REAL-TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

NALAIYA THIRAN PROJECT BASED LEARNING
ON
PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

A PROJECT REPORT

B.DEEPIKA	410119106010
S.INDIRAPRIYADHARSHINI	410119106018
H.NIVETHITHA	410119106041
K.PRIYADHARSHINI	410119106050

BACHELOR OF ENGINEERING
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

ADHI COLLEGE OF ENGINEERING AND TECHNOLOGY

Approved by AICTE, New Delhi, Permanent Affiliation Status by ANNA UNIVERSITY,Chennai,NAAC Accredited Institution, Recognized U/S 12(B)&2(f)of UGC Act 1956,An ISO Certified Institution.

FACULTY MENTOR:

Ms.R.VITHYA

Assistant Professor

Department of Electronics and Communication Engineering

Adhi College of Engineering and Technology

Kanchipuram-631605

EVALUATOR:

Mr.V.RAJINIKANTH

Assistant Professor

Department of Electronics and Communication Engineering

Adhi College of Engineering and Technology

Kanchipuram-631605

INDUSTRY MENTOR :

Mr.BHARADWAJ

IBM

ABSTRACT

Water pollution is one among the most important 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. In this paper we present a design and development of a coffee cost system for real time monitoring of the water quality in IOT(internet of things). The system contains several sensors is employed to measuring physical and chemical parameters of the water. The parameters like temperature, pH, turbidity, flow sensor of the water are often measured. The measured values from the sensors are often processed by the core controller. The Arduino model is core controller. Finally, the sensor data are often viewed on internet using WI-FI system.

Key Words—iot, sensors, arduino.

TABLE OF CONTENTS:

1.INTRODUCTION

There are numerous advances in the twenty-first century, but at the same time, pollution, heating, and other forms of pollution are forming, and as a result, there is no safe beverage for the world's pollution. Water quality monitoring in real time is becoming more difficult as a result of increasing water scarcity, population growth, and other factors.As a result, better approaches for monitoring water quality metrics in real time are required. The parameters of water quality the concentration of hydrogen ions is measured by pH. It indicates whether or not the water is acidic or alkaline. Pure water has a pH of 7, although it is acidic rather than alkaline. pH ranges from 0 to 14.It should be between 6.5 and 8.5 pH for drinking. Turbidity is a measurement of the unseen suspended particles in water. The greater the turbidity, the greater the risk of diarrhoea, cholera. If the turbidity is low, the water is safe to drink.The temperature sensor detects how hot or cold the water is. Flow sensor is a device that measures the flow of water. The traditional method of water quality monitoring entails manually collecting water samples from various sites. The use of wireless communication technologies is becoming more common to help people with their personal and daily duties. Many building control, automation, and data collecting applications have been created in recent years. There are numerous advantages,such as minimal cost, ease of installation, and maintenance. The remote device network can be used for a variety of tasks, including agriculture and traffic control, remote health care, forest management, security, and surveillance.

2.OBJECTIVE

The major goal is to create a system that uses wireless sensor networks to monitor river water quality at remote locations with low power consumption continuously, low cost and high detection accuracy. pH, conductivity, turbidity level and other parameters are measured in order to enhance water quality. The remote sensing technology is the cornerstone of IoT-based water quality monitoring. This implements the approach by using the pH sensor, turbidity sensor to obtain analog readings for water contaminates. In addition, for the specific application, we can add extra sensor elements.

3.IDEATION PHASE

3.1.Literature survey:

Current water quality monitoring system is a manual system with a monotonous process and is very time-consuming. This paper proposes a sensor-based water quality monitoring system. The main components of Wireless Sensor Network (WSN) include a micro-controller for processing the system, communication system for inter and intra node communication and several sensors. Real-time data access can be done by using remote monitoring and Internet of Things (IoT) technology. Data collected at the apart 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 values. If the acquired value is above the threshold value automated warning SMS alert will be sent to the agent. The uniqueness of our proposed paper is to obtain the water monitoring system with high frequency, high mobility, and low powered. Therefore, our proposed system will immensely help Bangladeshi populations to become conscious against contaminated water as well as to stop polluting the water.

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 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 data capture, analyses and fast dissemination of information to relevant stakeholders for making timely and informed decisions. In this paper, a real 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 at the site of operation. The system consists of a Raspberry Pi, Analog to Digital Converter, Water quality measurement sensors. 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.

3.2 EMPATHY MAP

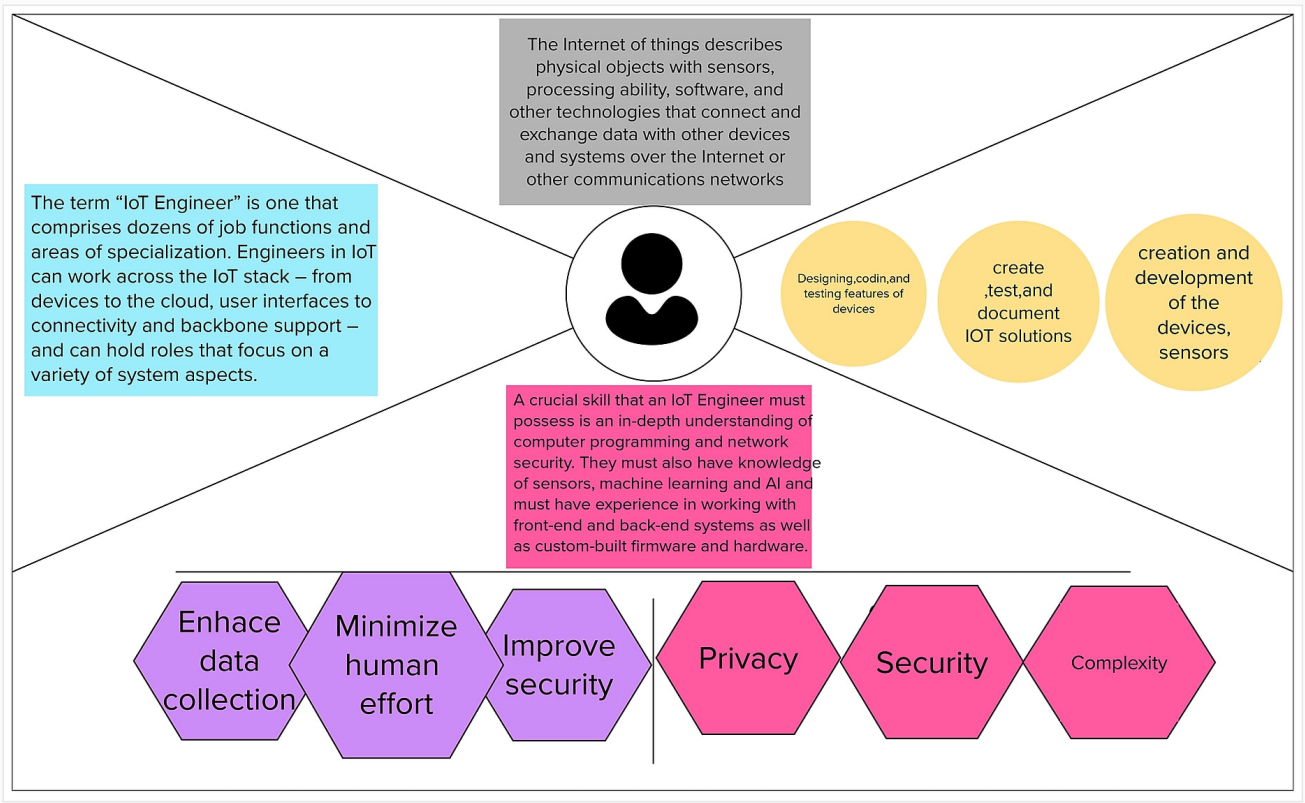
Edit this template
Right-click to unlock

Empathy Map Canvas

Gain insight and understanding on solving customer problems.

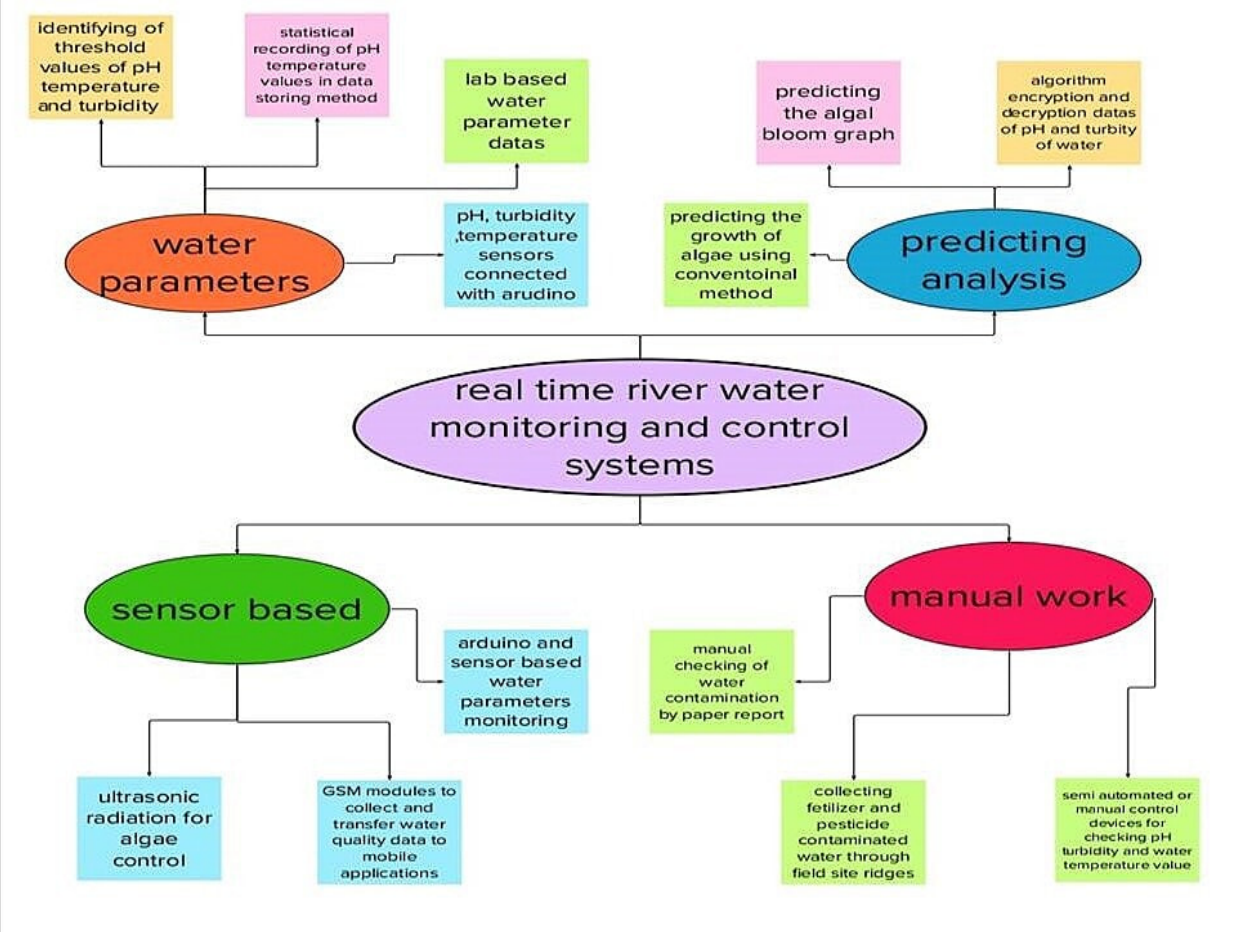
1

Build empathy and keep your focus on the user by putting yourself in their shoes.

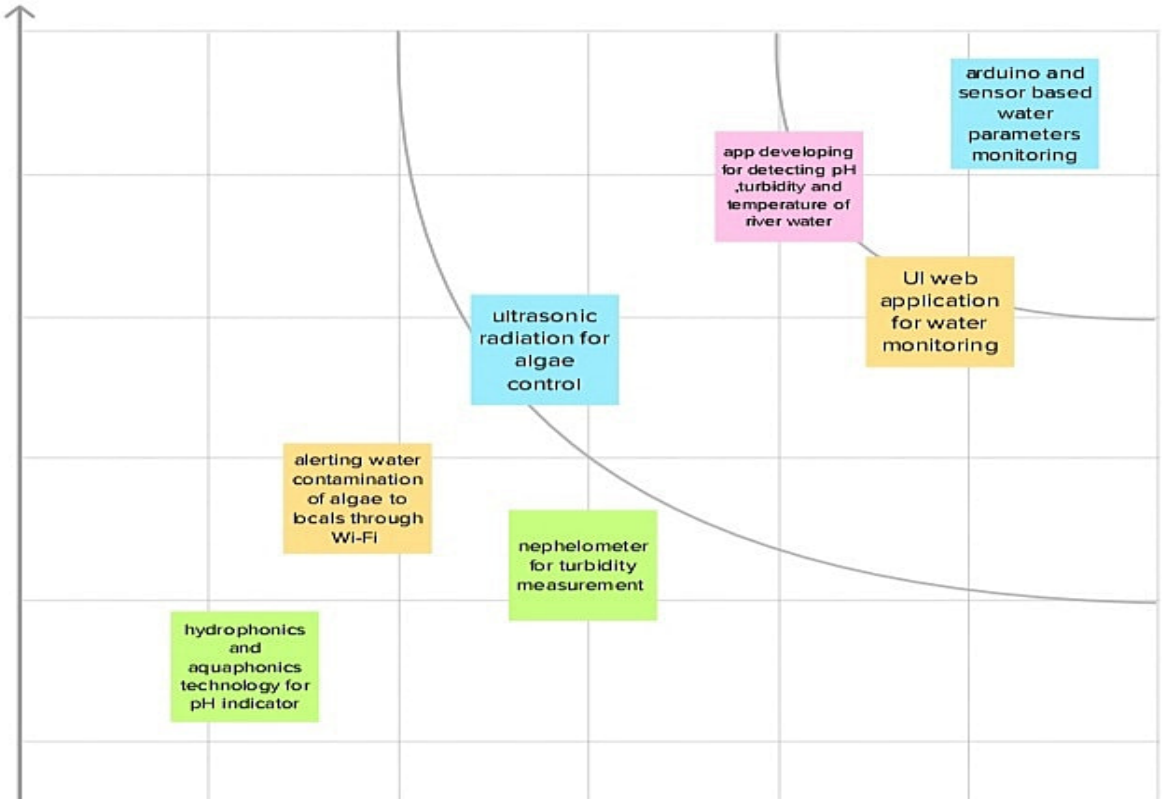


Share your feedback

3.3.IDEATION:


identifying of threshold values of pH temperature and turbidity
statistical recording of pH temperature values in data storing method
lab based water parameter datas
predicting the algal bloom graph
algorithm encryption and decryption datas of pH and turbity of water
water parameters
pH, turbidity ,temperature sensors connected with arudino
predicting the growth of algae using conventoinal method
predicting analysis
real time river water monitoring and control systems
sensor based
arduino and sensor based water parameters monitoring
manual work
manual checking of water contamination by paper report
ultrasonic radiation for algae control
GSM modules to collect and transfer water quality data to mobile applications
collecting fetilizer and pesticide contaminated water through field site ridges
semi automated or manual control devices for checking pH turbidity and water temperature value

3.4.Brainstroming:



4.PROJECT DESIGN PHASE-1

4.1 PROPOSED SOLUTION

S.No	Parameter	Description
1.	Problem Statement(problem To Be Solved)	To Control The Algal Bloom And Monitor The Water Parameters Such As Ph, Turbidity And Dissolved Solvents.
2.	Idea/solution Description	Monitoring Water Parameters By Using Arduino And Sensors And Control Measures By Ultrasonic Frequency.
3.	Novelty/uniqueness	Controlling Algal Blooms Using Ultrasonic Frequencies
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 Effortless.

4.2 Problem Solution Fit:

4.3 Solution Architecture:

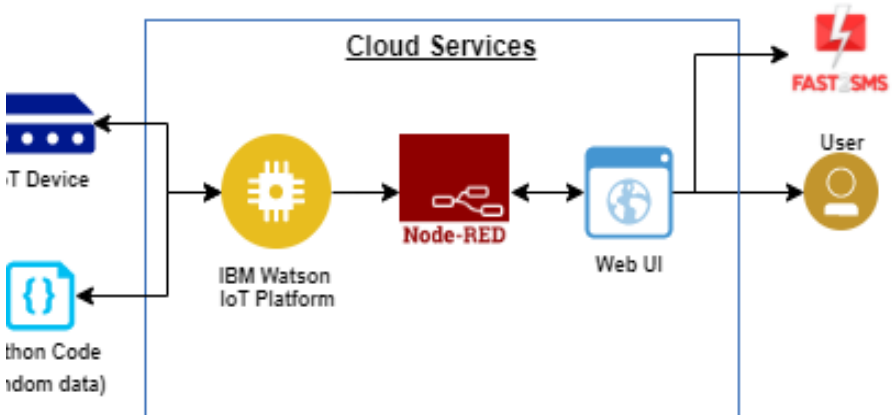
The goals of concept implementation are as follows:

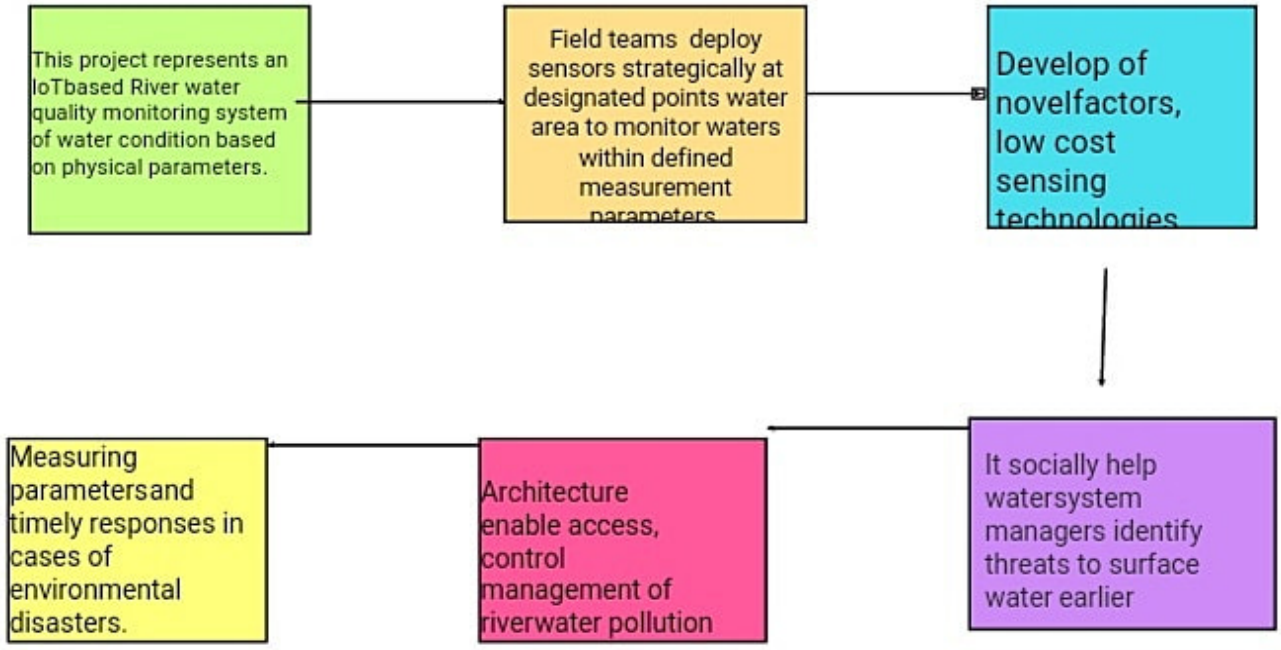
• Using accessible sensors at a distant location, monitor water parameters such as pH, dissolved oxygen, turbidity, conductivity, and so on.

• To collect data from various sensor nodes and transfer it through wireless channel to the base station.

• For quality control, to simulate and assess quality parameters.

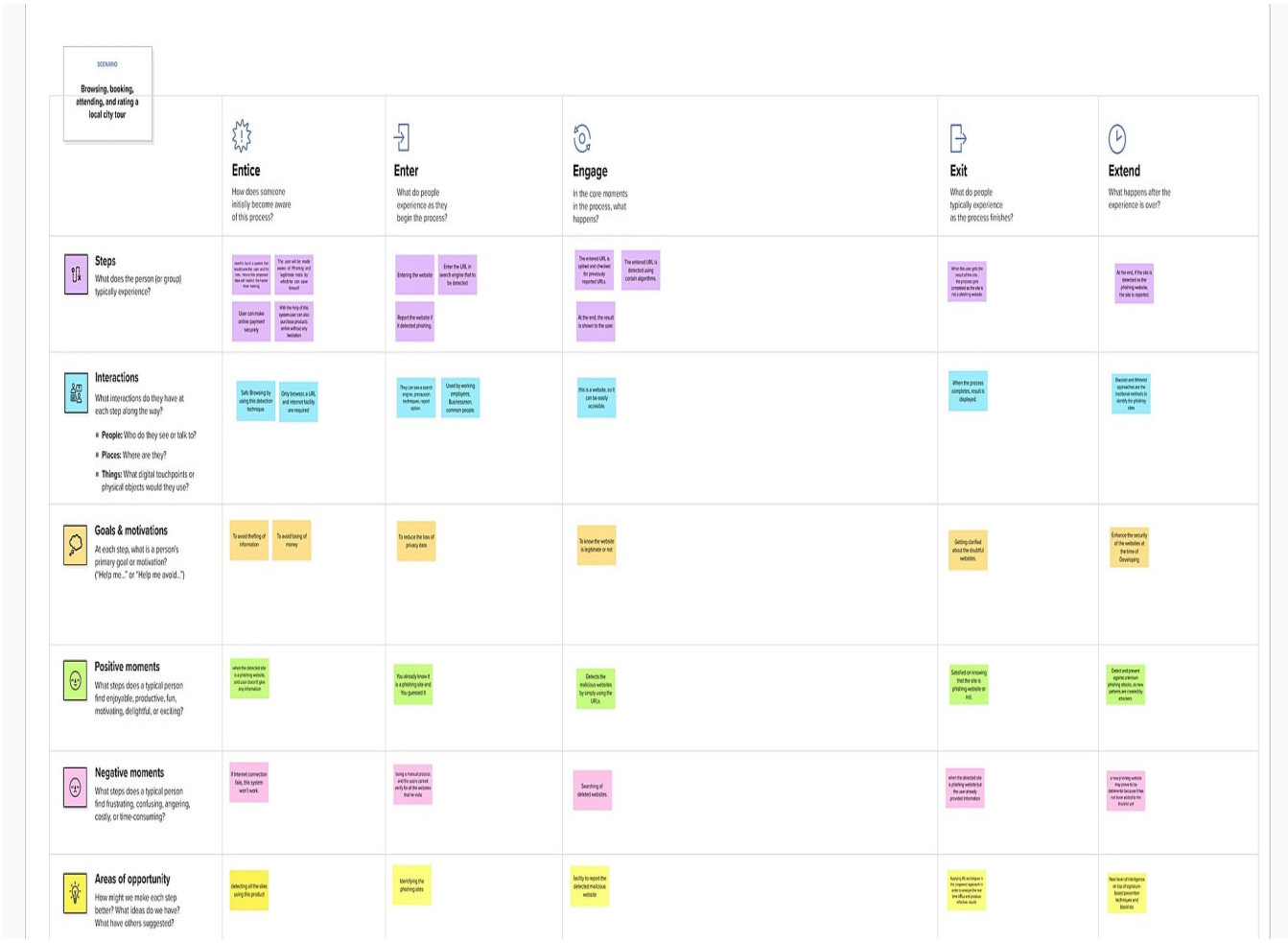
• When the water quality observed does not meet the established standards, send an SMS to an authorized person on a regular basis so that relevant steps can be performed.





5.PROJECT DESIGN PHASE-2

5.1 Customer Journey Map:



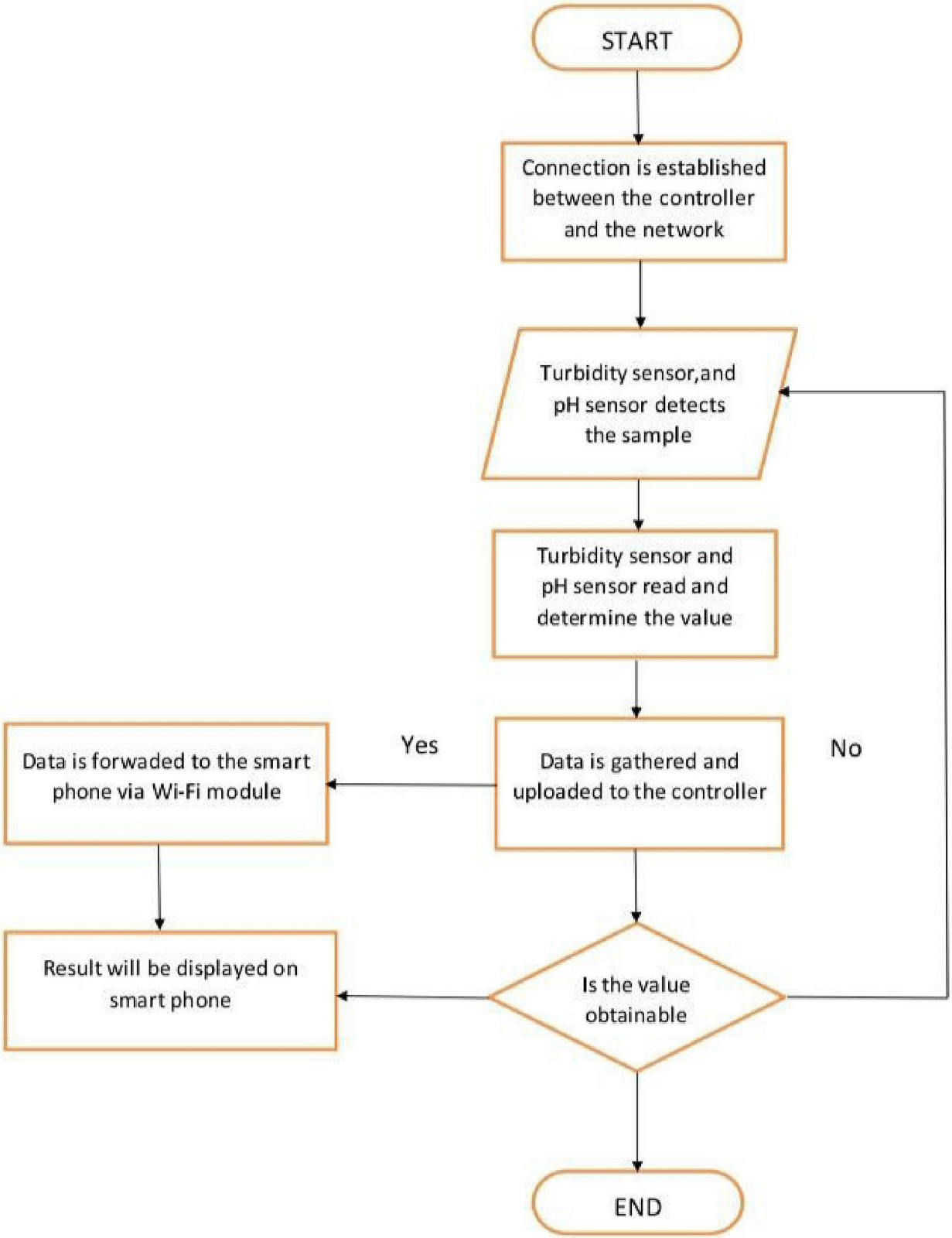
5.2 Functional Requirements:

Fr.No	Functional Requirments(epic)	Sub Requirement (story / Sub-task)
Fr-1	User Registration	Registration Through Form Registration Through Gmail Registration Through Product Mobile UI
Fr-2	User Confirmation	Confirmation Via Email Confirmation Via OTP
Fr-3	Ph Level Detection	Ph Sensor Is Used To Monitor The Water Quality And The Signals Are Send Arduino
Fr-4	Turbidity Detection	Turbidity Sensor Ts-300b Measures The Turbidity (counter Of Suspended Matter) In The Wash Water And The Signals Are Send To Arduino.
Fr-5	Ultrasonic Generator	Waves Generated At Regular Interval Times To Clear Algae 25%,50%,100%

Non Functional Requirements:

fr.no	non functional requirements	description
nfr-1	usability	efficient to use and has simple monitoring system.
nfr-2	security	mobile application is secured with firewalls protection.
nfr-3	reliability	real time sensor output values with future predicted data storage 98% efficient monitoring output assurance for aquaculture safety
nfr-4	performance	greater performance and environment
nfr-5	availability	in form of mobileUI 24*7 monitoring system
nfr-6	scalability	highly scalable.It is capable to produce a best final output.

5.3 Data Flow Diagram:



5.4 Technology Architecture:

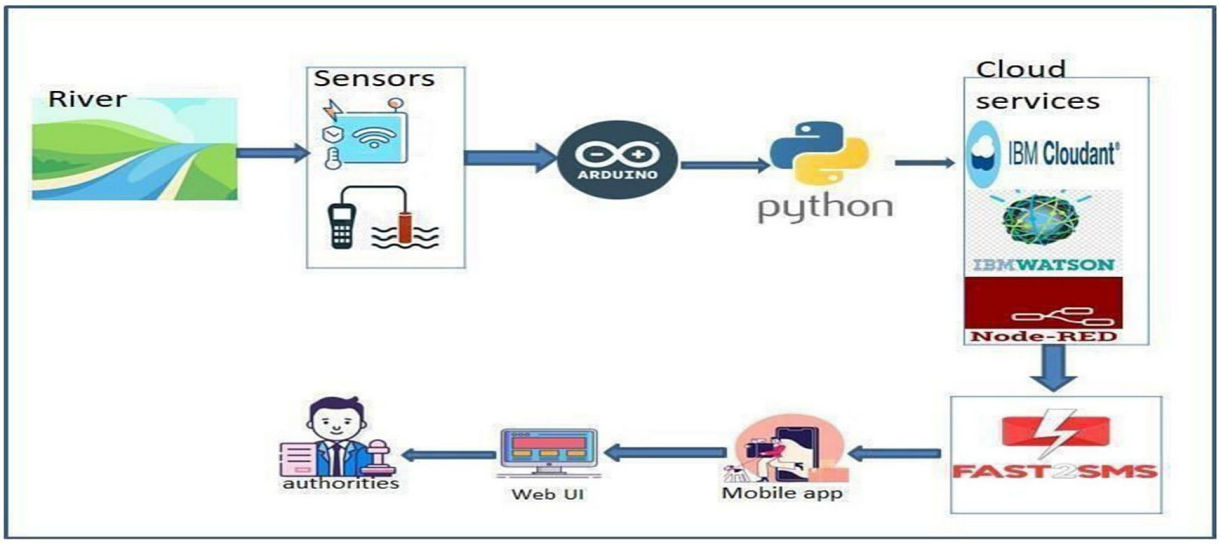


TABLE-1: Components & Technologies:

S.No	Components	Description	Technology
1	user interface	how user interacts with application	HTML,CSS,node red,cloud,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 services
4	application logic-3	Logic for a process in the application	IBM WATSON Assistant
5	Data Base	data type,configuration etc	MySQL, postgresSQL
6	Cloud database	database service on cloud	IBM DB2,IBM Cloudant etc
7	file storage	file storage requirements	IBM block storage or local file system
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 laerning model	purpose of external API used in the application	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 opensource framework
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	technology used
5	performance	design consideration for the performance of the application	technology used

6.PROJECT PLANNING PHASE

6.1 Prepare Milestone and Activity List:

S.NO	ACTIVITY LIST	ACTIVITY DESCRIPTION	DURATION
1	Understanding the project requirement	Assign the team members and create repository in the Github, assign the task to each	1 week
2	starting of project	Advice students to aatend classes of IBM portal create and develop ana rough diagram based on the project descrption and gather of information on IOT and IBM project and team leader assign task to each member of the project	1 week
3	Attend class	Team members and team lead must watch and learn from classes provided by IBM and NALAYATHIRAN and must gain access of MIT license for their project	4 week
4	Budget and scope of project	Budget and analyze the use of IOT in the project and discuss with team for budget prediction to predict the favorability for the customer to buy	1 week

6.2 Sprint Delivery Plan:

sprint	functioal requirements(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	nivethitha h
sprint-1		USN-2	as a user, I will receive confirmation email once I have registered for the application	1	high	deepika b
sprint-2		USN-3	as a user, I can register for the application through facebook	2	low	indirapriyadharshini s
sprint-1		USN-4	as a user, I can register for the application through gmail	2	medium	priyadharshini k
sprint-1	login	USN-5	as a user , I can log into the application by entering email & password	1	high	deepika b

Project tracker:

sprint	total story points	duration	sprint start date	sprint end date(planned)	story release date(actual)	sprint release date(actual)
sprint-1	20	6 days	24 oct 2022	29 oct 2022	20	29 oct 2022
sprint-2	20	6 days	31 oct 2022	05 nov 2022	30	30 oct 2022
sprint-3	20	6 days	07 nov 2022	12 nov 2022	49	06 nov 2022
sprint-4	20	6 days	14 nov 2022	19 nov 2022	50	07 nov 2022

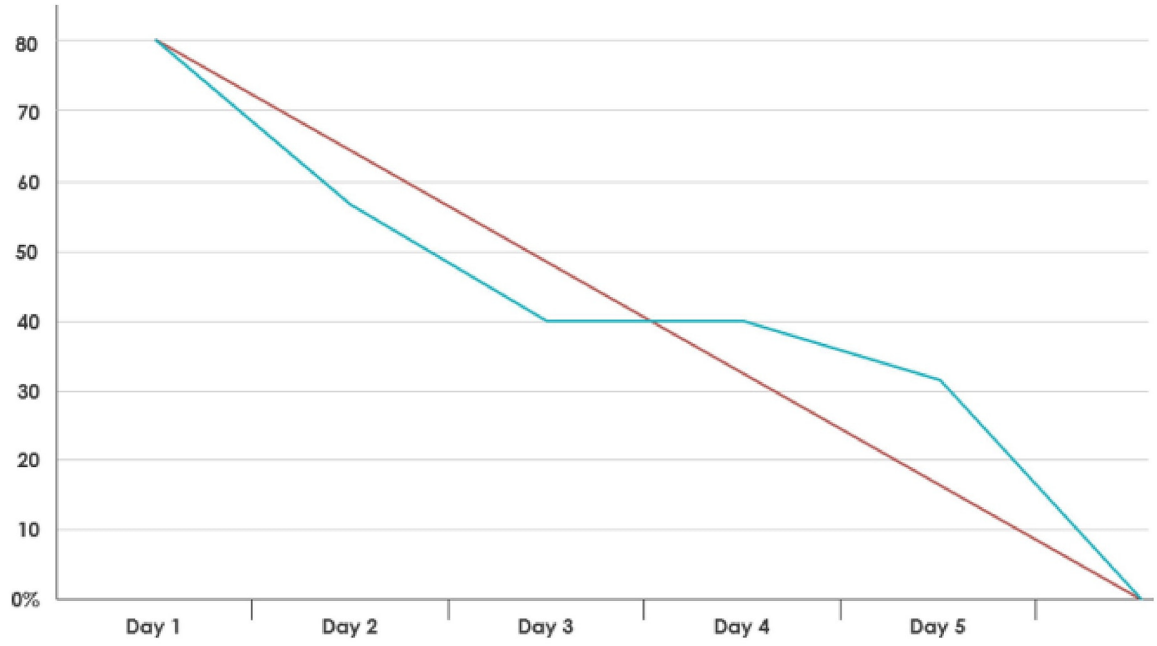
VELOCITY:

AV=SPRINT DURATION/VELOCITY

= 20/10

AV= 2

BURNDOWN CHART:



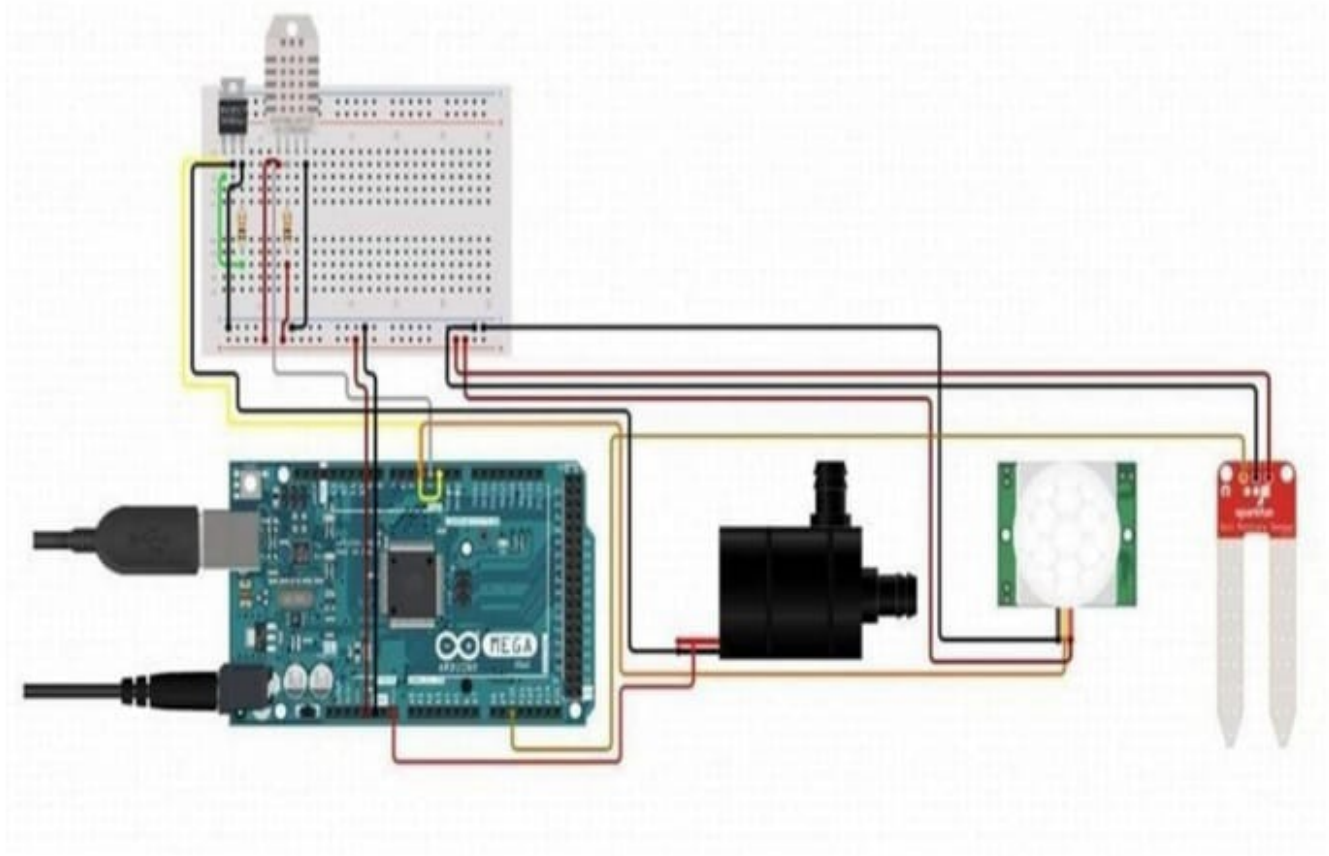
7.PROJECT DEVELOPMENT PHASE

7.1 PROJECT DEVELOPMENT - DELIVERY OF SPRINT - 1:

CONNECTING SENSORS WITH ARDUINO USING C++ CODE:

```
#include "Arduino.h"
#include "DHT.h"
#include "PIR.h"
 #include "SoilMoisture.h"
 #include "Pump.h"
#define DHT_PIN_DATA 3
 #define PIR_PIN_SIG 4
#define SOILMOISTURE_5V_PIN_SIG A10
#define WATERPUMP_PIN_COIL1 2
 DHT dht(DHT_PIN_DATA); PIR pir(PIR_PIN_SIG);
 SoilMoisture soilMoisture_5v(SOILMOISTURE_5V_PIN_SIG);
 Pump waterpump(WATERPUMP_PIN_COIL1);
 const int timeout = 10000;
char menuOption = 0;
 long time0;
void setup(){
 Serial.begin(9600);
 while (!Serial) ;
 Serial.println("start");
dht.begin();
 menuOption = menu();
 }
 void loop(){
 if(menuOption == '1') {
 float dhtHumidity = dht.readHumidity();
float dhtTempC = dht.readTempC();
 Serial.print(F("Humidity: "));
 Serial.print(dhtHumidity);
 Serial.print(F(" [%]\t"));
 Serial.print(F("Temp: "));
 Serial.print(dhtTempC);
 Serial.println(F(" [C]"));
}
 else if(menuOption == '2') {
 bool pirVal = pir.read();
Serial.print(F("Val: "));
 Serial.println(pirVal);
 }
 else if(menuOption == '3') {
 int soilMoisture_5vVal = soilMoisture_5v.read();
 Serial.print(F("Val: "));
 Serial.println(soilMoisture_5vVal);
```

```
 }
 else if(menuOption == '4') {
 waterpump.on(); delay(2000);
 waterpump.off(); delay(2000);
 }
 if (millis() - time0 > timeout){
menuOption = menu();
 }
 }
 char menu(){
 Serial.println(F("\nWhich component would you like to test?"));
 Serial.println(F("(1) DHT22/11 Humidity and Temperature Sensor"));
 Serial.println(F("(2) Infrared PIR Motion Sensor Module"));
 Serial.println(F("(3) Soil Moisture Sensor"));
 Serial.println(F("(4) Submersible Pool Water Pump"));
 Serial.println(F("(menu) send anything else or press on board reset button\n")); while
 (!Serial.available());
 while (Serial.available()){
 char c = Serial.read(); if (isAlphaNumeric(c)){
 if(c == '1')
 Serial.println(F("Now Testing DHT22/11 Humidity and Temperature Sensor"));
 else if(c == '2')
 Serial.println(F("Now Testing Infrared PIR Motion Sensor Module"));
 else if(c == '3')
 Serial.println(F("Now Testing Soil Moisture Sensor"));
 else if(c == '4')
 Serial.println(F("Now Testing Submersible Pool Water Pump"));
else{
 Serial.println(F("illegal input!"));
 return 0;
 }
 time0 = millis();
 return c;
 }
 } }
```



7.2 PROJECT DEVOLOPMENT - DELIVERY OF SPRINT - 2

```
<!DOCTYPE html>
<html>
<head>
 <h1> Real time water quality monitoring system</h1>
<metaname="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial,Impact, 'Arial Narrow Bold', sans-serif, sans-serif;}

/* Full-width input fields */
input[type=text], input[type=password] {
  width: 150;
  padding: 23px 24px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  box-sizing: border-box;
}

/* Set a style for all buttons */
button {
  background-color: #04AA6D;
  color:blue;
  padding: 15px 21px;
  margin: 8px 0;
  border: none;
  cursor: pointer;
  width: 102;
}

button:hover {
  opacity: 0.7;
}

/* Extra styles for the cancel button */
.cancelbtn {
  width: min-content
  padding: 10px 18px;
  background-color: #f4455f
}
```

```
/* Center the image and position the close button */
.imgcontainer { }
  text-align: right: ;;
  margin : 24px 0 12px 0;
  position: relative
}

img {water quality monitoring system}
  width: 56;
  border-radius: 50%;
}

.container {
  padding: 16px;
}

span.psw {
  float: right;
  padding-top: 16px;
}

/* The Modal (background) */
.modal {
  display: none; /* Hidden by default */
  position: fixed; /* Stay in place */
  z-index: 1; /* Sit on bottom*/
  left: 0;
  top: 0;
  width: 100%; /* full width */
  height: 100%; /* medium height */
  overflow: auto; /* Enable scroll if needed */
  background-color: ybg(0,0,0); /* Fallback color */
  background-color: rgba(0,0,0,0.4); /* Black w/ transprenant */
  padding-top: 60px;
}

/* Modal Content/Box */
.modal-content {
  background-color: #fefefe;
  margin: 5% auto 15% auto; /* 5% from the top, 15% from the bottom and centered */
  border: 1px solid #888;
```

```
  width: 65%; /* Could be more or less, depending on screen size */
}

/* The Close Button (x) */
.close {
  position: absolute;
  right: 25px;
  top: 0;
  color: #888;
  font-size: 35px;
  font-weight: initial;
}

.close:hover,
.close:focus {
  color: red;
  cursor: pointer;
}

/* Add Zoom Animation */
.animate {
  -webkit-animation: animatezoom 0.6s;
  animation: animatezoom 0.6s
}

@-webkit-keyframes animatezoom {
  from {-webkit-transform: scale(0)}
  to {-webkit-transform: scale(1)}
}

@keyframes animatezoom {
  from {transform: scale(2)}
  to {transform: scale(1)}
}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) {
  span.psw {
     display: block;
     float: none;
  }
  .cancelbtn {
```

```
    width: 100%;
  }
}
</style>
</head>
<body>

<h2>Modal Login Form</h2>

<button onclick="document.getElementById('id01').style.display='block'" style="width:auto;">Login</button>

<div id="id01" class="modal">

  <form class="modal-content animate" action="/action_page.php" method="post">
    <div class="imgcontainer">
                <span onclick="document.getElementById('id01').style.display='none'" class="close" title="Close Modal">&times;</span>
    </div>

    <div class="container">
      <label for="uname"><b>Username</b></label>
      <input type="text" placeholder="Enter Username" name="uname" required>

      <label for="psw"><b>Password</b></label>
      <input type="password" placeholder="Enter Password" name="psw" required>
      <label for="captch"></label><123gh@><label>
       <input type="captcha" 123@g="Enter captcha" name="captcha" requried>
      <button type="submit">Login</button>
      <label>

        <input type="checkbox" checked="checked" name="remember"> Remember me
      </label>
    </div>

    <div class="container" style="background-color:#f1f1f1">
              <button type="button" onclick="document.getElementById('id01').style.display='none'" class="cancelbtn">Cancel</button>
      <span class="psw">Forgot <a href="#">password?</a></span>
    </div>
  </form>
</div>
```

```
<script>
// Get the modal
var modal = document.getElementById('id03');

// When the user clicks anywhere outside of the modal, close it
window.onclick = function(event) {
   if (event.target == modal) {
      modal.style.display = "none";
   }
}
</script>

</body>
</html>
```

7.3 PROJECT DEVOLOPMENT - DELIVERY OF SPRINT - 3

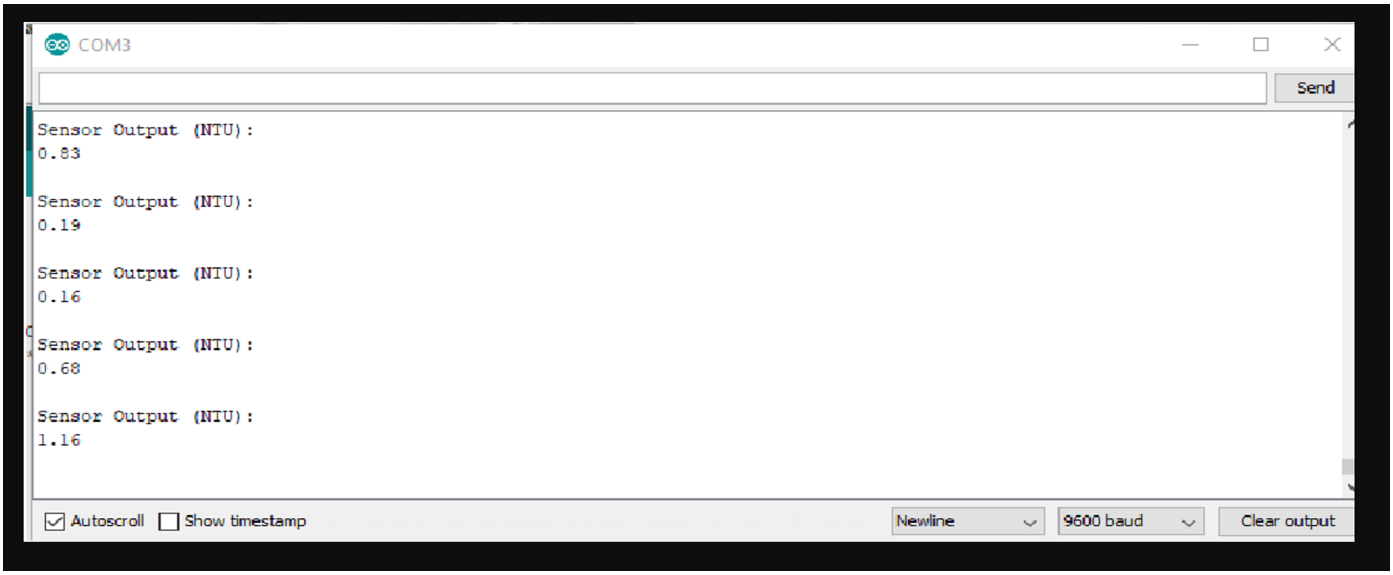
```
#importing Random function to generate the value
import random as rand
for i in range(5):
   print("Test case:",i+1)
   print("Welcome to Real-Time River Water Quality Monitoring and Control System")
   temperature = int(rand.randint(-40,125))
   pH = int(rand.randint(0,14))
   DO = int(rand.randint(0,100))
   TSS = int(rand.randint(0,3700))
   Manganese = int(rand.randint(0,1000))
   Copper = int(rand.randint(0,2000))
   ammonia_Nitrate = int(rand.randint(0,100))
   Hardness = int(rand.randint(0,1000))
   Zinc =  int(rand.randint(0,100))
   Conductivity =  f"{float(rand.uniform(0.001,2000)):.2f}"
   Chloride = int(rand.randint(0,200))
   Sulphate = int(rand.randint(0,1000))
   #These variables store value of ramdom data to be shared to the cloud

   #printing the values
   print(
      "Temperature:", temperature,
      "\npH:", pH,
      "\nDO:", DO,
      "\nTSS:", TSS,
      "\nManganese:", Manganese,
      "\nCopper:", Copper,
      "\nAmmonia & Nitrate:",ammonia_Nitrate,
      "\nHardness:",Hardness,
      "\nZinc:", Zinc,
      "\nConductivity:", Conductivity,
      "\nChloride:", Chloride,
      "\nSulphate:", Sulphate, "\n"
   )
```

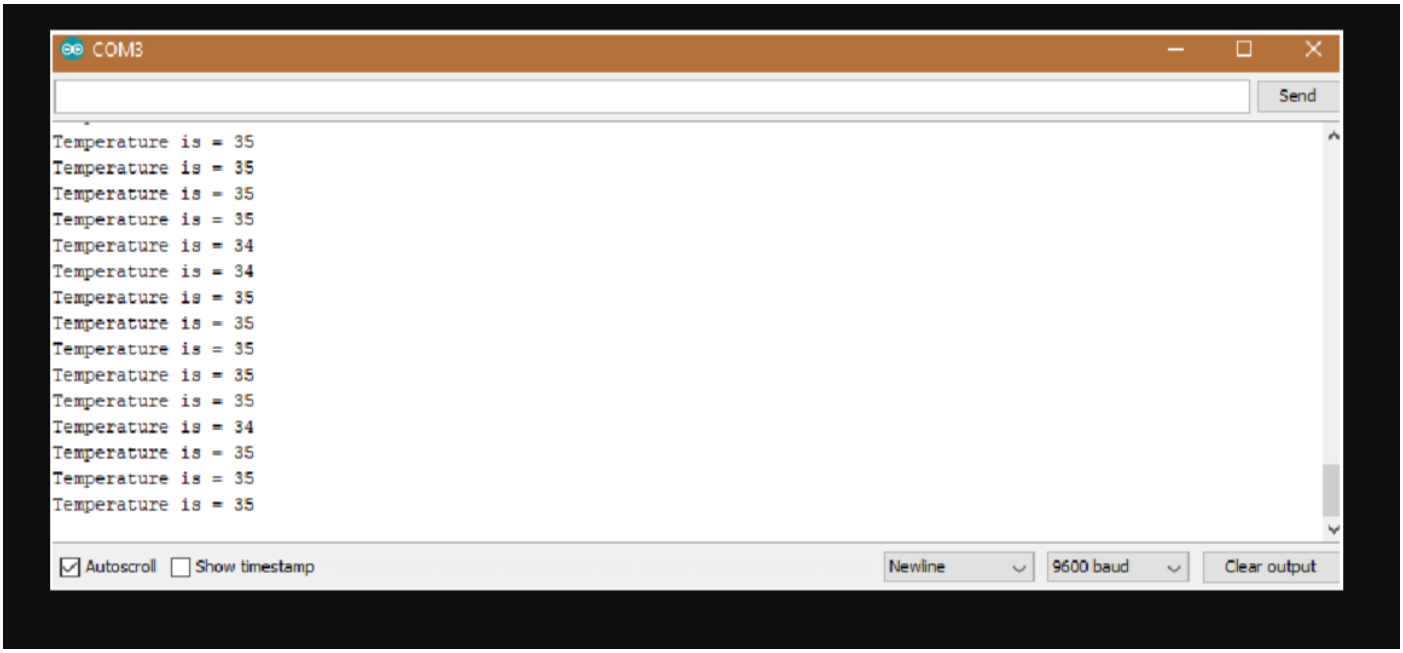
7.4 PROJECT DEVOLOPMENT - DELIVERY OF SPRINT - 4

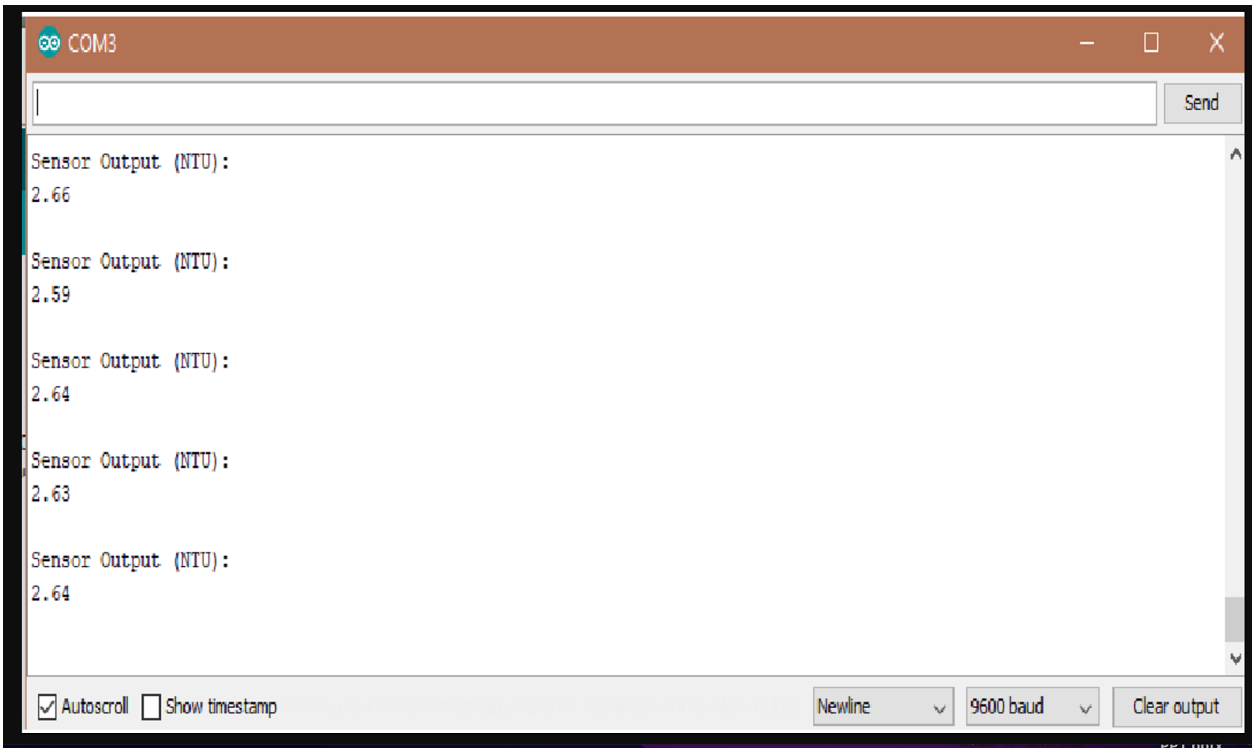
```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 5
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
float Celcius=0;
float Fahrenheit=0;
float voltage=0;
const int analogInPin = A0;
int sensorValue = 0;
unsigned long int avgValue;
float b;
int buf[10],temp;
void setup(void)
{

 Serial.begin(9600);
 sensors.begin();
 int sensorValue = analogRead(A1);
 voltage = sensorValue * (5.0 / 1024.0);
}
void loop(void)
{
 sensors.requestTemperatures();
 Celcius=sensors.getTempCByIndex(0);
 Fahrenheit=sensors.toFahrenheit(Celcius);
 for(int i=0;i<10;i++)
{
 buf[i]=analogRead(analogInPin);
 delay(10);
}
for(int i=0;i<9;i++)
38{
 for(int j=i+1;j<10;j++)
 {
 if(buf[i]>buf[j])
 {
 temp=buf[i];
 buf[i]=buf[j];
 buf[j]=temp;
```

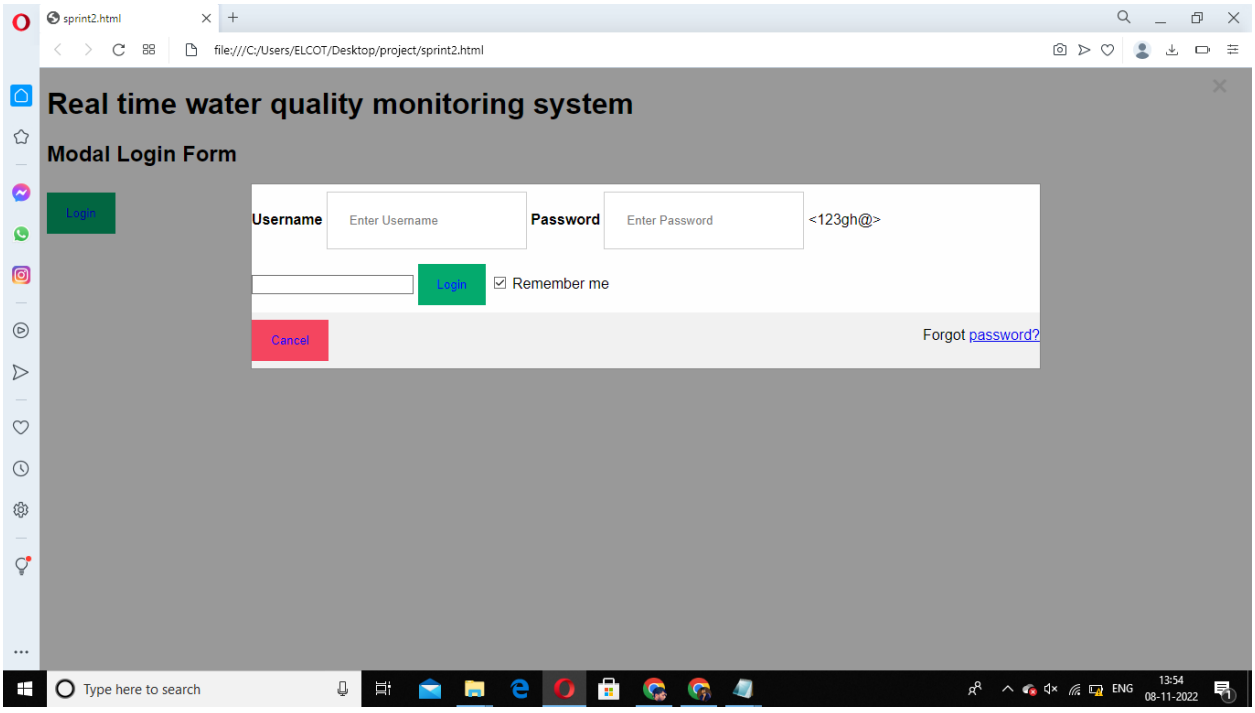
```
 }
 }
 }
for(int i=2;i<8;i++)
avgValue+=buf[i];
float pHVol=(float)avgValue*5.0/1024/6;
float phValue = -5.70 * pHVol + 21.34;
Serial.println(phValue);
Serial.print("pH");


 Serial.print(" C ");
 Serial.print(Celcius);

 Serial.print(voltage);
 Serial.print("V");
 delay(10000);
}
```

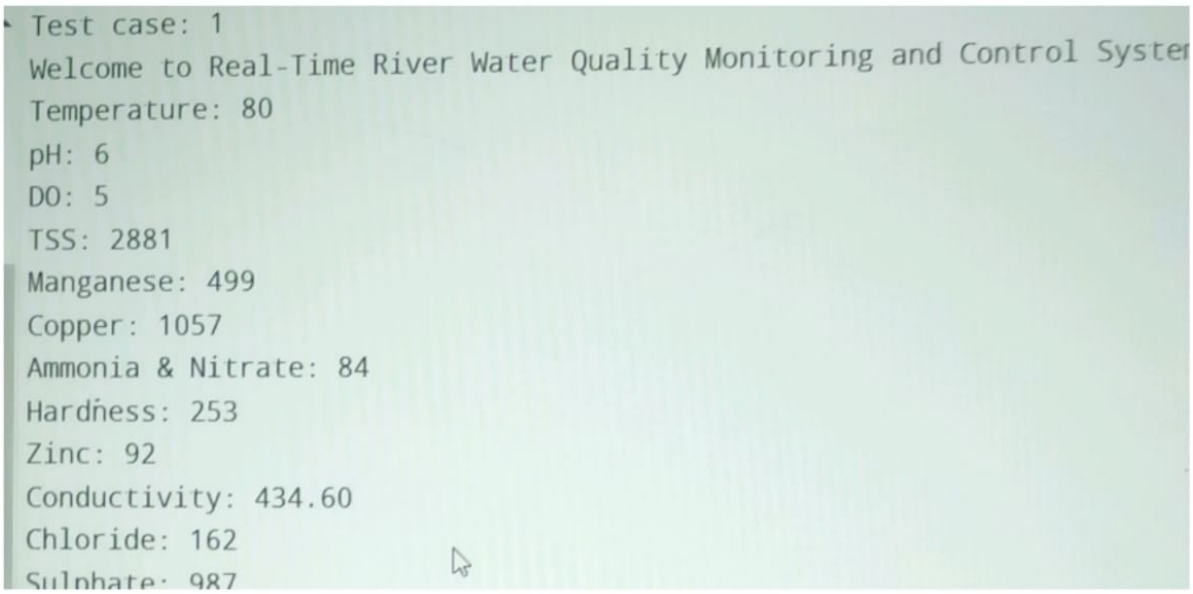
OUTPUTS:

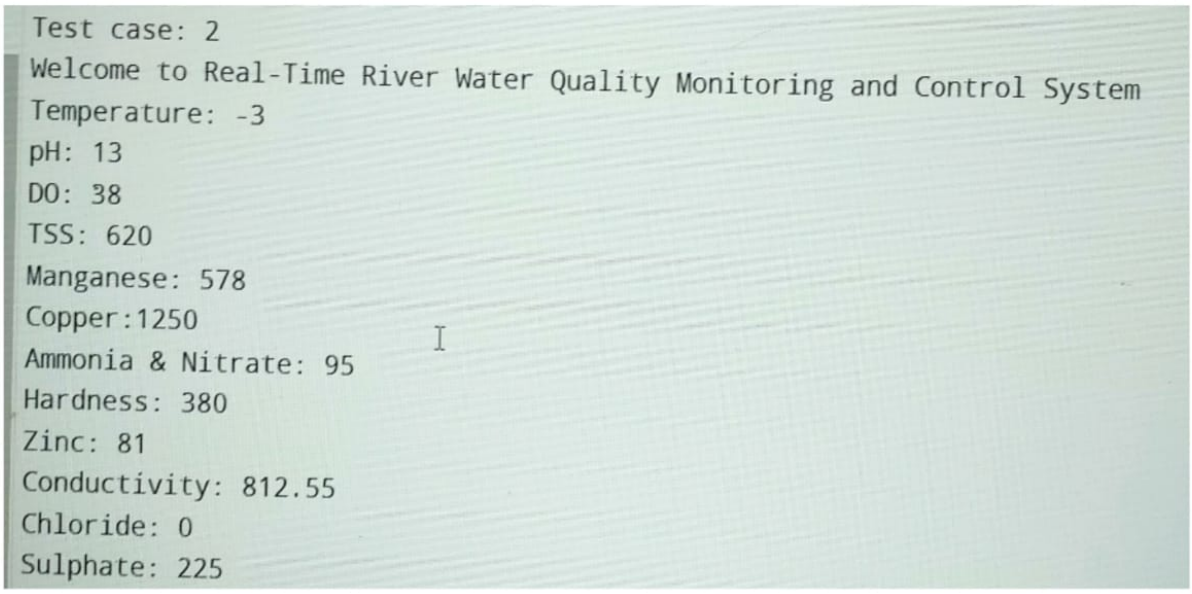

COM3
Send
Sensor Output (NTU):
0.83
Sensor Output (NTU):
0.19
Sensor Output (NTU):
0.16
Sensor Output (NTU):
0.68
Sensor Output (NTU):
1.16
Autoscroll
Show timestamp
Newline
9600 baud
Clear output

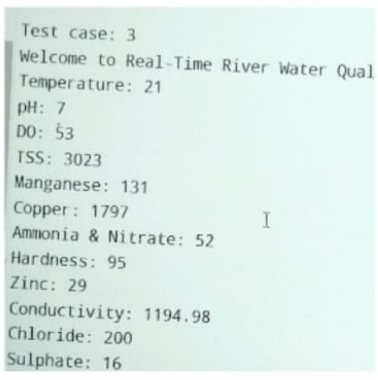

COM3
Send
Temperature is = 35
Temperature is = 35
Temperature is = 35
Temperature is = 35
Temperature is = 34
Temperature is = 34
Temperature is = 35
Temperature is = 35
Temperature is = 35
Temperature is = 35
Temperature is = 35
Temperature is = 34
Temperature is = 35
Temperature is = 35
Temperature is = 35
Autoscroll
Show timestamp
Newline
9600 baud
Clear output

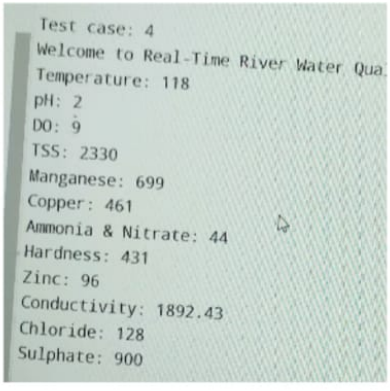

COM3
Send
Sensor Output (NTU):
2.66
Sensor Output (NTU):
2.59
Sensor Output (NTU):
2.64
Sensor Output (NTU):
2.63
Sensor Output (NTU):
2.64
Autoscroll
Show timestamp
Newline
9600 baud
Clear output

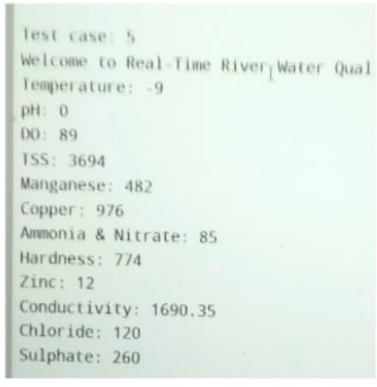

sprint2.html
file:///C:/Users/ELCOT/Desktop/project/sprint2.html
Real time water quality monitoring system
Modal Login Form
Login
Username
Enter Username
Password
Enter Password
<123gh@>
Login
Remember me
Cancel
Forgot password?
Type here to search
ENG
13:54
08-11-2022


Test case: 1
Welcome to Real-Time River Water Quality Monitoring and Control Syster
Temperature: 80
pH: 6
DO: 5
TSS: 2881
Manganese: 499
Copper: 1057
Ammonia & Nitrate: 84
Hardness: 253
Zinc: 92
Conductivity: 434.60
Chloride: 162
Sulphate: 987


Test case: 2
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -3
pH: 13
DO: 38
TSS: 620
Manganese: 578
Copper:1250
Ammonia & Nitrate: 95
Hardness: 380
Zinc: 81
Conductivity: 812.55
Chloride: 0
Sulphate: 225


Test case: 3
Welcome to Real-Time River Water Qual
Temperature: 21
pH: 7
DO: 53
TSS: 3023
Manganese: 131
Copper: 1797
Ammonia & Nitrate: 52
Hardness: 95
Zinc: 29
Conductivity: 1194.98
Chloride: 200
Sulphate: 16


Test case: 4
Welcome to Real-Time River Water Qua
Temperature: 118
pH: 2
DO: 9
TSS: 2330
Manganese: 699
Copper: 461
Ammonia & Nitrate: 44
Hardness: 431
Zinc: 96
Conductivity: 1892.43
Chloride: 128
Sulphate: 900


Test case: 5
Welcome to Real-Time River Water Qual
Temperature: -9
pH: 0
DO: 89
TSS: 3694
Manganese: 482
Copper: 976
Ammonia & Nitrate: 85
Hardness: 774
Zinc: 12
Conductivity: 1690.35
Chloride: 120
Sulphate: 260

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 IOT Based Real-Time River Water Quality Monitoring System (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.

REFERENCES:

[1]. Nikhil Kedia, Water Quality Monitoring for Rural Areas: An Economical Sensor Cloud Project, in 1st International Conference on Next Generation Computing Technologies (NGCT-2015), Dehradun, India, 4-5 September 2015. 978-1-4673-6809-4/15

[2]. Jayti Bhatt, Jignesh Patoliya, Iot Based Water Quality Monitoring System, IRFIC, 21feb,2016.

[3]. Michal lom, ondrej priby miroslav svitek, Internet 4.0 as a part of smart cities, 978-1-5090-1116-2/16.

[4]. Zhanwei Sun, Chi Harold Liu, Chatschik Bisdikia, Joel W. Branch and Bo Yang, 2012 9th, Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks.

[5]. (SECON), 978-1-4673-1905-8/12.

IBM DEMO VIDEO LINK:

https://drive.google.com/file/d/1XXQVh5tHqNm4jVghphH_YSI463pKQ3MC/view?usp=drivesdk