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

GERALD RUBAN	19110026
ASWATH S	19110009
LAKSHMANA KUMAR	19110048
BENHER CHRISTOPHER	19110011

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

HINDUSTHAN COLLEGE OF ENGINEERING AND TECHOLOGY

Approved by AICTE, New Delhi, Accredited with 'A' Grade by NAAC

(An Autonomous Institution, Affiliated to Anna University, Chennai)

COIMBATORE – 641 032

November 2022

ANNA UNIVERSITY :: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report titled "**Real-Time River Water Quality Monitoring and Control System by NALAIYA THIRAN PROJECT BASED LEARNING Program"**, is the bonafide work of **GERALD RUBAN (19110026), ASWATH S (19110009), LAKSHMANA KUMAR (19110048), BENHER CHRISTOPHER (19110011)** who carried out the work under faculty mentor and industry mentor supervision, for the partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY.**

Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

DECLARATION

I, hereby declare that the Project work entitled "**Real-Time River Water Quality Monitoring and Control System by NALAIYA THIRAN PROJECT BASED LEARNING Program**" submitted to the IBM November 2022 in partial fulfilment for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY,** is the report of the original project work done by us under the guidance of Mrs. BEAULAH DAVID (Faculty Mentor), Assistant Professor, Department of B.TECH IT, Hindusthan College of Engineering and Technology, Coimbatore.

NAME **SIGNATURE**

GERALD RUBAN
(Team Leader)

I certify that the declaration made by the above candidate is true.

SIGNATURE
FACULTY MENTOR
Mrs. BEAULAH DAVID ,
ASSISTANT PROFESSOR,
Dept. B.TECH -IT.
HiCET,Coimbatore-641032

ABSTRACT

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 system consists of several sensors which is used to measure physical and chemical parameters of the water. The main components of Wireless Sensor Network (WSN) include a microcontroller 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.

Now a day's Internet of things (IoT) is an innovative technological phenomenon. It is shaping today's world and is used in different fields for collecting, monitoring and analysis of data from remote locations. IoT integrated network if everywhere starting from smart cities, smart power grids, and smart supply chain to smart wearable. Though IoT is still under applied in the field of environment it has huge potential. It can be applied to detect forest fire and early earthquake, reduce air population, monitor snow level, prevent landslide, and avalanche etc. Moreover, it can be implemented in the field of water quality monitoring and controlling system. Water quality monitoring has gained more interest among researchers in this twenty-first century. Numerous works are either done or ongoing in this topic focusing on various aspects of it. The key theme of all the projects was to develop an efficient, cost-effective, real-time water quality monitoring system which will integrate wireless sensor network and internet of things. In this research, we monitor the physical and chemical parameters of water bodies inside Chittagong city by using an IoT based sensor network.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1. INTRODUCTON

The Internet of Things (IoT) is a system that allows devices to be connected and remotely monitored across the Internet. In the last years, the IoT concept has had a strong evolution, being currently used in various domains such as real-time river water quality monitoring and control system, telemedicine, industrial environments, etc. According to Human Rights Watch, twenty million people in our country are still drinking water contaminated with arsenic. The World health Organization (WHO) has also stated this crisis as "the largest mass poisoning of a population in history". To reduce the water related diseases and prevent water population, we have to measure water parameters such as pH, turbidity, conductivity, temperature etc. Traditional methodology of water monitoring requires collecting data from various sources manually. Afterwards samples will be sending to laboratory for testing and analyzing. In order to save time consumption and decrease manual effort my testing equipment's will be placed in any water source. As a result, this model can detect pollution remotely and take necessary actions.

1.2.COMPANY PROFILE

International Business Machines Corporation (IBM) is a technology company engaged in providing hybrid cloud and artificial intelligence (AI) solutions. It offers integrated solutions and products that use data and information technology (IT) in industries and business processes. Its segments include Software, Consulting, Infrastructure and Financing. Software segment consists of two business areas: Hybrid Platform & Solutions, which includes software to help clients operate, manage, and optimize their IT resources and business processes within hybrid, multi-cloud environments, and Transaction Processing, which includes software that supports clients' mission-critical, on-premises workloads in various sectors. Consulting segment is engaged in business transformation, technology consulting and application operations. Infrastructure segment is engaged in hybrid infrastructure and infrastructure support. Financing segment is engaged in client financing and commercial financing business

CHAPTER 2

OBJECTIVE

2. OBJECTIVE

Project based learning are generally thought of to be reserved for college students looking to gain experience in a particular field. However, the 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 analyzed to improve the water quality. The main objective of this project is to

- Gain knowledge of Watson IoT Platform.
- Connecting IoT devices to the Watson IoT platform and exchanging the sensor data.
- Gain knowledge on Cloudant DB
- Creating a Web Application through which the user interacts with the device.

This project makes the human work much easier.

2.1.TECHNOLOGY

The Internet of things (IoT) describes physical objects with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks. IoT is a giant, digitally connected universe of billions of physical devices around the world; “things” that collect and share data about how they’re used and the environment around them. These objects are embedded with internet connectivity, software, sensors, and other hardware that enable them to connect and exchange data with other systems and devices over the web. IoT extends the power of the internet beyond smartphones and computers to ordinary household objects such as lightbulbs, locks, smart microwaves, wearable fitness devices, sophisticated industrial tools, and self-driving cars, affording them a higher degree of analytical and computing capabilities

CHAPTER 3

IDEATION PHASE

3. IDEATION PHASE

3.1 Literature Survey

REAL-TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM LITERATURE SURVEY

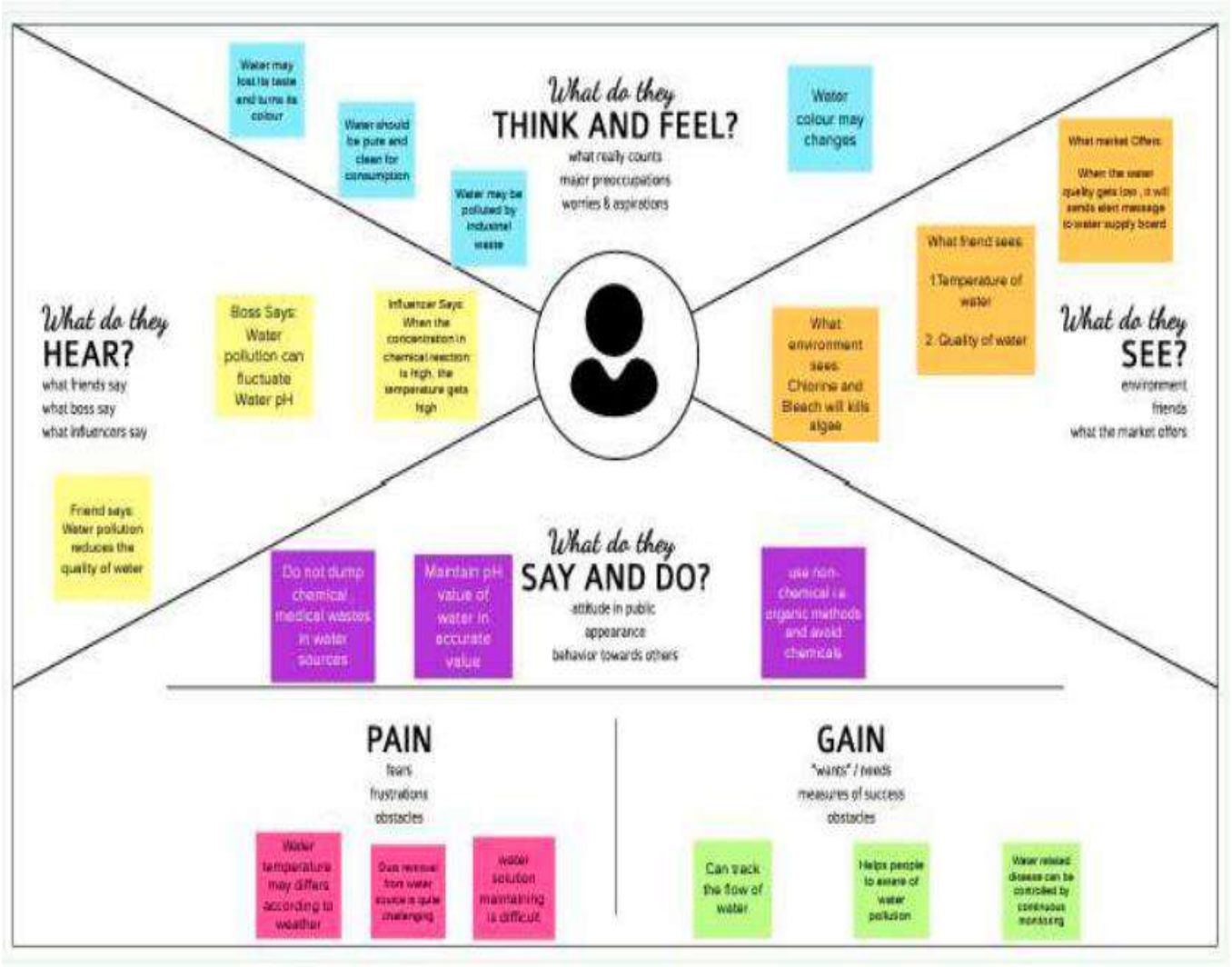
The Purpose of this chapter to review the previous of Researchers on the Real-time River water quality monitoring and control system using Internet of things. This chapter will present on continuous monitoring of river water quality at remote places using wireless sensor networks. Current water quality 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 microcontroller 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. To design a good quality model, we reviewed out different existing system developed by researchers. Different authors have proposed distinguished models to check water quality by analysing the parameters such as temperature, pH and conductivity, and so on. By considering all these points, we designed a smart water monitoring system which can perform all these monitoring functions. Stephen Brosnan investigated a WSN to collect real time water quality parameters (WQP). Quio Tie-Zhn, developed online water quality monitoring system based on GPRS/GSM. The information was sent by means of GPRS network, which helped to check remotely the WQP. Kamal Alameh presented web based WSN for monitoring water pollution using ZigBee and WiMAX networks. The system collected, processed measured data from sensors, and directed through ZigBee gateway to the web server by means of WiMAX network to monitor quality of water from large distances in real time. Dong, He developed WQM system based on WSN. The remote sensor was based on ZigBee network. WSN tested WQP and sent data to Internet using GPRS. With the help of Web, information was gathered at remote server. Vijayakumar et al., designed a low-cost system design for real time water quality monitoring in IoT utilizes sensors to check many important physical and chemical parameters of water. The parameters such

as turbidity, temperature, pH, dissolved oxygen conductivity of water can be measured. In our project, we proposed a water quality monitoring system based on IoT.

Reference:

[1] ChoZinMyint, Lenin Gopal and Yan Lin Aung," Reconfigurable smart water quality monitoring system in iot environment", IEEE Internatinal Conference on Information Systems (ICIS),978-1-5090-5507-4/17, May 2017. [2] Sona Pawara, Siddhi Nalam, Saurabh Mirajkar, Shruti GujarVaishali Nagmoti," Remote Monitoring of Waters Quality from Reservoirs", 2017 2nd International Conference for Convergence in Technology (I2CT). [3] Francesco A, Fliippo A, Carlo G C, Anna M L," A Smart sensor network for sea water quality monitoring, IEEE Sensors J 15(5):2514-2522, May 2015. [4] S. P. Gorde, M. V. Jadhav "Assessment of Water Quality Parameters: A Review", S. P. Gorde et al Int. Journal of Engineering Research and Applications, ISSN: 2248-9622, Vol. 3, Issue 6, Nov-Dec 2013, pp.2029-2035. [5] S. Geetha and S. Gouthami," Internet of things enabled real time water quality monitoring system", Springer open (2017) 2:1 DOI 10.1186/s40713-017-0005-y. [6] AainaVenkateshwaran, HarshaMendha, Prof. PritiBadar, "An IoT based system for water quality monitoring", International Journal of Innovation Research in Computer and Communication Engineering, Vol.5, Issue 4, April 2017. [7] VaishanviVDaigavane, Dr. M A Gaikwad," Water quality monitoring system based on IoT", Advances in wireless and mobile communications, ISSN 0973-6972 Volume 10, Number 5,2017, pp. 1107-1116. [8] Aravinda S. Rao, Stephen Martial, JayavardhanaGubbi, MarimuthuPalani Swami, "Design of lowcost autonomous water quality monitoring system", 2013 IEEE, pp. 14-19. [9] Cheng-Liang Lai, Chien-Lun Chiu "Using image processing technology for water quality monitoring system", July2011 IEEE, pp. 1856-1861. [10] Niel Andre Cleote, Reza Malekian and Lakshmi Nair," Design of smart sensors for real-time water quality monitoring,", vol 13, no. 9, September 2014 IEEE. [11] K. S. Adu-Manu, C. Tapparello, W. Heinzelman, F. A. Katsriku, and J.-D. Abdulai, "Water quality monitoring using wireless sensor networks: Current trends and future research directions," ACM Transactions on Sensor Networks (TOSN), vol. 13, p. 4, 2017.

3.2 Empathy Map



3.3 Ideation

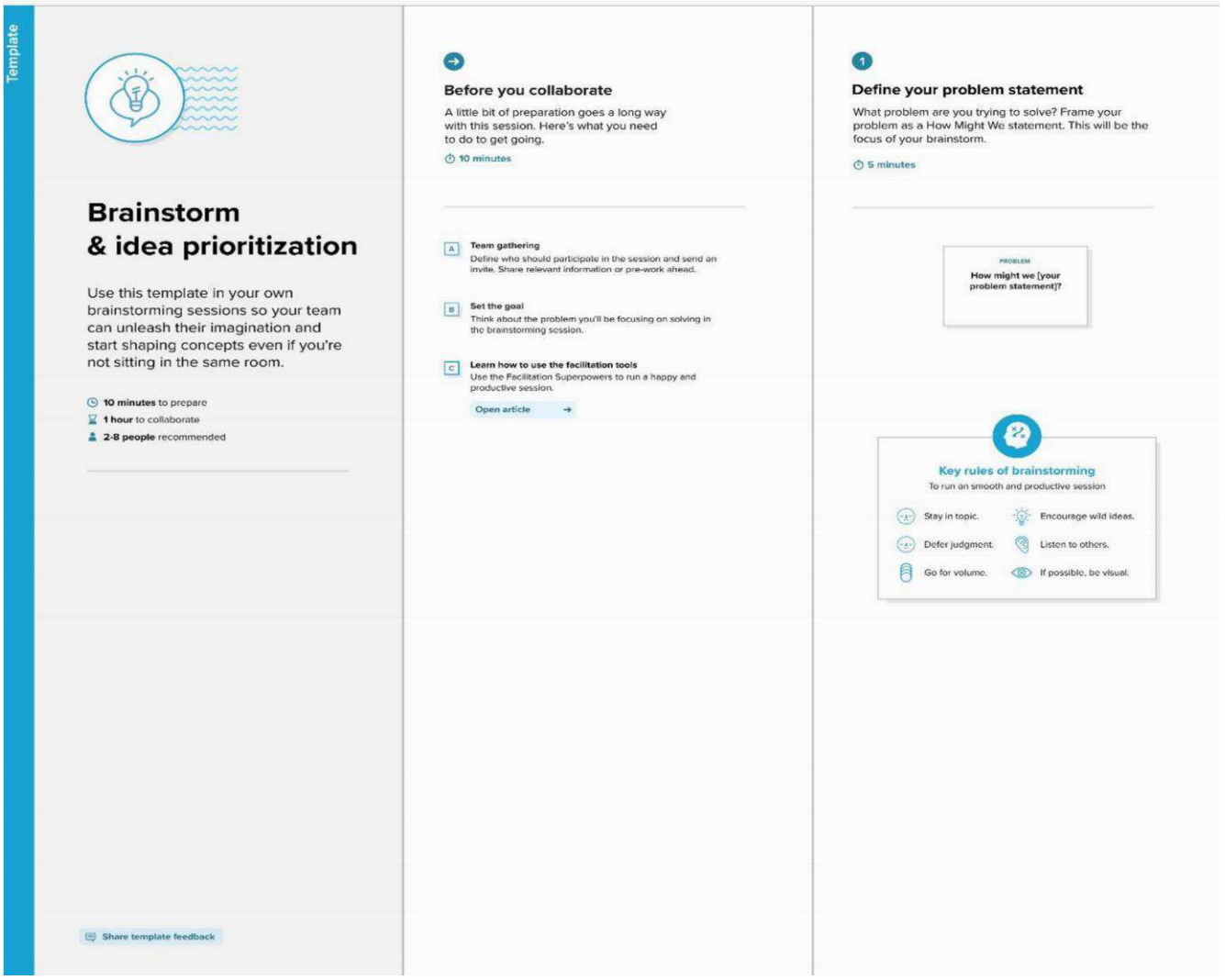
Ideation Phase

Brainstorm & Idea Prioritization Template

Date	21 SEPTEMBER 2022
Team ID	PNT2022TMID10251
Project Name	IOT Based Real-Time River Water Quality Monitoring and Control System
Maximum Marks	4 Marks

Brainstorm & Idea Prioritization

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



Step-2: Brainstorm, Idea Listing and Grouping

2

Brainstorm

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

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

GERALD RUBAN

Monitor the quality of water.	Testing the quality from remote location.	Arduino Controller is used to generate readings
Water pollution can be investigated.	Current system is time consuming	Water is an important factor of our ecosystem.
A wireless communication system is sufficient.	pH, turbidity and temperature will be the sensors used.	Collected data can be stored in cloud platfrom.

LAKSHMANA KUMAR

Real-time database is used as cloud server.	Collected data is analyzed and results are updated.	Authorizations are responsible for giving alerts about the current quality of water.
Alerts can be provided as SMS.	Cloud Data can be retrieved anywhere anytime.	Different sensors can be used to access the water quality.
Current state of water is due to man made activities.	Monitoring water quality monitoring is very important for maintaining ecosystem and livelihood.	This system also helps in maintaining the water quality.

ASWATH S

Ensuring the quality of water before using it is the best.	Water quality monitoring system is cost efficient.	Determining the quality of water reveals the health consequences that may happen.
Existing water quality monitoring system is high power consuming and high cost.	This system is more accurate than the existing system.	Predictions can be done over the cloud data.
Remote monitoring of water quality is time saving.	Testing the quality from remote location.	It also helps in reducing the risk of causing many deceases.

BENHER CHRISTOPHER

Usage of different sensors to analyze the water quality	Testing the quality from remote location.	Machine learning algorithms are used to draw conclusions on water quality.
Existing water quality monitoring system is a manual system.	Monitoring water quality plays an important role in determining whether the water is consumable or not.	The existing system is time consuming where the proposed system is not.
Advanced and automated sensor can give detailed insight about water quality.	This system of water quality monitoring is an automated system.	Whenever the quality of water exceeds the normal level, the user will be notified accurately.

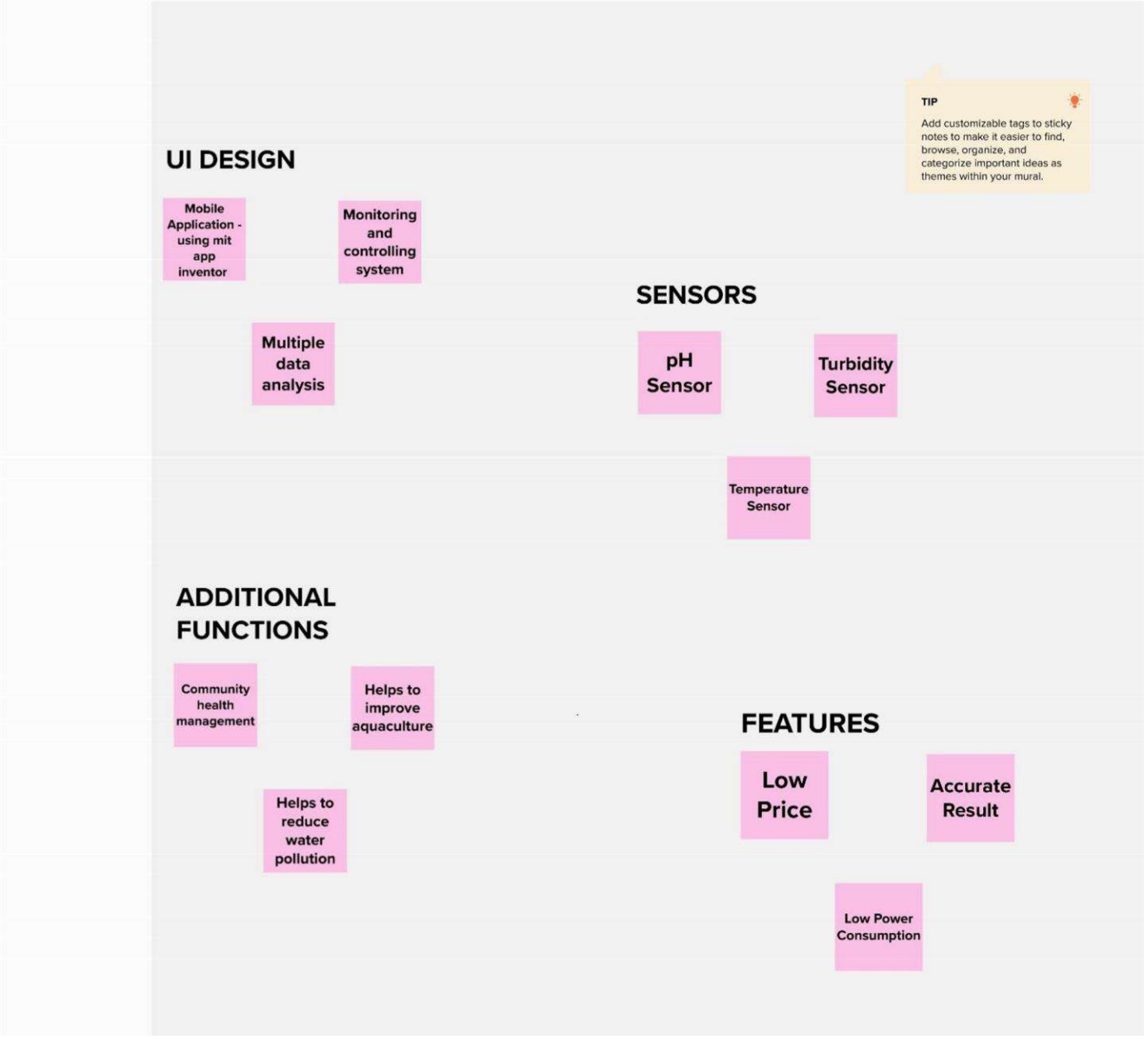
Step-3: Idea Prioritization

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

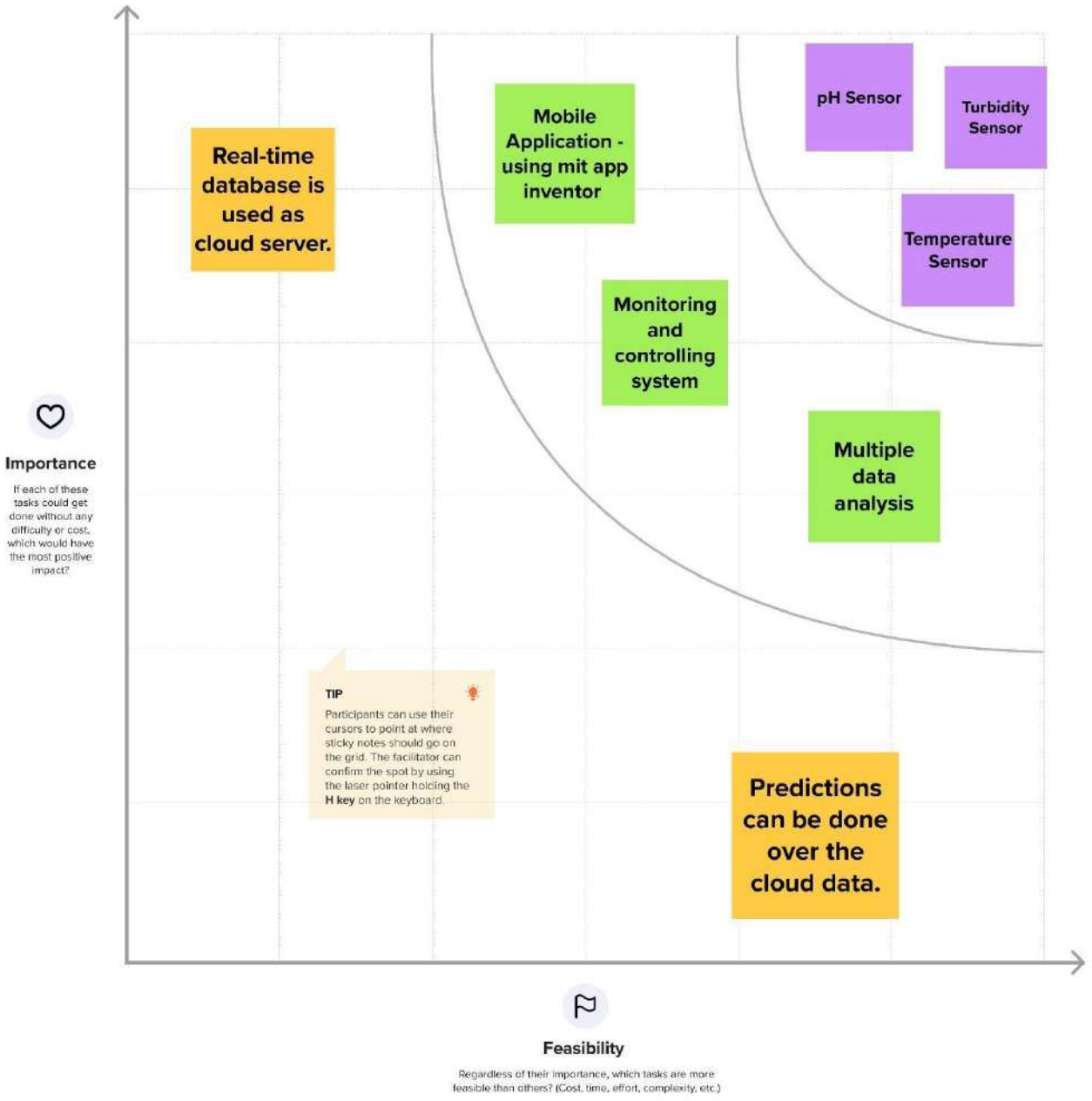


4

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



3.4 Problem Statement

Ideation Phase
Define the Problem Statements

Date	21 SEPTEMBER 2022
Team ID	PNT2022TMID10251
Project Name	Project - IOT based Real-Time River Water Quality Monitoring and Control System.
Maximum Marks	2 Marks

I am (Customer)	An authorised person who is supposed to ensure the safety of common people.	Common people living on Earth who consume water in their day-to-day life for different purpose. So, it is necessary for authorizations to confirm that the water is safe.
I'm trying to	Monitor the quality of water.	Wants to monitor the water consumed every day and check whether the water is contaminated or pure, its pH, temperature and salinity.
But	The existing model is not accurate.	Accuracy is very important because the quality of water is a very sensitive content that can damage the lives of people and nature.
Because	The existing model is high cost, high power consumption and also it is not automated.	It is difficult to be alert about the lack of quality of water at every single time.

Which makes me feel	Concerned about the safety of people.	If the authorized people are not aware of the lack of water quality then they won't be able to give alerts to people consuming that water, which will lead to a serious problem.

CHAPTER 4

PROJECT DESIGN

PHASE 1

4. PROJECT DESIGN PHASE 1

Project Design Phase-I

Proposed Solution Template

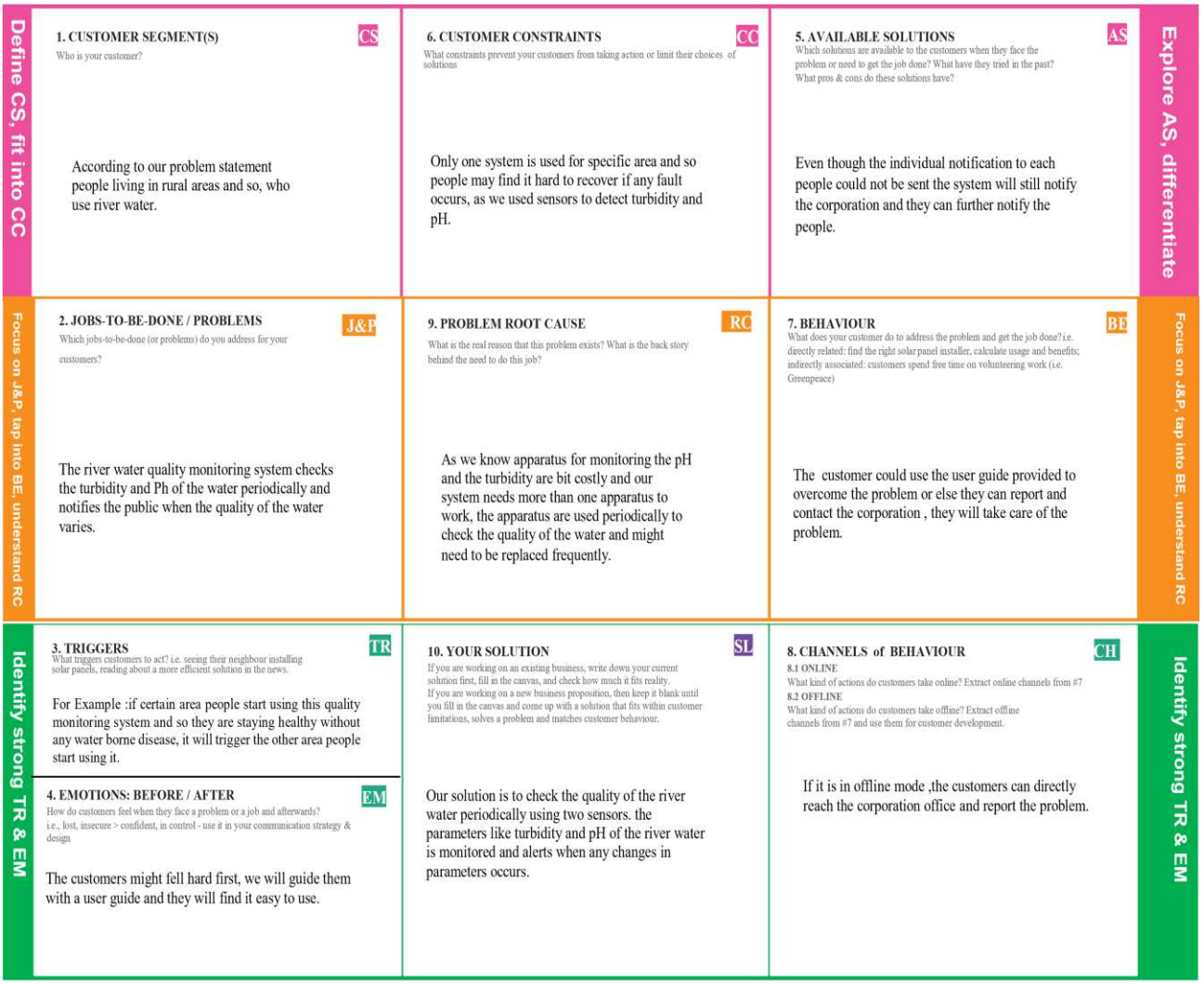
Date	20 October 2022
Team ID	**PNT2022TMID10251**
Project Name	Project - Real-Time River Water Quality Monitoring And Control
Maximum Marks	2 Marks

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Often people and other living organisms are suffered due to unavailability of pure usable water. Due to this health hazards and other infections are spreaded among people. In order to secure them it is necessary to develop a system to handle the quality of water. This can also help the people to have an idea on drinkable water.
2.	Idea / Solution description	• So, to start this, we just need to know or have an idea on the chemical composition of water or simply the nature of water • Based on timely taken analysis we can find the nature of water . • Use a random location on taking the amount of chemicals and impurities present in water.
3.	Novelty / Uniqueness	Low investment and maintainace cost , This system developed is useful and creates an ease of pure water consumption for natives as well as other beings.
4.	Social Impact / Customer Satisfaction	• This helps the people to save time and energy as they can get pure river water with ease. • Building an effective system that can be create as a product for best water quality and control system.
5.	Business Model (Revenue Model)	Many other parts of the world and rural parts of the village are expecting this technology that can greatly facilitate the river water quality management system.
6.	Scalability of the Solution	The process of operating is easy and it can designed according to customer needs.

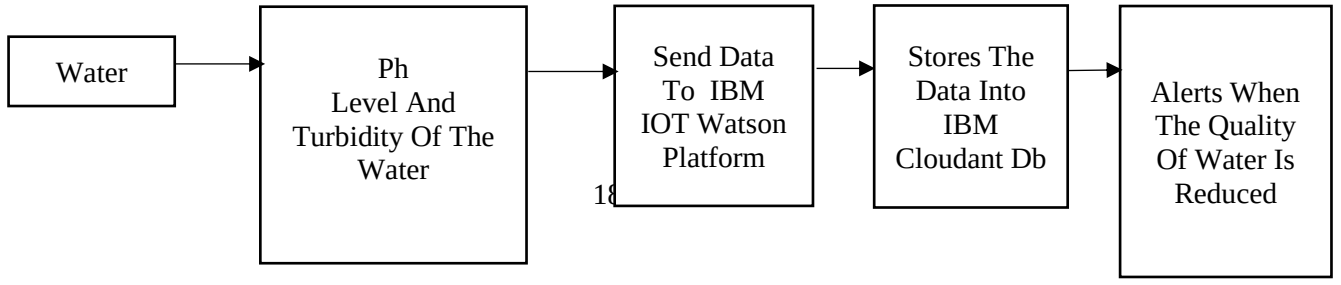
4.2 Problem Solution Fit



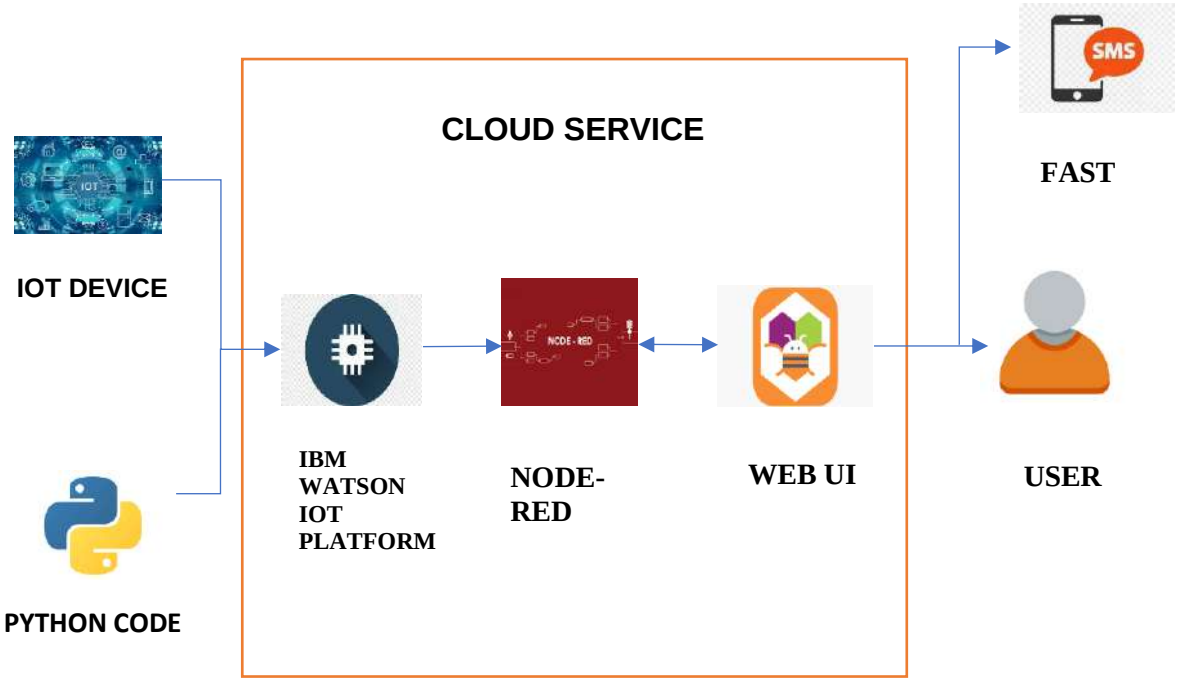
4.3 Solution Architecture

Date	21 October 2022
Team ID	**PNT2022TMID10251**
Project Name	Project - Real-Time River Water Quality Monitoring and Control System
Maximum Marks	4 Marks

Solution Architecture:



Example –Solution Architecture Diagram:

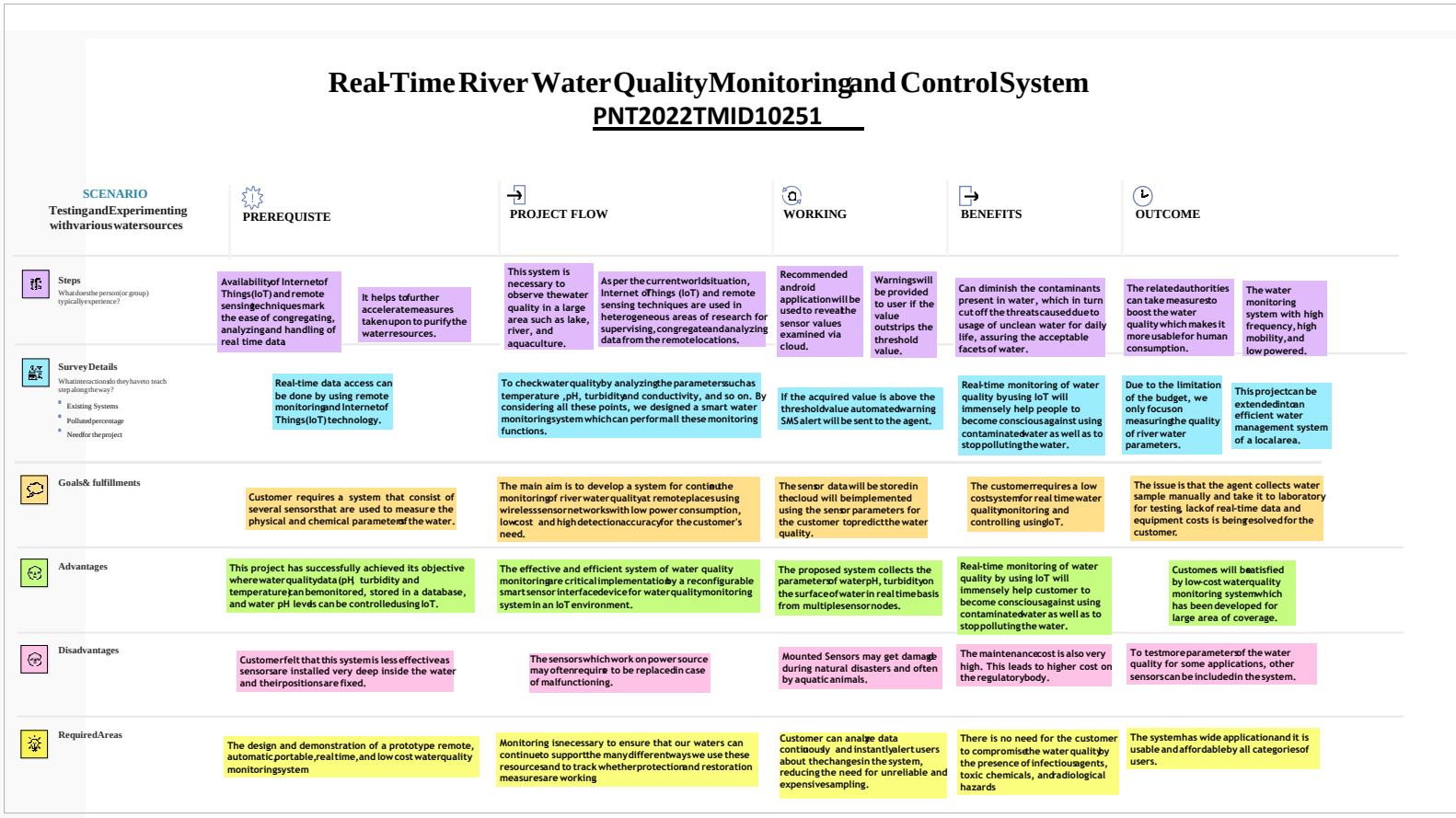


PROJECT DESCRIPTION:

- River water quality can be monitored by the web application.
- The web application and the user are interfaced.
- The pH level and the turbidity of the water can be monitored.
- If the water quality is not good then the authorities get alerted by the message.

Real-Time River Water Quality Monitoring and Control System

PNT2022TMID10251



5.2 Requirement Analysis

Date	08 October 2022
Team ID	PNT2022TMID10251
Project Name	Real Time River Water Quality Monitoring And Control System
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	1. Registration through Gmail. 2. Registration through mobile number.
FR-2	User Confirmation	1. Confirmation via Email. 2. Confirmation via OTP.
FR-3	User access	1. Accepting all the terms and conditions. 2. Confirmation of recaptcha.
FR-4	User mode	Online
FR-5	User alert	Alert SMS to the registered mobile number if the measured value crosses the threshold value.

Non-functional Requirements:

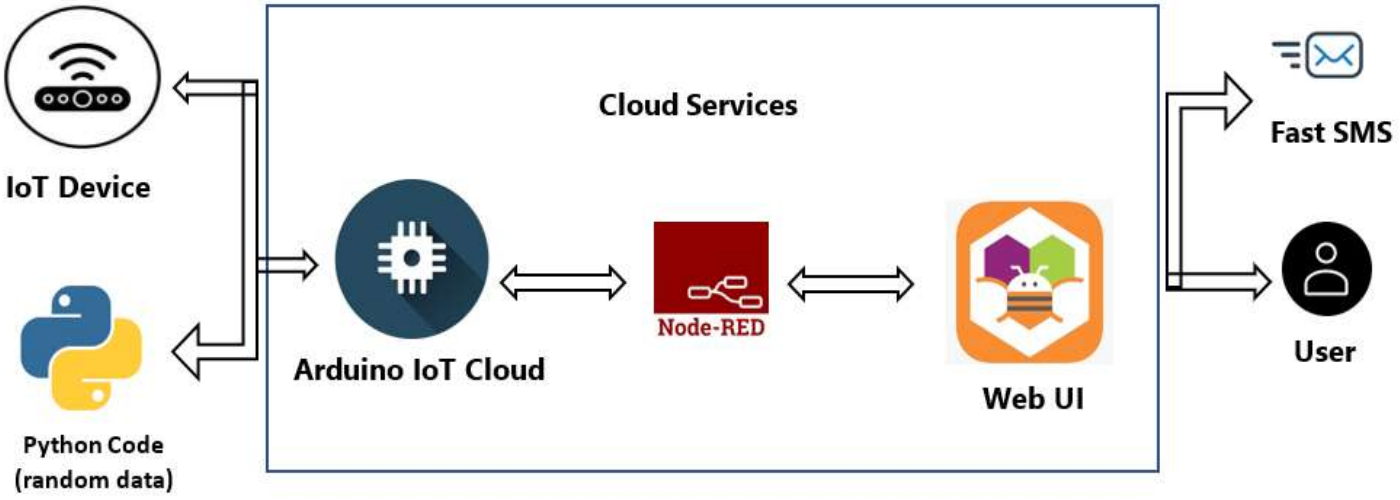
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	1. Easy to use. 2. Effective, Efficient, Engaging, Error tolerant. 3. Easy to learn.
NFR-2	**Security**	1. Accepting Terms and Conditions. 2. Confirmation via Email and OTP. 3. Confirmation via recaptcha. 4. Strong cryptography skills. 5. Software security architects also have experience with malware, intrusion detection and prevention and firewalls.
NFR-3	**Reliability**	1. Great user interface. 2. Software operating without failure while in a specified environment over a set duration of time.
NFR-4	**Performance**	Fast loading of the result time and high performance.
NFR-5	**Availability**	Easy installation.
NFR-6	**Scalability**	1. Optimizing SQL queries and implementing indexing strategies. 2. By building articles and authors into a single query, we can dramatically reduce the volume of queries we're running.

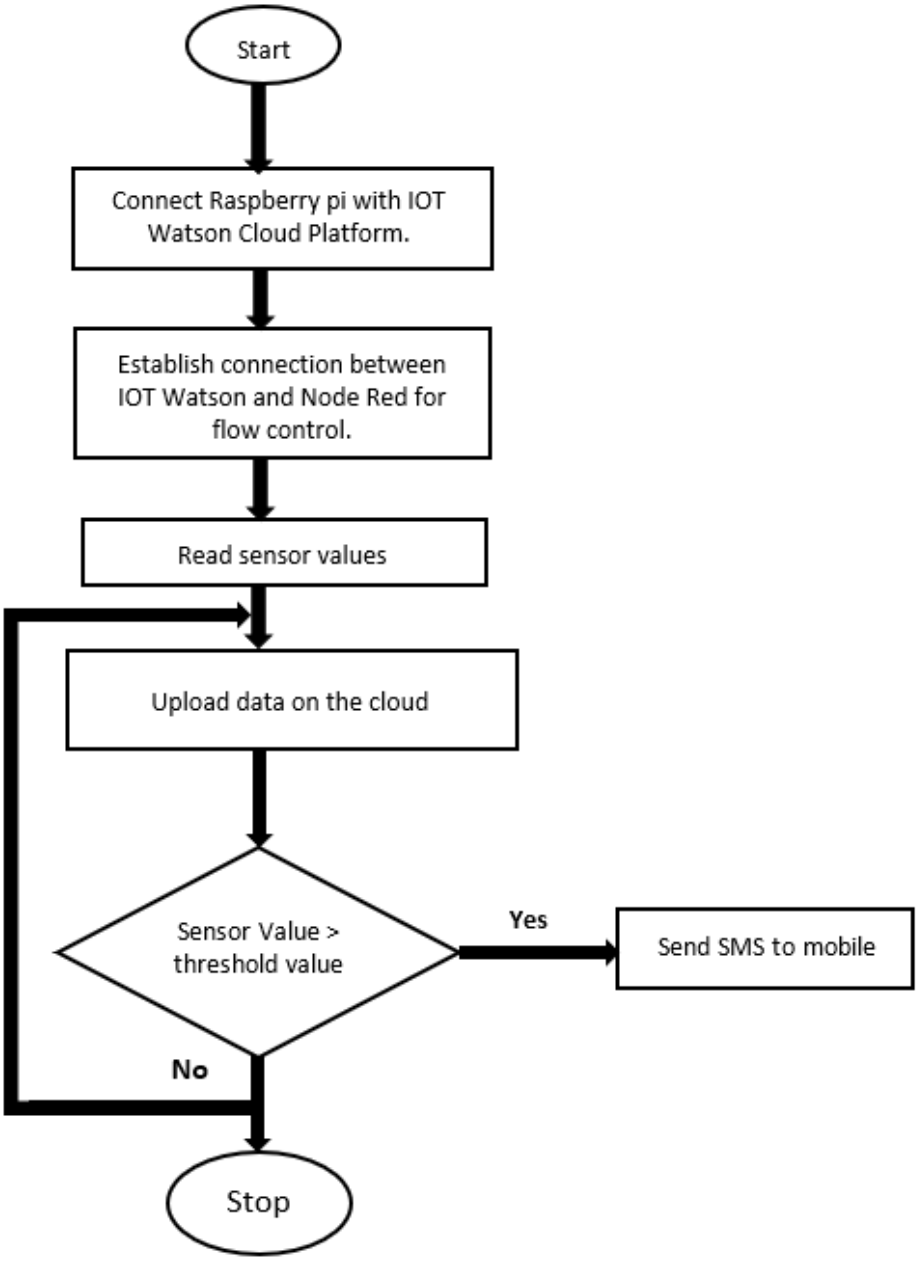
5.3 Data Flow Diagrams

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within



a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.


Start
Connect Raspberry pi with IOT Watson Cloud Platform.
Establish connection between IOT Watson and Node Red for flow control.
Read sensor values
Upload data on the cloud
Sensor Value > threshold value
Yes
Send SMS to mobile
No
Stop

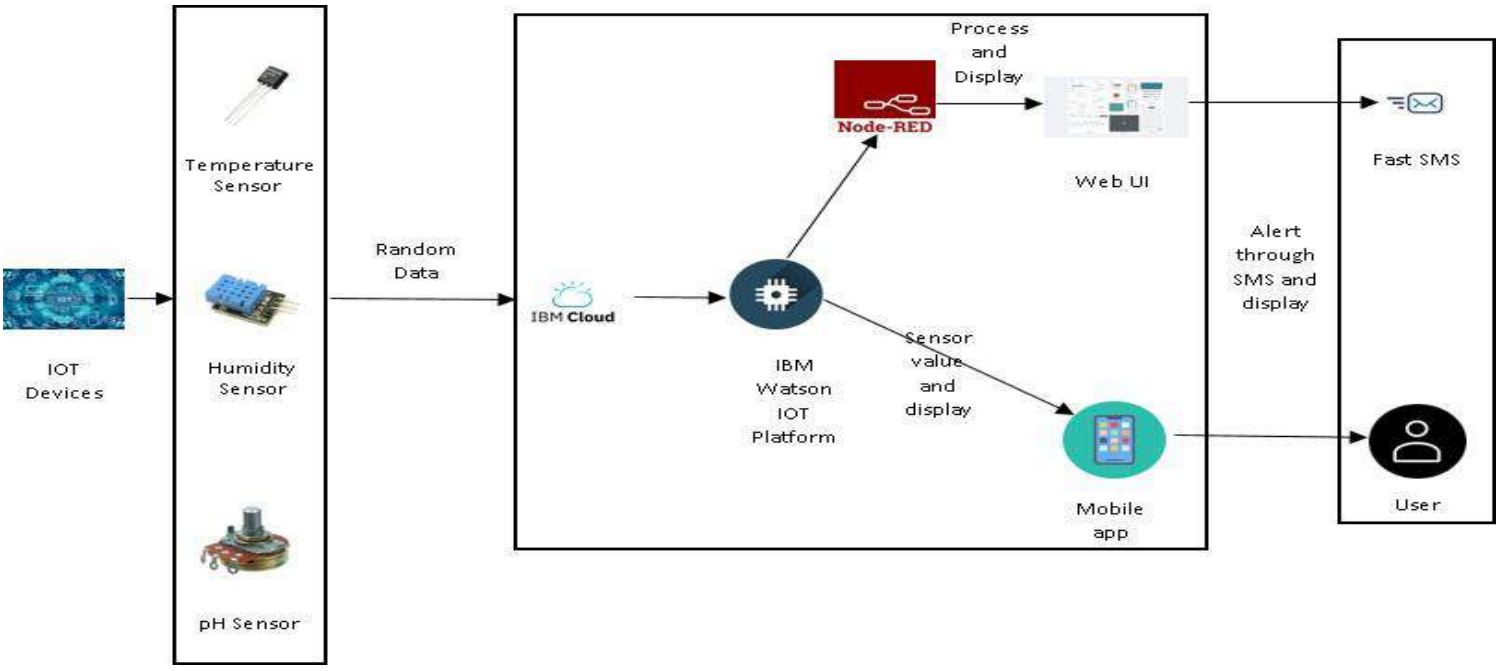
User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard.	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application.	I can receive confirmation email & click confirm.	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook.	I can register & access the dashboard with Facebook Login.	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail.		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password.		High	Sprint-1
	Dashboard	USN-6	As a user, I can login to the dashboard and search the access account and receive mail.			
Customer (Web user)	Login	UI	As a user, I need to create an account by providing all the necessary information.		Medium	Sprint-1
Customer Care Executive	Registration	UX	As a customer I need to register for application's care executive.	I can register and access the account.	High	Sprint-1
Administrator	Confirmation		As a customer I need to confirm mail once registered for the web user.		High	Sprint-1

5.4 Technology Stack

Date	08 November 2022
Team ID	PNT2022TMID10251
Project Name	Project - Real-Time River Water Quality Monitoring and Control System
Maximum Marks	4 Marks



Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Table-1 : Components & Technologies:

S.No	Component	Description	Technology

1.	User Interface	Web UI, Mobile App	Node – Red, Kubernetes, MIT mobile app inventor
2.	Application Logic-1	Generate random data	Python
3.	Application Logic-2	Generate random sensor data	IBM Watson IOT Platform
4.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant,
5.	External API-1	Send SMS to customer	Fast SMS API
6.	Infrastructure (Server / Cloud)	Application Deployment on Cloud	Cloud Foundry, Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	the open-source tools we utilized to create our project	Node – Red, IBM Cloudant, IBM Watson IOT Platform
2.	Security Implementations	Use of a login page with a user's unique username and password on a web interface optimized for mobile devices and computers with adjustable screen sizes	Password protection in MIT App
3.	Scalable Architecture	optimized for mobile devices and computers with adjustable screen sizes	Node – Red (Web UI)
4.	Availability	accessible to users through both a web UI and a mobile app	Node – Red(Web UI), MIT App(Mobile App)
5.	Performance	Give precise results and a prompt warning in the event of water contamination	Node – Red(Web UI), MIT App(Mobile App)

CHAPTER 6

PROJECT

PLANNING PHASE

6 PROJECT PLANNING PHASE

6.1 Milestone and Activity List

Date	04 November 2022
Team ID	PNT2022TMID10251
Project Name	Project - IOT Based Real-Time River Water Quality Monitoring and Control System
Maximum Marks	8 Marks

Title	Description	Date
Literature Survey on The Selected Project and Information Gathering	A Literature Survey is a compilation summary of research done previously in the given topic. Literature survey can be taken from books, research paper online or from any source	17 September 2022
Prepare Empathy Map	Empathy Map is a visualization tool which can be used to get a better insight of the customer.	18 September 2022
Ideation-Brainstorming	Brainstorming is a group problem solving session where ideas are shared, discussed and organized among the team members	19 September 2022
Define Problem Statement	A Problem Statement is a concise description of the problem or issues a project seeks to address. The problem statement identifies the current state, the desired future state and any gaps between the two	20 September 2022

Problem Solution Fit	This helps us to understand the thoughts of the customer their likes, behaviour, emotions etc.	26 September 2022
Proposed Solution	Proposed solution shows the current solution and it helps is going towards the desired result until it is achieved.	26 September 2022
Solution Architecture	Solution Architecture is a very complex process i.e., it has a lot of sub-processes and branches. It helps in understanding the components and features to complete our project.	26 September 2022
Customer Journey	It helps us to analyse from the perspective of a customer, who uses our project.	08 October 2022
Functional Requirement	Here functional and non-functional requirements are briefed. It has specific features like usability, security, reliability, performance, availability and scalability.	08 October 2022
Data Flow Diagrams	Data Flow Diagram is a graphical or visual representation using a standardised set of symbols and notations to describe business operations through data movement.	14 October 2022
Technology Architecture	Technology Architecture is a better defined version of solution architecture. It helps us analyse and understand various technologies that need to be implemented in the project.	14 October 2022

Prepare Milestone & Activity List	It helps us to understand and evaluate our own progress and accuracy so far.	22 October 2022
Sprint Delivery Plan	Sprint planning is an event in scrum that kicks off the sprint. The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved.	In Progress

6.2 Sprint Delivery Plan

Date	04 November 2022
Team ID	PNT2022TMID10251
Project Name	Project - IOT Based Real-Time River Water Quality Monitoring and Control System
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1		US-1	Creating IBM Cloud and using its services.	6	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-1		US-2	Configure the IBM cloud service and creating IoT platform.	4	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-1		US-3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, hence Launching IBM Watson IoT platform.	5	Low	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-1		US-4	In order to connect the IoT device to the IBM Cloud, create a device in the IBM Watson IoT Platform and get the device credentials.	5	Medium	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-2		US-1	Configure the connection security and create API keys that are used in the NODE-RED service for accessing the IBM IoT Platform.	10	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2		US-2	Create a Node-RED service.	10	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-3		US-1	Develop a python script to publish random sensor data such as temperature, turbidity and pH to the IBM IoT Platform.	7	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-3		US-2	After developing python code, commands are received just print the statements which represent the control of the devices.	5	Medium	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-3		US-3	Publish data to the IBM Cloud.	8	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-4		US-1	Create Web UI in Node-RED.	10	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER
Sprint-4		US-2	Configure the Node-RED flow to receive data from the IBM IoT Platform and also use Cloudant DB nodes to store the received sensor data in Cloudant DB.	10	High	GERALD RUBAN ASWATH S LAKSHMAN BENHER

Project Tracker, Velocity & Burndown Chart: (4 Marks)

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	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

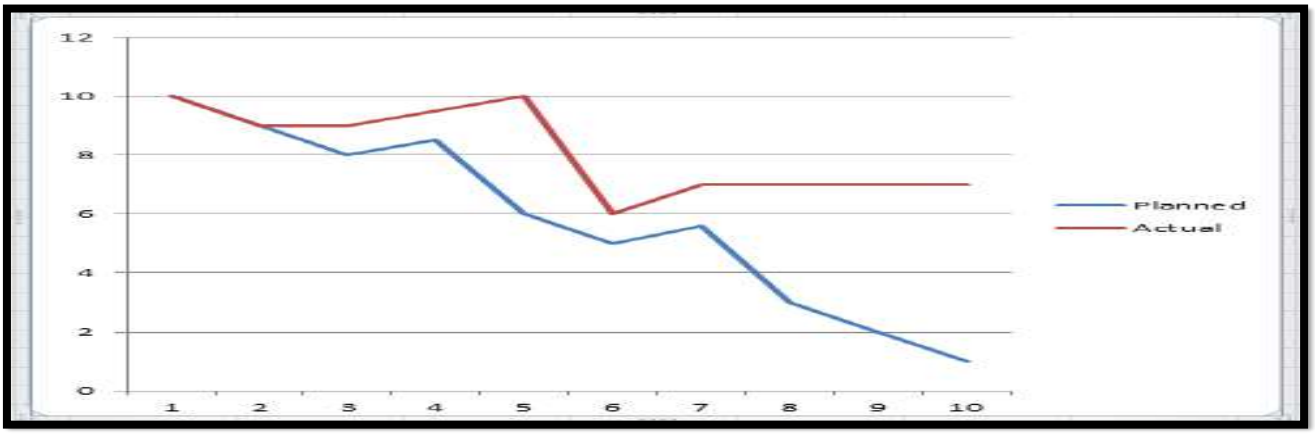
Velocity:

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

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

Burndown Chart:

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

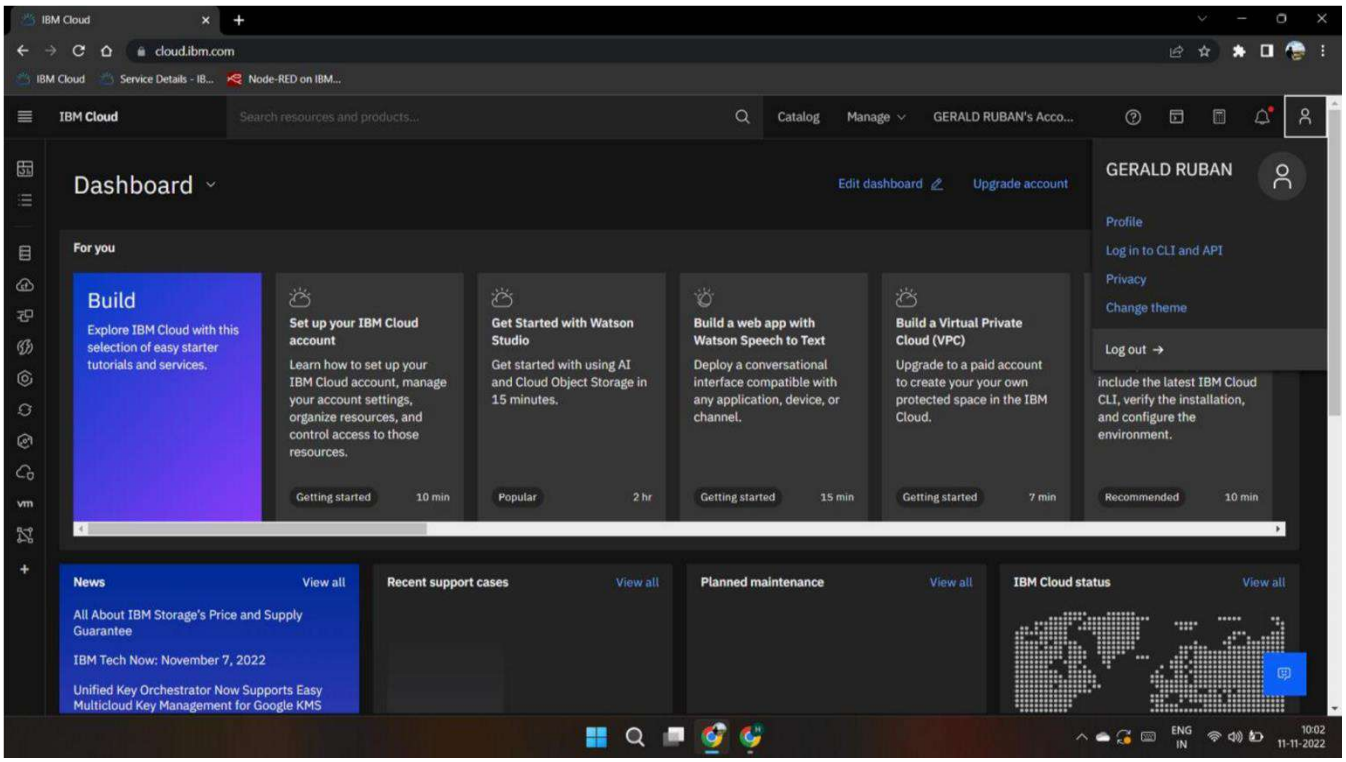


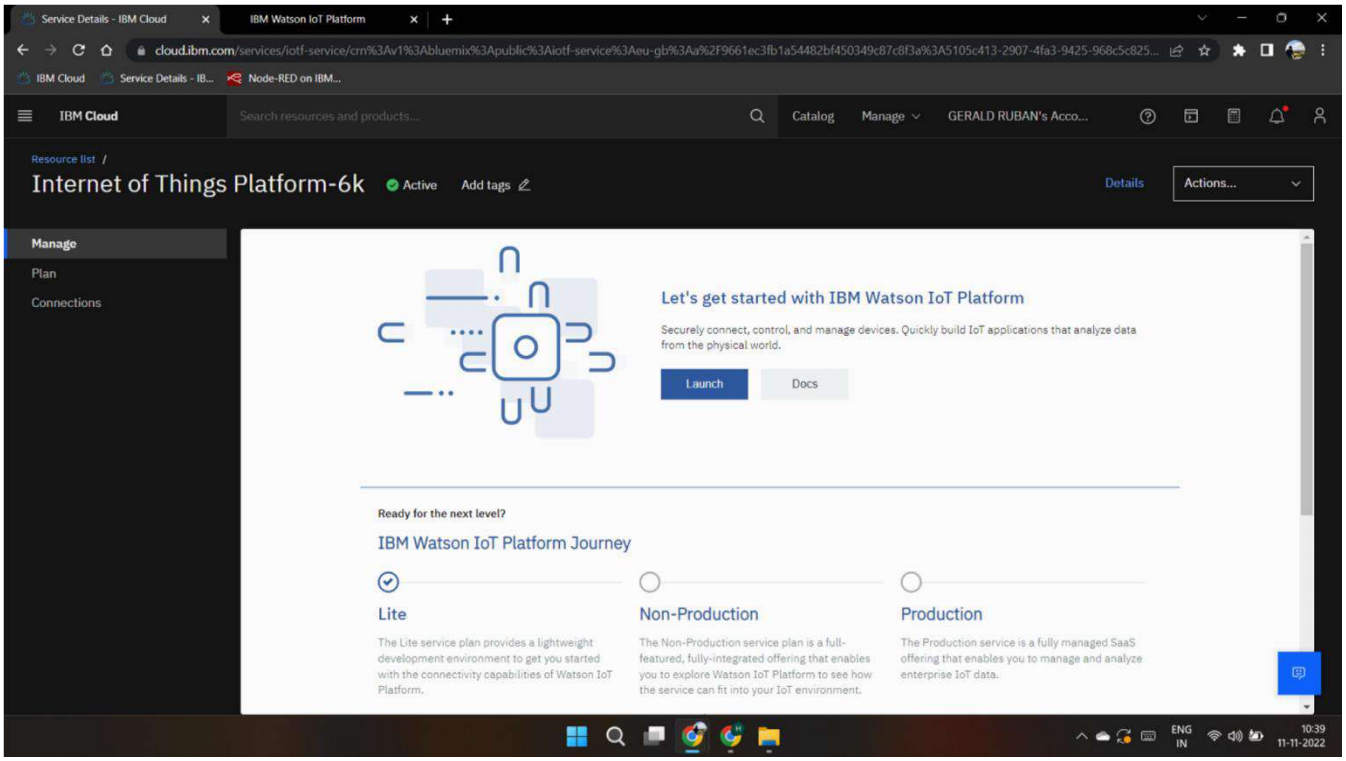
CHAPTER 7
PROJECT DEVELOPMENT PHASE

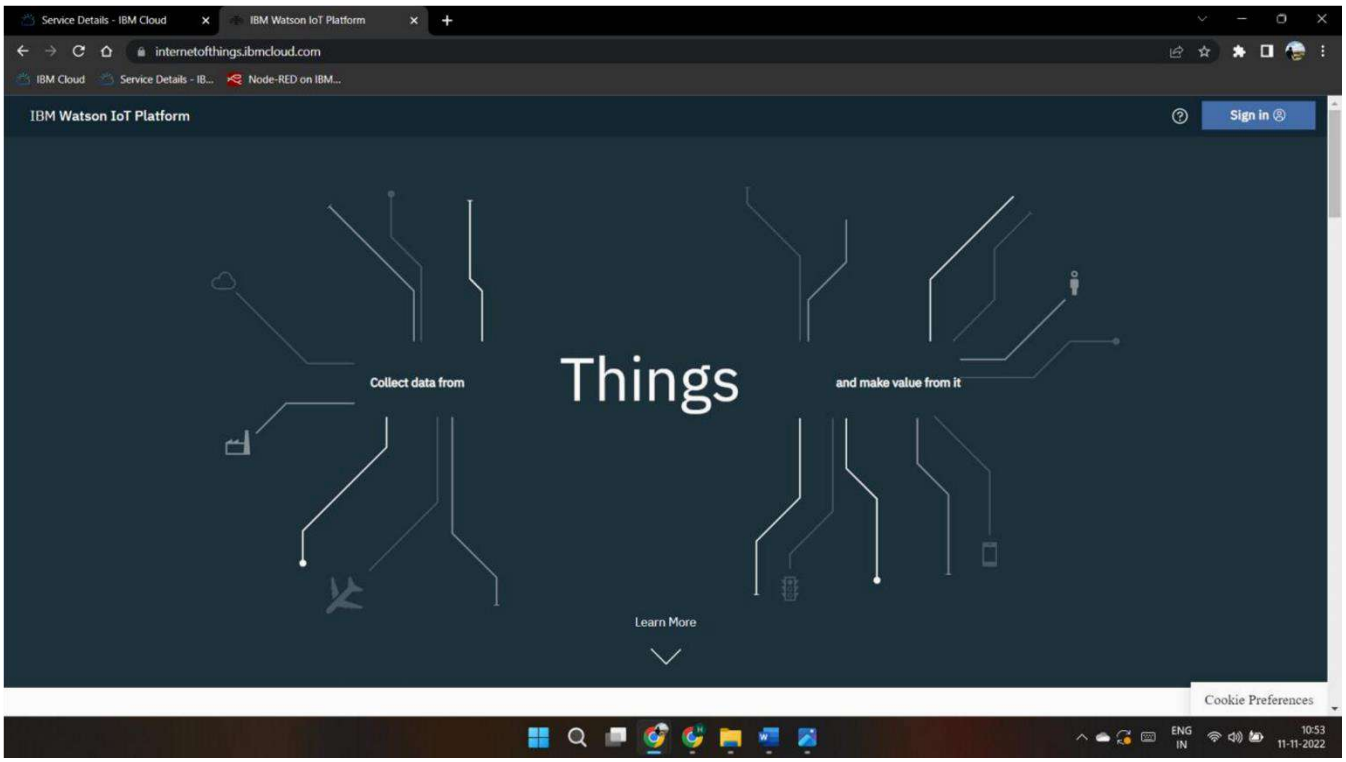
7.1 Project Development - Delivery of Sprint – 1

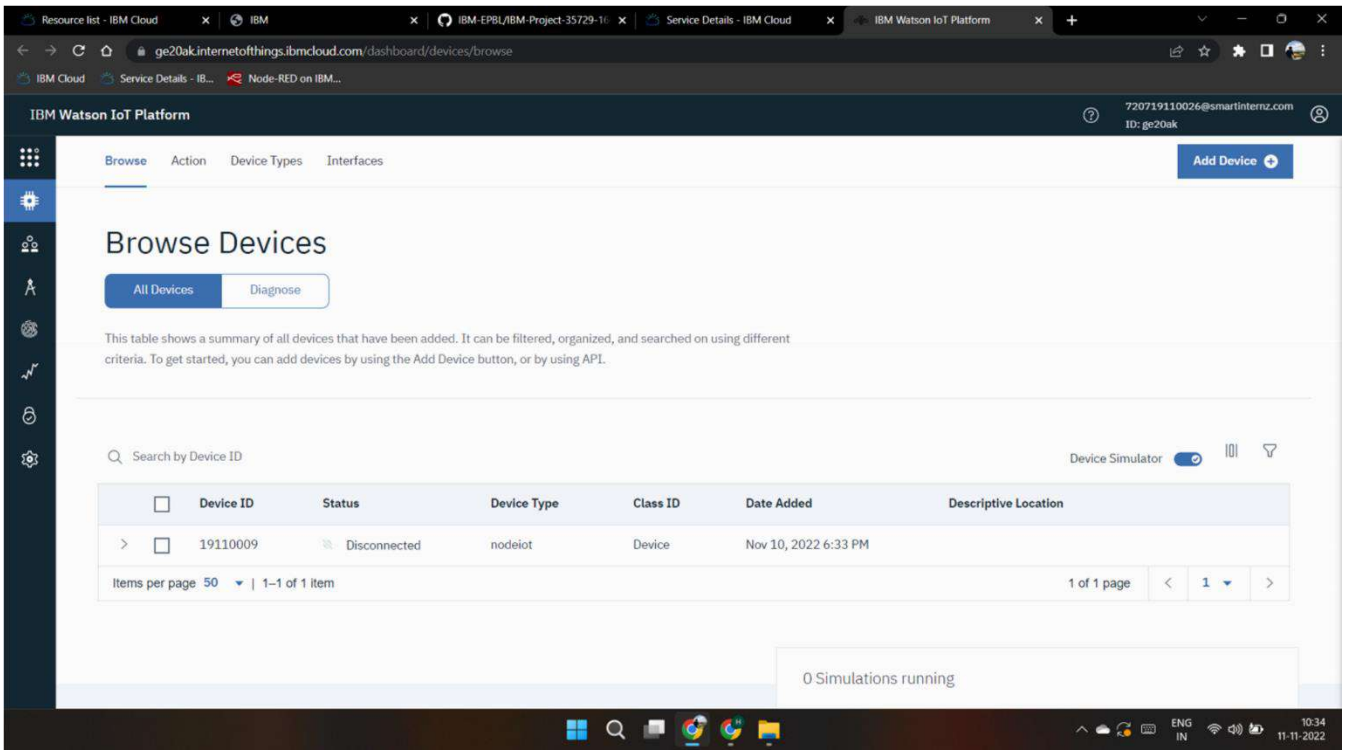
CREATE IBM WATSON IOT PLATFORM AND DEVICE

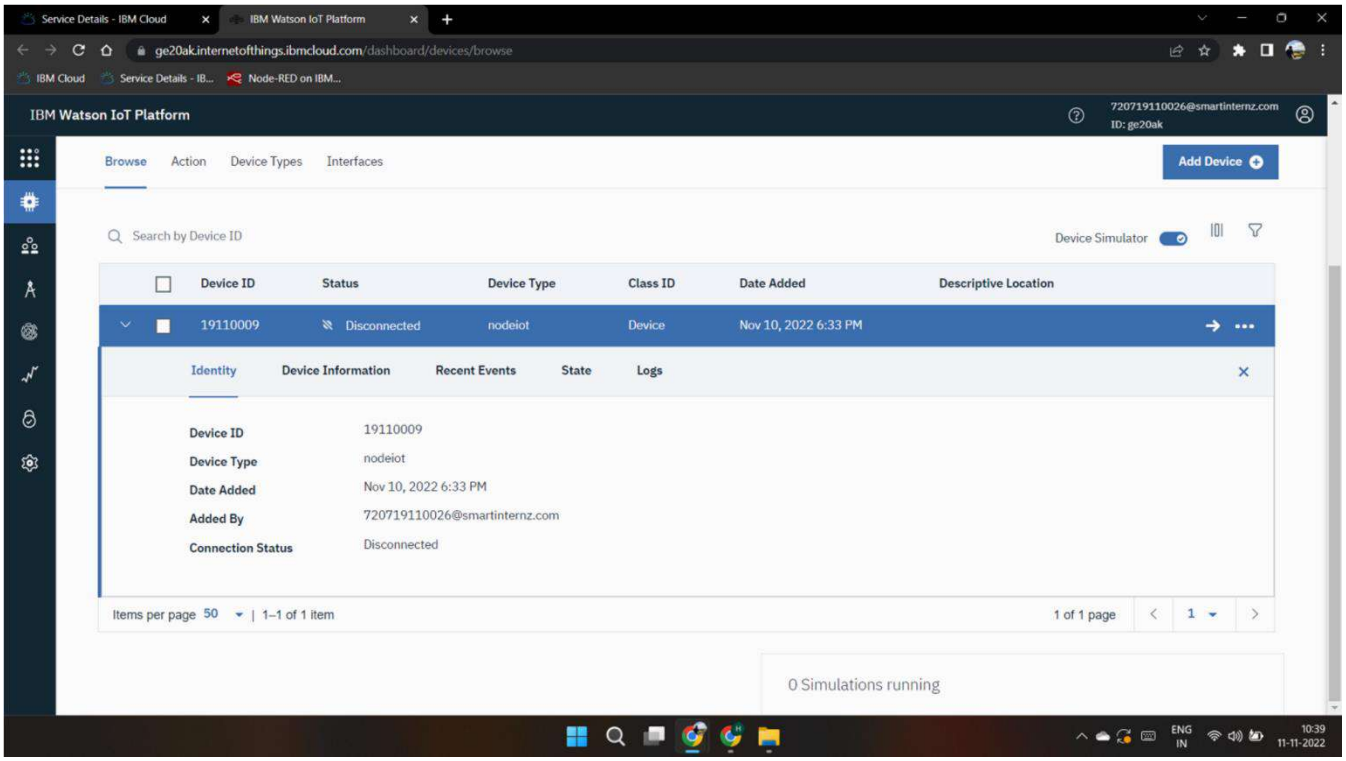
TEAM ID	PNT2022TMID10251
PROJECT TITLE	REAL TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM
TEAM LEADER	**GERALD RUBAN**
TEAM MEMBER 1	**ASWATH S**
TEAM MEMBER 2	**LAKSHMANA KUMAR**
TEAM MEMBER 3	**BENHER CHRISTOPHER A**


IBM Cloud
cloud.ibm.com
IBM Cloud
Search resources and products...
Catalog
Manage
GERALD RUBAN's Acco...
Dashboard
Edit dashboard
Upgrade account
For you
Build
Explore IBM Cloud with this selection of easy starter tutorials and services.
Set up your IBM Cloud account
Learn how to set up your IBM Cloud account, manage your account settings, organize resources, and control access to those resources.
Getting started
10 min
Get Started with Watson Studio
Get started with using AI and Cloud Object Storage in 15 minutes.
Popular
2 hr
Build a web app with Watson Speech to Text
Deploy a conversational interface compatible with any application, device, or channel.
Getting started
15 min
Build a Virtual Private Cloud (VPC)
Upgrade to a paid account to create your your own protected space in the IBM Cloud.
Getting started
7 min
include the latest IBM Cloud CLI, verify the installation, and configure the environment.
Recommended
10 min
GERALD RUBAN
Profile
Log in to CLI and API
Privacy
Change theme
Log out
News
View all
All About IBM Storage's Price and Supply Guarantee
IBM Tech Now: November 7, 2022
Unified Key Orchestrator Now Supports Easy Multicloud Key Management for Google KMS
Recent support cases
View all
Planned maintenance
View all
IBM Cloud status
View all
ENG
IN
10:02
11-11-2022


Service Details - IBM Cloud
IBM Watson IoT Platform
cloud.ibm.com/services/iotf-service/crn%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-gb%3Aa%2F9661ec3fb1a54482bf450349c87c8f3a%3A5105c413-2907-4fa3-9425-968c5c825...
IBM Cloud
Search resources and products...
Catalog
Manage
GERALD RUBAN's Acco...
Resource list /
Internet of Things Platform-6k
Active
Add tags
Details
Actions...
Manage
Plan
Connections
Let's get started with IBM Watson IoT Platform
Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.
Launch
Docs
Ready for the next level?
IBM Watson IoT Platform Journey
Lite
The Lite service plan provides a lightweight development environment to get you started with the connectivity capabilities of Watson IoT Platform.
Non-Production
The Non-Production service plan is a full-featured, fully-integrated offering that enables you to explore Watson IoT Platform to see how the service can fit into your IoT environment.
Production
The Production service is a fully managed SaaS offering that enables you to manage and analyze enterprise IoT data.
ENG
IN
10:39
11-11-2022


Service Details - IBM Cloud
IBM Watson IoT Platform
internetofthings.ibmcloud.com
IBM Cloud
Service Details - IB...
Node-RED on IBM...
IBM Watson IoT Platform
Sign in
Collect data from
Things
and make value from it
Learn More
Cookie Preferences
ENG
IN
10:53
11-11-2022


IBM Watson IoT Platform
720719110026@smartinternz.com
ID: ge20ak
Browse
Action
Device Types
Interfaces
Add Device
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
19110009
Disconnected
nodeiot
Device
Nov 10, 2022 6:33 PM
Items per page 50
1–1 of 1 item
1 of 1 page
0 Simulations running


IBM Watson IoT Platform
720719110026@smartinternz.com
ID: ge20ak
Browse
Action
Device Types
Interfaces
Add Device
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
19110009
Disconnected
nodeiot
Device
Nov 10, 2022 6:33 PM
Identity
Device Information
Recent Events
State
Logs
Device ID
19110009
Device Type
nodeiot
Date Added
Nov 10, 2022 6:33 PM
Added By
720719110026@smartinternz.com
Connection Status
Disconnected
Items per page 50
1–1 of 1 item
1 of 1 page
0 Simulations running

SPRINT -1

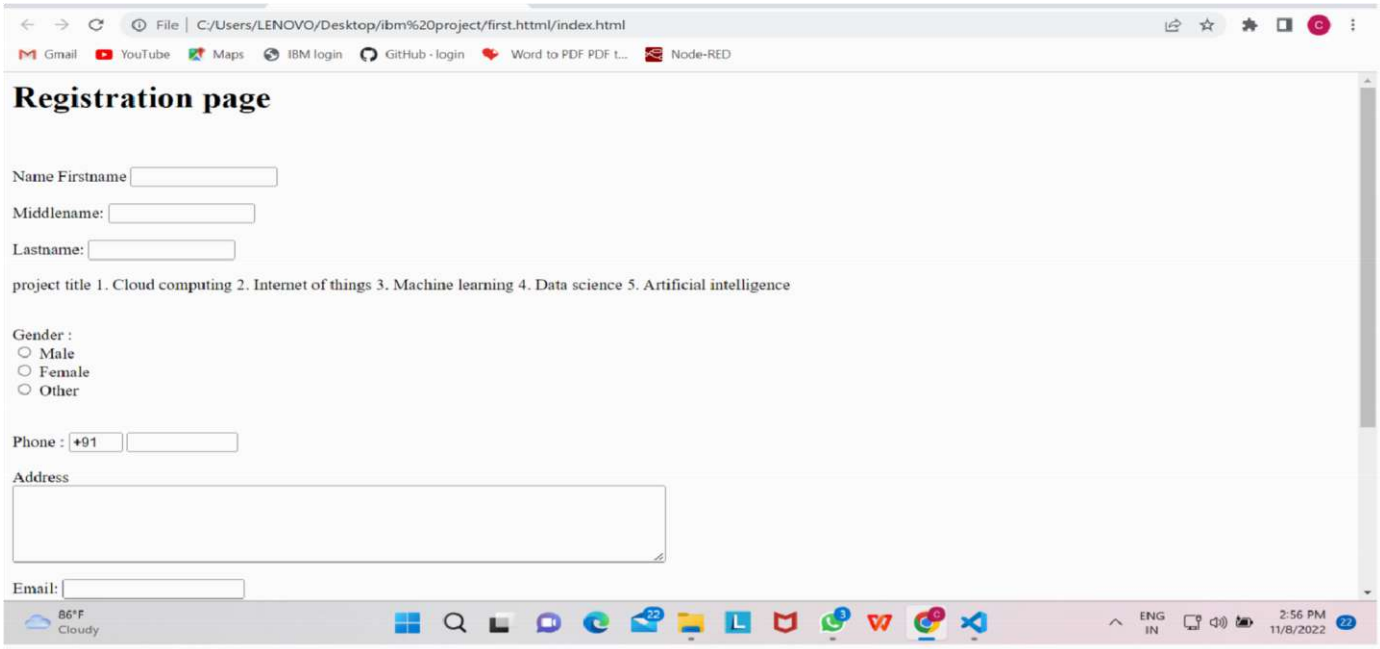
Registration Page

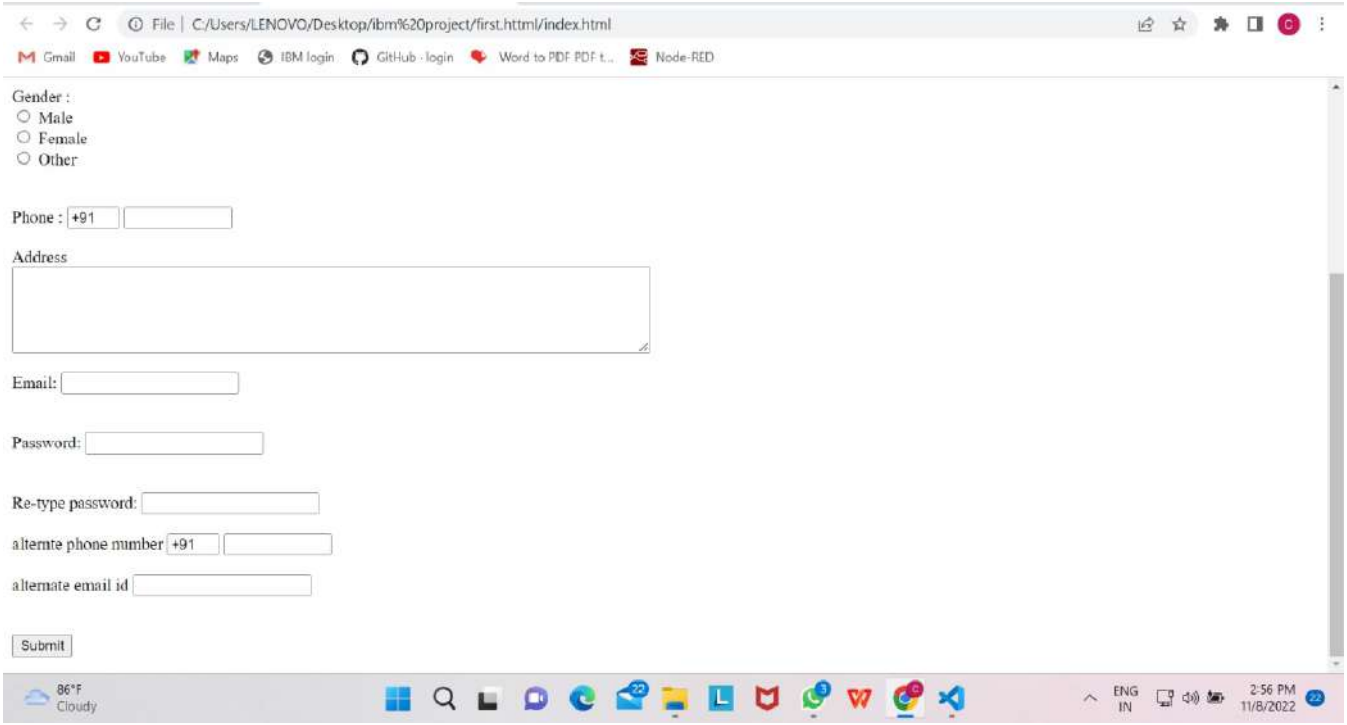
```
<html>
    <head>
    <title>
    Registration Page
    </title>
    <h1>Registration page</h1>
    </head>
    <body>
    <br>
    <br>
    <form>
    Name
    <label> Firstname </label>
    <input type="text" name="firstname" size="15"/> <br> <br>
    <label> Middlename: </label>
    <input type="text" 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>




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>

<input type="button" value="Submit"/>
 </form>
<body>
 <html>
```

SPRINT-1

Registration Page Output





7.2 Project Development - Delivery of Sprint – 2

SPRINT-2

LOGIN PAGE

HTML CODE:

```
<!DOCTYPE html>
<html>
<head>
 <h1> Real time River water quality monitoring and Control 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;
}
```

```css
/* 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) {
```

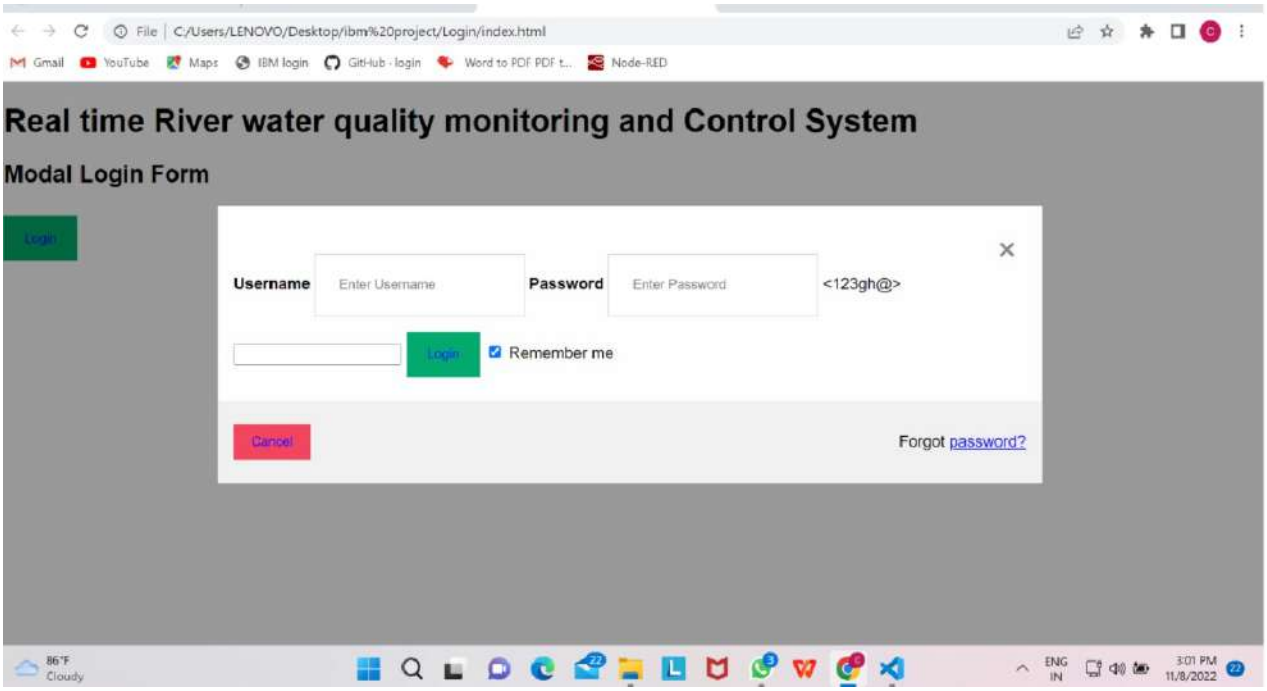
```
        modal.style.display = "none";
    }
}
</script>

</body>
</html>
```

SPRINT-2

LOGIN PAGE

OUTPUT

OUTPUT:



7.3 Project Development - Delivery of Sprint – 3

SPRINT-3

PYTHON CODE

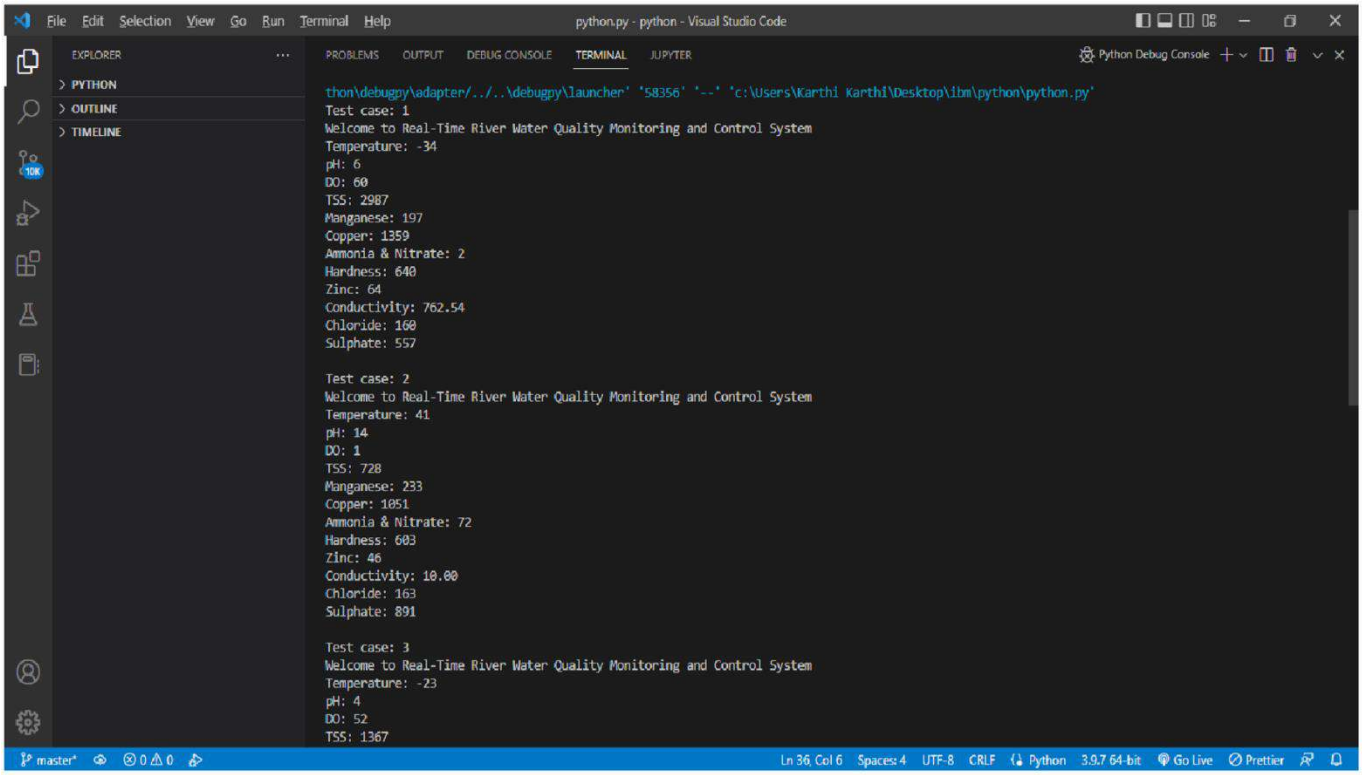
```
#importing Random function to generate the value
import random as rand


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,
```

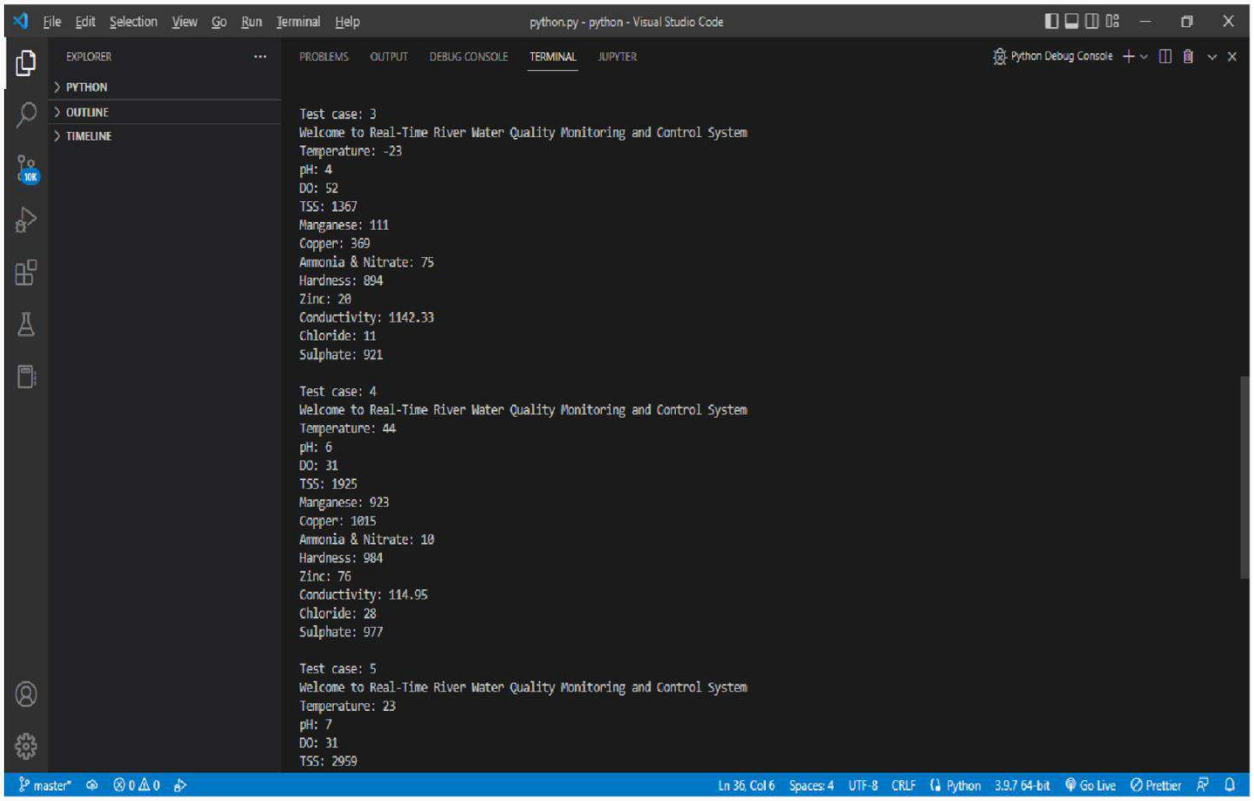
```
        "\nDO:", DO,
        "\nTSS:", TSS,
        "\nManganese:", Manganese,
        "\nCopper:", Copper,
        "\nAmmonia & Nitrate:",ammonia_Nitrate,
        "\nHardness:",Hardness,
        "\nZinc:", Zinc,
        "\nConductivity:", Conductivity,
        "\nChloride:", Chloride,
        "\nSulphate:", Sulphate, "\n"
    )
```

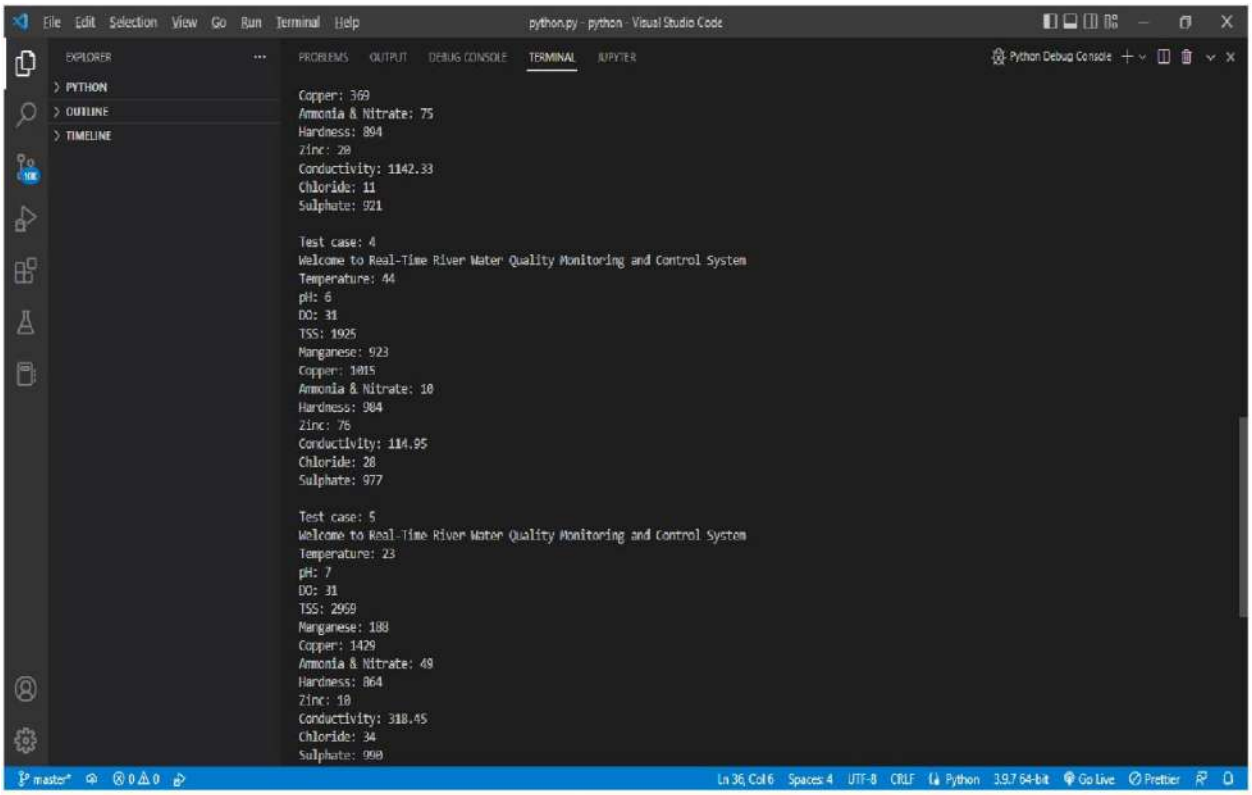
SPRINT-3

PYTHON CODE

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
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console
PYTHON
OUTLINE
TIMELINE
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
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console
PYTHON
OUTLINE
TIMELINE
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

7.4 Project Development - Delivery of Sprint – 4

SPRINT-4

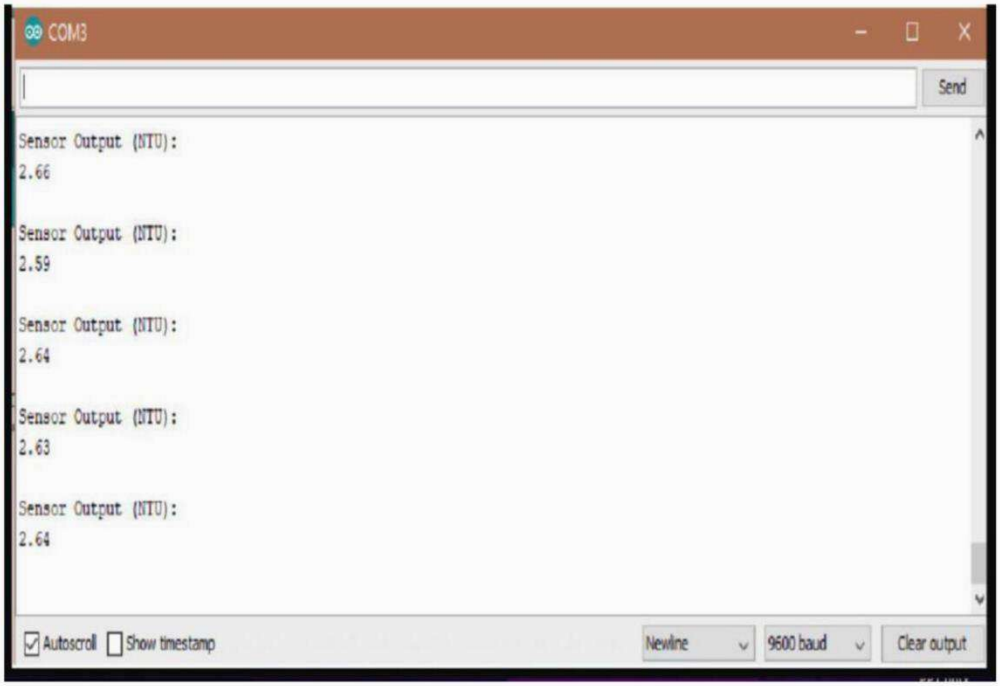
ARDUINO CODE

```
#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++)
{
 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);
}
```

ARDUINO OUTPUT



SPRINT-4

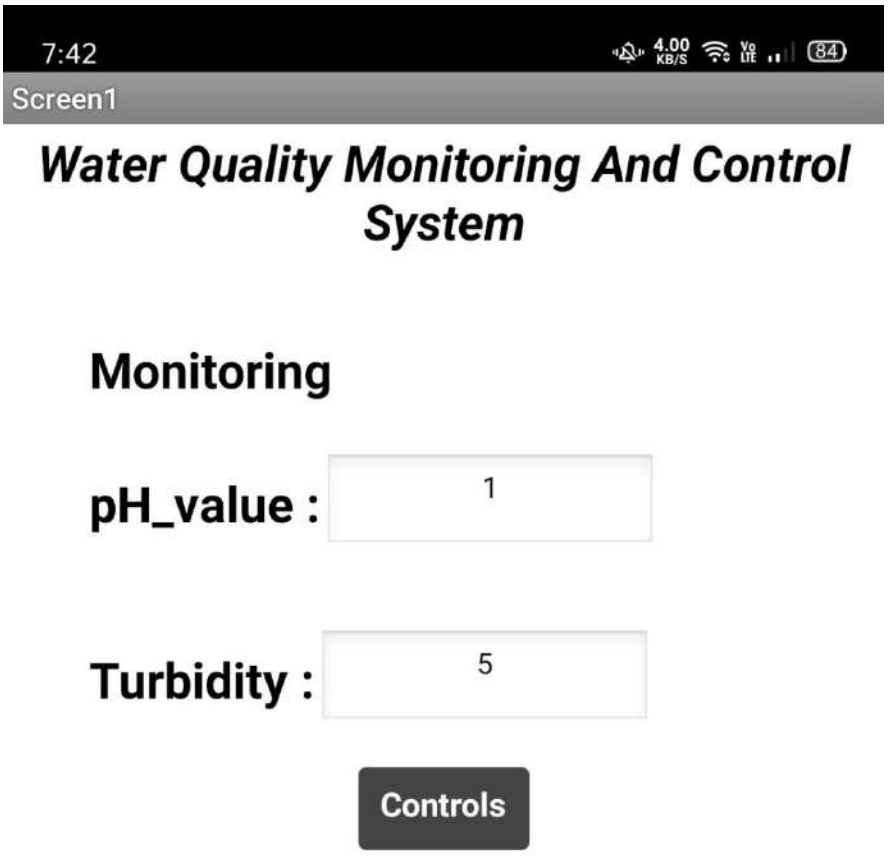
CODE IMPLEMENTATION

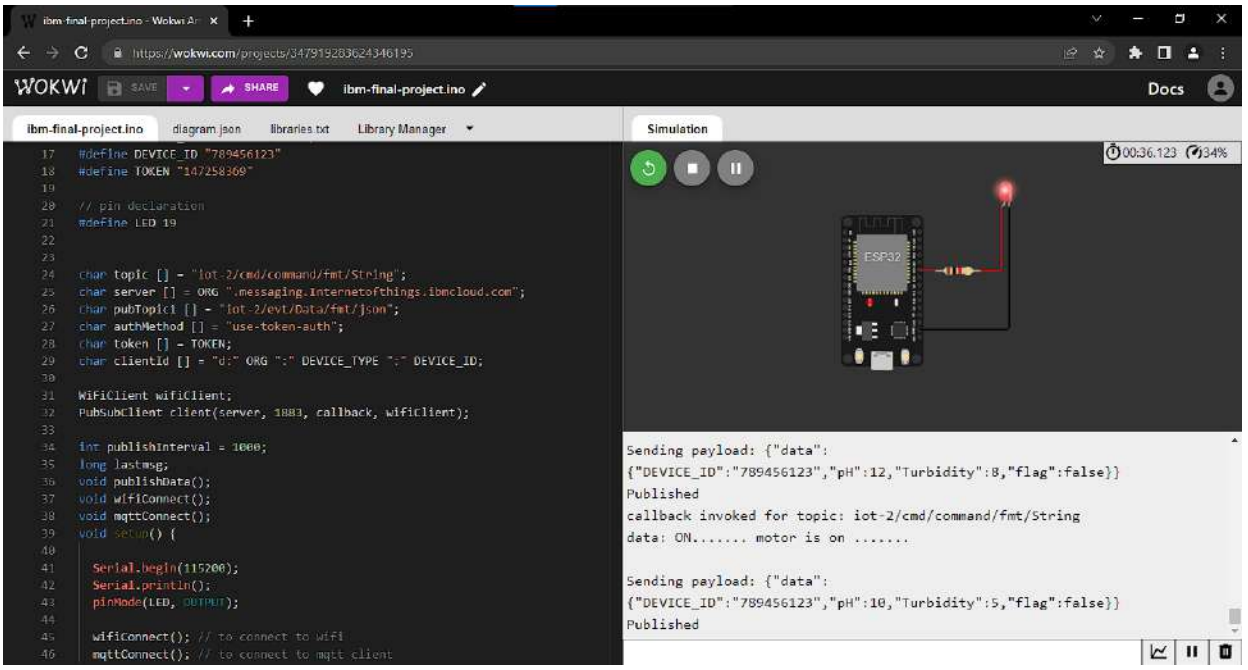
```
import serial

import time

import csv


import numpy as np

import matplotlib.pyplot as plt ser

= serial.Serial('/COM6',9600)

ser_bytes = ser.readline(10) print

(ser_bytes) ser.flushInput()


while True:

try:

ser_bytes = ser.readline()

decoded_bytes = float(ser_bytes[0:len(ser_bytes)-2].decode("utf-8")) print(decoded_bytes)

temp = float(decoded_bytes(1:3)) turb =

float(decoded_bytes(4:6)) pH =

float(decoded_bytes(6:8)) with

open("test_data.csv","a") as f: writer =
```

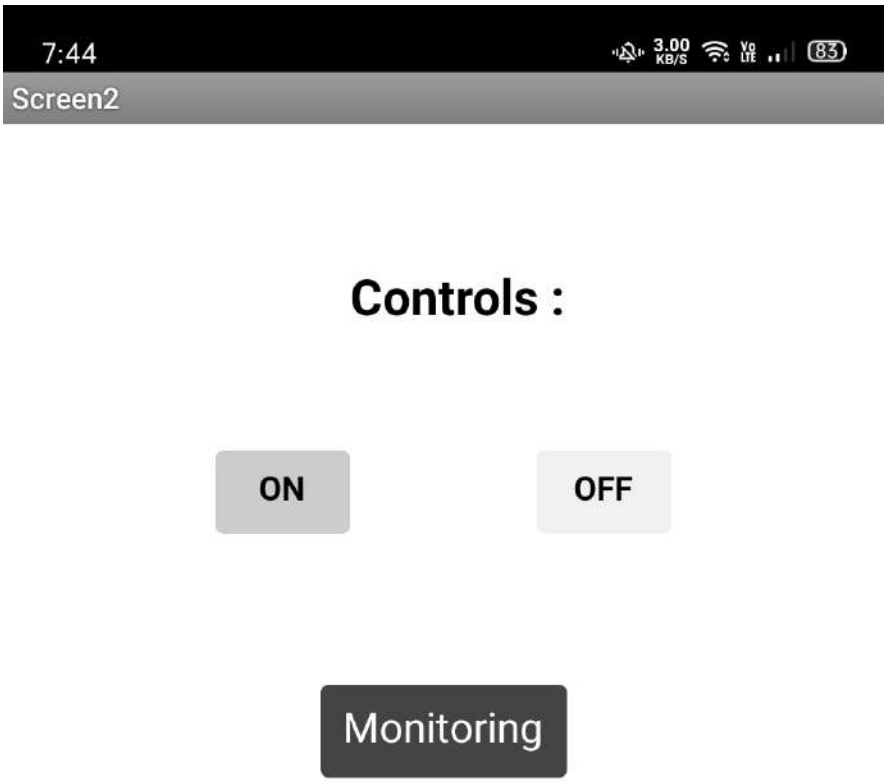
```
csv.writer(f,delimiter=",")
writer.writerow([time.time(),decoded_bytes])
except: print("Keyboard Interrupt") ser.close()

break()
    t = np.arange(0.0, 2.0, 0.01) s
    = 1 + np.sin(2*np.pi*t)
    plt.plot(t, s) plt.xlabel('time
    (s)') plt.ylabel('Celsisus (C)')
    plt.title('Temperature')
    plt.grid(True)
    plt.savefig("Temperature.png")
    plt.show()

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++)

{
```

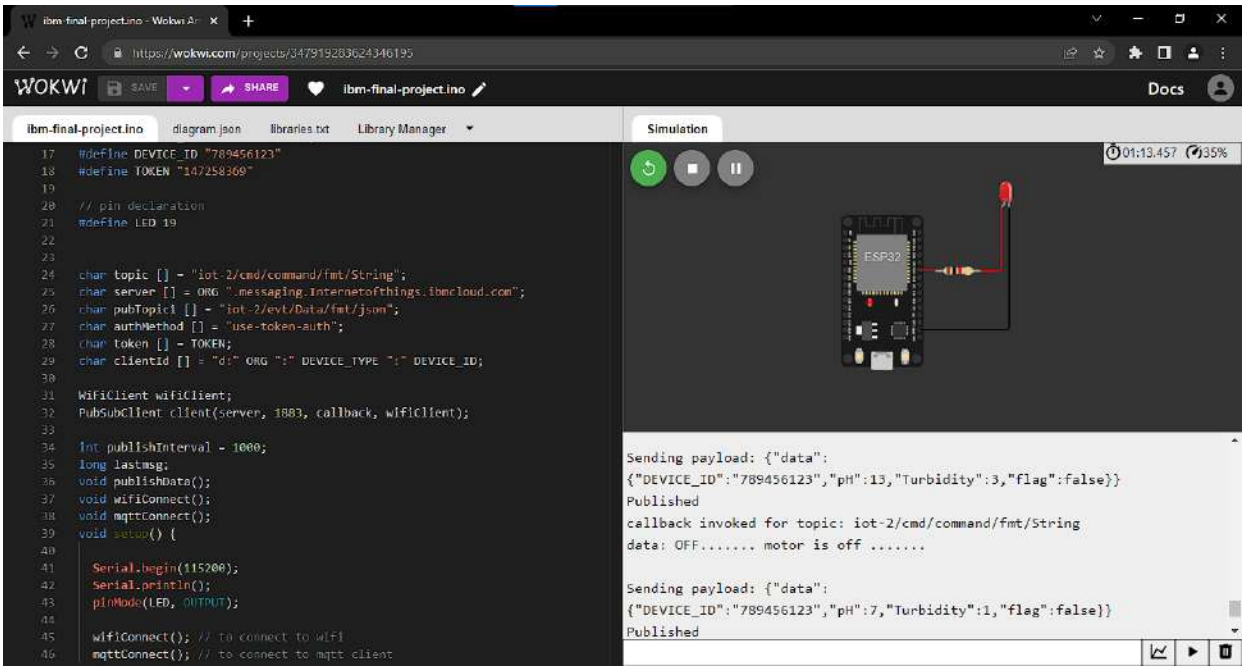
```
buf[i]=analogRead(analogInPin); delay(10);

}

for(int i=0;i<9;i++)

{

for(int j=i+1;j<10;j++)

{

if(buf[i]>buf[j])

{

    temp=buf[i];

    buf[i]=buf[j];

    buf[j]=temp;

}

n = 256

X = np.linspace(-np.pi, np.pi, 256, endpoint=True) C,S

= np.cos(X), np.sin(X)

plt.plot(X, C)

plt.plot(X,S) plt.show()
```

```
print ("Visualization of real time sensor Data.") print("/n")

while True:

try:

ser_bytes = ser.readline()

    decoded_bytes = float(ser_bytes[0:len(ser_bytes)2].decode("utf-8")) print(decoded_bytes) temp =

   float(decoded_bytes(1:3)) turb = float(decoded_bytes(4:6)) pH = float(decoded_bytes(6:8)) with

   open("test_data.csv","a") as f: writer = csv.writer(f,delimiter=",")

   writer.writerow([time.time(),decoded_byt es]) except: print("Keyboard Interrupt") ser.close()

break()

t = np.arange(0.0, 2.0, 0.01) s = 1 + np.sin(2*np.pi*t) plt.plot(t, s)
```


7:42
4.00 KB/S
84
Screen1
Water Quality Monitoring And Control System
Monitoring
pH_value :
1
Turbidity :
5
Controls


ibm-final-project.ino - Wokwi Ar
https://wokwi.com/projects/347919283624346195
WOKWi
SAVE
SHARE
ibm-final-project.ino
Docs
ibm-final-project.ino
diagram.json
libraries.txt
Library Manager
#define DEVICE_ID "789456123"
#define TOKEN "147258369"
// pin declaration
#define LED 19
char topic [] = "iot-2/cmd/command/fmt/String";
char server [] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1 [] = "iot-2/evt/Data/fmt/json";
char authMethod [] = "use-token-auth";
char token [] = TOKEN;
char clientId [] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback, wificlient);
int publishInterval = 1000;
long lastmsg;
void publishData();
void wifiConnect();
void mqttConnect();
void setup() {
Serial.begin(115200);
Serial.println();
pinMode(LED, OUTPUT);
wifiConnect(); // to connect to wifi
mqttConnect(); // to connect to mqtt client
Simulation
00:36.123
34%
ESP32
Sending payload: {"data":
{"DEVICE_ID":"789456123","pH":12,"Turbidity":8,"flag":false}}
Published
callback invoked for topic: iot-2/cmd/command/fmt/String
data: ON..... motor is on .....
Sending payload: {"data":
{"DEVICE_ID":"789456123","pH":10,"Turbidity":5,"flag":false}}
Published


7:44
3.00 KB/S
Screen2
Controls :
ON
OFF
Monitoring


ibm-final-project.ino - Wokwi Ar
https://wokwi.com/projects/347919283624346195
WOKWi
SAVE
SHARE
ibm-final-project.ino
Docs
ibm-final-project.ino
diagram.json
libraries.txt
Library Manager
Simulation
01:13.457
35%
ESP32
#define DEVICE_ID "789456123"
#define TOKEN "147258369"
// pin declaration
#define LED 19
char topic [] = "iot-2/cmd/command/fmt/String";
char server [] = ORG ".messaging.internetofthings.ibmcloud.com";
char pubTopic1 [] = "iot-2/evt/Data/fmt/json";
char authMethod [] = "use-token-auth";
char token [] = TOKEN;
char clientId [] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback, wifiClient);
int publishInterval = 1000;
long lastmsg;
void publishData();
void wifiConnect();
void mqttConnect();
void setup() {
Serial.begin(115200);
Serial.println();
pinMode(LED, OUTPUT);
wifiConnect(); // to connect to wifi
mqttConnect(); // to connect to mqtt client
Sending payload: {"data":
{"DEVICE_ID":"789456123","pH":13,"Turbidity":3,"flag":false}}
Published
callback invoked for topic: iot-2/cmd/command/fmt/String
data: OFF..... motor is off .....
Sending payload: {"data":
{"DEVICE_ID":"789456123","pH":7,"Turbidity":1,"flag":false}}
Published

CONCLUSION

Real-time monitoring of water quality by using IoT will immensely help people to become conscious against using contaminated water as well as to stop polluting the water. The research is conducted focusing on monitoring river water quality in real-time. Therefore, IoT is appeared to be a better solution as reliability, scalability, speed, and persistence can be provided.

IoT devices use various types of sensors to collect data about turbidity, ORP, temperature, pH, conductivity, etc. Of river water continuously. Also, IoT devices have capability to stream the array of collected data wirelessly to the remote Data Aggregator Server in the cloud. Moreover, the volume of semi structured data increases with time in such a velocity that only the Big Data Analytics applications can efficiently store and analyze the data constantly. Due to the limitation of the budget, we only focus on measuring the quality of river water parameters. This project can be extended into an efficient water management system of a local area. Moreover, other parameters which wasn't the scope of this project such as total dissolved solid, chemical oxygen demand and dissolved oxygen can also be quantified. So the additional budget is required for further improvement of the overall system.

REFERENCES

1. K. S. Adu-Manu, C. Tapparello, W. Heinzelman, F. A. Katsriku, and J.-D. Abdulai, "Water quality monitoring using wireless sensor networks: Current trends and future research directions," ACM Transactions on Sensor Networks (TOSN), vol. 13, p. 4, 2017.
2. B. Chen, Y. Song, T. Jiang, Z. Chen, B. Huang, and B. Xu, "Real-time estimation of population exposure to PM2.5 using mobile- and station-based big data," Int J Environ Res Public Health, vol. 15, Mar 23 2018
3. B. Paul, "Sensor based water quality monitoring system," BRAC University, 2018.
4. K. Andersson and M. S. Hossain, "Smart Risk Assessment Systems using Belief-rule-based DSS and WSN Technologies", in 2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace and Electronic Systems, VITAE 2014 : Co-located with Global Wireless Summit, Aalborg, Denmark 11-14 May 2014, 2014.
5. S. Thombre, R. U. Islam, K. Andersson, and M. S. Hossain, "IP based Wireless Sensor Networks : performance Analysis using Simulations and Experiments", Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, vol. 7, no. 3, pp. 53–76, 2016
6. K. Andersson and M. S. Hossain, "Heterogeneous Wireless Sensor Networks for Flood Prediction Decision Support Systems", in 2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) : 6th IEEE INFOCOM International Workshop on Mobility Management in the Networks of the Future World, 2015, pp. 133–137.
7. S. Thombre, R. U. Islam, K. Andersson, and M. S. Hossain, "Performance Analysis of an IP based Protocol Stack for WSNs", in Proceedings of the 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2016, pp. 691–696
8. M. Z. Abedin, A. S. Chowdhury, M. S. Hossain, K. Andersson, and R. Karim, "An Interoperable IP based WSN for Smart Irrigation Systems", presented at the 14th Annual IEEE Consumer Communications & Networking Conference, Las Vegas, 8-11 January 2017, 2017.
9. M. Z. Abedin, S. Paul, S. Akhter, K. N. E. A. Siddiquee, M. S. Hossain, and K.

Andersson, "Selection of Energy Efficient Routing Protocol for Irrigation Enabled by Wireless Sensor Networks", in Proceedings of 2017 IEEE 42nd Conference on Local Computer Networks Workshops, 2017, pp. 75–81.

10. N. Chilamkurti, S. Zeadally, A. Vasilakos, and V. Sharma, "Cross-layer support for energy efficient routing in wireless sensor networks," Journal of Sensors, vol. 2009, 2009.
11. H. R. Maier and G. C. Dandy, "The use of artificial neural networks for the prediction of water quality parameters," Water resources Research, vol. 32, pp. 1013-1022, 1996
12. T. Mahmud, K. N. Rahman, and M. S. Hossain, "Evaluation of Job Offers Using Evidential Reasoning", Global Journal of Computer Science and Technology, Vol. 13, No. 6, 2013, pp. 41-50.
13. M. S. Hossain, K. Andersson, and S. Naznin, "A Belief Rule Based Expert System to Diagnose Measles under Uncertainty", in Proceedings of the 2015 International Conference on Health Informatics and Medical Systems (HIMS'15), 2015, pp. 17–23.
14. M. S. Hossain, S. Rahaman, A.-L. Kor, K. Andersson, and C. Pattison, "A Belief Rule Based Expert System for Datacenter PUE Prediction under Uncertainty", IEEE Transactions on Sustainable Computing, vol. 2, no. 2, pp. 140–153, 2017.
15. M. Z. Abedin, N. A. Chandra, D. Prashengit, D. Kaushik, and M. S. Hossain, "License Plate Recognition System Based On Contour Properties and Deep Learning Model" in Proceedings of the IEEE Region 10 Humanitarian Technology Conference, 2017, pp. 590-593.