PROJECT REPORT

Real-Time River Water Quality Monitoring and Control System

Submitted by

KAVIYA P – 715519106021

PREETHI T– 715519106039

PRANEETHA S S– 715519106036

RAMANYA U S – 715519106042

INDEX

ADVANTAGES & DISADVANTAGES

CONCLUSION

FUTURE SCOPE

APPENDIX

Source Code

GitHub & Project Demo Link

INTRODUCTION

1.1 PROJECT OVERVIEW

Water pollution is a foremost global problem which needs ongoing evaluation and adaptation of water resource directorial principle at the levels of international down to individual wells. It has been studied that water pollution is the leading cause of mortalities and diseases worldwide.River water pollution is one of the biggest fears for the green globalization. In order to ensure the safe supply of the drinking water the quality needs to be monitor in real time. The system consists of several sensors is used to measuring physical and chemical parameters of the water. The parameters such as temperature,PH, turbidity, flow sensor of the water can be measured.The most important function of the system is to ensure that the data collected from the sensor reflects the actual aquatic scenario and the data are transmitted and delivered as a display of information in the web or as a short message service (SMS) sent to pre-identified key users in a timely manner.

1.2 PURPOSE

Monitoring provides objective evidence necessary to make sound decisions on managing river water quality today and in the future.River water quality monitoring is used to alert us to current,on going and emerging problems and to determine compliance with river water standards and to protect the other beneficial uses of water.

LITERATURE SURVEY

1.3 PROBLEM STATEMENT

Water is a finite resource that is necessary for agriculture, industry and the survival of all living things on the planet, including humans. Many people are unaware of the need of drinking adequate amounts of water on a daily basis. Many unregulated methods waste more water. Poor water allocation, inefficient consumption, lack of competent and integrated water management are all factors that contribute to this problem. Therefore, efficient use and water monitoring are potential constraint for home or office water management system.

1.4 REFERENCES

1) **IoT-based System for Real-time Water Pollution Monitoring of Rivers**
   Mohammad Ariful IslamKhan;Mohammad AkidulHoque; Sabbir Ahmed

   IEEE September 2021

INFERENCE:

The research proposes a system to remotely monitor the water quality of a river so that the authorities can gather better insights about the condition of that particular river and predict the critical future phenomena.

2) **Design and Implementation of Real Time Approach for The Monitoring of Water Quality Parameters**

   Siti Aishah Binti Makhtar;Norhafizah Binti Burham;Anees Bt Abdul Aziz

IEEE - June 2022

INFERENCE:

Access to safe drinking water is essential to nurturing human life on earth. Polluted air and unsanitary water can cause health problems. Unhygienic water can cause stomach and health-related problems.

3) **An IoT Based Smart Water Quality Monitoring System using Cloud**

Ajith Jerom B.; R. Manimegalai; R. Manimegalai

IEEE – April 2020**3)**

INFERENCE:

Other sources of pollution include agricultural runoff and unregulated small scale industry that results in polluting, most of the rivers, lakes and surface water in India. In this paper, An IoT Based Smart WaterQuality Monitoring Systemusing Cloud and Deep Learning is proposed to monitor the quality of the water in water-bodies.

4) **IoT Based Real-time River Water Quality Monitoring System**

Mohammad Salah UddinChowdury, Talha BinEmran

Science Direct – 2018

INFERENCE:

This paper proposes a sensor-based water quality monitoring system. The main components of Wireless SensorNetwork (WSN) include a microcontroller for processing the system, communication system for inter and intra node communication and several sensors.

5) **A Development and Implementation of Water Quality Assessment Monitoring (WQAM) System using the Internet of Things (IoT) in Water Environment**

Muhammad Farhan Johan, S. Abdullah, A. Zanal Saurabh S. Soman,Hamidreza Zareipour, Om Malik JEVA

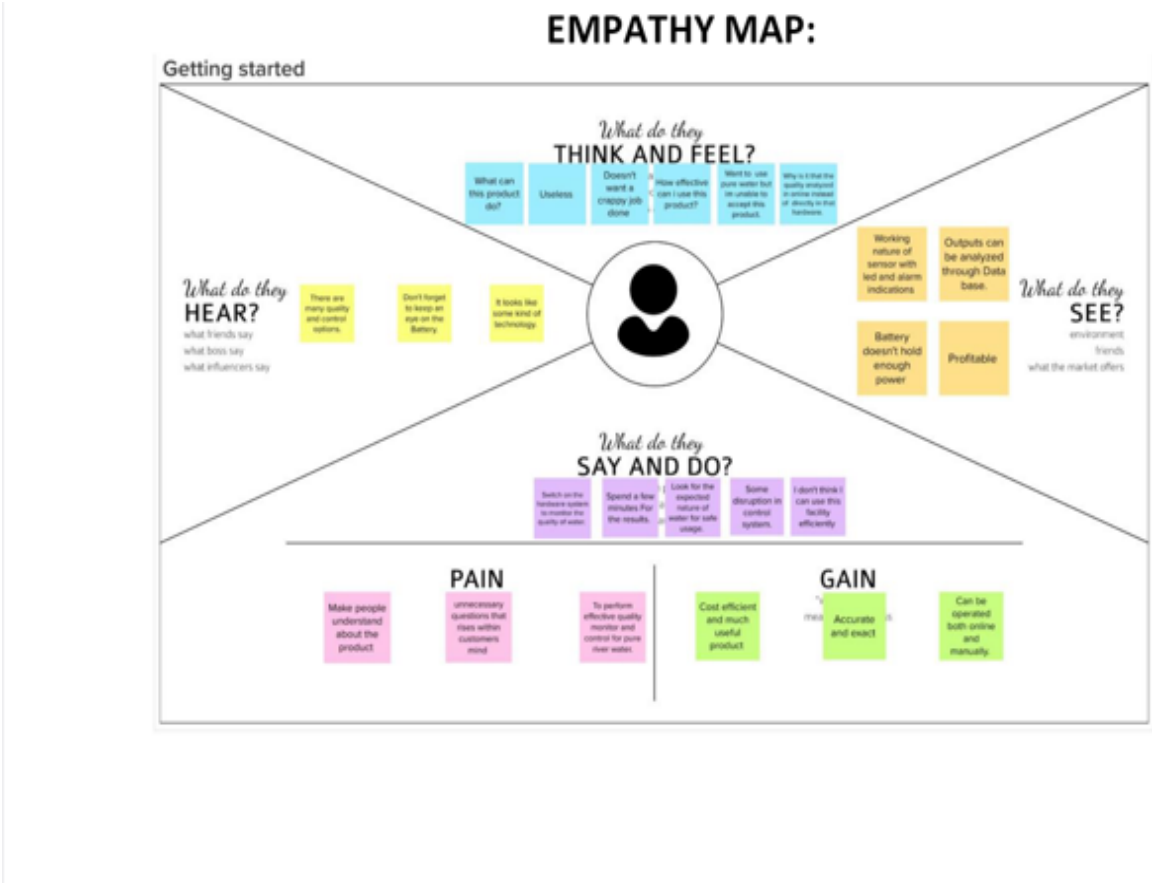
- 23 November 2021

INFERENCE:

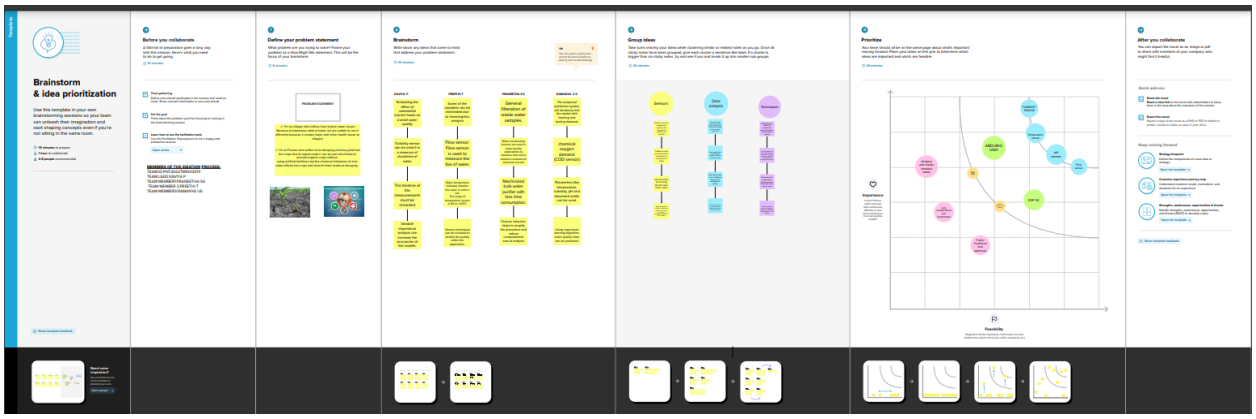
This paper presents the development and implementation of Water Quality Assessment and Monitoring (WQAM) system. The system development used.Wi-Fi enabled microcontroller to connect with the IoT environment and store the data in the IoT cloud server.

IDEATION AND PROPOSED SOLUTION

1.6 EMPATHY MAP



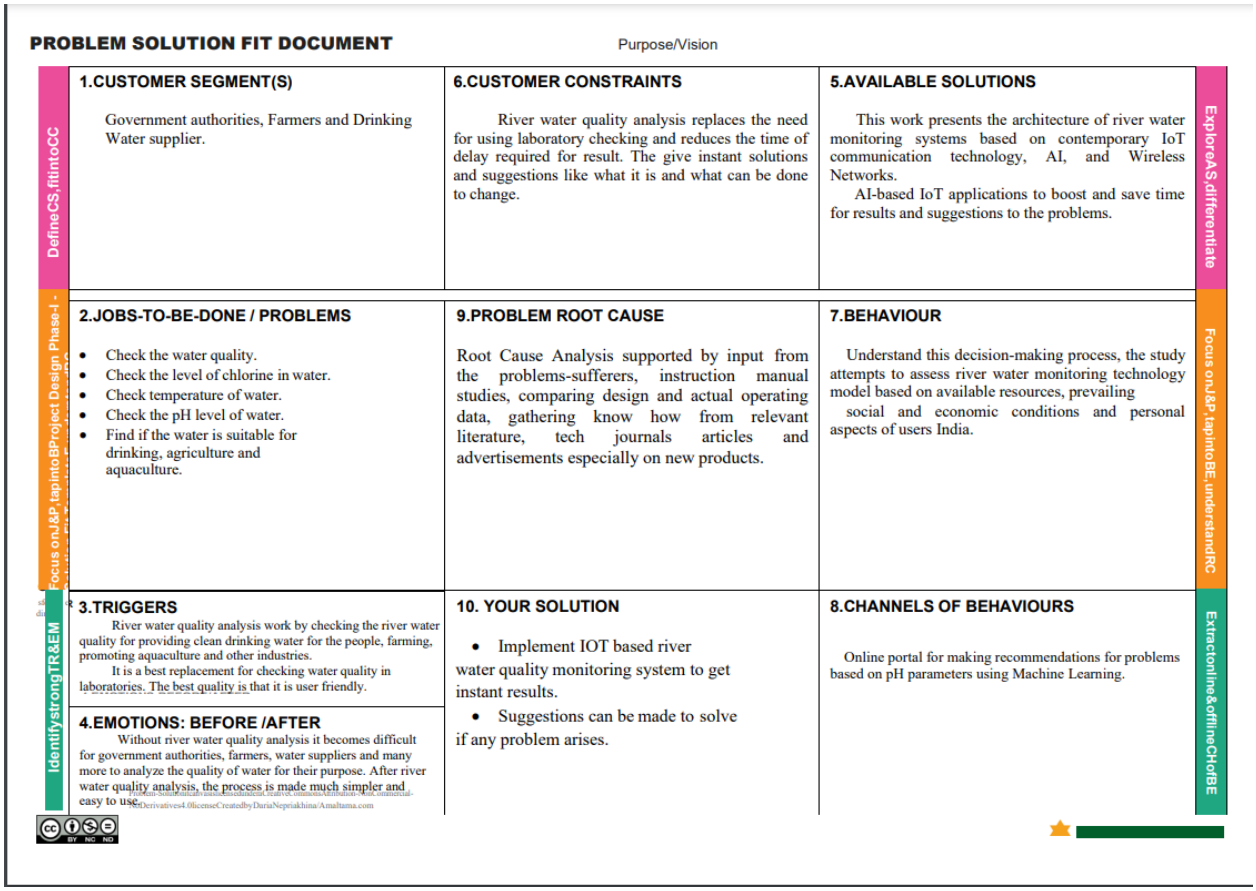
1.7 IDEATION AND BRAINSTORMING



1.8 PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	To Control the impure water issues & to avoid decaying crops due to unhealthy water.
2.	Idea / Solution description	Monitoring water parameters by using Arduino and Sensors and control measures by ultrasonic frequency.
3.	Novelty / Uniqueness	Using Wireless sensor network to monitor the quality of water
4.	Social Impact / Customer Satisfaction	People come to know about the quality of water.
5.	Business Model (Revenue Model)	Water Monitoring and Control Model
6.	Scalability of the Solution	The process of operating this Model is very easy.

1.9 PROBLEM SOLUTION FIT



PROBLEM SOLUTION FIT DOCUMENT

Purpose/Vision

DefineCS,fitintoCC

1.CUSTOMER SEGMENT(S)

Government authorities, Farmers and Drinking Water supplier.

6.CUSTOMER CONSTRAINTS

River water quality analysis replaces the need for using laboratory checking and reduces the time of delay required for result. The give instant solutions and suggestions like what it is and what can be done to change.

5.AVAILABLE SOLUTIONS

This work presents the architecture of river water monitoring systems based on contemporary IoT communication technology, AI, and Wireless Networks.

AI-based IoT applications to boost and save time for results and suggestions to the problems.

ExploreAS,differentiate

Focus onJ&P,tapintoBProject Design Phase-I -

2.JOBS-TO-BE-DONE / PROBLEMS

- Check the water quality.
- Check the level of chlorine in water.
- Check temperature of water.
- Check the pH level of water.
- Find if the water is suitable for drinking, agriculture and aquaculture.

9.PROBLEM ROOT CAUSE

Root Cause Analysis supported by input from the problems-sufferers, instruction manual studies, comparing design and actual operating data, gathering know how from relevant literature, tech journals articles and advertisements especially on new products.

7.BEHAVIOUR

Understand this decision-making process, the study attempts to assess river water monitoring technology model based on available resources, prevailing social and economic conditions and personal aspects of users India.

Focus onJ&P,tapintoBE,understandRC

IdentifystrongTR&EM

3.TRIGGERS

River water quality analysis work by checking the river water quality for providing clean drinking water for the people, farming, promoting aquaculture and other industries.

It is a best replacement for checking water quality in laboratories. The best quality is that it is user friendly.

4.EMOTIONS: BEFORE /AFTER

Without river water quality analysis it becomes difficult for government authorities, farmers, water suppliers and many more to analyze the quality of water for their purpose. After river water quality analysis, the process is made much simpler and easy to use.

10. YOUR SOLUTION

- Implement IOT based river water quality monitoring system to get instant results.
- Suggestions can be made to solve if any problem arises.

8.CHANNELS OF BEHAVIOURS

Online portal for making recommendations for problems based on pH parameters using Machine Learning.

ExtractonlineA&offlineCHofBE

REQUIREMENT ANALYSIS

1.10 FUNCTIONAL REQUIREMENT ANALYSIS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User registration	Registration with Gmail Create an account By Following the instructions
FR-2	User Authentication	The credentials are accessible only to the authorized users to access the model.
FR-3	User Confirmation	Confirmation viaAlarm Confirmation via SMS
FR-4	Interface sensor	Interface sensor-temperature sensor, turbidity sensor, etc. If contaminated water is present in the river, it gives alarm.
FR-5	Accessing datasets	Datasets are retrieved from Cloudant DB server
FR-6	Mobile application	Can see water is contaminated or not. Can control the motor to stop the flow of contaminated water

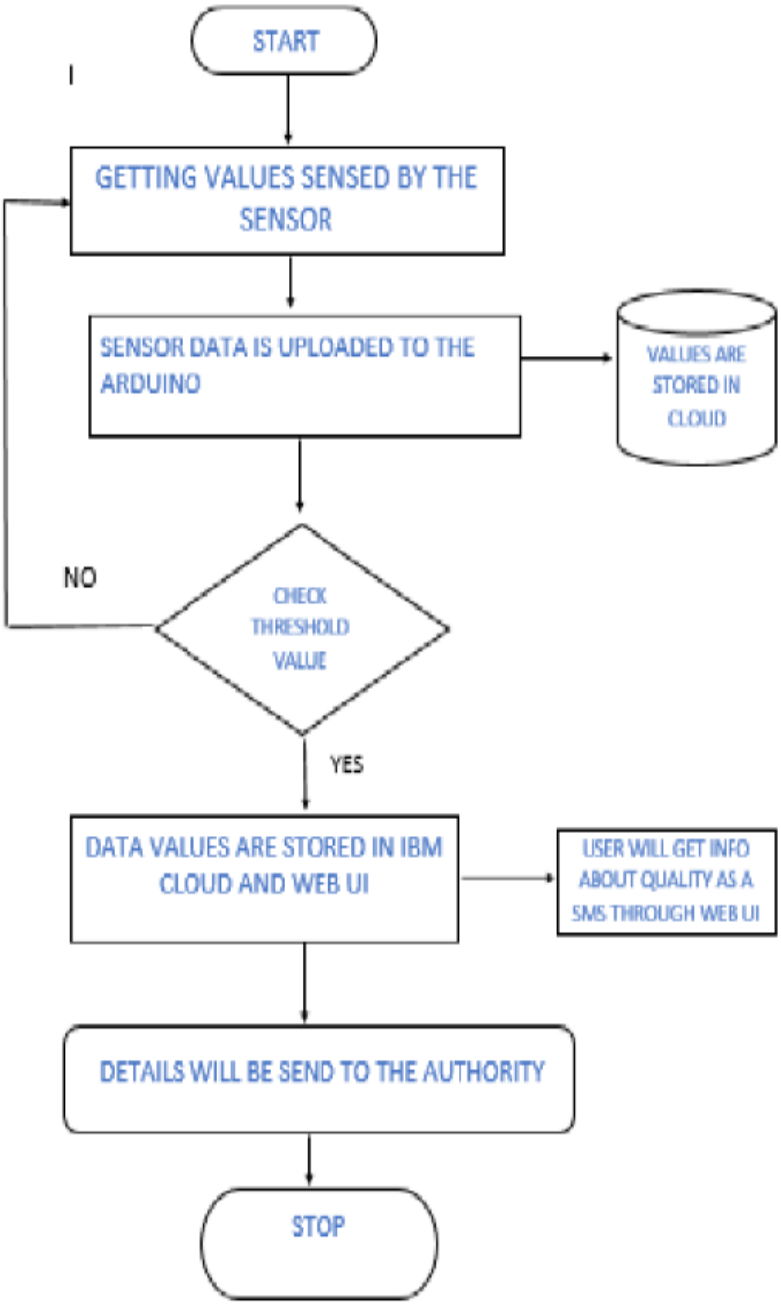
1.11 NON FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of the proposed solution

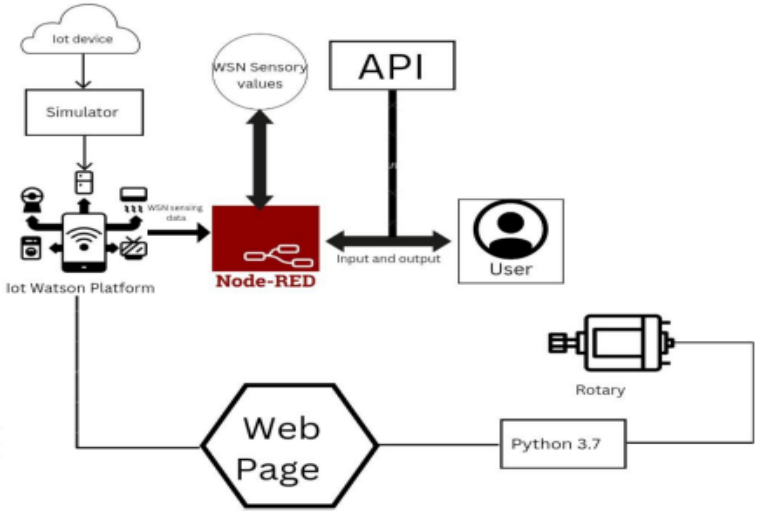
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The smart protection system defines that this project helps people to protect the drinking water andagriculture.
NFR-2	Security	We have designed this project to secure the waterfrom contaminated water or chemical or sewage.
NFR-3	Reliability	This project will help people's in protecting theirwater and save them from several diseases
NFR-4	Performance	IOT devices and sensors are used to alert the station control person by a message when water in the river is contaminated and not suitable for drinking.
NFR-5	Availability	By developing and deploying resilient hardware and software we can protect the river from contamination chemicals, sewage etc.... and also thereby can alert the people about any contamination if happened and also can protect them from several diseases. This project can be implemented in every river across the country
NFR-6	Scalability	This project used to collect real time information in water and measure quality

PROJECT DESIGN

1.12 DATA FLOW DIAGRAM



1.13 SOLUTION AND TECHNICAL ARCHITECTURE



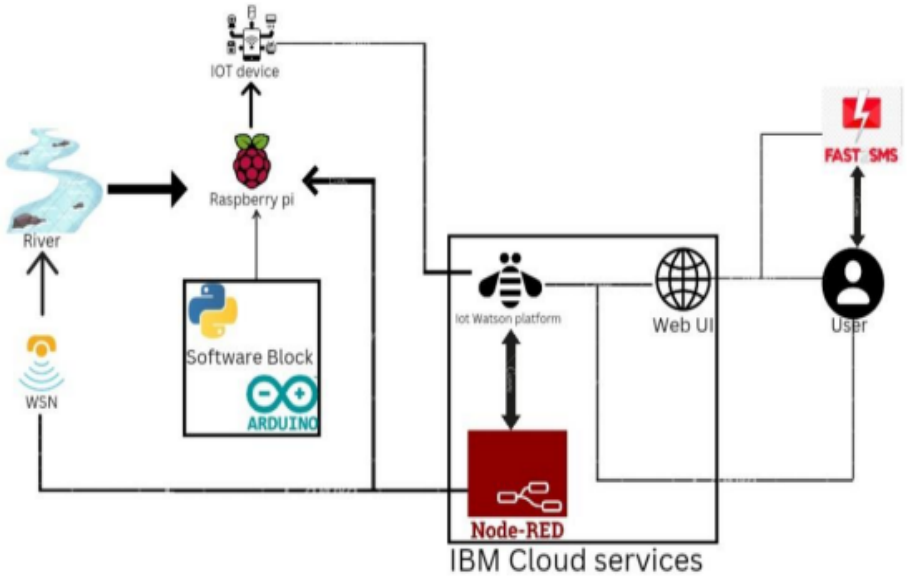


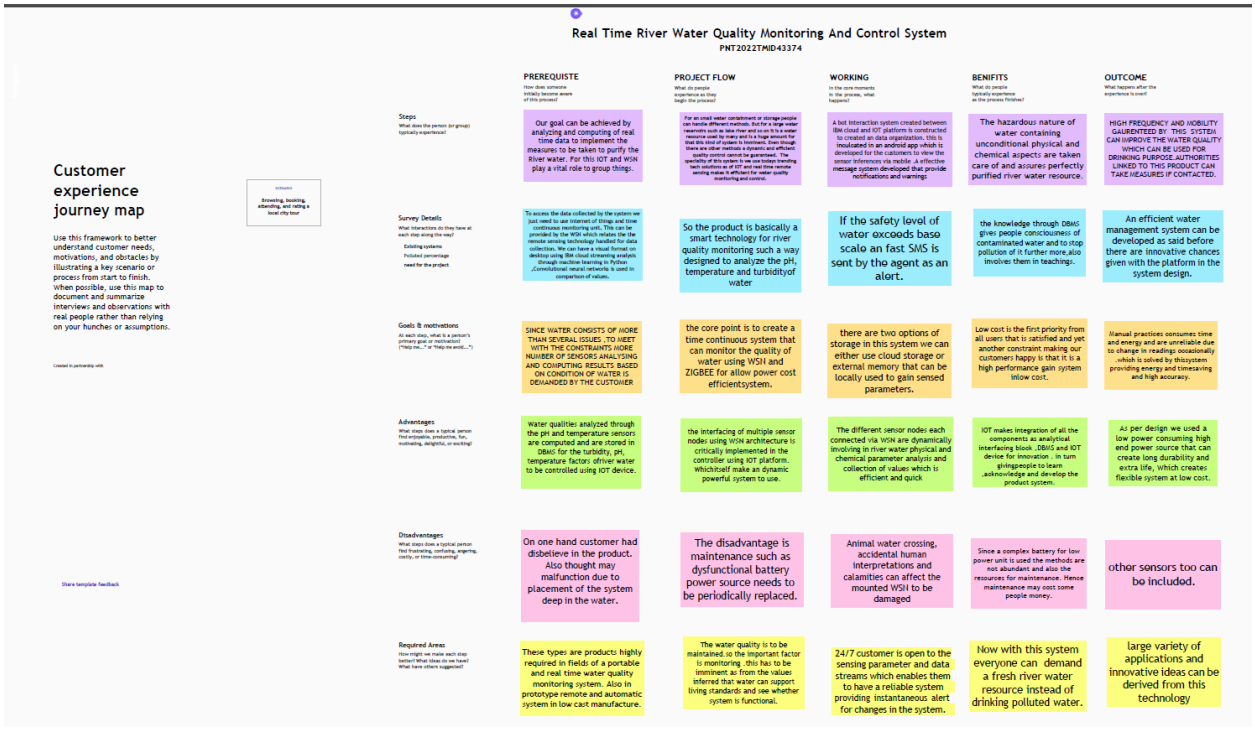
Table-1 : Components & Technologies:

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, Hive mqtt 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.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Technology of Open source framework Wokwi, IBMwatson, NodeRed
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Technology used
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc	Technology used

1.14 USER STORIES



PROJECT PLANNING & SCHEDULING

1.15 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	P.Kaviya, T. Preethi
Sprint-2	confirmation	USN-2	As a user, I will receive confirmation emailonce I have registered for the application	1	High	Praneetha S S Ramanya U S
	IBM cloud service access		Get access to ibm cloud services	2	High	
Sprint-3	Create the IBM Watson IOT and Device Settings	USN-6	To create the IBM watson platform and integrate the microcontroller with it ,to send the sensed data and cloud	2	High	RamanyaU S, Preethi T, Kaviya P, Praneetha S S

	Create a Nodered Service	USN-7	TO create a nodered service to intergrate the IBM watson along with the web UI	2	Medium	
	Create a Web UI	USN-8	To create a web UI,to access the data from the cloud and display all the parameters	2	Medium	
	To develop a python code	USN-9	Create a python code to sense the physical quantity and store data.	2	Medium	
Sprint-4	Publish data to cloud	USN-10	Publish data that is sensed by the microcontroller to the cloud	3	High	P. Kaviya, T.Preethi, Praneetha S S ,Ramanya U S
	Fast SMS service	USN-11	Use fast SMS to send alert message to once the parameters like pH,turbidity and temperature goes beyond the threshold	3	High	

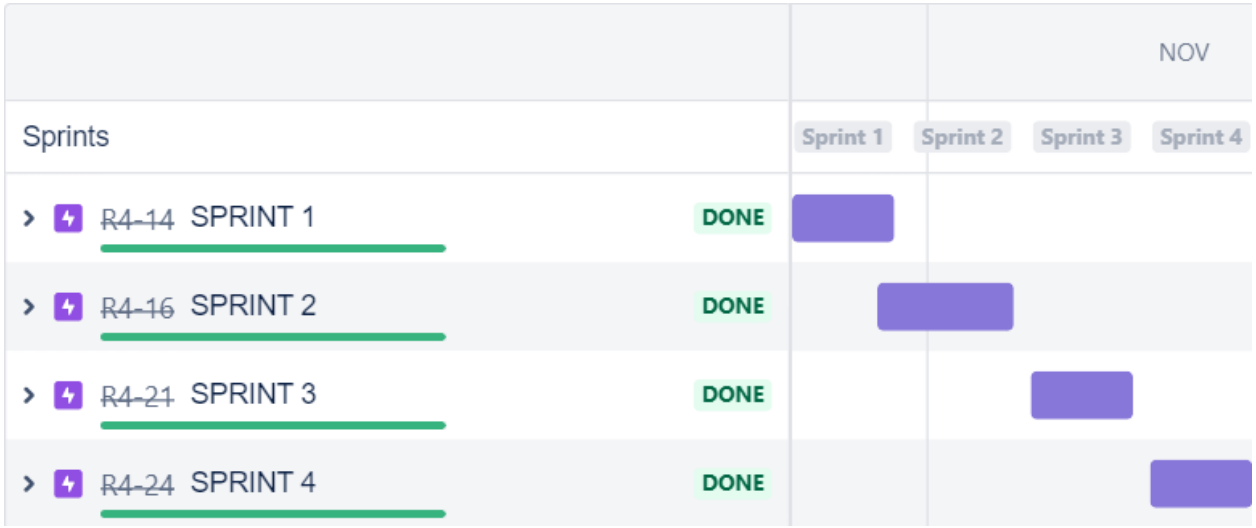
	Testing	USN-12	Testing of project and final deliverables	3	Medium	

1.16 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	4 Days	24 Oct 2022	27 Oct 2022	20	29 Oct 2022
Sprint-2	20	5 Days	28 Oct 2022	01 Nov 2022	20	04 Nov 2022
Sprint-3	20	8 Days	02 Nov 2022	09 Nov 2022	20	11 Nov 2022
Sprint-4	20	9 Days	10 Nov 2022	18 Nov 2022	20	19 Nov 2022

1.17 REPORTS FROM JIRA

ROADMAP:



AV = Sprint duration / Velocity =20/10

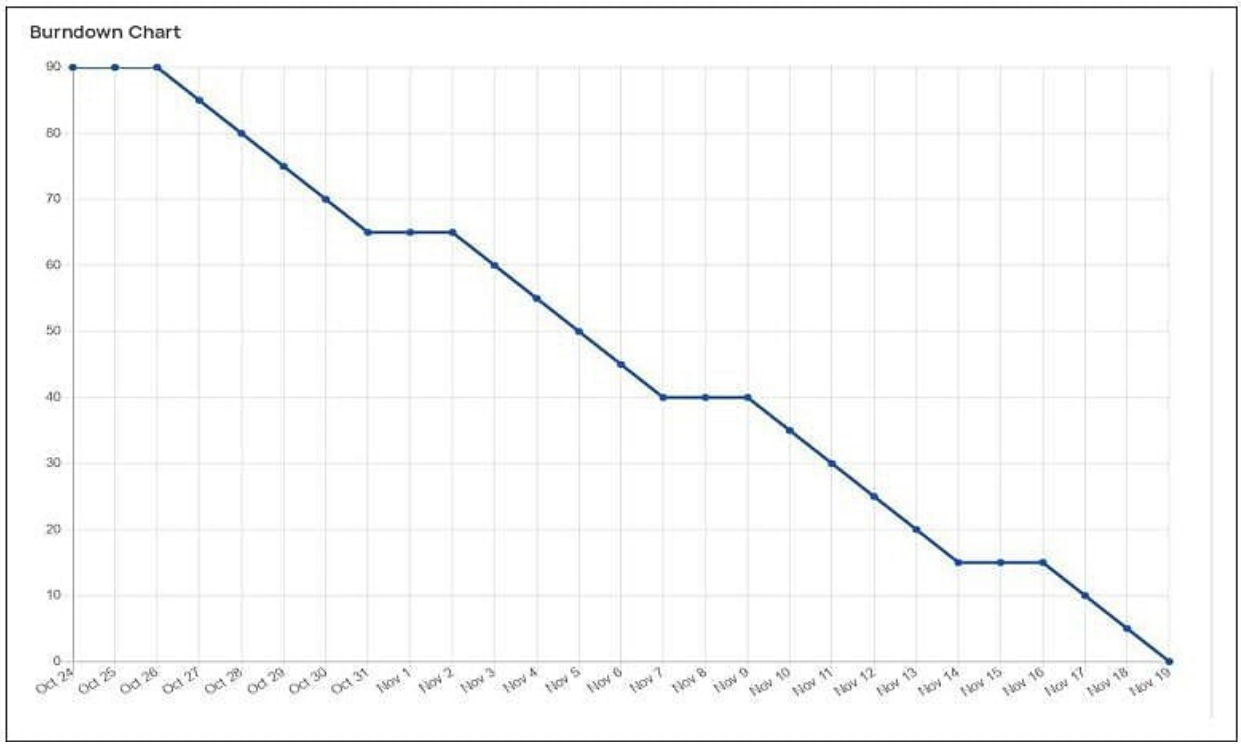
= 2

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

Burndown Chart:

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



CODING AND SOLUTIONING

DEVELOPMENT OF PYTHON SCRIPT

Importing the library functions:

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

IBM watson device credentials:

```
organization="gdkgkx"
deviceType="kprp"
deviceid="2222"
authMethod="token"
authToken="na)UXp4FWOjf1iJhOn"

```

Generate random values for pH and turbidity:

```
def myCommandCallback(cmd):
    print ("command received: %s" % cmd.data)
    if(cmd.data['command']=="MOTOR_ON"):
        print('motoron')
    elif(cmd.data['command']=="MOTOR_OFF"):
        print('motoroff')
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 sending data:

```
 deviceCli.connect()

 while True:
     pH=random.randint(0,14)
     turb=random.randint(0,250)
     temp=random.randint(0,40)

     data={'pH':pH,'Turbidity':turb,'Temperature':temp}
     print(data)
     def myOnPublishCallBack():
         print("pH Value of Water %s " %pH)
         print("Turbidity Value of Water %s " %turb)
         print("Temperature Value of Water %s " %temp)
     success=deviceCli.publishEvent("IoTSensor","json",data,qos=0,on_publish=myOnPublishCallBack)
     if not success:
         print ("Not connected to IoTF")
     time.sleep(2)

     deviceCli.commandCallback=myCommandCallback
```

Disconnect the device from the cloud:

```
 deviceCli.connect()
```

TESTING

1.21 USER ACCEPTANCE TESTING

1. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

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	4	1	2	10
Fixed	10	1	5	17	33
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	2	3
Won't Fix	0	3	3	1	7
Totals	24	13	17	25	79

2. Test case Analysis:

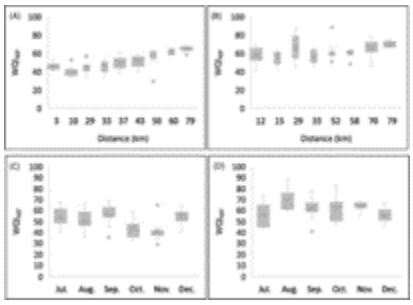
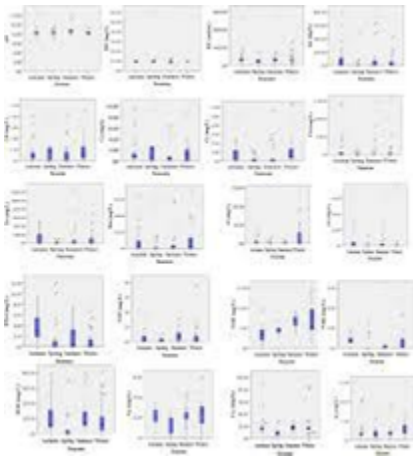
This report shows the number of test cases that have passed, failed, and untested.

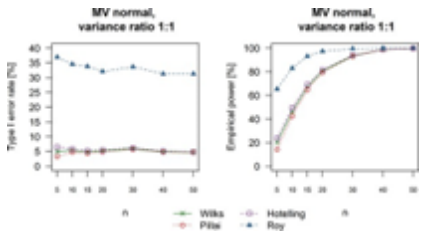
Section	Total Cases	Not Tested	Fail	Pass
Print Engine	15	0	0	15
Client Application	45	0	0	45
Security	1	0	0	1

Outsource Shipping	2	0	0	2
Exception Reporting	10	0	0	10
Final Report Output	4	0	0	4
Version Control	3	0	0	3

RESULTS

1.23 PERFORMANCE METRICS

S.no	Parameter	Values	Screenshot
1.	Model summary	Real time river water quality monitoring system is based on iot which is implemented such a way for best product performance.	
2.	Accuracy	Training accuracy- While training the start point may be front end or back end.so there is no disturbance while training as objects,modules and methods are perfectly implemented. Validation accuracy-	

		Risk management is immediate and efficient as risk resources act immediately.its because resources are implemented long before testing of products.	
3.	Confidence level(only yolo project)	Class detected- Yes Confidence score- 90%	

ADVANTAGES AND DISADVANTAGES

Any system of its own has both advantages and disadvantages. The advantages and disadvantages of this model are as follows

ADVANTAGES:

Using this system has many advantages. Some of them are:

- can provide valuable data on the condition of a particular body of water.
- allowing you to accurately create a water management plan with the data.
- analyse data continually and instantly alert users to changes in system.
- allowing easy access of all the data in one place, accessible via any enabled device.
- protect human health and avoid the costs related to medical care, productivity loss, and even loss of life.

DISADVANTAGES:

Using this system has some disadvantages. Some of them are:

- Not reliable for long distance can only apply to single source of water.
- The system is less effective as sensors are installed very deep inside the water and their positions are fixed
- The sensors are very expensive. Moreover their maintenance cost is also very high. This leads to higher cost on the regulatory body.
- The sensors which work on power source may often required to be replaced in case of malfunctioning.
- Mounted Sensors may get damage during natural disasters and often by aquatic animals.

CONCLUSION

Water turbidity, PH, and temperature are monitored using a water detection sensor that has a unique advantage and is already connected to a GSM network. The technology can automatically monitor water quality, is low-cost, and does not require personnel to be on duty.

As a result, water quality testing will most likely be more cost-effective, convenient, and quick. The method is very adaptable. This system may be used to monitor different water quality metrics by simply replacing the matching sensors and modifying the required software packages. The procedure is straightforward. The system can be expanded to track hydrologic, air pollution, industrial, and agricultural output, among other things. It is widely used and has a large number of applications. Keeping embedded devices in the environment for monitoring allows the environment to protect itself (i.e., smart environment).

This will necessitate the deployment of sensor devices in the environment for data collection and processing. We can bring the environment to life by placing sensor devices in it, allowing it to communicate with other things over the network. The end user will then have access to the collected data and analysis results via Wi-Fi.

FUTURE SCOPE

The future scope of this project is monitoring environmental conditions, drinking water quality, treatment and disinfection of waste water etc. This system could also be implemented in various industrial processes. The system can be modified according to the needs of the user and can be implemented along with lab view to monitor data on computers.

Main Improvements can be done in:

- Connections between nodes and base station are via WSN, while connections among different base stations are via Ethernet. The Ethernet can also be connected to Internet so that users can login to the system and get real time water quality data faraway Process Improvement
- The wireless data acquisition from remote places and database storage is the supporting structure of the system which can be used for further research studies like soil content analysis using different simulators.
- Finally, the prototype system with a single sensor node and base station is designed and implemented. Real-time water quality data can be seen from a GUI window in PC.

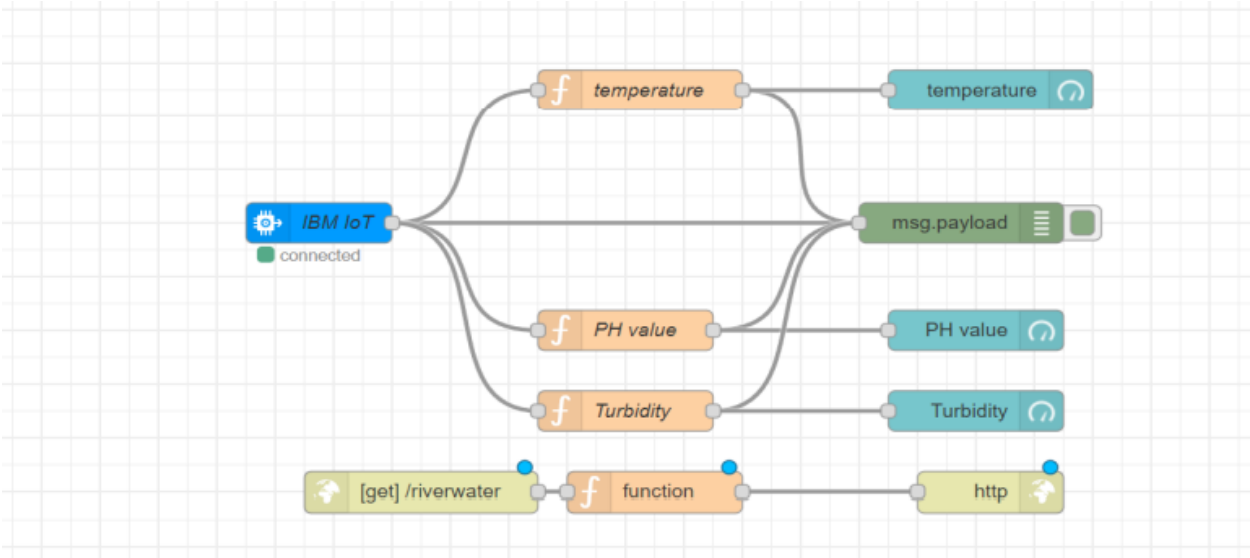
APPENDIX

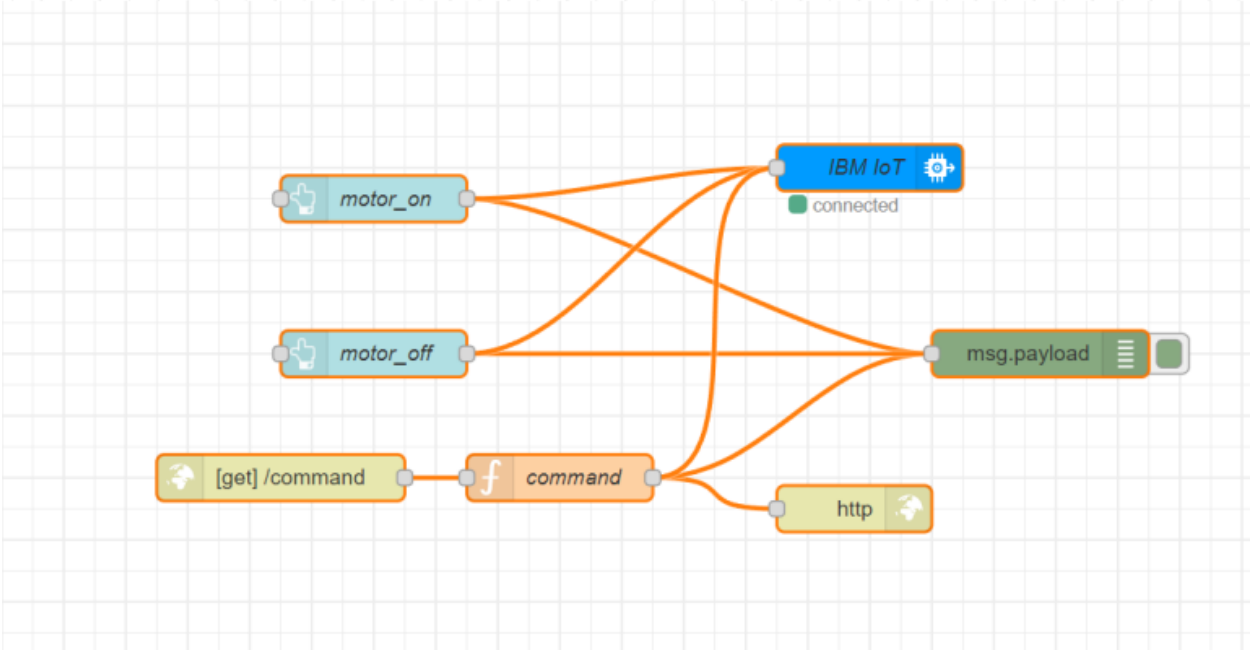
SOURCE CODE

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

#ibm watson device credentials

organization="gdkgkx"
deviceType="kprp"
deviceid="2222"
authMethod="token"
authToken="na)UXp4FWOjf1iJhOn"

#generate random values for pH and turbity


def myCommandCallback(cmd):
    print ("command received: %s" % cmd.data)
    if(cmd.data['command']=="MOTOR_ON"):
        print('motoron')
    elif(cmd.data['command']=="MOTOR_OFF"):
        print('motoroff')
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 sending data

deviceCli.connect()

while True:
    pH=random.randint(0,14)
    turb=random.randint(0,250)
    temp=random.randint(0,40)

    data={'pH':pH,'Turbidity':turb,'Temperature':temp}
    print(data)
    def myOnPublishCallBack():
        print("pH Value of Water %s " %pH)
        print("Turbidity Value of Water %s " %turb)
        print("Temperature Value of Water %s " %temp)
    success=deviceCli.publishEvent("IoTSensor","json",data,qos=0,on_publish=myOnPublishCallBack)
    if not success:
        print ("Not connected to IoTF")
    time.sleep(2)

    deviceCli.commandCallback=myCommandCallback


#disconnect the device from the cloud

deviceCli.connect()
```

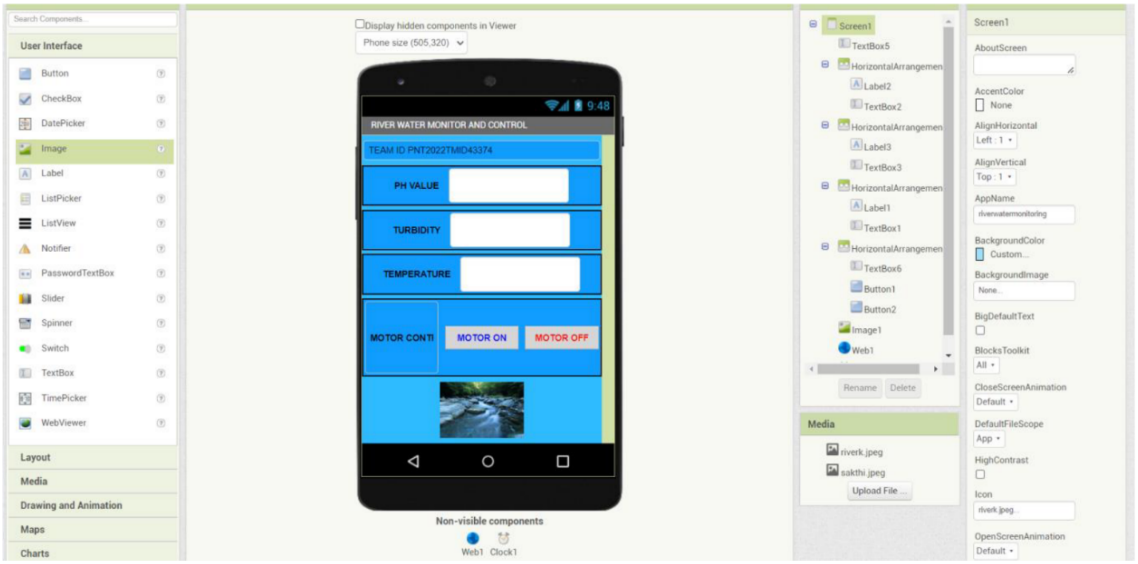
NODE RED


temperature
temperature
IBM IoT
connected
msg.payload
PH value
PH value
Turbidity
Turbidity
[get] /riverwater
function
http

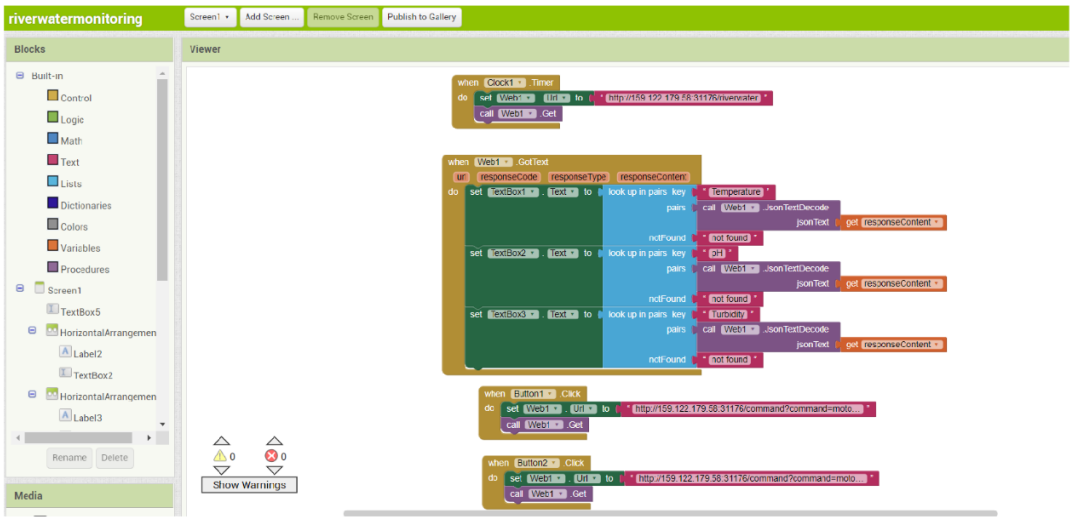

motor_on
IBM IoT
connected
motor_off
msg.payload
[get] /command
command
http

MOBILE APP USING MIT APP INVENTOR:

Design Your UI To Display The Water Turbidity, PH Values, Temperature values



Configure The Application To Receive The Data From Cloud



GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-42399-1660661456

DEMO LINK:

https://drive.google.com/file/d/1kO1WRaEDn2bA5lrm8cNUPQ9xL9u5j5zU/view?usp=share_link