water.</p> <p>To identify PH level of the water can be monitored.</p> <p>To measure Water temperature</p>	9. PROBLEM ROOT CAUSE RC <p>Lack of training for data management and statistical tools.</p> <p>Lack of software to analyse the data for trend analyses and data validation.</p>	7. BEHAVIOUR BE <p>To recognize the water quality,check ph level,water temperature,param eters.</p> <p>The water quality can be maintainable.</p>	
3. TRIGGERS TR <ul style="list-style-type: none">The benefit of water quality standards is that they protect state waters for the ways that we want and need to use them.		10. YOUR SOLUTION SL <ul style="list-style-type: none">System must be low-cost,most efficient as well as processing,sending and viewing data on cloud through wifi modules.To collect data from the cloud and develop a web application to present the condition of water and to recommend according to condition it will alert or not.		8.CHANNELS of BEHAVIOR CH 8.1 ONLINE <ul style="list-style-type: none">To intimate the people through message.If the water condition is good or bad. 8.2 OFFLINE <ul style="list-style-type: none">It processes the pH levels,water temperature,an d parameters of the water.It includes a number of sensors to test the water's quality based on factors including pH, temperature conductivity, turbidity, and arduino.
4. EMOTIONS: BEFORE / AFTER EM BEFORE: No knowledge about water quality, unaware of health issues. AFTER: knowledge about water quality,awareness about health issues.				

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution

FR-1

Functional Requirement (Epic) - User Requirement

Sub Requirement (Story / Sub-Task) - Monitoring water quality, water flow and temperature to control water pollution and algal bloom.

FR-2

Functional Requirement (Epic) - User Registration

Sub Requirement (Story / Sub-Task) - Manual Sign-Up using a website or Gmail.

FR-3

Functional Requirement (Epic) - User Confirmation

Sub Requirement (Story / Sub-Task) - OTP authentication through phone messages, email, notices, paper and confirmation.

FR-4

Functional Requirement (Epic) - Product Implementation

Sub Requirement (Story / Sub-Task) - Installing the product to monitor water quality for checking the confirmation by using websites.

FR-5

Functional Requirement (Epic) - Payment option

Sub Requirement (Story / Sub-Task) - Bank transfer, Debit cards, UPI Method.

FR-6

Functional Requirement (Epic) - Product Feedback

Sub Requirement (Story / Sub-Task) - Through the websites, phone conversation, and Gmail.

4.2 Non-Functional requirements

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

NFR-1

Non-Functional Requirement - Usability

Description - This application is used to describe the product and easy to access the product by the guidelines.

NFR-2

Non-Functional Requirement - Security

Description - This application security ensures the website by building a firewall and two step verification. Only can access by authorization person by given user id and password or otp verification.

NFR-3

Non-Functional Requirement - Reliability

Description - To maintain the product conditions and update the version of the product is up-to-date. System update and software update are possible to increase various features and durability.

NFR-4

Non-Functional Requirement - Performance

Description - This application collects the data of river water to provide accurate value. Using this method, we can alert the locality right on time. This application is user friendly and can be accessed by both end-users and management.

NFR-5**Non-Functional Requirement - Availability**

Description - Depending on the requirement of the user, all required functions will be offered. when the user requests a feature or makes a message, all features made available in places where users like to know about it.

NFR-6**Non-Functional Requirement - Scalability**

Description - Regardless of size, the product must fill the entire river's space. The product is based on monitoring water quality, flow, humidity, and temperature as well as controlling algal blooms.

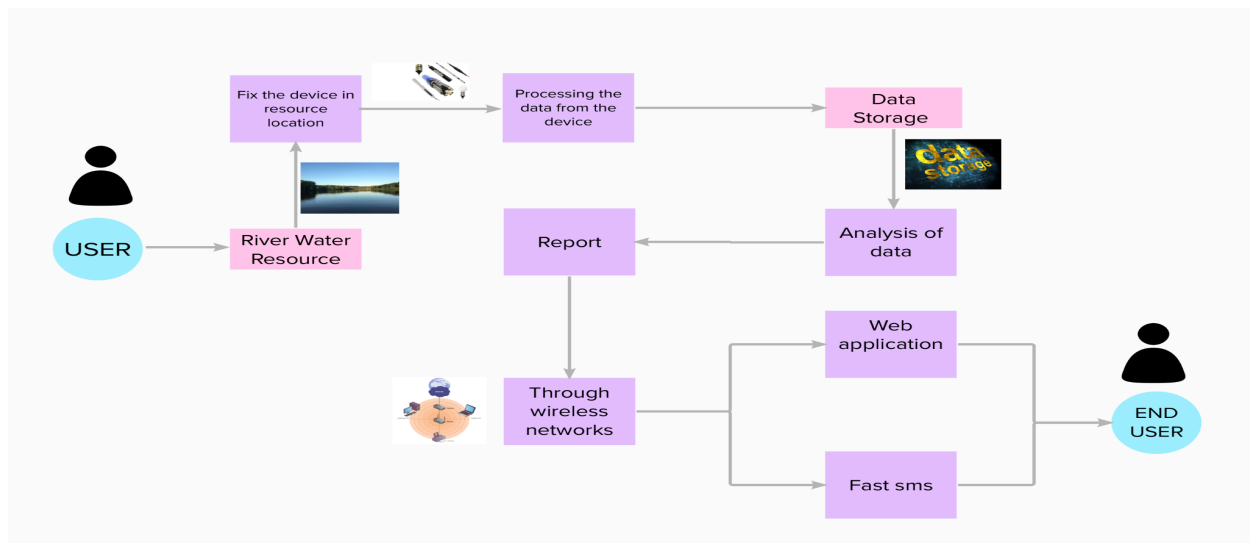
CHAPTER 5

PROJECT DESIGN

5.1 Data Flow Diagrams

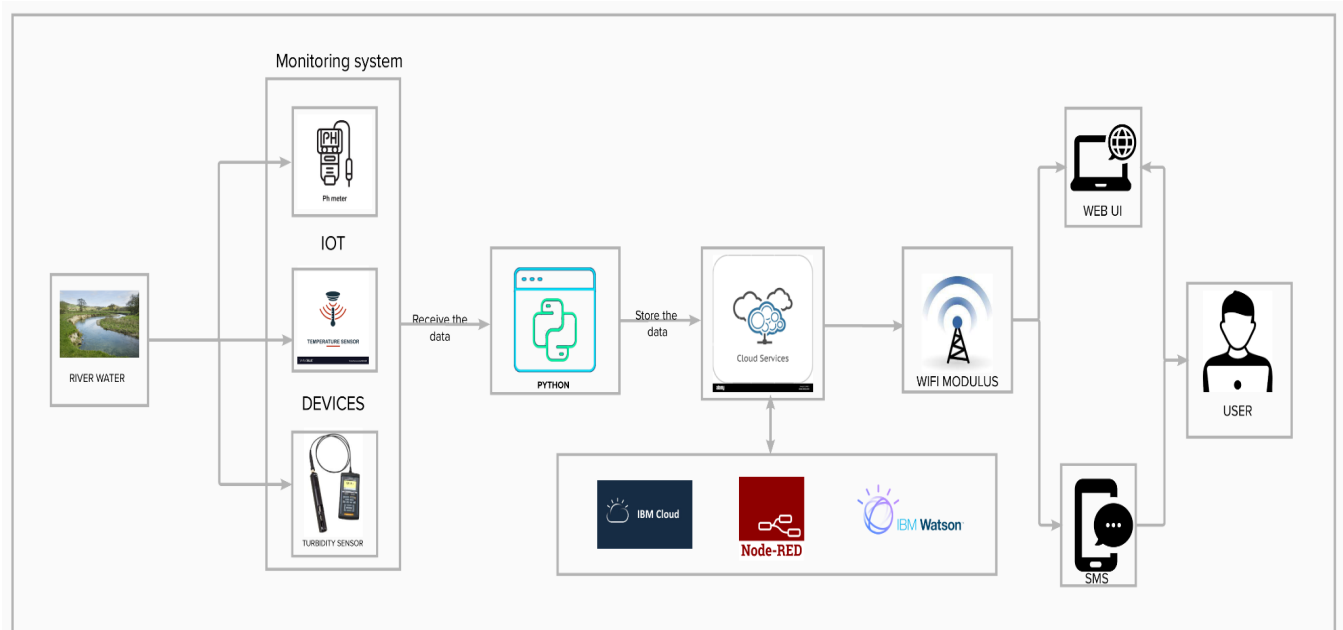
A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That’s why DFDs remain so popular after all these years.

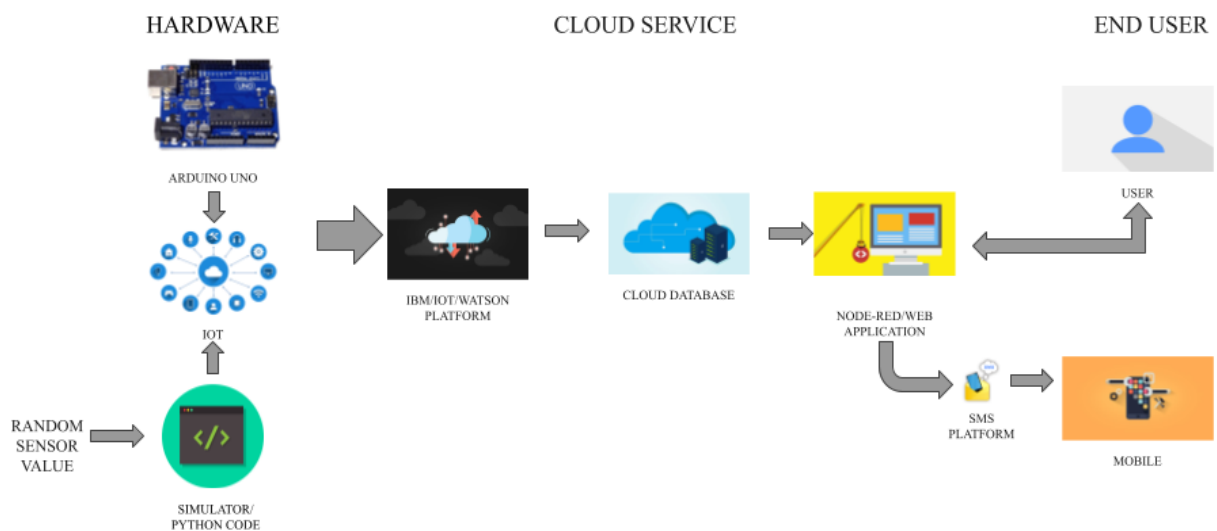


5.2 Solution & Technical Architecture

Technical architecture:



Solution architecture:



5.3 User Stories

Sprint-1

USN-1 :

User Story / Task : The device wants to maintain clean river water for a good environment.

Priority : High

USN-2 :

User Story / Task : To maintain the river water always to be clean

Priority : High

USN-3 :

User Story / Task : A farmer who raises crops for planting and yielding

Priority : High

USN-4 :

User Story / Task : 12% of water is needed for manufacturing business like paint and coating.

Priority : High

Sprint-2

USN-5 :

User Story / Task : Government forms NSS camps in each school and college for monthly once to clean the river water.

Priority : Medium

Sprint-3

USN-6 :

User Story / Task : As a user, I can register for the application by entering my email, password, and confirming my password. To get details about river water.

Priority : High

Sprint-4

USN-7 :

User Story / Task : As a user, we can receive a message from the administration about conditions of river water.

Priority : High

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint-1

USN-1 :

User Story / Task : As a user, I can register for the application by entering my email, password, and confirming my password

Priority : High

Story Points : 2

Team Members : Vinayaga Moorthy R, Vimalkarthik J

USN-2 :

User Story / Task : As a user, I can register for the application through Facebook

Priority : Low

Story Points : 2

Team Members : Vishal K, Charan Rohith D

USN-3 :

User Story / Task : As a user, I can register through the gmail or email

Priority : Low

Story Points : 2

Team Members : Vinayaga Moorthy R, Vishal K

Sprint-2

USN-4 :

User Story / Task : As a user, I will receive confirmation email once I have registered for the application

Priority : High

Story Points : 2

Team Members : Charan Rohith D, Vimalkarthik J

USN-5 :

User Story / Task : As a user, I can log into the application by entering email & password

Priority : High

Story Points : 2

Team Members : Vishal K, Vimalkarthik J

Sprint-3

USN-6 :

User Story / Task : As a user,I can view the sensor readings

Priority : High

Story Points : 2

Team Members : Vinayaga Moorthy R, Charan Rohith D

USN-7 :

User Story / Task : As a user, I can register for the application by entering my email, password, and confirming my password

Priority : Medium

Story Points : 2

Team Members : Vinayaga Moorthy R, Vimalkarthik J

Sprint-4

USN-8 :

User Story / Task : As a user,I get the notification about the water quality conditions.

Priority : High

Story Points : 2

Team Members : Charan Rohith D, Vishal K

USN-9 :

User Story / Task : As a user, I can get the messages about the water quality conditions.

Priority : High

Story Points : 2

Team Members : Vinayaga Moorthy R, Vimalkarthik J

6.2 Sprint Delivery Schedule

Sprint-1

- **Total Story Points - 6**
- **Duration - 6 Days**
- **Sprint Start Date - 24 Oct 2022**
- **Sprint End Date (Planned) - 29 Oct 2022**
- **Story Points Completed (as on Planned End Date) - 6**
- **Sprint Release Date (Actual) - 29 Oct 2022**

Sprint-2

- **Total Story Points - 4**
- **Duration - 6 Days**
- **Sprint Start Date - 31 Oct 2022**
- **Sprint End Date (Planned) - 05 Nov 2022**
- **Story Points Completed (as on Planned End Date) - 4**
- **Sprint Release Date (Actual) - 31 Oct 2022**

Sprint-3

- **Total Story Points - 6**
- **Duration - 6 Days**
- **Sprint Start Date - 07 Nov 2022**
- **Sprint End Date (Planned) - 12 Nov 2022**
- **Story Points Completed (as on Planned End Date) - 6**
- **Sprint Release Date (Actual) - 07 Nov 2022**

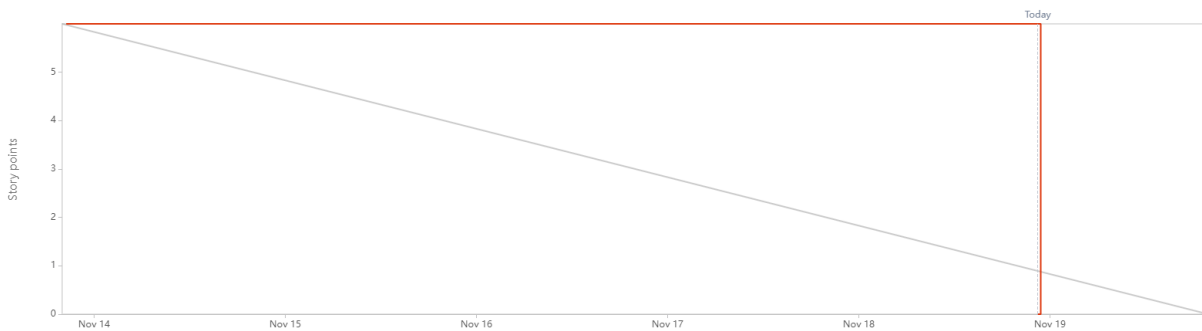
Sprint-4

- **Total Story Points - 6**
- **Duration - 6 Days**
- **Sprint Start Date - 14 Nov 2022**
- **Sprint End Date (Planned) - 19 Nov 2022**
- **Story Points Completed (as on Planned End Date) - 6**
- **Sprint Release Date (Actual) - 14 Nov 2022**

6.3 Reports from JIRA

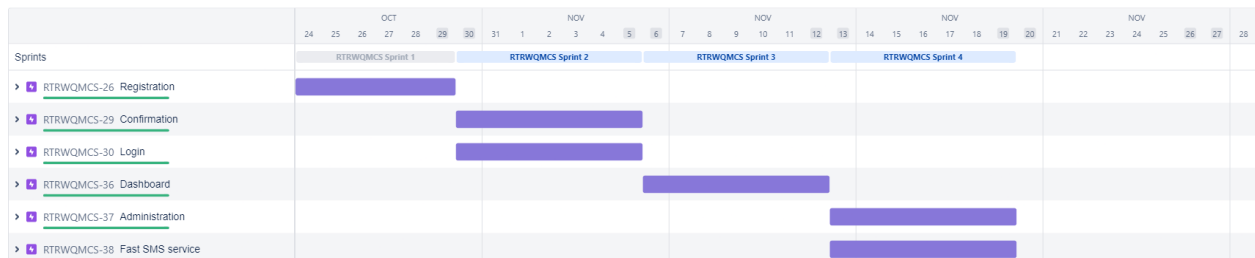
Burndown chart report :

A burndown chart is a graphical representation of work left to do versus time and completed work. It is often used in agile software development methodologies such as scrum, jira. However burndown charts can be applied to any project containing measurable time.



Roadmap report :

It provides the details about the project completion status ,the work yet to be completed in four ways like days,months,weeks,quarters.



CHAPTER 7

CODING & SOLUTIONING

7.1 Feature 1

Details Acquisition feature

Here the user can get their details regarding their water quality and they can get into a conclusion with the help of data.

CODE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
from twilio.rest import Client

#Provide your IBM Watson Device Credentials
organization = "uyyqeq"
deviceType = "12345"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO
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 ("please send proper command")

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

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

# Connect and send a datapoint "hello" with value "world" into the cloud as an event
of type "greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11
    pH = random.randint(1, 14)
    temp=random.randint(90,110)
    Humid=random.randint(60,100)

```

```

data = {'pH': pH, 'temp' : temp, 'Humid': Humid }
def SMS():
    message = Client.messages.create(
        body="ALERT!! THE WATER QUALITY IS DEGRADED",
        from_=keys.twilio_number,
        to = keys.target_number)
    print(message.body)
    if temperature>70 or pH<6 or Humidity>500:
        SMS()
#print data
def myOnPublishCallback():
    print ("Published pH= %s" % pH,"Published Temperature = %s C" % temp,
"Humidity = %s %%" % Humid, "to IBM Watson")

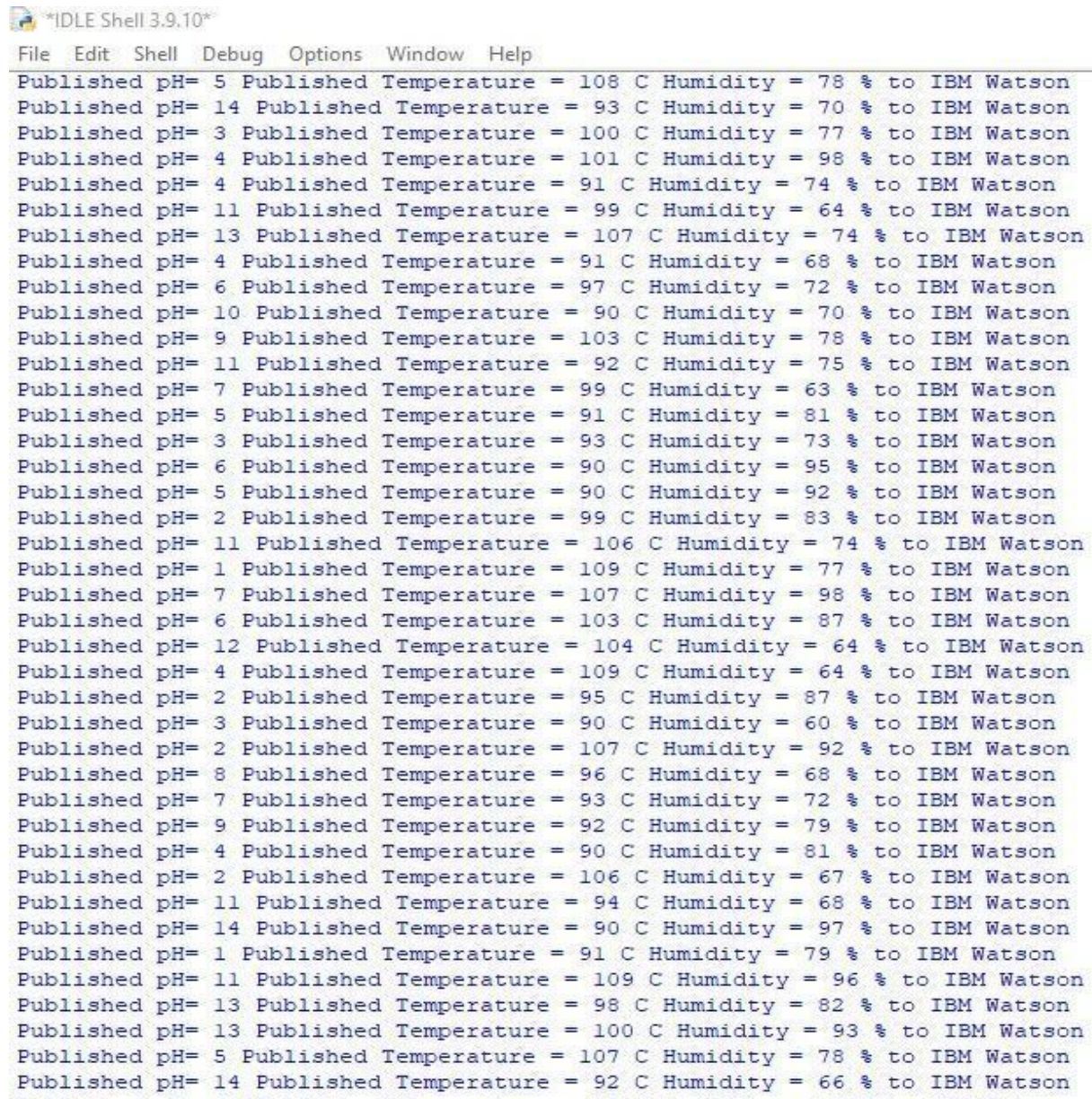
    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTTF")
        time.sleep(10)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

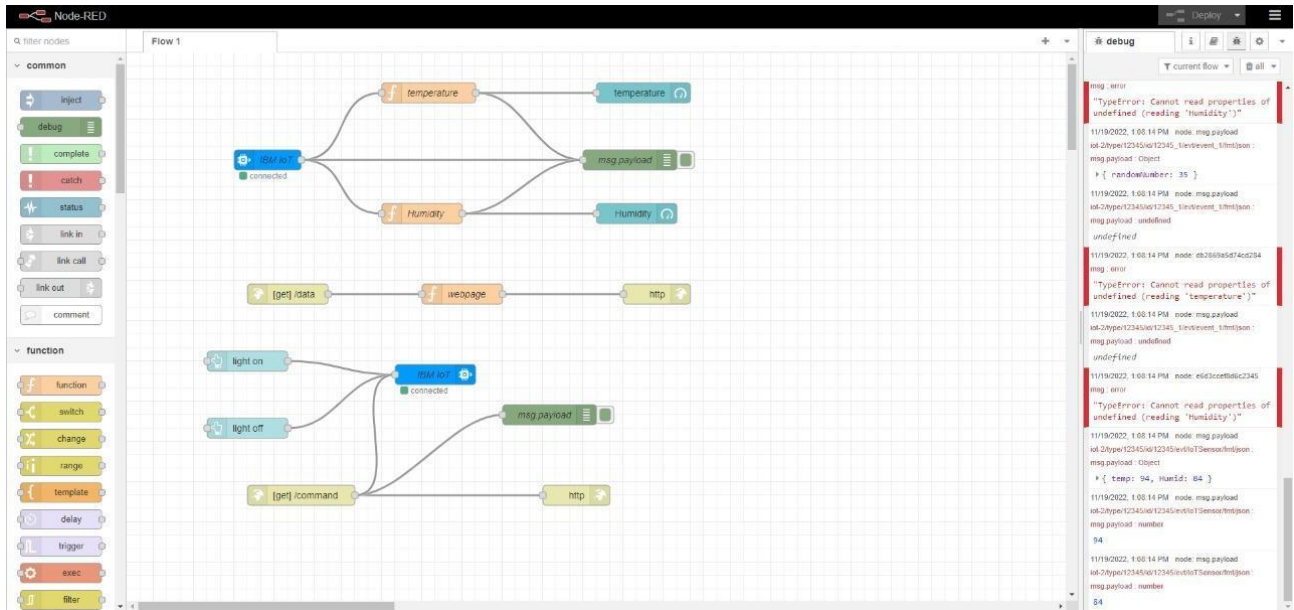
FEATURE SCREENSHOT:



The screenshot shows the IDLE Shell 3.9.10* application window. The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main text area displays a list of 40 published data points, each consisting of a pH value, a temperature in Celsius, a humidity percentage, and the source 'to IBM Watson'. The data points are as follows:

pH	Temperature (C)	Humidity (%)	Source
5	108	78	to IBM Watson
14	93	70	to IBM Watson
3	100	77	to IBM Watson
4	101	98	to IBM Watson
4	91	74	to IBM Watson
11	99	64	to IBM Watson
13	107	74	to IBM Watson
4	91	68	to IBM Watson
6	97	72	to IBM Watson
10	90	70	to IBM Watson
9	103	78	to IBM Watson
11	92	75	to IBM Watson
7	99	63	to IBM Watson
5	91	81	to IBM Watson
3	93	73	to IBM Watson
6	90	95	to IBM Watson
5	90	92	to IBM Watson
2	99	83	to IBM Watson
11	106	74	to IBM Watson
1	109	77	to IBM Watson
7	107	98	to IBM Watson
6	103	87	to IBM Watson
12	104	64	to IBM Watson
4	109	64	to IBM Watson
2	95	87	to IBM Watson
3	90	60	to IBM Watson
2	107	92	to IBM Watson
8	96	68	to IBM Watson
7	93	72	to IBM Watson
9	92	79	to IBM Watson
4	90	81	to IBM Watson
2	106	67	to IBM Watson
11	94	68	to IBM Watson
14	90	97	to IBM Watson
1	91	79	to IBM Watson
11	109	96	to IBM Watson
13	98	82	to IBM Watson
13	100	93	to IBM Watson
5	107	78	to IBM Watson
14	92	66	to IBM Watson

7.2 Feature 2



OUTPUT:

```
{"temperature":104,"Humidity":81,"pH":7}
```

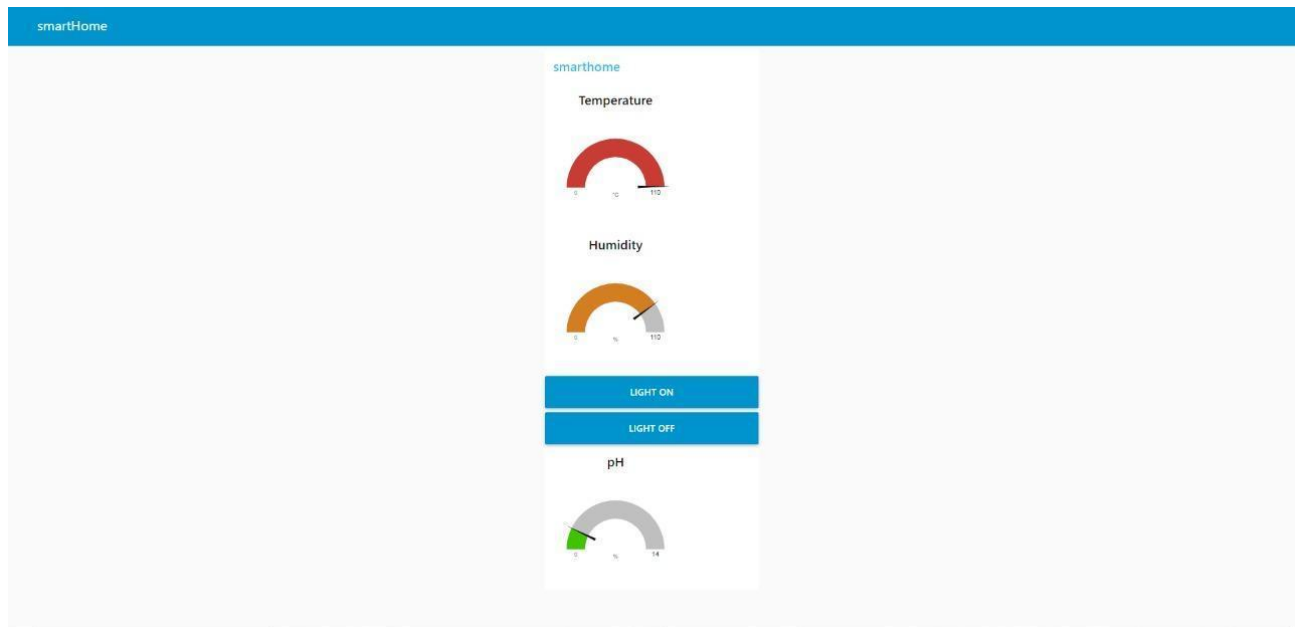
CHAPTER 8

TESTING

8.1 Test Cases

- We want to check the water condition whether drinkable or not based on data received.
- If the water conditions are normal and drinkable water nothing needs to be done.
- If the water conditions are not normal, water is polluted or spoiled and needs to be changed and alert the people.

8.2 User Acceptance Testing



(i) DASHBOARD

pH 14
Temperature 99
Humidity 65

LIGHT ON

LIGHT OFF

(ii) APP

CHAPTER 9

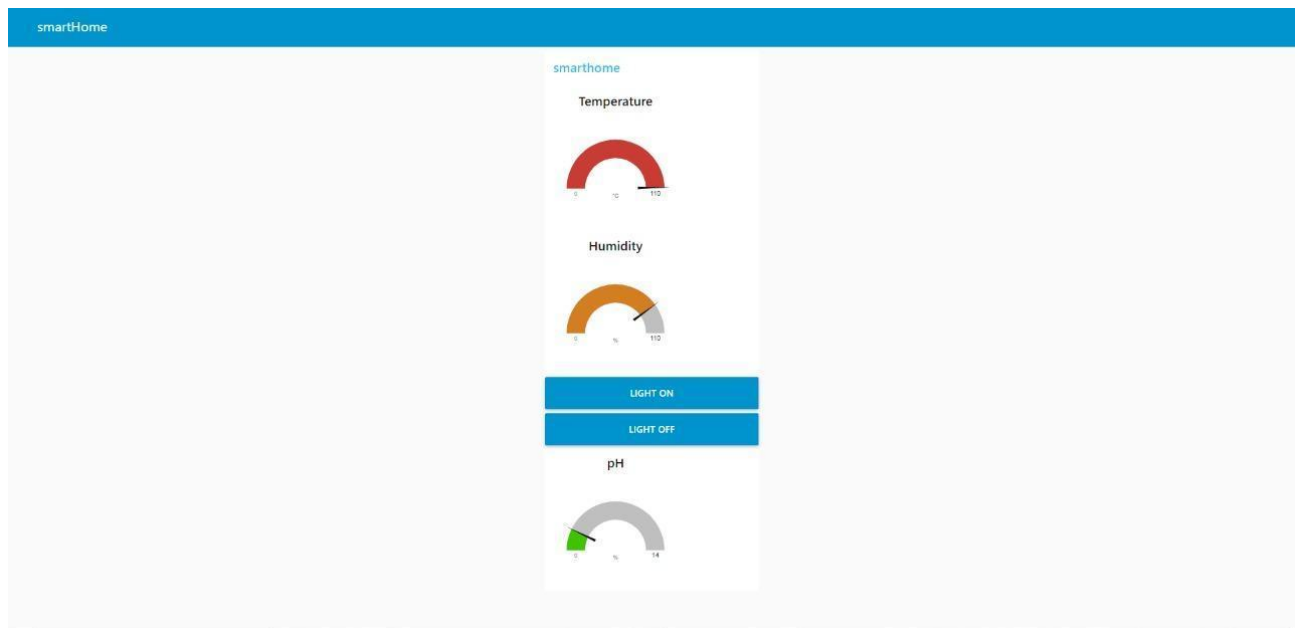
RESULTS

9.1 Performance Metrics

With this user interface users can easily predict the water quality based on their sensor's value and the process of alerting system and web application. Users are also able to know their water quality.

Some sample images of the output are provided below:

Sensors value of water quality in gauge:



Sensors value of water quality:

pH

14

Temperature 99

Humidity 65

LIGHT ON

LIGHT OFF

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES

- No need for manual collection
- Ease of use
- Cheap and quick process

DISADVANTAGES

- Collect samples manually.
- Complicated methodology.
- Time consuming.
- Low measurement precision.
- High cost.
- Lack of real-time monitoring

CHAPTER 11

CONCLUSION

Monitoring of Turbidity, PH & Temperature of Water makes use of water detection sensors with unique advantages. It is low in cost and does not require people on duty. To implement this, we need to deploy the sensor devices in the environment for collecting the data and analysis. By deploying sensor devices in the environment, we can bring the environment into real life.

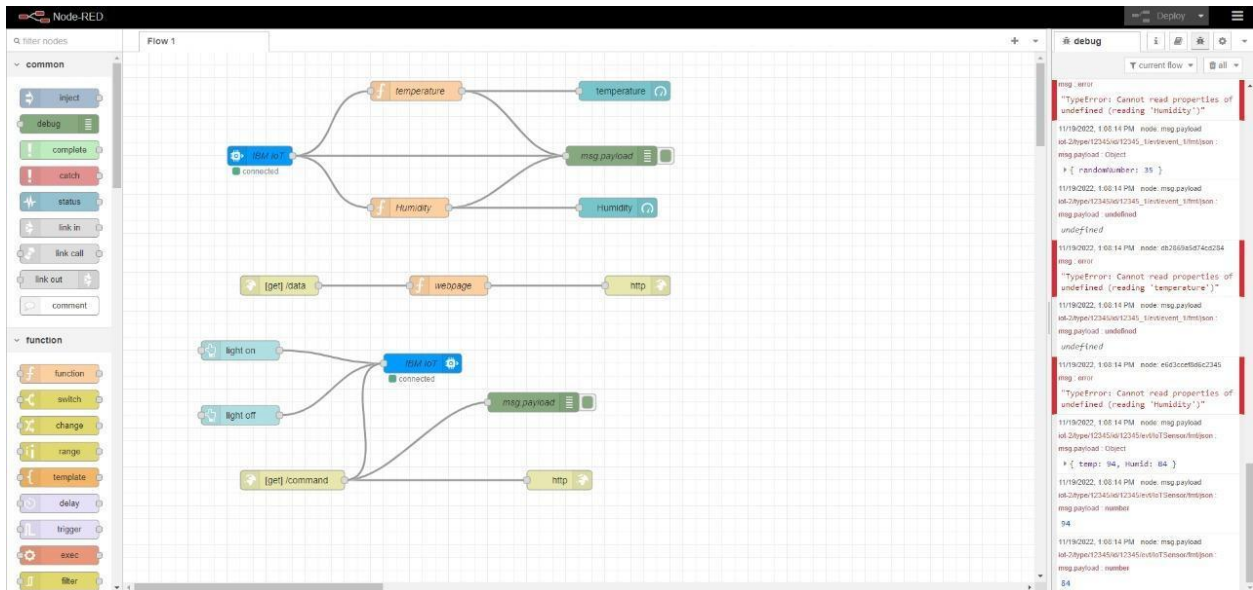
This work presents a review of the recent works carried out by the researchers in order to make water quality monitoring systems smart, low powered and highly efficient such that monitoring will be continuous and alerts/notifications will be sent to the concerned authorities for further processing.

CHAPTER 12

FUTURE SCOPE

- In the future we will use more IOT concepts in this project.
- Detecting the more parameters for the most secure purpose.
- Increase the parameters by addition of multiple sensors.
- By interfacing the relay we control the supply of water.

NODE-RED CONFIGURATION:



IBM IOT WATSON PLATFORM:

IBM Watson IoT Platform

vinayagamacus@gmail.com

ID: uaynq

Browse

Action

Device Types

Interfaces

Add Device

Browse Devices

All Devices

Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Q

Search by Device ID

Device Simulator

	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
>	1234	Disconnected	abcd	Device	Nov 19, 2022 11:37 AM		vinayagamacus@gmail.com	
▼	12345	Connected	12345	Device	Nov 13, 2022 8:36 PM		vinayagamacus@gmail.com	→ ...

Identity

Device Information

Groups

Recent Events

State

Logs

X

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

Event	Value	Format	Last Received
IoTSensor	{"pH":3,"temp":104,"Humid":91}	json	a few seconds ago
IoTSensor	{"pH":11,"temp":98,"Humid":91}	json	a few seconds ago
IoTSensor	{"pH":6,"temp":102,"Humid":84}	json	a few seconds ago
IoTSensor	{"pH":7,"temp":97,"Humid":81}	json	a few seconds ago

1 Simulation running

SOURCE CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
from twilio.rest import Client
#Provide your IBM Watson Device Credentials
organization = "uyyqeq"
deviceType = "12345"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"
# Initialize GPIO
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 ("please send proper command")
```

```

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

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

# Connect and send a datapoint "hello" with value "world" into the cloud as an event
of type "greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11
    pH = random.randint(1, 14)
    temp=random.randint(90,110)
    Humid=random.randint(60,100)

    data = {'pH': pH, 'temp' : temp, 'Humid': Humid }
    def SMS():
        message = Client.messages.create(
            body="ALERT!! THE WATER QUALITY IS DEGRADED",
            from_=keys.twilio_number,
            to = keys.target_number)
        print(message.body)

```

```

        if temperature>70 or pH<6 or Humidity>500:
            SMS()
        #print data
    def myOnPublishCallback():
        print ("Published pH= %s" % pH,"Published Temperature = %s C" % temp,
"Humidity = %s" % Humid, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(10)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

GITHUB:

<https://github.com/IBM-EPBL/IBM-Project-12889-1659498368>

DEMO LINK:

https://drive.google.com/file/d/1thUZNpT1F6hU293UiOum5W2sn1ZpSBNx/view?usp=share_link

