Project Report Format

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. pH, conductivity, turbidity level, etc. are the limits that are analysed to improve the water quality.

2. SURVEY LITERATURE

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 References

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

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

http://cgwb.gov.in/.

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

2.3Problem Statement Definition

Due to the fast growing urbanization supply of safe river water is a challenge for the every city authority. Water can be any 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 the 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.

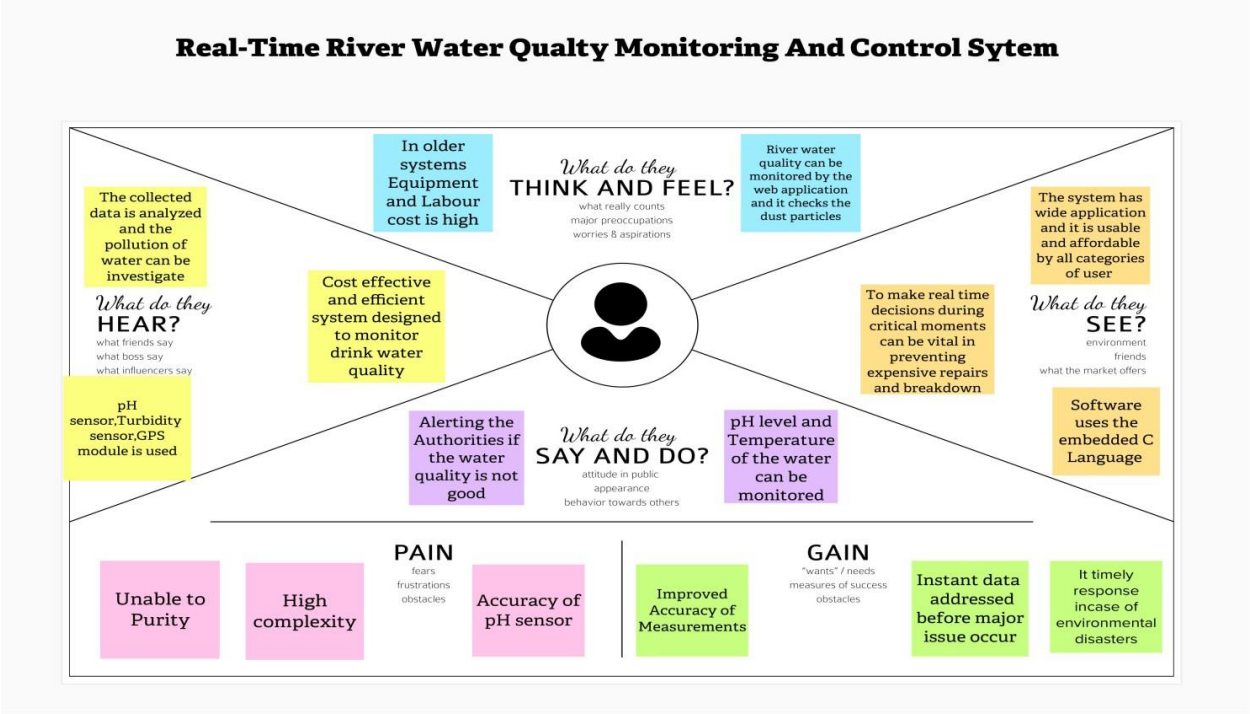
3. IDEATION & PROPOSED SOLUTION:

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.

Empathy Map:



3.2 Ideation & Brainstorming

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

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

Monitor and control the water quality

Key rules of brainstorming

To run an smooth and productive session

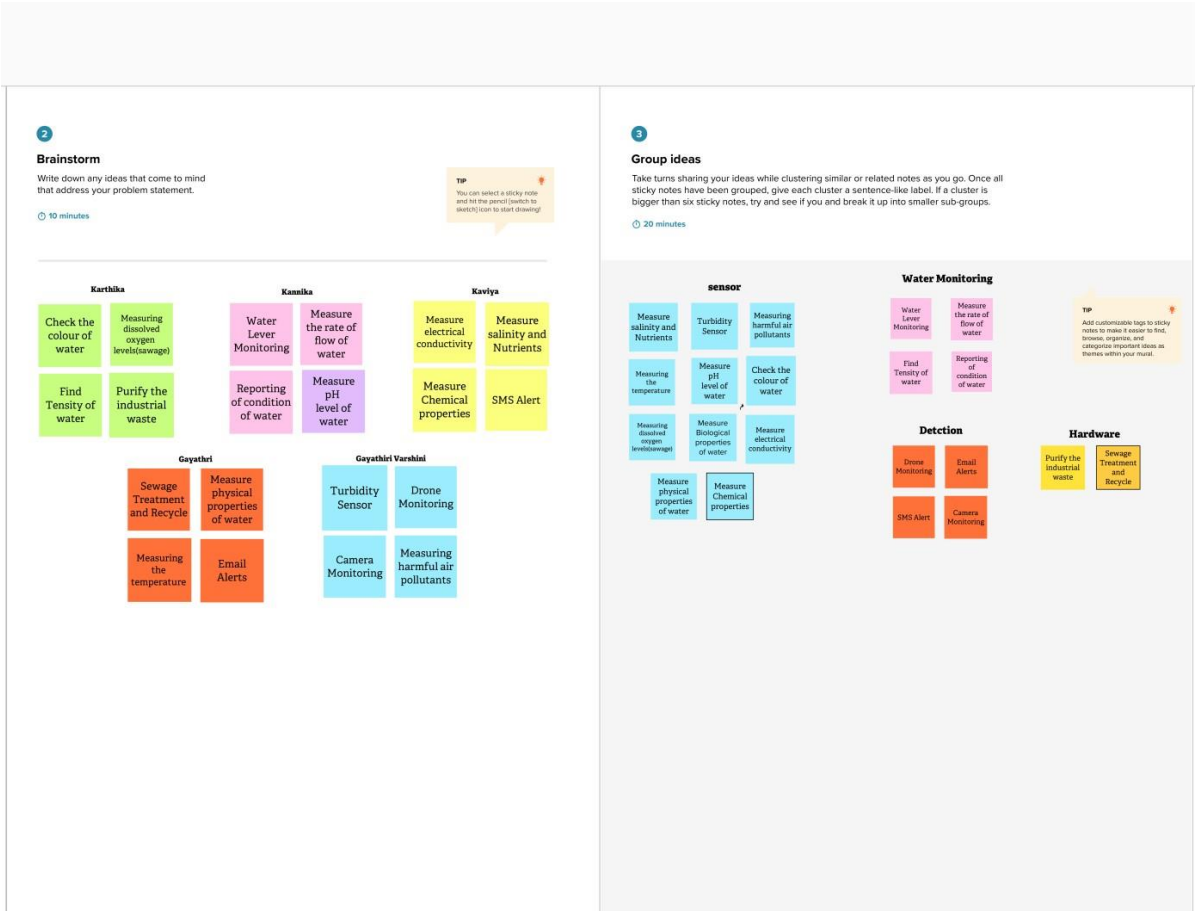
- Stay in topic.
- Defer judgment.
- Go for volume.
- Encourage wild ideas.
- Listen to others.
- If possible, be visual.

Need some inspiration?

See a finished version of this template to kickstart your work.

Open example →

Step-2: Brainstorm, Idea Listing and Grouping

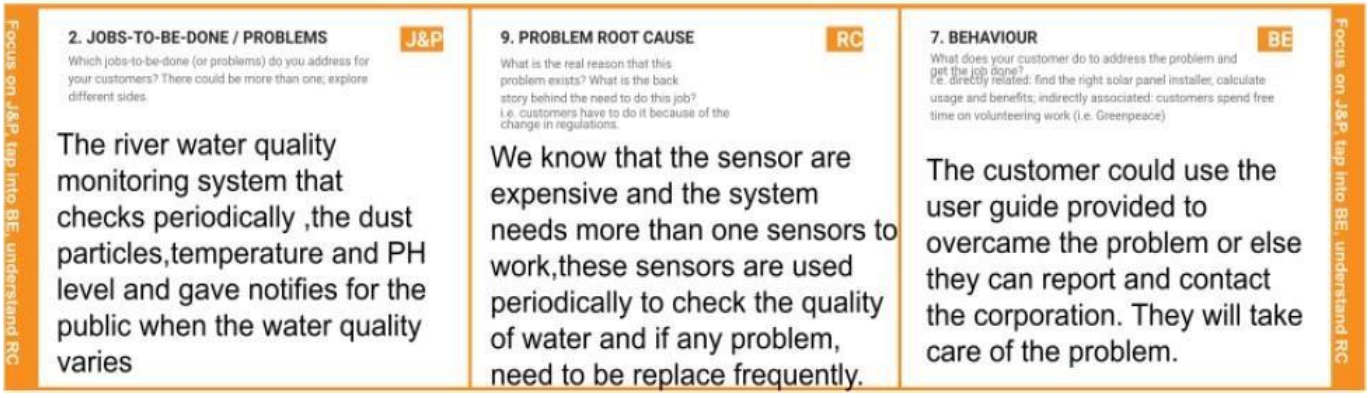


Step-3: Idea Prioritization

3.3 Proposed Solution:

S .No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Massive growth of algae called eutrophication leads to pollution(monitoring and controlling the quality of river water)
2.	Idea / Solution description	Detecting the dust particles , PH level of water, Dissolved oxygen and temperature to be monitored and altering the authorities if water quality is not good.
3.	Novelty / Uniqueness	River water quality can be monitored by web application. Quality parameter will track continuously with standard measurements.
4.	Social Impact / Customer Satisfaction	Localities will not get suffered by poor quality of water by alerting them when the water quality is not good.
5.	Business Model (Revenue Model)	Water quality monitoring system by aeron systems for industrial water treatment plant, river bodies, aqua forming ,digital loggers.
6.	Scalability of the Solution	Measuring of real time values and continuous monitoring helps in maintaining the quality of water.

3.4 Problem Solution Fit:



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

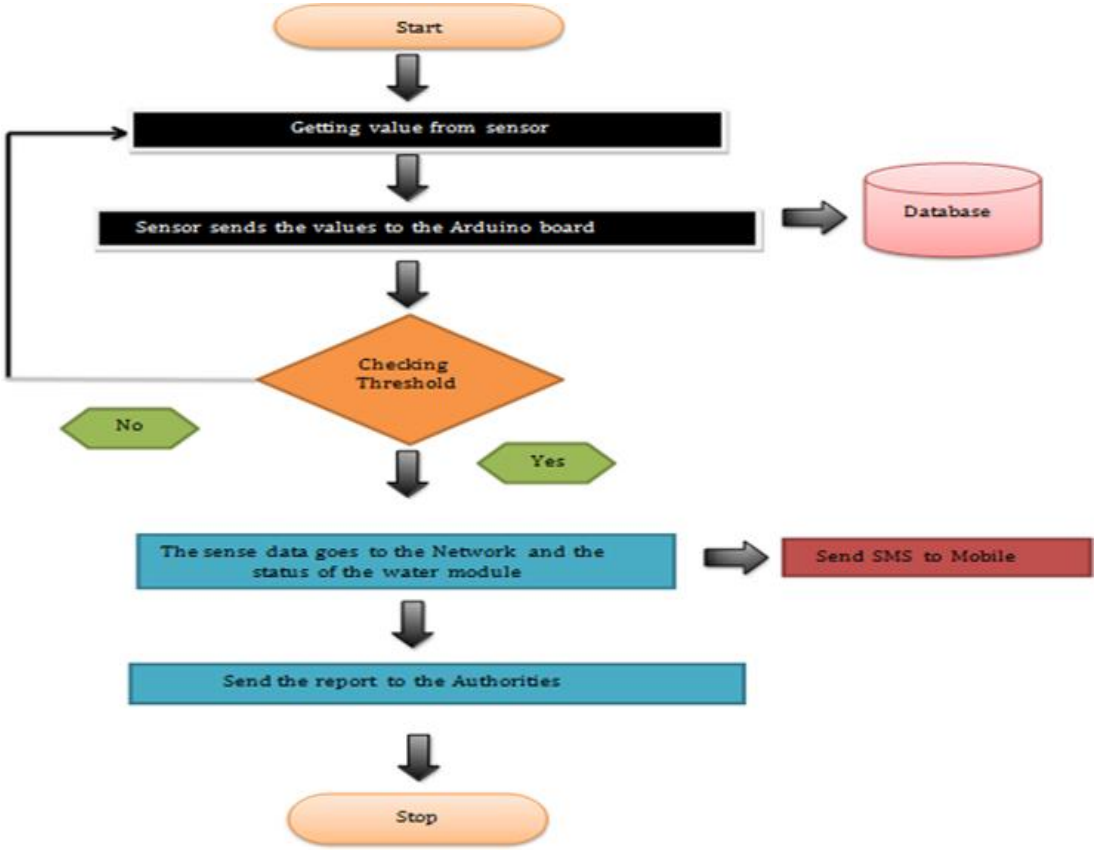
4.2 Non-functional Requirements:

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 without 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 and technical architecture:

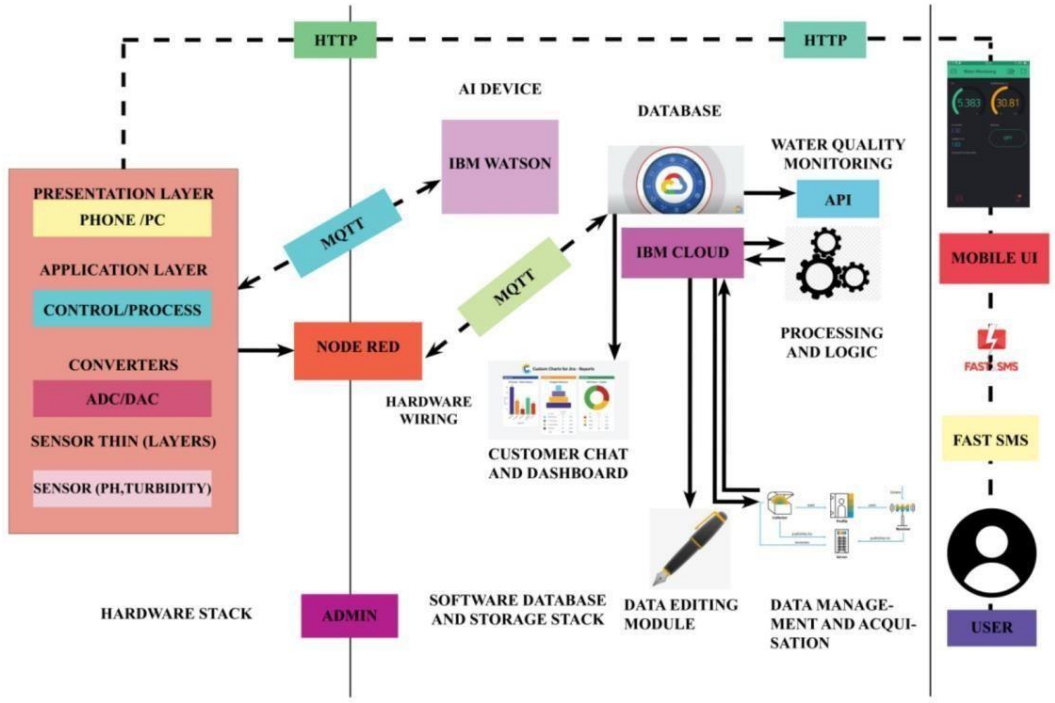


Table-1: Components &Technologies:

S .No	Component	Description	Technology
1.	User Interface	Mobile UI	HTML ,CSS ,java script
2.	ApplicationLogic-1(mobile application)	Scale meter is introduced to monitor the water parameters	Java
3.	ApplicationLogic-2(AI Application)	For predicting future values of water quality range	IBM Watson Assistant
4.	Database	Data Type	NOSQL.
5.	Cloud Database	Database Service on Cloud	IBM Cloud ant
6.	File Storage	File storage requirements: ContainerPlatformVersion4.6	IBM Block Storage
7.	ExternalAPI-1	The data is used to compare the values for Sensor with threshold values	IBM water quality API
8.	ExternalAPI-2	For the locals and authorities to know the water quality	Mobile API,

9.	Machine Learning Model(node-red)	For interfacing hardware and software application (a virtual wiring tool)	Platform: Node.js
10.	Infrastructure(Server/Cloud)	Application Deployment on cloud Server Configuration :application client- bond	IBM cloud

Table-2 :Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Bootstrap	CSS
2.	Security Implementations	MQTT,CoAP,DTLS,6LoWPAN	Encryptions ,OWASP
3.	Scalable Architecture	The scalability of architecture(3–tier)	IOT and mobile application
4.	Availability	Distributed servers	IBM cloud and Watson
5.	Performance	Use of cache ,better performance	Fast SMS application

6. Project planning & scheduling:

6.1. Sprint planning & estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Procurement of Hardware requirements (if needed)	USN-1	Procurement of quality sensors and actuators, microcontroller that will be required to sense the physical parameters like pH, turbidity and Temperature.	2	High	GAYATHRI.M, KARTHIKA.K
	Create IBM Cloud Services	USN-2	Creation of an IBM Cloud account and registering a device.	2	High	
	Configure the IoT device in IBM Cloud.	USN-3	Creation and registering of a device	1	Medium	
Sprint-2	Development of the Python code in IDLE, Install all required libraries like ibmiotf.	USN-4	To develop the Python Code to generate random values of pH , Temperature and turbidity values along with their units.	1	Medium	GAYATHIRI VARSHINI.R, KAVIYA.S,

	Create a IBM Watson IoT service and Publish the values generated by python code to Cloud.	USN-5	To create the IBM Watson IoT Platform and integrate the microcontroller with it, to send the sensed data on cloud	1	High	KANNIKA.C
Sprint-3	Create a Node Red Service	USN-6	To create a node red service to integrate the IBM Watson along with the Web UI	2	Medium	KANNIKA.C
	Create a Web UI	USN-7	To create a Web UI, to access the data from the cloud and display all parameters.	2	Medium	GAYATHIRI VARSHINI.R
	Generate a link to Interface the node red service with the Web UI/Mobile app	USN-8	Generate Link to interface the services.	3	High	GAYATHRI.M
Sprint-4	Design a Mobile App, to display pH, Temperature and turbidity values	USN-9	To design a Android App using MIT App inventor, to display pH, Temperature and turbidity values.	2	High	KARTHIKA
	Fast-SMS Service	USN-10	Use Fast SMS to send alert messages once the parameters like pH, Turbidity and temperature goes beyond the threshold	3	High	KAVIYA,S
	Product Testing	USN-11	Testing of project and final deliverables	3	Medium	GAYATHRI.M

6.2. SPRINT DELIVERY SCHEDULE:

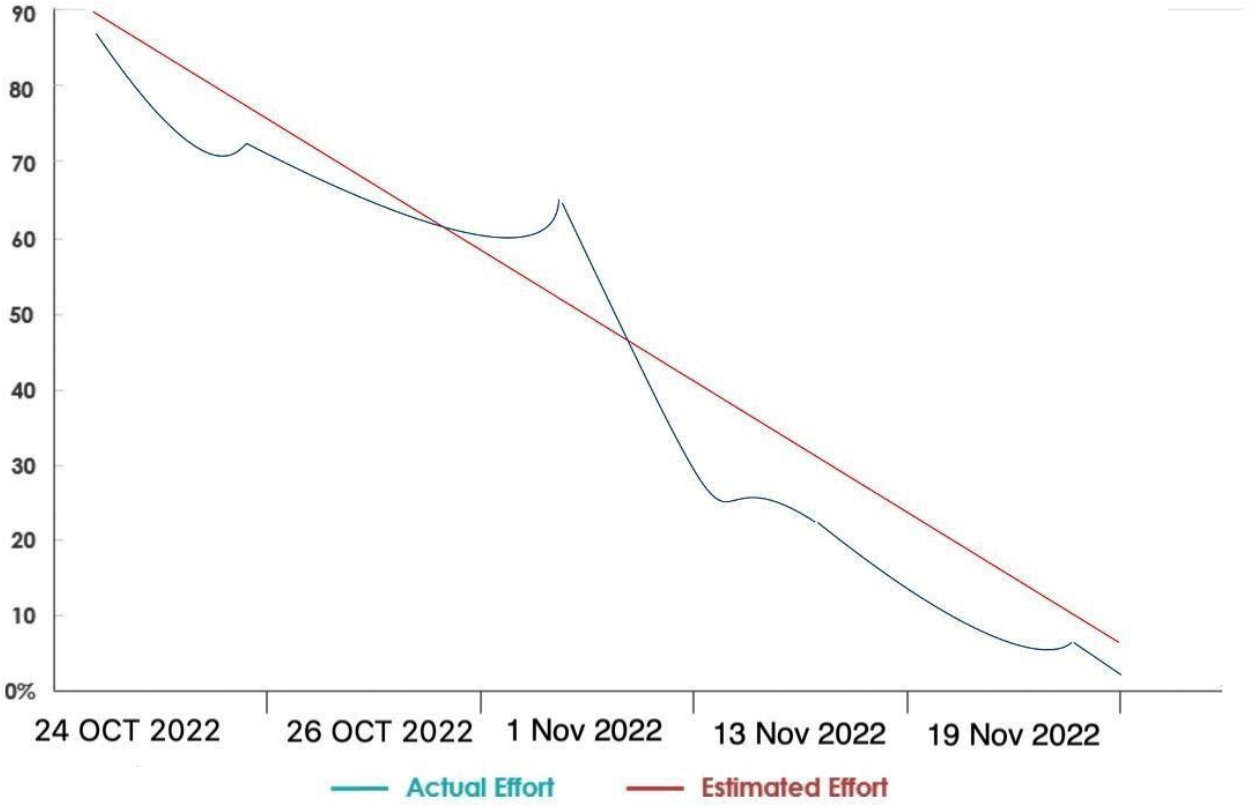
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	30 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	5 Nov 2022	40	06 Nov 2022
Sprint-3	20	6 Days	5 Nov 2022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



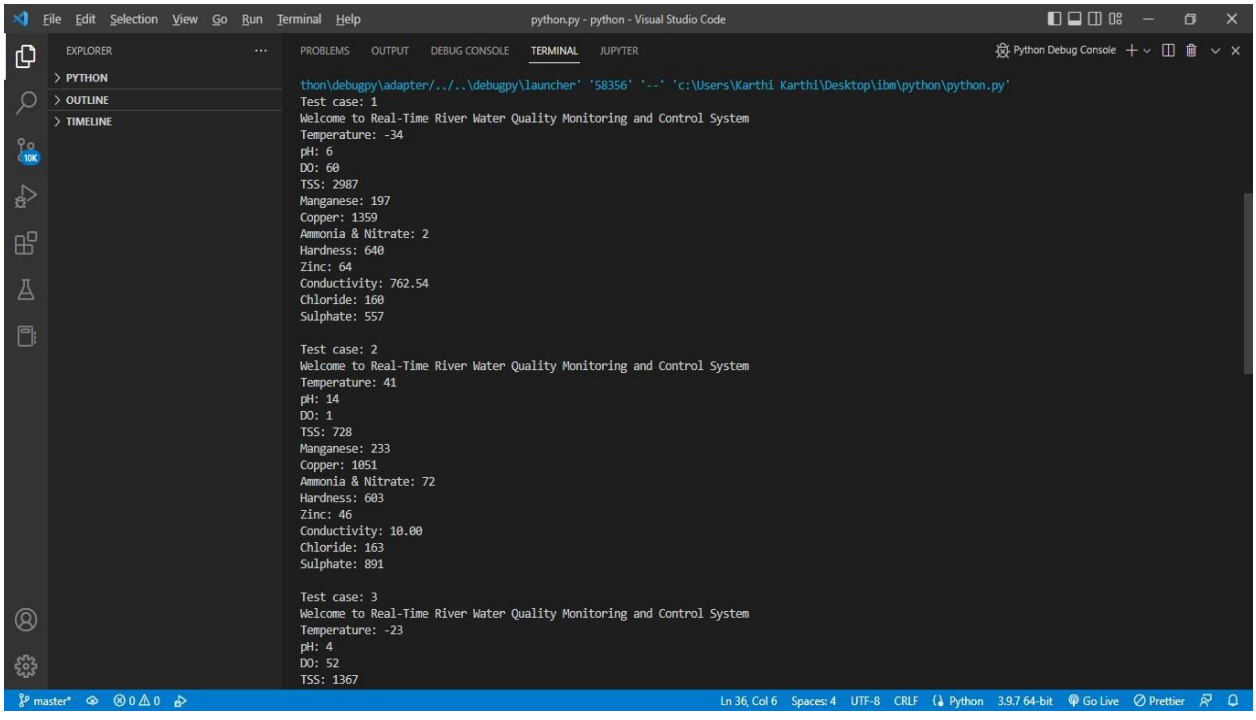
7. CODING & SOLUTIONING:

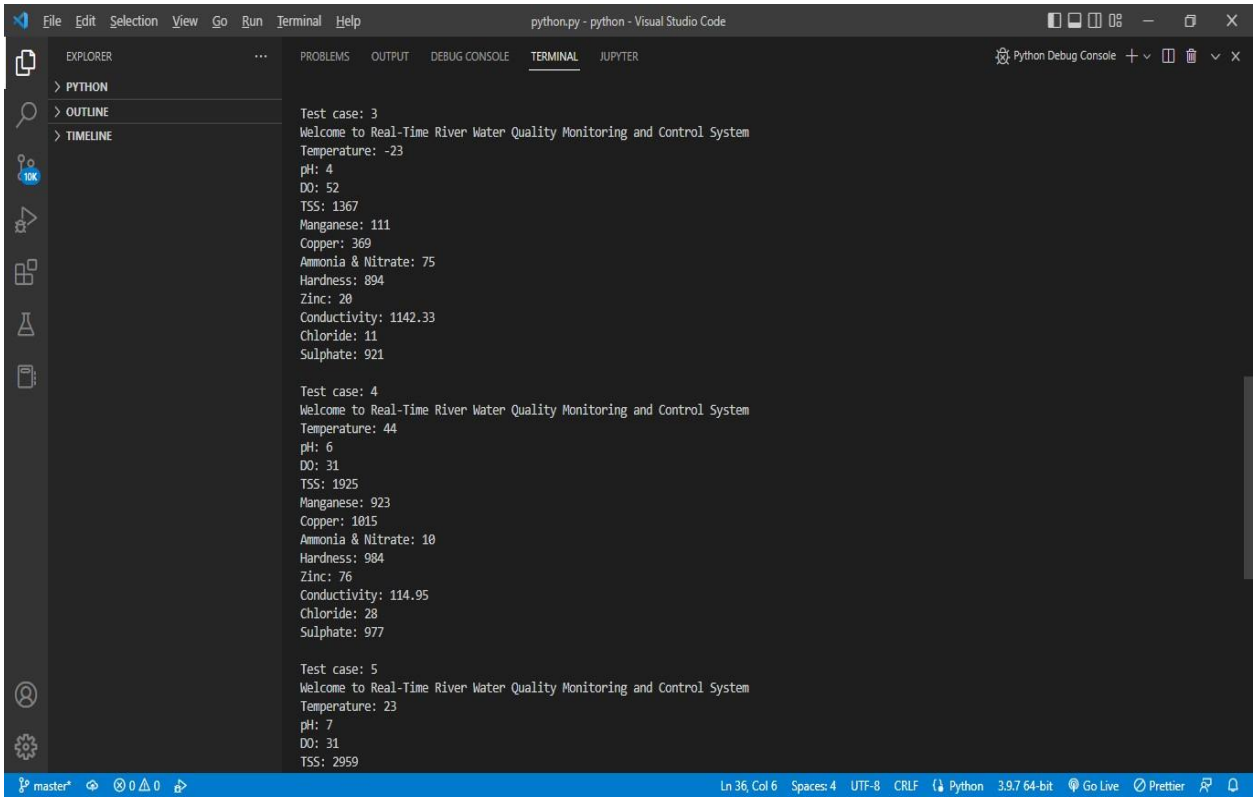
7.1 Feature 1

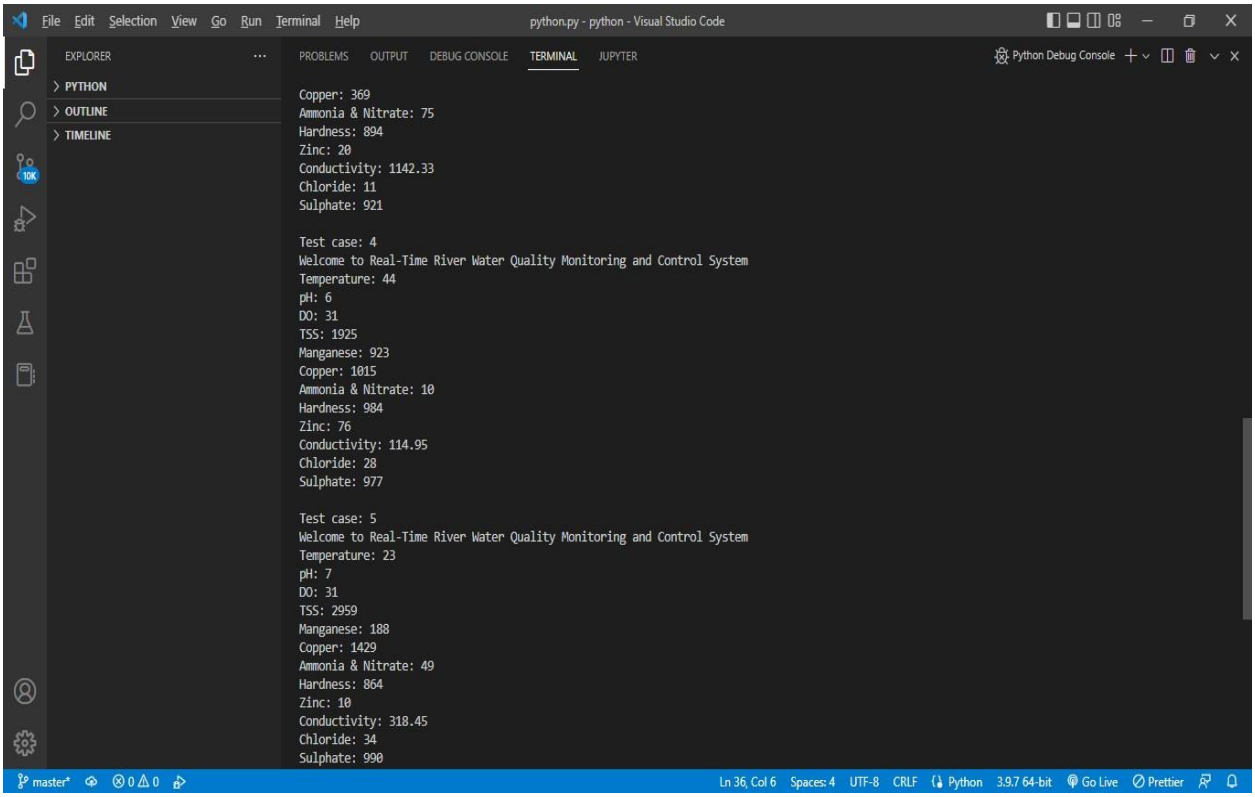
```
#importing Random function to generate the value
import random as rand


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
```

```
    print(

        "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
thon\debugpy\adapter/../..\debugpy\launcher' '58356' '--' 'c:\Users\Karthi Karthi\Desktop\ibm\python\python.py'
Test case: 1
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -34
pH: 6
DO: 60
TSS: 2987
Manganese: 197
Copper: 1359
Ammonia & Nitrate: 2
Hardness: 640
Zinc: 64
Conductivity: 762.54
Chloride: 160
Sulphate: 557
Test case: 2
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 41
pH: 14
DO: 1
TSS: 728
Manganese: 233
Copper: 1051
Ammonia & Nitrate: 72
Hardness: 603
Zinc: 46
Conductivity: 10.00
Chloride: 163
Sulphate: 891
Test case: 3
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -23
pH: 4
DO: 52
TSS: 1367
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
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">
          <spanclass="dotthree"></span>
          <spanclass="gender">Prefernottosay</span>
          </label>
        </div>
      </div>
      <divclass="button">
        <inputtype="submit"value="Register">
      </div>
    </form>
```

Section Break(Next Page)

```
    </div>
  </div>

</body>
</html>
```

```
 @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-
   size: 25px; font-
   weight: 500; position:
   relative;
}
.container .title::before{
   content: "";
   position:absolute;left:
```

CSS CODE:

```
    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;
```

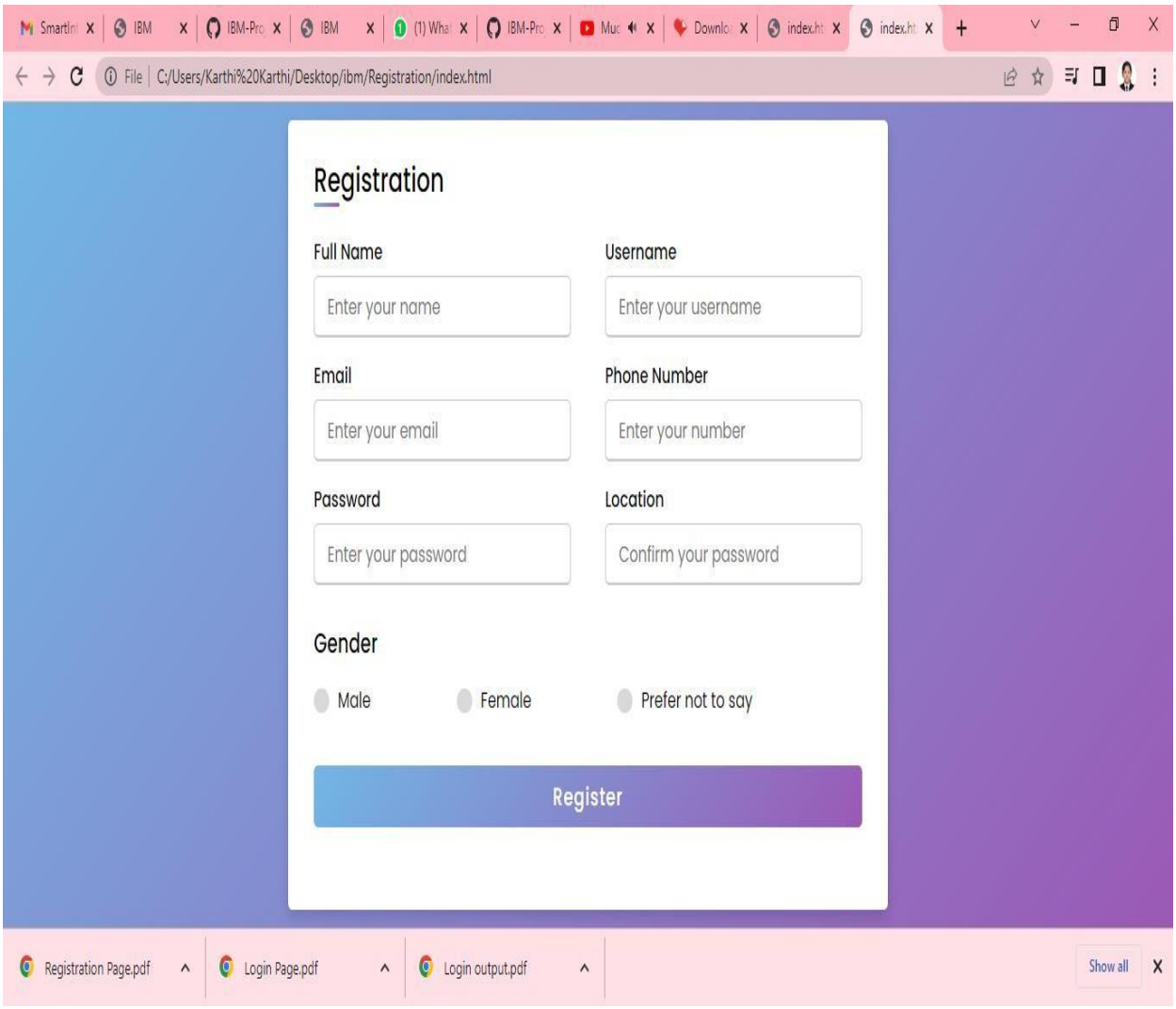
```
    radius:5px;padding-
    left:15px;
    border: 1px solid #ccc;border-
    bottom-width:
    2px;transition:all0.3sease;
}
.user-details.input-boxinput:focus,
.user-details .input-box input:valid{border-
    color:#9b59b6;
}
 form.gender-details.gender-title{font-
    size:20px;
    font-weight:500;
 }
 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;
  }
  form .button{height:
    45px;margin:35px0
  }
  form.buttoninput{heigh
    t:100%;
    width: 100%;border-
    radius:5px;border:
    none;color:#fff;
    font-size: 18px;font-
    weight: 500;letter-
    spacing:1px;cursor:point
```

```
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	PNT2022TMID42440		
				Project Name	Project - Real time River waterquality monitoring and control system		
				Maximum Marks	4 marks		
Test case ID	**Feature Type**	**Component**	**Test Scenario**	**Pre-Requisite**	**Steps To Execute**		**Tes**
Home Page_TC_01	Functional	Home Page	Used to take the user into registerpage		1.Click on Sign Up Button (If NotRegistered)		
Home Page_TC_02	Functional	Home Page	Used to take the user into Loginpage		1.Click on Sign In Button (IfRegistered)		
Register Page_TC_O 1	Functional	Register page	Used to register the user in theapplication	Username & Password	1.Enter Username & Password2.Click on Register button	Username: rithick password:123	
Register Page_TC_O2	Functional	Register page	Used to go back to Home Page		1.Click on Back button		
LoginPage_TC_O1	Functional	Login page	Used to Login into th Application	Username & Password	1.Enter Username & Password2.Click on Login button	Username: dineshpasswo 1234	
LoginPage_TC_O2	Functional	Login page	Used to go back to Home Page		1.Click on Back button		
User Page1_TC_01	Functional	User Page1	It shows the temperature andhumidity value	Temperature value and Humidityvalue	1.Enter Username & Password2.Click on Login button	1. Temperatu 2. Humidity	
User Page1_TC_01	Functional	User Page1	Used to go back to Home Page		1.Click on Logout button		
User Page2_TC_01	Functional	User Page2	Used to go back to User Page1		1.Click on Back button		

8 .2. 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 enabledsolution.

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. Moreover 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"
                    )
```

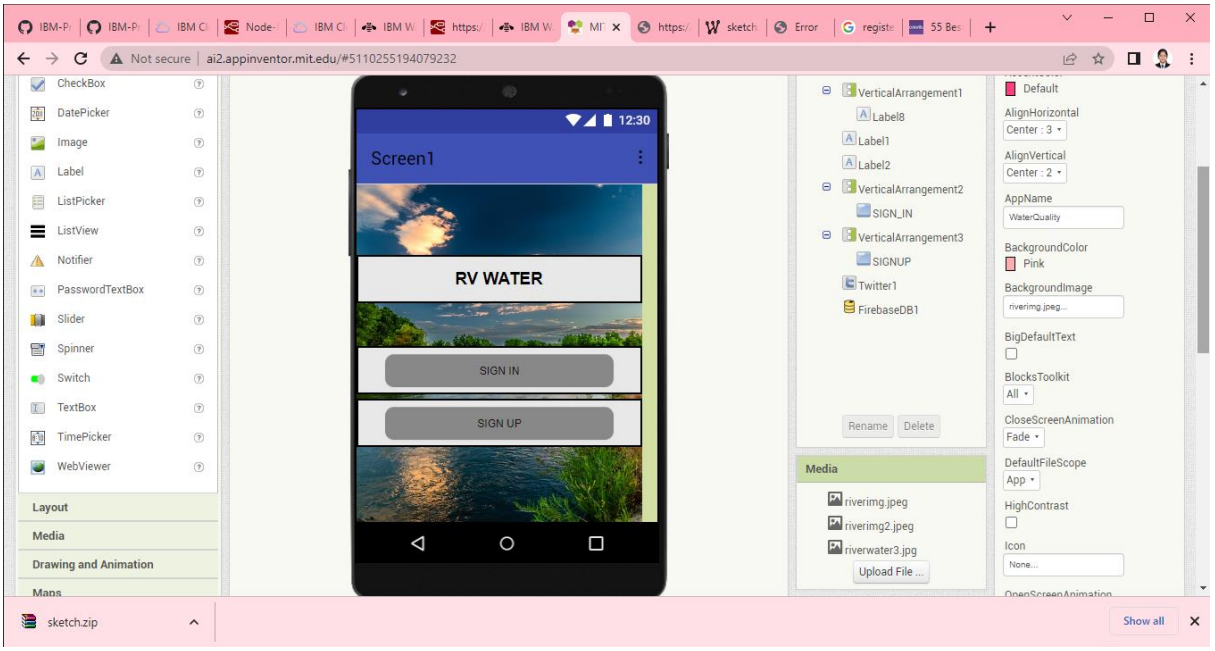
4. Aurdino:

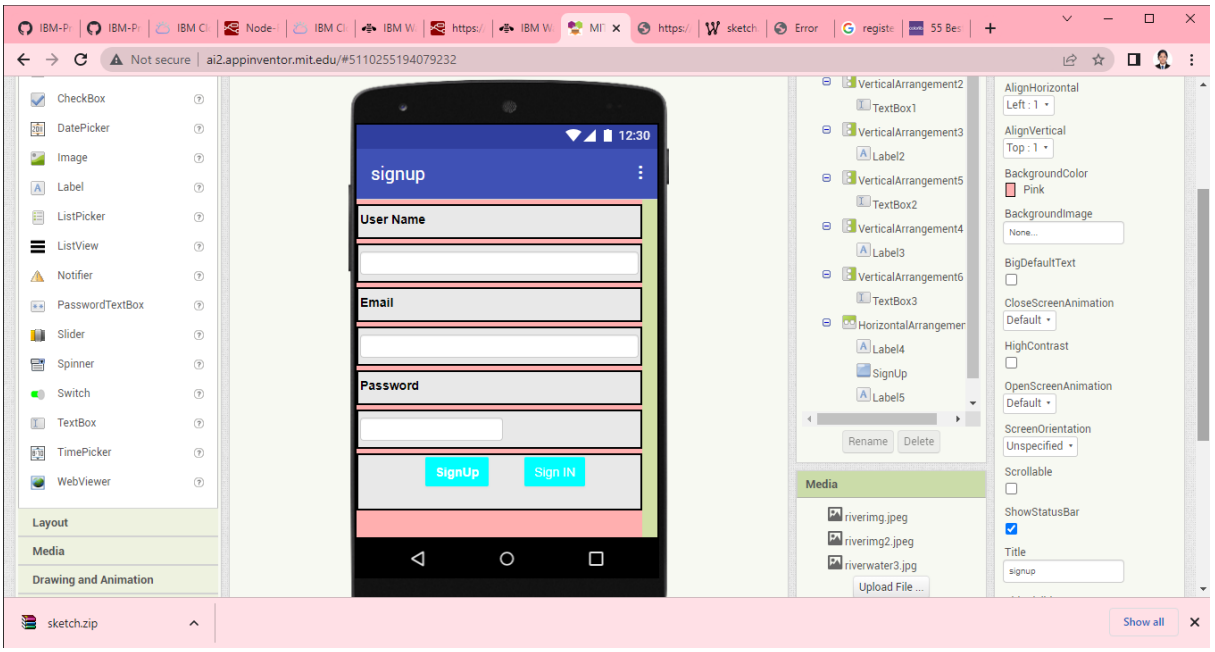
```
#include
<OneWire.h>
                    #include <DallasTemperature.h>
                    #define ONE_WIRE_BUS 5
                    OneWire oneWire(ONE_WIRE_BUS);
                    DallasTemperature sensors(&oneWire);
                    float Celcius=0;
                    float Fahrenheit=0;
                    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;
```

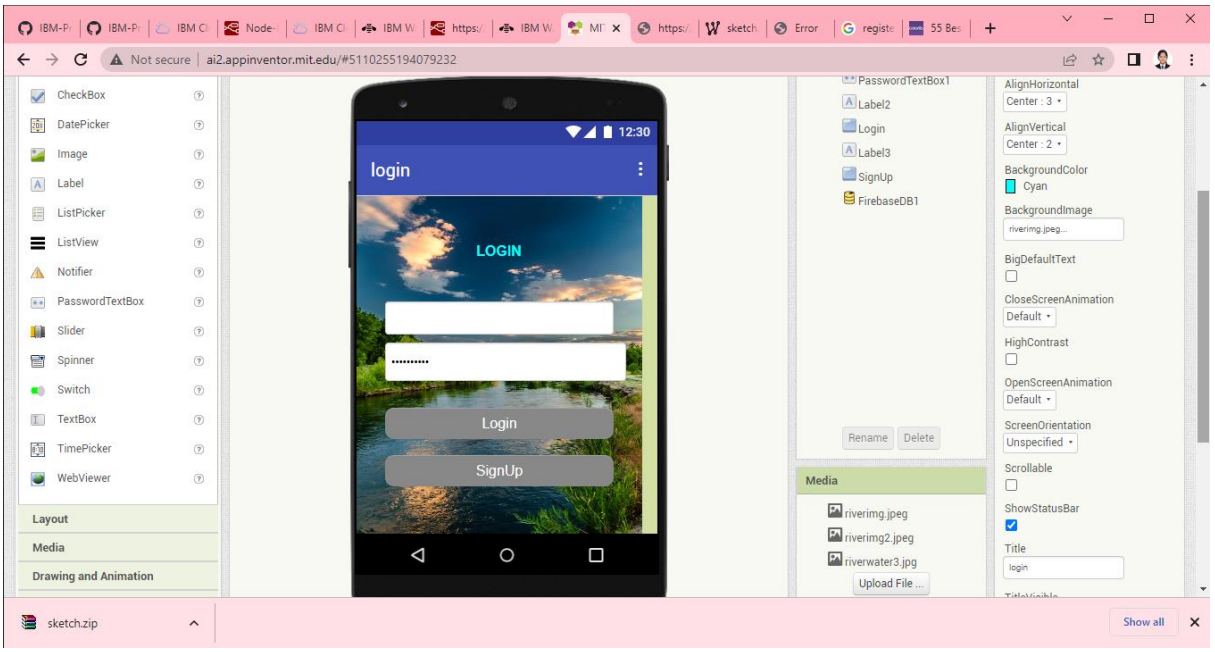
```
  }
   }
}
for(int i=2;i<8;i++)
avgValue+=buf[i];
float pHVol=(float)avgValue*5.0/1024/6;
float phValue = -5.70 * pHVol + 21.34;
Serial.println(phValue);
Serial.print("pH");
```

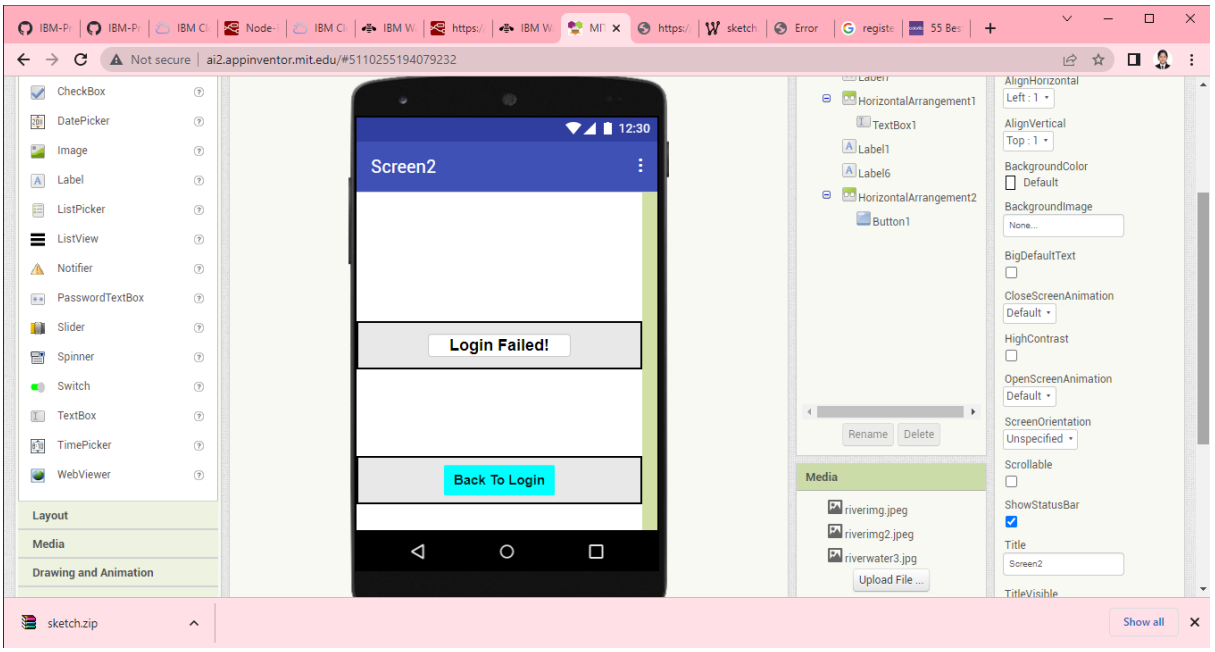
```
Serial.print(" C ");
        Serial.print(Celcius);
       Serial.print(voltage);
Serial.print("V");
delay(10000);
}
```

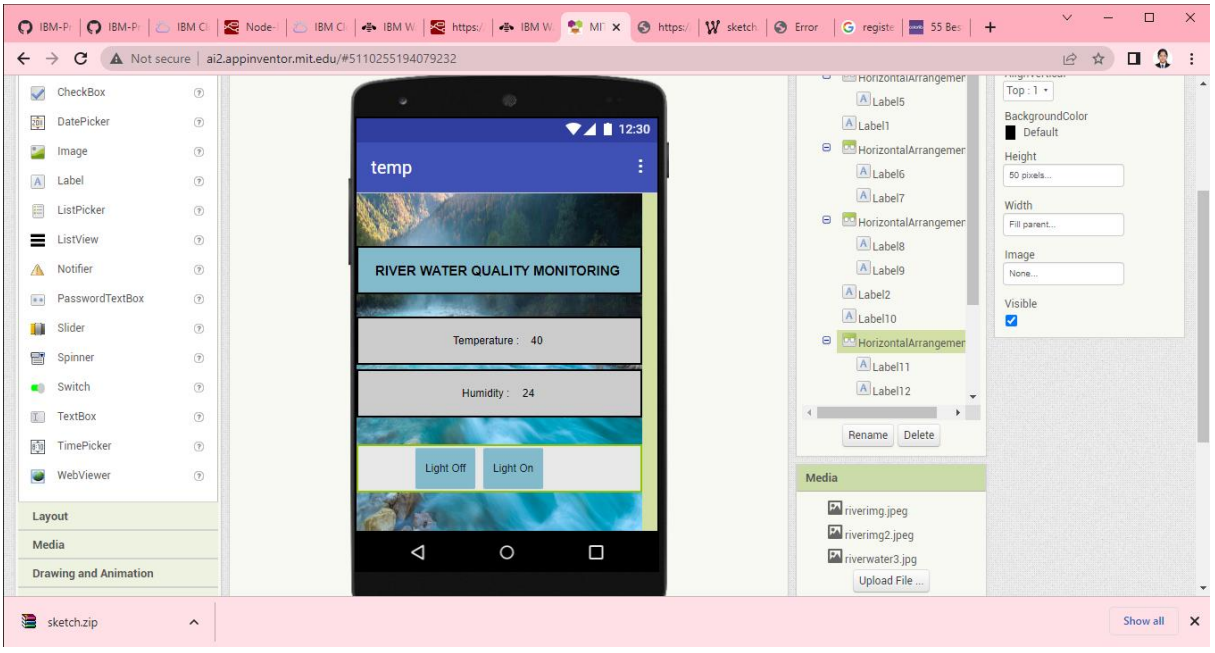
13.2 GITHUB AND PROJECT DEMO LINK:

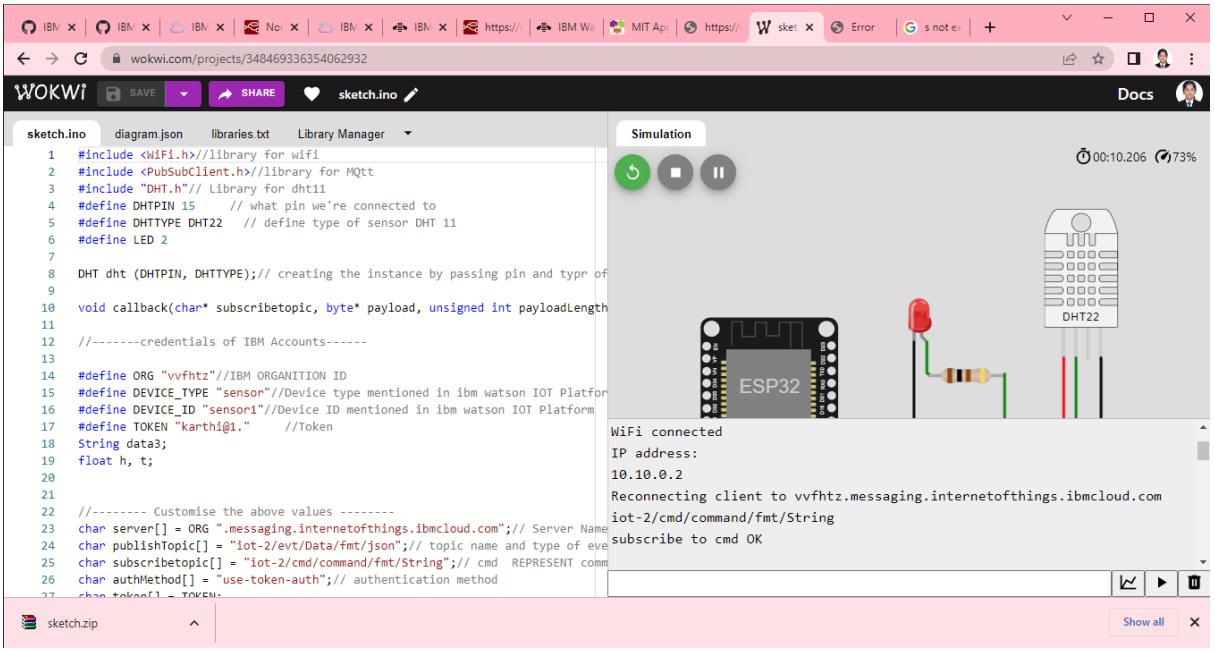



Not secure | ai2.appinventor.mit.edu/#5110255194079232
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
12:30
signup
User Name
Email
Password
SignUp
Sign IN
VerticalArrangement2
TextBox1
VerticalArrangement3
Label2
VerticalArrangement5
TextBox2
VerticalArrangement4
Label3
VerticalArrangement6
TextBox3
HorizontalArrangemer
Label4
SignUp
Label5
Rename
Delete
Media
riverimg.jpeg
riverimg2.jpeg
riverwater3.jpg
Upload File ...
AlignHorizontal
Left : 1
AlignVertical
Top : 1
BackgroundColor
Pink
BackgroundImage
None...
BigDefaultText
CloseScreenAnimation
Default
HighContrast
OpenScreenAnimation
Default
ScreenOrientation
Unspecified
Scrollable
ShowStatusBar
Title
signup
sketch.zip
Show all


Not secure | ai2.appinventor.mit.edu/#5110255194079232
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
12:30
login
LOGIN
Login
SignUp
PasswordTextBox1
Label2
Login
Label3
SignUp
FirebaseDB1
Rename
Delete
Media
riverimg.jpeg
riverimg2.jpeg
riverwater3.jpg
Upload File ...
AlignHorizontal
Center : 3
AlignVertical
Center : 2
BackgroundColor
Cyan
BackgroundImage
riverimg.jpeg...
BigDefaultText
CloseScreenAnimation
Default
HighContrast
OpenScreenAnimation
Default
ScreenOrientation
Unspecified
Scrollable
ShowStatusBar
Title
login
sketch.zip
Show all


Not secure | ai2.appinventor.mit.edu/#5110255194079232
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
12:30
Screen2
Login Failed!
Back To Login
HorizontalArrangement1
TextBox1
Label1
Label6
HorizontalArrangement2
Button1
Rename
Delete
Media
riverimg.jpeg
riverimg2.jpeg
riverwater3.jpg
Upload File ...
AlignHorizontal
Left : 1
AlignVertical
Top : 1
BackgroundColor
Default
BackgroundImage
None...
BigDefaultText
CloseScreenAnimation
Default
HighContrast
OpenScreenAnimation
Default
ScreenOrientation
Unspecified
Scrollable
ShowStatusBar
Title
Screen2
sketch.zip
Show all


Not secure | ai2.appinventor.mit.edu/#5110255194079232
CheckBox
DatePicker
Image
Label
ListPicker
ListView
Notifier
PasswordTextBox
Slider
Spinner
Switch
TextBox
TimePicker
WebViewer
Layout
Media
Drawing and Animation
12:30
temp
RIVER WATER QUALITY MONITORING
Temperature : 40
Humidity : 24
Light Off
Light On
Label5
Label1
Label6
Label7
Label8
Label9
Label2
Label10
Label11
Label12
Rename
Delete
Media
riverimg.jpeg
riverimg2.jpeg
riverwater3.jpg
Upload File ...
Top : 1
BackgroundColor
Default
Height
50 pixels...
Width
Fill parent...
Image
None...
Visible
sketch.zip
Show all


wokwi.com/projects/348469336354062932
WOKWI
SAVE
SHARE
sketch.ino
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15 // what pin we're connected to
#define DHTTYPE DHT22 // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength
//-------credentials of IBM Accounts------
#define ORG "vvfhtz"//IBM ORGANITION ID
#define DEVICE_TYPE "sensor"//Device type mentioned in ibm watson IOT Platfor
#define DEVICE_ID "sensor1"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "karthi@1." //Token
String data3;
float h, t;
//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of eve
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT comm
char authMethod[] = "use-token-auth";// authentication method
Simulation
00:10.206
73%
ESP32
DHT22
WiFi connected
IP address:
10.10.0.2
Reconnecting client to vvfhtz.messaging.internetofthings.ibmcloud.com
iot-2/cmd/command/fmt/String
subscribe to cmd OK
sketch.zip
Show all