CHRIST THE KING
ENGINEERING COLLEGE
Karamadai, Coimbatore – 641 104
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

Date	19-11-2022
Team ID	PNT2022TMID42431
Project Name	Project - Real time River water quality monitoring and control system
Maximum Marks	4 MARKS

PROJECT REPORT

1. INTRODUCTION

1.1PROJECT OVERVIEW:

River water is used as drinking water is a very precious commodity for all human beings .The system consists of several sensors which are used for measuring physical and chemical parameters of water. The parameters such as temperature, pH, and dissolved oxygen of the water can be measured. Using River water which this system a person can detect pollutants from a water body from anywhere in the world.

1.2 PURPOSE:

The main aim is to develop a system for continuous monitoring of river water quality at remote places using wireless sensor networks with low power consumption, low-cost and high detection accuracy. P H, conductivity, turbidity level, etc. are the limits that are analyzed to improve the water quality.

2. LITERATURE SURVEY:

2.1 PROBLEM:

- It is difficult to collect the water samples from all the area of the water body.
- It is difficult to proceed.
- Manual practices are time consuming and there is not enough facilitating technologies.

Frustrated and answerable for people.

2.2 REFERENCE:

1.https://doi.org/10.1186/s40713-017-0005-y

2.https://www.researchgate.net/publication/318695965_GPRS_based_river_water_level_monitoring_and_measuring_system

3. http://cgwb.gov.in/.

4.https://www.researchgate.net/publication/305781494_SCADA_system_for_real-time_measuring_and_evaluation_of_river_water_quality

2.3 PROBLEM STATEMENT DEFENITION:

Due to the fast growing urbanization supply of safe river water is a challenge for the every city authority. Water can be polluted any time. So the water we reserved in the water tank at our roof top or basement in our society or apartment may not be safe. Still in India most of the people use simple water purifier that is not enough to get surety of pure water. Sometimes the water has dangerous particles or chemical mixed and general purpose water purifier cannot purify that. And it's impossible to check the quality of water manually in every time. So an automatic real-time monitoring system is required to monitor the health of the water reserved in our water tank of the society or apartment. So it can warn us automatically if there is any problem with the reserved water. And we can check the quality of the water anytime and from anywhere. By keeping this mind we designed this system especially for residential areas.

3. IDEATION & PROPOSED SOLUTION

3.1EMPATHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a

user’s behaviours and attitudes.

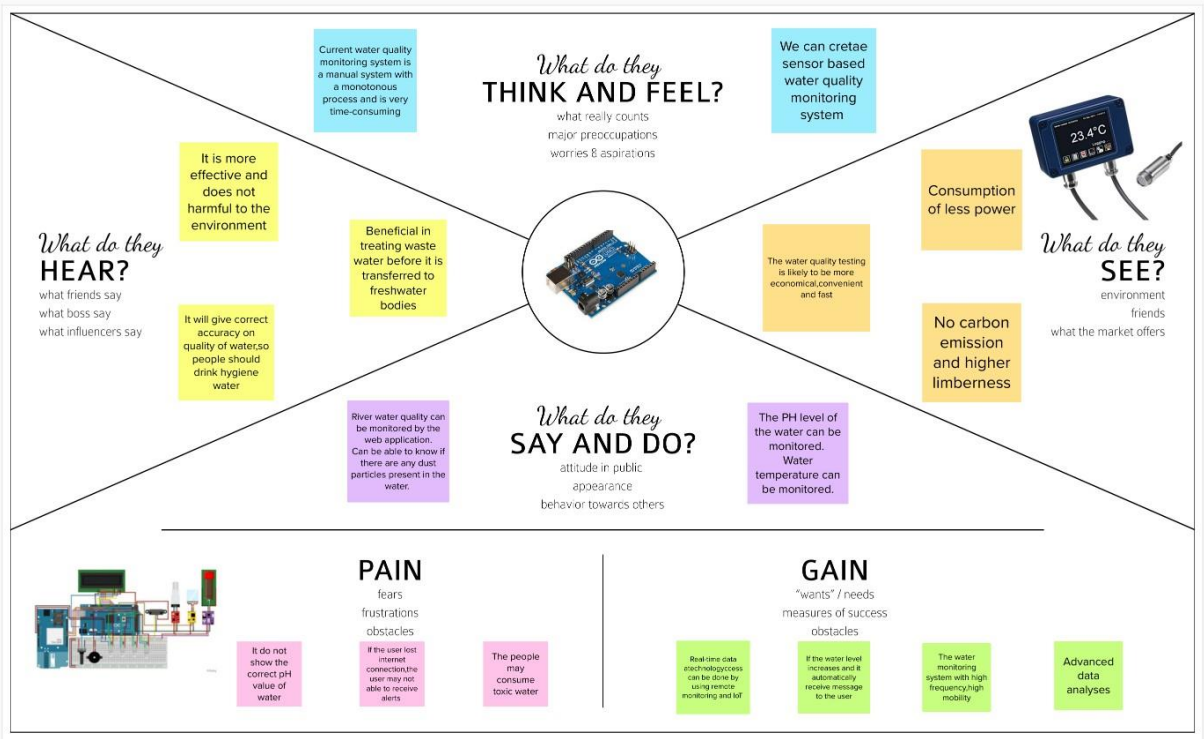
It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

3.2 IDEATION & BRAINSTROMING:

Brainstorm & Idea Prioritization Template:

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

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

Share template feedback

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

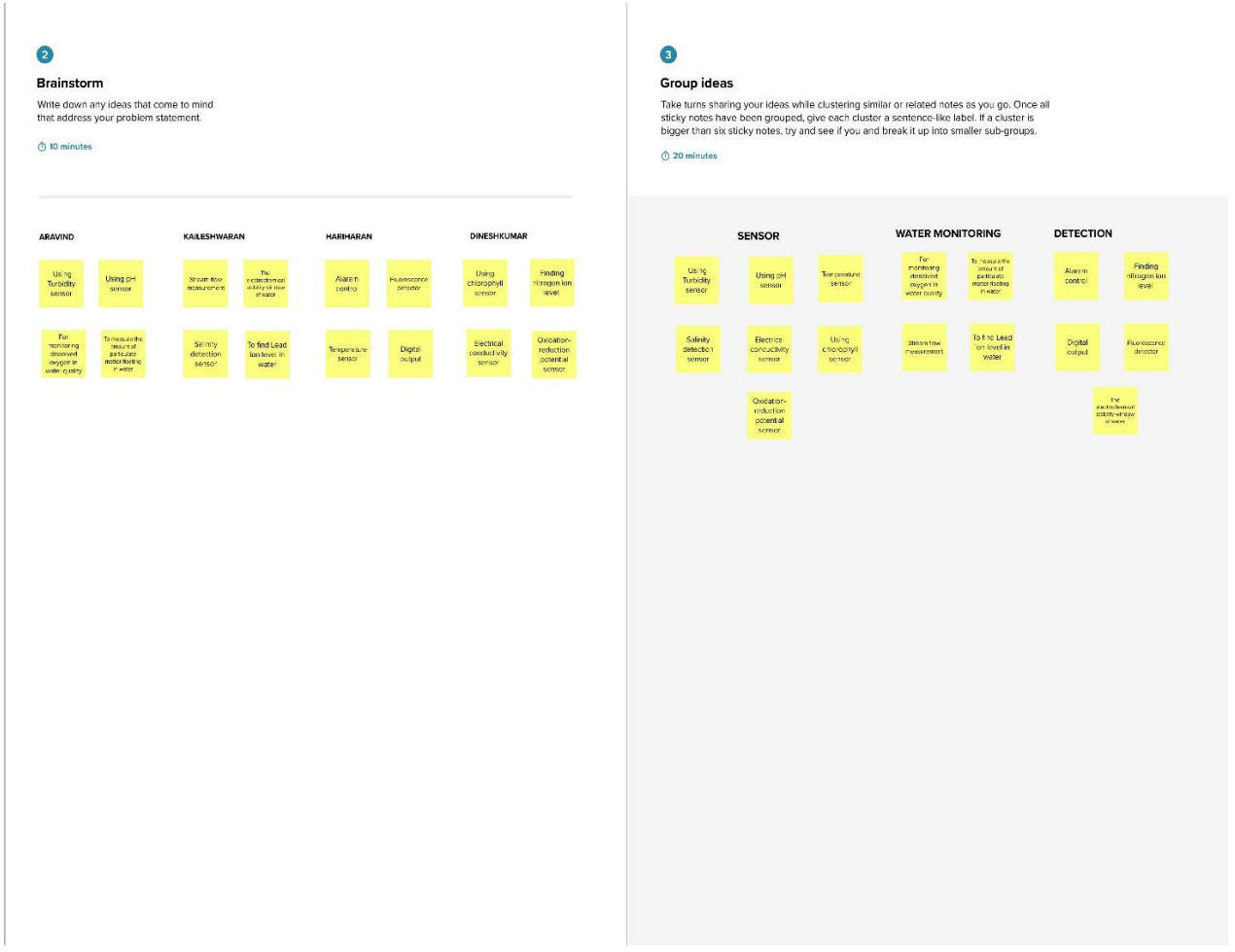
How might we [your problem statement]?

Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Defer judgment.
- Go for volume.
- Encourage wild ideas.
- Listen to others.
- If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization

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

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

3.3 PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	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 very easy.

3.4 PROBLEM SOLUTION FIT:


Define CS, fit into CC
1. CUSTOMER SEGMENT(S)
CS
Who is your customer?
According to our problem statement, people living in rural areas and so, who uses river water.
6. CUSTOMER CONSTRAINTS
CC
What constraints prevent your customers from taking action or limit their choices of solutions?
Only one system is used for specific area and so people may find it hard to recover if any fault occurs, as we used sensors to detect temperature and pH.
5. AVAILABLE SOLUTIONS
AS
Which solutions are available to the customers when they face the problem need to get the job done? What have they tried inthe past? What pros & cons do these solutions have?
Eventhough the individual notifications to each people could not be sent, the system will still notify the corporation and they can further notify the people.
Explore AS, differentiate
Focus on J&P, tap into
2. JOBS-TO-BE-DONE / PROBLEMS
J&P
Which jobs-to-be-done (or problems) do you addressfor your customers?
The river water quality monitoring system checks the temperature and pH of the water periodically and notifies the public when the quality of the water varies.
9. PROBLEM ROOT CAUSE
RC
What is the real reason that this problem exists? What is the back story behind the need to do this job?
As we know sensors are bit costly and our system needs more than one sensors to work. The sensors are used periodically to check the quality of the water and might need to be replaced frequently.
7. BEHAVIOUR
BE
What does your customer do to address the problem andget the job done?
The customer could use the user guide provided to overcome the problem or else they can report and contact the corporation. They will take care of the problem.
Focus on J&P, tap int C
Identify strong TR & EM
3. TRIGGERS
TR
What triggers customers to act? i.e. seeing their neighbour installing
For Example : If certain area people start using this quality monitoring system and so they are staying healthy without any water borne diseases, it will trigger the other area people start using it.
4. EMOTIONS: BEFORE / AFTER
EM
How do customers feel when they face a problem or ajob and afterwards?
The customers might feel hard first, we will guide them with a user guide and they will find it easy to use.
10. YOUR SOLUTION
SL
If you are working on an existing business, write down your current solution first,fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill inthe canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.
Our Solution is to check the quality of the river water periodically using two sensors. The parameters like temperature and pH of the river water is monitored and alerts when any changes in the parameters occus.
8. CHANNELS of BEHAVIOUR
CH
8.1 ONLINE
What kind of actions do customers take online?
If it is in online mode, they can use the helpline number to contact the authorities.
8.2 OFFLINE
What kind of actions do customers take offline?
If it is in offline mode, the customers can directly reach the corporation office and report the problem.
Extract online & offline CH of BE

4. REQUIREMENT ANALYSIS:

4.1 FUNCTIONAL REQUIREMENT:

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

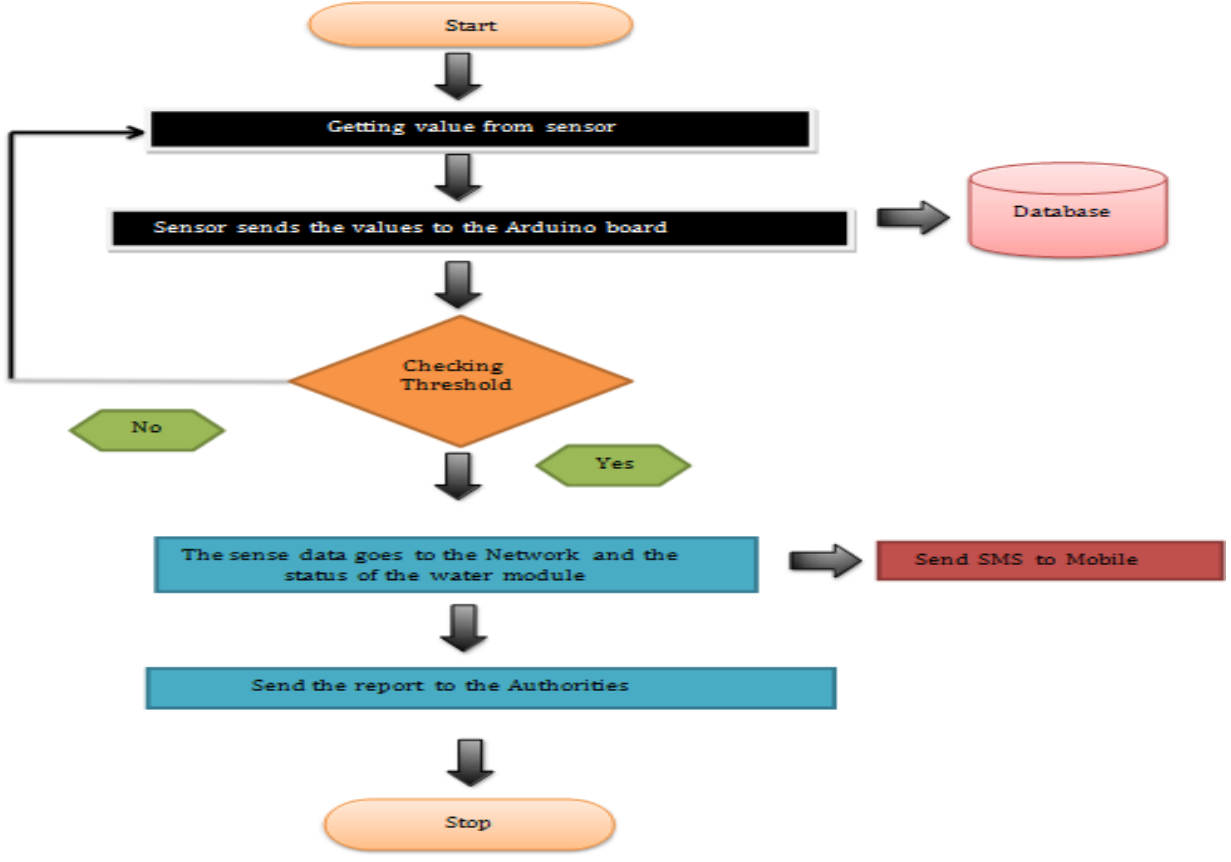
4.2 NONFUNCTIONAL REQUIREMENT:

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Load time for user interface screens shall not be more than 2 seconds.
NFR-2	**Security**	User account is password protected Account creation done only after email verification
NFR-3	**Reliability**	Users can access their account 98% of the time with out failure

NFR-4	**Performance**	Load time for user interface screens shall not be more than 2seconds. Login info verified within 10 seconds.
NFR-5	**Availability**	Maximum down time will be about 4hours
NFR-6	**Scalability**	System can handle about 1000 users at any given time

5 .PROJECT DESIGN

5.1 DATA FLOW DIAGRAM:



5.2 SOLUTION & TECHNICAL ARCHITECTURE:

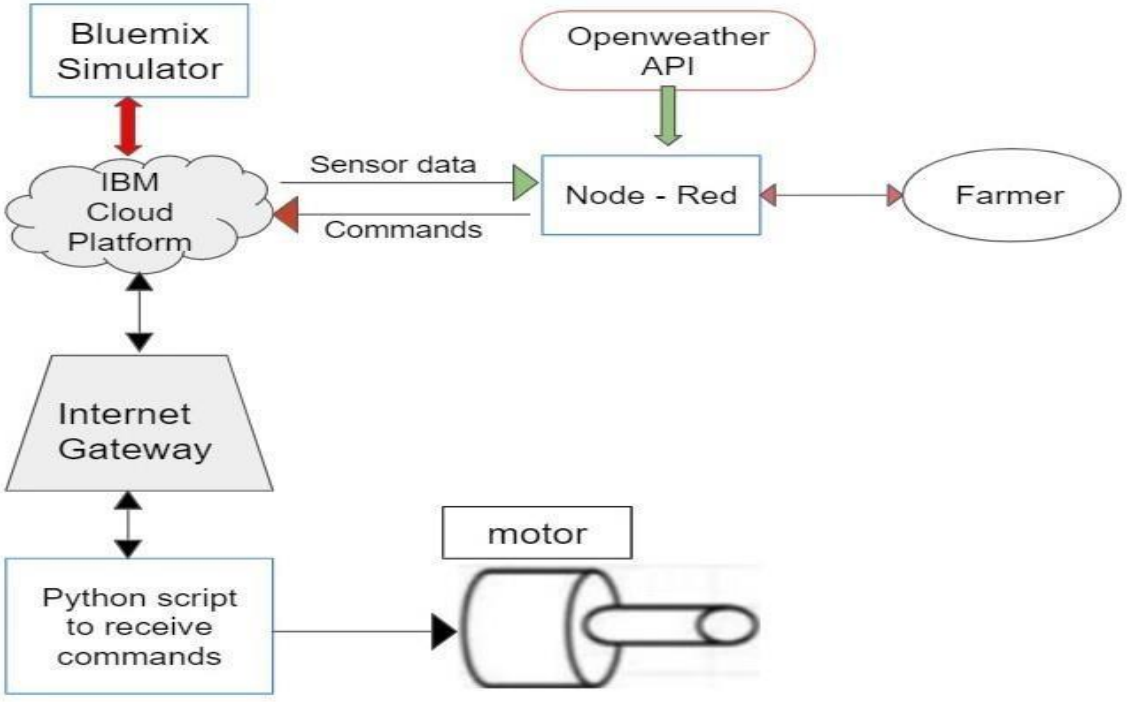


Table-1: Components &Technologies:

S.No	Component	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	BM WATSON Assistant
5.	Database	Data Type, Configurations 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 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 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
	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 PLANING & SCHEDULING:

6.1 SPRINT PLANING & 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.	3	High	Aravind, Kaileshwaran
Sprint-1	Registration via Facebook	USN-2	As a user, I can register for the application through Facebook	3	High	Aravind, Kaileshwaran
Sprint-1	registration via Gmail	USN-3	As a user, I can register for the application through Gmail	2	Medium	Aravind, Kaileshwaran
Sprint-2	Confirmation	USN-4	As a user I will receive confirmation email once I have registered for the application	3	High	Kaileshwaran, Dineshkumar
Sprint-2	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	Kaileshwaran, Dineshkumar
Sprint-2	IBM cloud service	USN-6	Get access to IBM cloud services	3	High	Kaileshwaran, Dineshkumar

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Create IBM Watson and device settings	USN-7	To create the IBM Watson IOT platform and Intergrate the microcontroller with it to send sensed data to cloud	3	High	Hariharan, Aravind
Sprint-3	Create node red service	USN-8	To create a node red service to integrate theIBM Watson along with Web UI	1	Low	Hariharan, Aravind
Sprint-3	Create Web UI	USN-9	To create Web UI to access the data from cloudAnd display all parameters	3	High	Hariharan, Aravind
Sprint-3	To develop a python code	USN-10	Create python code to sense the physical quantity and store data	2	Medium	Hariharan, Aravind
Sprint-4	Publish data to cloud	USN-11	Publish data that is sensed by the microcontroller to the cloud	3	High	Dineshkumar, Hariharan
Sprint-4	Fast SMS service	USN-12	Use fast SMS to send alert message once the parameters like ph , turbidity and temperature goes beyond the threshold	2	Medium	Dineshkumar, Hariharan
Sprint-4	Testing	USN-13	Testing of project and final deliverables	3	High	Dineshkumar, Hariharan

6.2. SPRINT DELIVERY SCHEDULE:

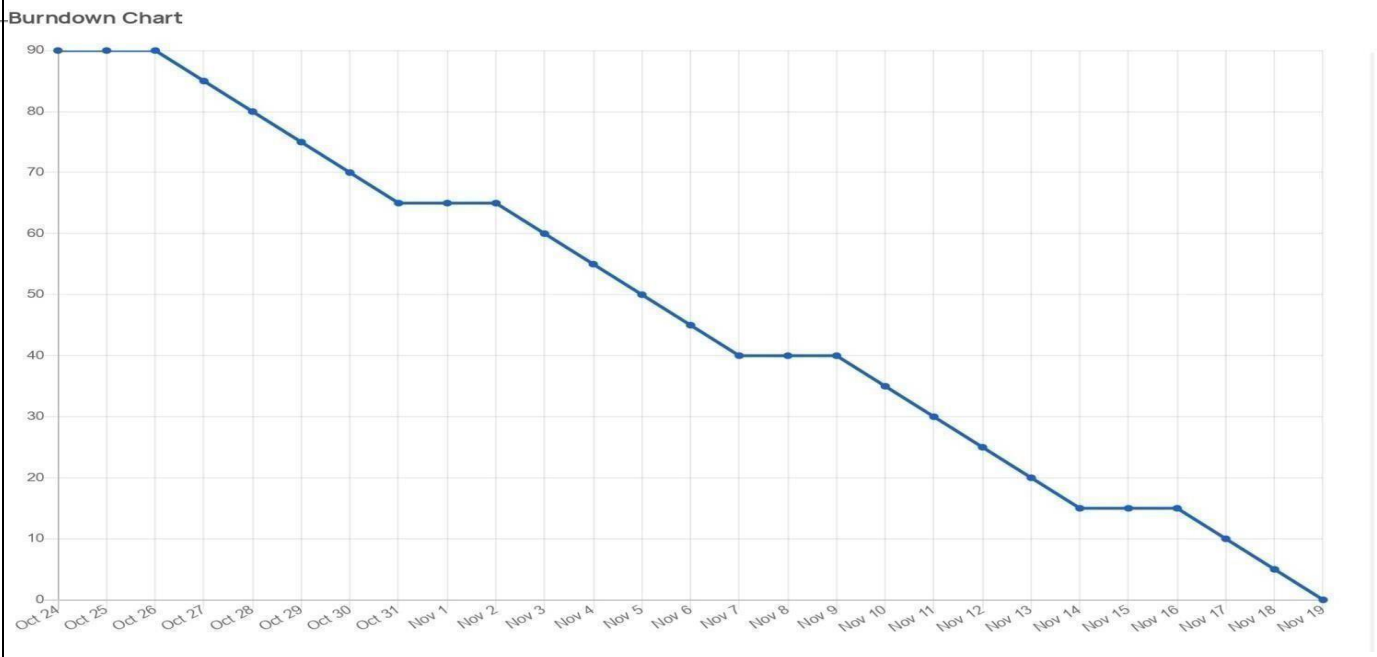
Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on** **Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	20	6 Days	24 Oct 2022	30 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	5 Nov 2022	40	06 Nov 2022
Sprint-3	20	6 Days	5 Nov 2022	12 Nov 2022	60	12 Nov 2022
Sprint-4	20	6 Days	12 Nov 2022	19 Nov 2022	80	19 Nov 2022

Velocity:

Imagine we have a10-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)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.3 REPORT FROM JIRA:



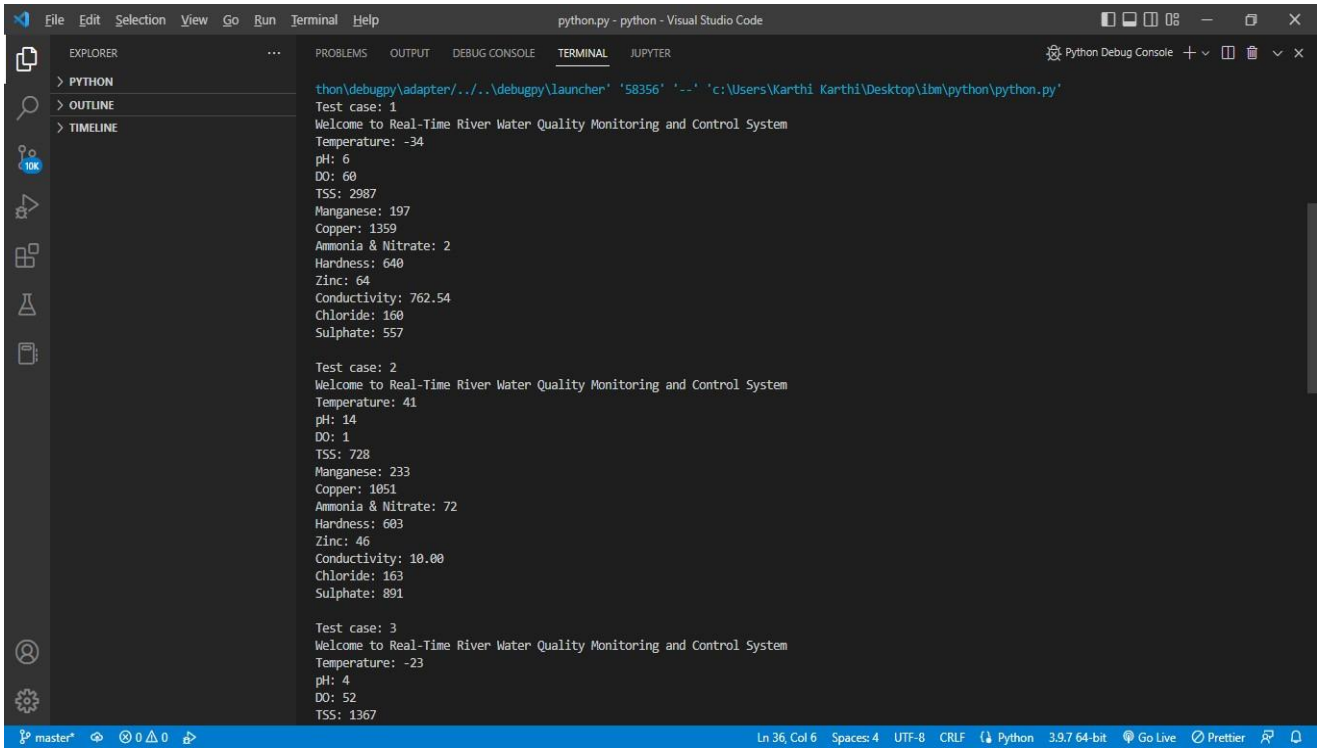
7. CODING & SOLUTIONING:

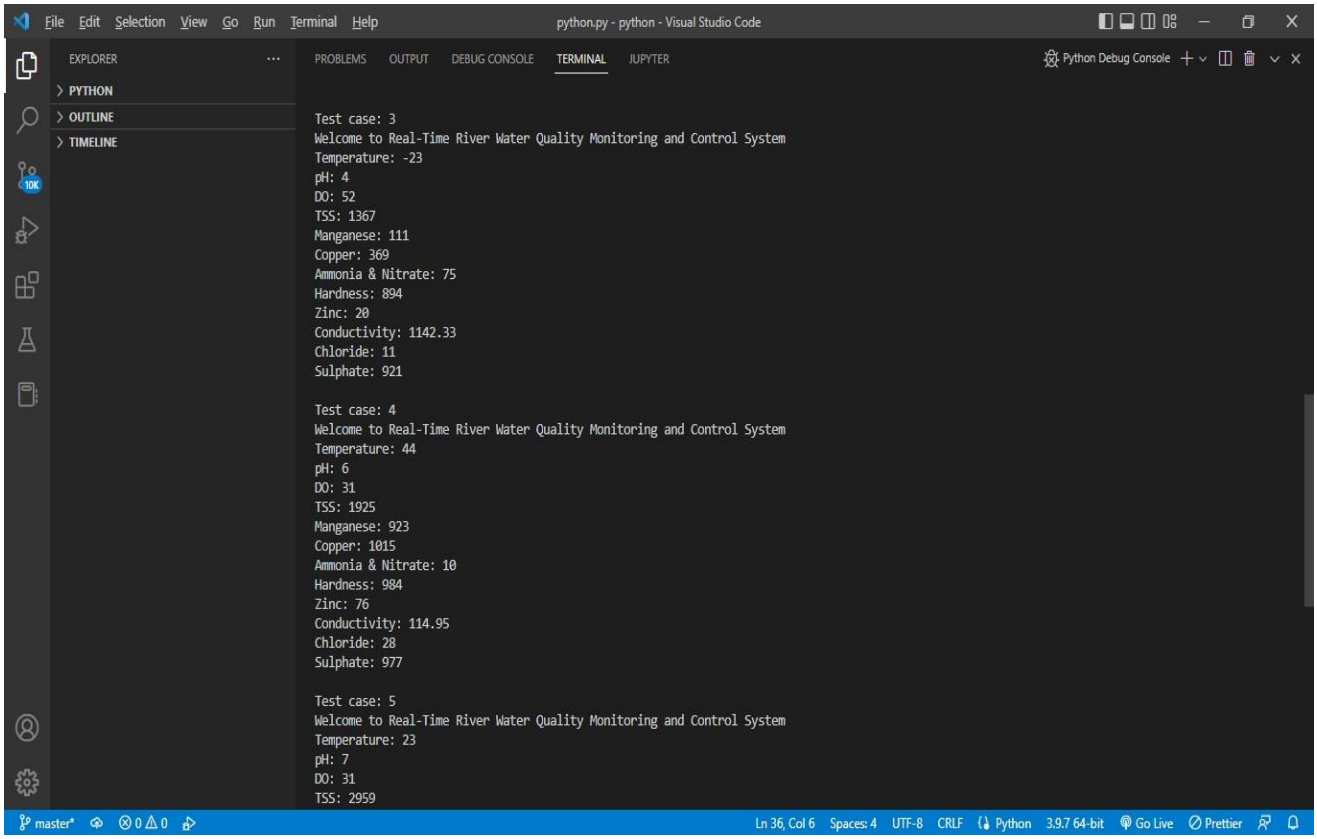
7.1 FEATURE 1:

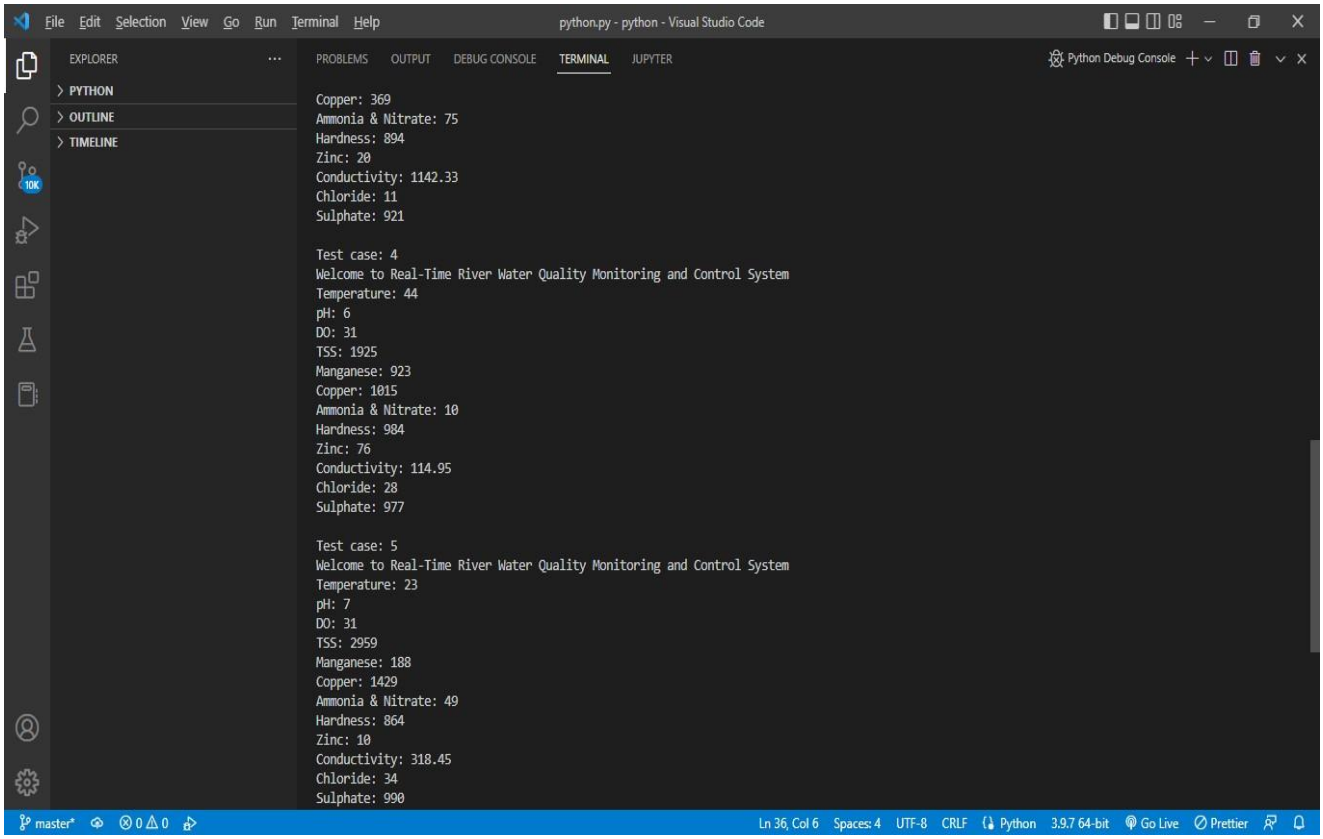
```
#importing Random function to generate the value
import random as rand


in range(5): print("Test
case:",i+1)
        print("Welcome to Real-Time River Water Quality Monitoring and ControlSystem")
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 valuesprint(
              "Temperature:", temperature,
              "\npH:", pH,
```

```
        "\nDO:", DO,
        "\nTSS:", TSS,
        "\nManganese:", Manganese,
        "\nCopper:", Copper,
        "\nAmmonia & Nitrate:",ammonia_Nitrate,
        "\nHardness:",Hardness,
        "\nZinc:", Zinc, "\nConductivity:",
    )
```

OUTPUT:




File Edit Selection View Go Run Terminal Help
python.py - python - Visual Studio Code
EXPLORER
PYTHON
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console
Test case: 3
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -23
pH: 4
DO: 52
TSS: 1367
Manganese: 111
Copper: 369
Ammonia & Nitrate: 75
Hardness: 894
Zinc: 20
Conductivity: 1142.33
Chloride: 11
Sulphate: 921
Test case: 4
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 44
pH: 6
DO: 31
TSS: 1925
Manganese: 923
Copper: 1015
Ammonia & Nitrate: 10
Hardness: 984
Zinc: 76
Conductivity: 114.95
Chloride: 28
Sulphate: 977
Test case: 5
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 23
pH: 7
DO: 31
TSS: 2959
master*
Ln 36, Col 6 Spaces: 4 UTF-8 CRLF Python 3.9.7 64-bit Go Live Prettier


File Edit Selection View Go Run Terminal Help
python.py - python - Visual Studio Code
EXPLORER
PYTHON
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console
Copper: 369
Ammonia & Nitrate: 75
Hardness: 894
Zinc: 20
Conductivity: 1142.33
Chloride: 11
Sulphate: 921
Test case: 4
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 44
pH: 6
DO: 31
TSS: 1925
Manganese: 923
Copper: 1015
Ammonia & Nitrate: 10
Hardness: 984
Zinc: 76
Conductivity: 114.95
Chloride: 28
Sulphate: 977
Test case: 5
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 23
pH: 7
DO: 31
TSS: 2959
Manganese: 188
Copper: 1429
Ammonia & Nitrate: 49
Hardness: 864
Zinc: 10
Conductivity: 318.45
Chloride: 34
Sulphate: 990
master*
Ln 36, Col 6 Spaces: 4 UTF-8 CRLF Python 3.9.7 64-bit Go Live Prettier

7.2 FEATURE 2 :

```
 <!DOCTYPE html>

<!-- Created By CodingLab - www.codinglabweb.com -->
<html lang="en" dir="ltr">
   <head>
<meta charset="UTF-8">
<!---<title> Responsive Registration Form | CodingLab </title>--->
<link rel="stylesheet" href="style.css">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
<body>
   <div class="container">
<div class="title">Registration</div>
<div class="content">
          <form action="#">
            <div class="user-details">
              <div class="input-box">
                 <span class="details">Full Name</span>
```

```
            <divclass="input-box">
               <spanclass="details">Username</span>
               <inputtype="text"placeholder="Enteryourusername"required>
            </div>
            <divclass="input-box">
               <spanclass="details">Email</span>
               <inputtype="text"placeholder="Enteryouremail"required>
            </div>
            <divclass="input-box">
               <spanclass="details">PhoneNumber</span>
               <inputtype="text"placeholder="Enteryournumber"required>
            </div>
            <divclass="input-box">
               <spanclass="details">Password</span>
               <inputtype="text"placeholder="Enteryourpassword"required>
            </div>
            <divclass="input-box">
               <spanclass="details">Location</span>
               <inputtype="text"placeholder="Confirmyourpassword"required>
            </div>
         </div>
         <divclass="gender-details">
            <inputtype="radio" name="gender"id="dot-1">
            <inputtype="radio" name="gender"id="dot-2">
            <inputtype="radio" name="gender"id="dot-3">
            <spanclass="gender-title">Gender</span>
            <divclass="category">
               <labelfor="dot-1">
               <spanclass="dotone"></span>
               <spanclass="gender">Male</span>
            </label>
            <labelfor="dot-2">
               <spanclass="dottwo"></span>
               <spanclass="gender">Female</span>
            </label>
            <labelfor="dot-3">
               <spanciass="dotthree"></span>
               <spanclass="gender">Prefernottosay</span>
```

```
    </div>
  </div>

</body>
</html>
```

CSS CODE:

```
 @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600;70
0&display=swap');

*{
   margin: 0;
   padding: 0;
   box-sizing: border-box;
   font-family: 'Poppins',sans-serif;
}
body{
   height: 100vh;
   display: flex;
   justify-content: center;align-
   items: center; padding:
   10px;
   background: linear-gradient(135deg, #71b7e6, #9b59b6);
}
.container{
   max-width: 700px; width:
   100%; background-color:
   #fff;padding: 25px 30px;
   border-radius: 5px;
   box-shadow: 0 5px 10px rgba(0,0,0,0.15);
}
.container .title{ font-
```

```
    width: 30px;border-
    radius:5px;
    background:linear-gradient(135deg,#71b7e6,#9b59b6);
}
.contentform.user-
    details{display:flex;
    flex-wrap:wrap;
    justify-content: space-
    between;margin:20px012px0;
}
form .user-details .input-box{margin-
    bottom:15px;
    width:calc(100%/2-20px);
}
form.input-
    boxspan.details{display:block;
    font-weight:
    500;margin-bottom:5px;
}
.user-details .input-box
    input{height:45px;
    width: 100%;outline:
    none;font-size:
    16px;border-
    radius:5px;padding-
    left:15px;
    border: 1px solid #ccc;border-
    bottom-width:
    2px;transition:all0.3sease;
}
.user-details.input-boxinput:focus,
.user-details .input-box input:valid{border-
    color:#9b59b6;
}
  form.gender-details.gender-title{font-
    size:20px;
    font-weight:500;
  }
  form .category{display:
      flex;width:
      80%;margin:14px0;
      justify-content:space-between;
  }
  form.categorylabel{ ---------- Section Break (Next Page) ----------
```

```
    radius:5px;padding-
    left:15px;
    border: 1px solid #ccc;border-
    bottom-width:
    2px;transition:all0.3sease;
}
.user-details.input-boxinput:focus,
.user-details .input-box input:valid{border-
    color:#9b59b6;
}
  form.gender-details.gender-title{font-
    size:20px;
    font-weight:500;
  }
  form .category{display:
      flex;width:
      80%;margin:14px0;
      justify-content:space-between;
  }
  form.categorylabel{
```

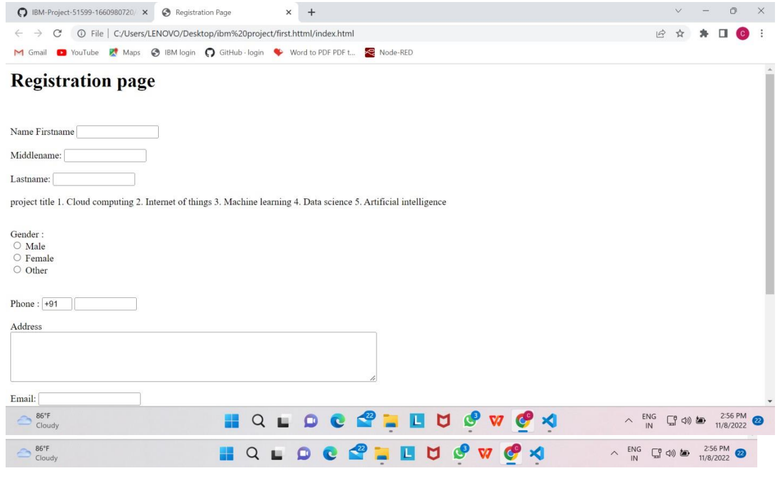
Section Break(Next Page)

```
    display: flex;align-
    items:center;cursor:poin
    ter;
}
form.categorylabel.dot{height:1
  8px;
  width: 18px;border-
  radius:50%;margin-
  right:10px;
  background:#d9d9d9;
  border: 5px solid
  transparent;transition:all0.3sease;
}
#dot-1:checked ~ .category label .one,#dot-
2:checked ~ .category label .two,#dot-
3:checked~.categorylabel.three{
    background:
    #9b59b6;border-
    color:#d9d9d9;
}
forminput[type="radio"]{display:
    none;
}
```

```
t:100%;
width: 100%;border-
radius:5px;border:
none;color:#fff;
font-size: 18px;font-
weight: 500;letter-
spacing:1px;cursor:point
er;
transition:all0.3sease;
background:linear-gradient(135deg,#71b7e6,#9b59b6);
```

```
    max-width:100%;
}
form .user-details .input-box{margin-
        bottom:15px;width:100%;
    }
    form
         .category{width
        :100%;
    }
    .content form .user-details{max-
        height: 300px;overflow-y:scroll;
    }
    .user-details::-webkit-scrollbar{width:5px;
    }
    }
    @media(max-width:459px){
    .container .content .category{flex-
        direction:column;
    }
}
```

OUTPUT:



8. TESTING:

8.1: TEST CASES

Date	3-Nov-22
Team ID	PNT2022TMID42431
Project Name	Project - Real time River water quality monitoring and control system
Maximum Marks	4 marks

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
Home Page_TC_01	Functional	Home Page	Used to take the user into register page		1.Click on Sign Up Button (If Not Registered)		Application should show below UI elements: a.email text box b.password text box c.register button d.back button	Working as expected	Pass				Beta Tester
Home Page_TC_02	Functional	Home Page	Used to take the user into Login page		1.Click on Sign In Button (If Registered)		Application should show below UI elements: a.email text box b.password text box c.login button d.back button	Working as expected	Pass	Steps are not clear to follow		BUG-1234	Beta Tester
Register Page_TC_O1	Functional	Register page	Used to register the user in the application	Username & Password	1.Enter Username & Password 2.Click on Register button	Username: rithick password:1234	Redirct to login page after register	Working as expected	Pass				Beta Tester
Register Page_TC_O2	Functional	Register page	Used to go back to Home Page		1.Click on Back button		Redirect to home page	Working as expected	Pass				Beta Tester
LoginPage_TC_O1	Functional	Login page	Used to Login into th Application	Username & Password	1.Enter Username & Password 2.Click on Login button	Username: dinesh password: 1234	Application should show 'Incorrect email or password ' validation message(If not Registered), and redirect to user page 1(If Registered)	Working as expected	Pass				Beta Tester
LoginPage_TC_O2	Functional	Login page	Used to go back to Home Page		1.Click on Back button		Redirect to home page	Working as expected	Pass				Beta Tester
User Page1_TC_01	Functional	User Page1	It shows the temperature and humidity value	Temperature value and Humidity value	1.Enter Username & Password 2.Click on Login button	1. Temperature- 20 Degree 2. Humidity - 40%	Application should show below UI elements: a.Temperature text box b.Humidity text box c.Add button d.view button e.back button	Working as expected	Pass				Beta Tester
User Page1_TC_01	Functional	User Page1	Used to go back to Home Page		1.Click on Logout button		Redirect to the Home Page	Working as expected	Pass				Beta Tester
User Page2_TC_01	Functional	User Page2	Used to go back to User Page1		1.Click on Back button		Redirect to the user page 1	Working as expected	Pass				Beta Tester

8.2 USER ACCEPTANCE TESTING:

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. 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	20	4	2	3	29
Duplicate	3	0	3	0	7
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	36	14	13	26	86

3. 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	7	0	0	7
Client Application	11	0	0	11
Security	0	0	0	0
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS:

9. 1. PERFORMANCE MATRICS:

S. NO	Parameter	Performance
1.	Response Time	0.2s (Average of 10 trials)
2.	Workload	500 users (Calculated based on Cloud Space)
3.	Revenue	Individual users and water industries.
4.	Efficiency	Simple and straightforward workflow, which makes theprocess efficient.
5.	Down Time	Almost no down time due to IBM Cloud enabled solution.

10. ADVANTAGES AND DISADVANTAGES:

10.1 Advantages:

- Water-quality monitoring is used to alert us to current, ongoing, and emerging problems.

- Water quality testing can provide valuable data on the condition of a particular
- body of water, and whether it may need special treatment before use.
- Examine factors such as the pH level, nutrient levels, amount of dissolved oxygen and temperature level are all useful in understanding the health of a water body.

10.2 Disadvantages:

- The system is less effective as sensors are installed very deep inside the water and their positions are fixed.
- The sensors are very expensive. More over their maintenance cost is also very high.

11. CONCLUSION:

This project developed for water quality maintenance is very beneficial for safeguarding public health and also adds to the clean environment. The automation of this water monitoring and control process remove the need of manual labour and thus saves time and money .The automation of the system makes the control and monitoring process more efficient and effective Real time monitoring on mobile phone which is possible through the interface PLC with Arduino and Bluetooth module allows remote controlling of the system.

12. 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.

13. APPENDIX:

13.1 SOURCE CODE:

UI CODE:

Code 1.

```
<html>
        <head>
        <title>
        Registration Page
        </title>
        </head>
        <body>
        <br>
        <br>
        <form>
        name
        <label> Firstname </label>
        <input type="text" name="firstname" size="15"/> <br> <br>
        <label> Middlename: </label>
        <input type="text" name="middlename" size="15"/> <br> <br>
```

```
<label> Lastname: </label>
<input type="text" name="lastname" size="15"/> <br> <br>
</select>
project title
1.<label> cloud computing </label>
2.<label> internet of things </label>
3.<label> machine learning </label>
4.<label> data science </label>
5.<label> artificial intelligence </label>
 <br>
<br>
<br>
<label>
Gender :
</label><br>
<input type="radio" name="male"/> Male <br>
 <input type="radio" name="female"/> Female <br>
<input type="radio" name="other"/> Other
<br>
<br>
 <br>
<label>
Phone :
</label>
<input type="text" name="country code"   value="+91"
size="2"/>
<input type="text" name="phone" size="10"/> <br> <br>
Address
<br>
```

```
<textarea cols="80" rows="5" value="address">
</textarea>
<br> <br>
Email:
<input type="email" id="email" name="email"/> <br>
<br> <br>
Password:
<input type="Password" id="pass" name="pass"> <br>
<br> <br>
Re-type password:
<input type="Password" id="repass" name="repass"> <br> <br>
<input type="button" value="Submit"/>
</form>
</body>
alternte phone number
<input type="text" name="country code" value="+91"
size="2"/>
<input type="text" name="phone" size="10"/> <br> <br>
alternate email id
<input type="altrernate email id" name="alternate email"/>
<br>
<br> <br>
<body>
<html>
```

Code 2.

```
<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>
```

3. Python Script:

```
#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"
                    )
```

4. Aurdino:

```
#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(analogInPi
                                        n);
                     delay(10);
                    }
                    for(int i=0;i<9;i++)
                    {
                     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);
}
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-7857-1658901010/tree/main/RIVER%20WATER%20QUALITY%20MONITORING

PROJECT DEMO LINK:

https://drive.google.com/file/d/13DO9UtiYdbl1XJy0jhokjabgyTLU-O_6/view?usp=drivesdk