Real-Time River Water Quality Monitoring and Control System

TEAM ID	PNT2022TMID45388
PROJECT NAME	real-time river water quality monitoring and control
DATE	18 november 2022

TEAM MEMBERS:

Team Leader : ABSARKHAN A
Team member : ASLAM BASHA S
Team member : BALA MURUGAN M

INTRODUCTION:

PROJECT OVERVIEW:

Real Time-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. 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.

PURPOSE:

The uniqueness of our project is to obtain the water monitoring system with high frequency, high mobility, and low powered. Therefore, our proposed system will *immensely help the populations to become conscious against contaminated water as well as to stop polluting the water.*

LITERATURE SURVEY:

LITERATURE PAPER TITLE	AUTHOR	OBJECTIVE
IOT based real timeriver water qualitymonitoring system(August 19,2019)	*Elsevier B.V.*	*The main objective of this paper is to access data by the remotemonitoring and IOTtechnology.If the acquired value is abovethe threshold value automated warning SMS alert will send to the agent*
Design and Development of Real-Time Water Quality Monitoring System (October 18,2019)	*Meghana M,Kiran Kumar B MDivyaKiran Ravikant Verma*	*This paperpresents a systemthat is developed to measure the parameters of water such as turbidity dissolved solvents PH andtemperature.The sensors areinterfaced with ArduinoUNO and raspberry Pi for data processing and transmission.This data is transmitted through Wi-Fito the remoteplace*

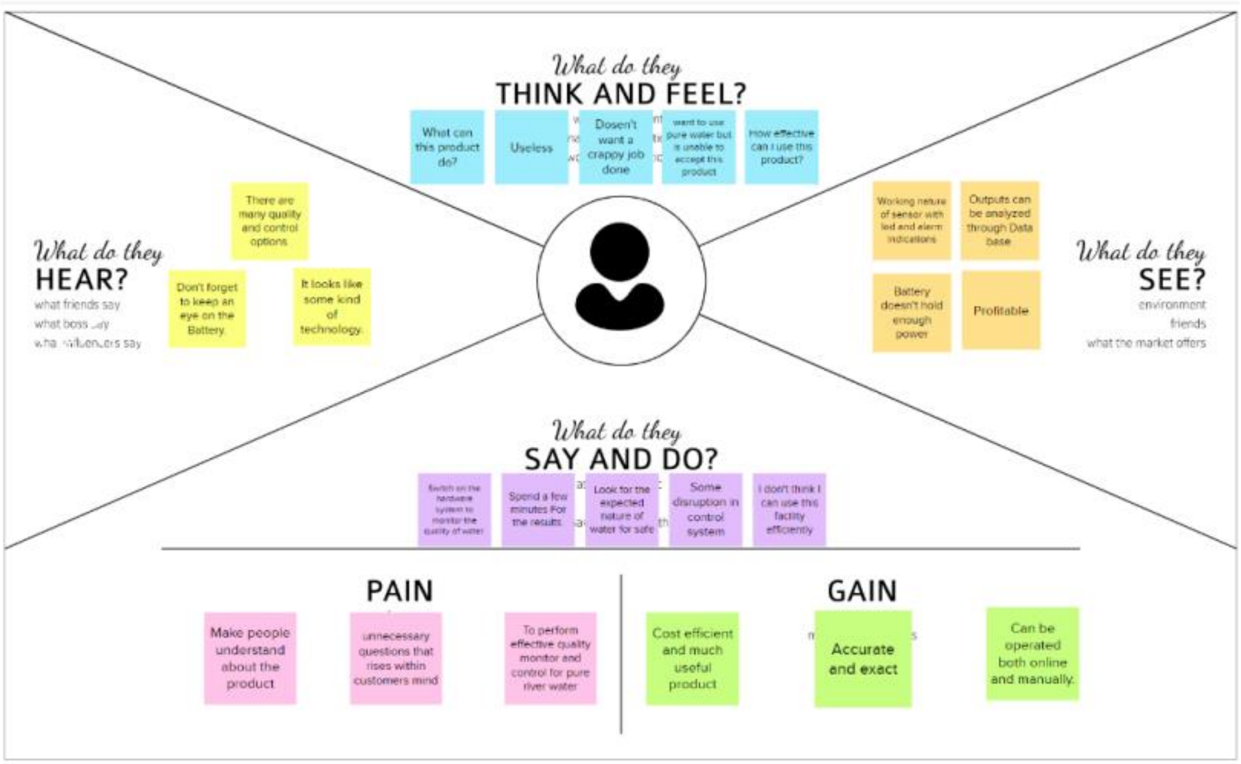
Ultrasonic as a greenchemistry for bacterial and algal controlin drinking watertreatment source (20 September2020)	*Nourhan F.AliZenat M.kamel S.Z.Wahba*	*The treatment process is done using ultrasonic waves at a frequency of 20,40 and60KHz at different time intervals namely 15,30,45 and 60 minutes*
Improved Cyanobacteria Removalfrom Harmful Algae Blooms by Two-Cycle,Low-Frequency, Low- Density, and Short- Duration Ultrasonic Radiation(29 August 2020)	*Haocai Huang Gang WuChaowu Sheng Wu Jiannan Danhua Li Hangzhou Wang*	*This paperhas a proposed cyanobacteria removal method based on two applications oflow frequency, low density andshort duration and ultra sonic radiation for calculating theeffectiveness of ultrasonicradiation is done by algaeremoval rate/ultrasonic dosage*

PROBLEM STATEMENT DEFINITION:

The most frequent water quality issue is due to the high content of iron and magnesium content in raw water of treated water. Water quality disorders occur as a result of changes in the color of the water that turns yellow to a dark brown color. The color change is due to action chemical reactions that are used in the water treatment process at the Treatment Plant (Kasan, 2006). This water treatment diagnostic and auditing process still uses manual methods, where water will be measured and the quality index will be clinically measured inside the laboratory. Besides, low pH levels cause fish killed by stressing animals system and causing physical damage, which in turn makes them more vulnerable to disease. Water is the most important source of survival for all beings on earth. Therefore, water safety issues are a very important issue. Consumer complaints and reports made by the relevant government departments indicate that consumers are dissatisfied with the quality of water supplied . Hence, a concept in which equipment, machines, sensors and devices are connected to the Internet and there is *data collection and transfer through the network developed to follow the river water quality index. Integration of the elements of sustainability and IR4.0 through the Internet of Thing by adopting electronic and Internet applications of Thing has a very positive impact to refresh the approach to lesions in Malaysia. This tool is produced by reading recording function to evaluate the quality level through the special sensor of Internet of Thing. And the data obtained can be used for analysis, recording, display and it is a warning to the JPS about the health status of the river that is chemically dissolved.*

IDEATION AND PROPOSED SOLUTION:

EMPATHYMAP AND CANVAS:



What do they
THINK AND FEEL?

What can this product do?

Useless

Dosen't want a crappy job done

want to use pure water but is unable to accept this product

How effective can i use this product?

What do they
HEAR?
what friends say
what boss say
what influencers say

There are many quality and control options

Don't forget to keep an eye on the Battery.

It looks like some kind of technology.

What do they
SEE?
environment
friends
what the market offers

Working nature of sensor with led and alarm indications

Outputs can be analyzed through Data base

Battery doesn't hold enough power

Profitable

What do they
SAY AND DO?

Switch on the hardware system to monitor the quality of water

Spend a few minutes For the results

Look for the expected nature of water for safe

Some disruption in control system

I don't think I can use this facility efficiently

PAIN

Make people understand about the product

unnecessary questions that rises within customers mind

To perform effective quality monitor and control for pure river water

GAIN

Cost efficient and much useful product

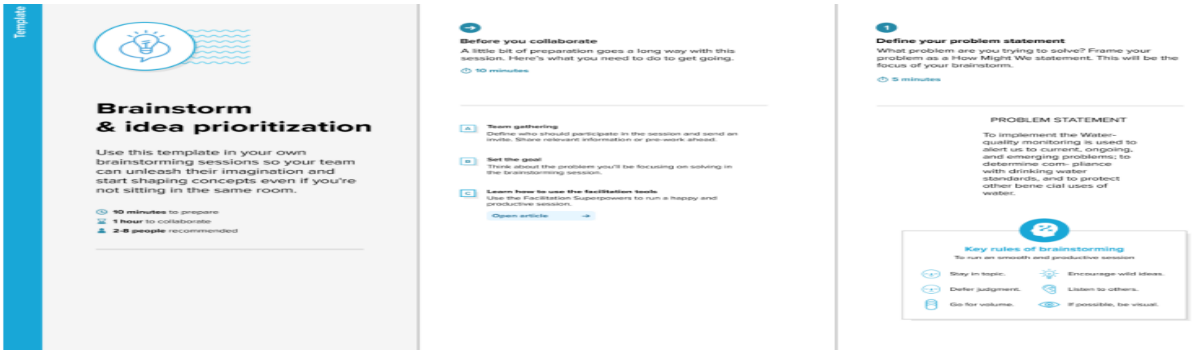
Accurate and exact

Can be operated both online and manually.

IDEATION AND BRAINSTROMING:

Brainstorm & Idea Prioritization Template:

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

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 STATEMENT

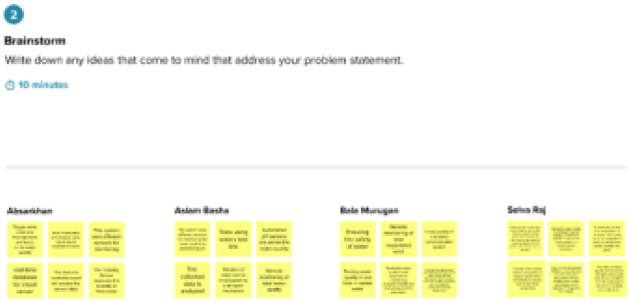
To implement the Water-quality monitoring is used to alert us to current, ongoing, and emerging problems; to determine com- pliance with drinking water standards, and to protect other bene cial uses of water.

Key rules of brainstorming

To run an smooth and productive session

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

Step-2: Brainstorm, Idea Listing and Grouping



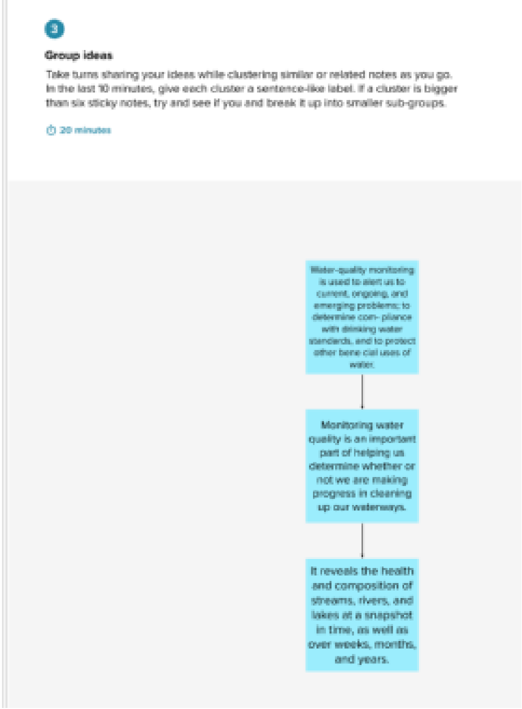
2

Brainstorm

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

10 minutes

Absarkhan
real-time database for cloud server
Aslam Basha
Tests using waters test kits
Automated pH sensors are sense the water quality
The collected data is analyzed
Remote monitoring of total water quality
Bala Murugan
Ensuring the safety of water
Remote monitoring of total suspended solid
Testing water quality in real time in normal water
Selva Raj



3

Group ideas

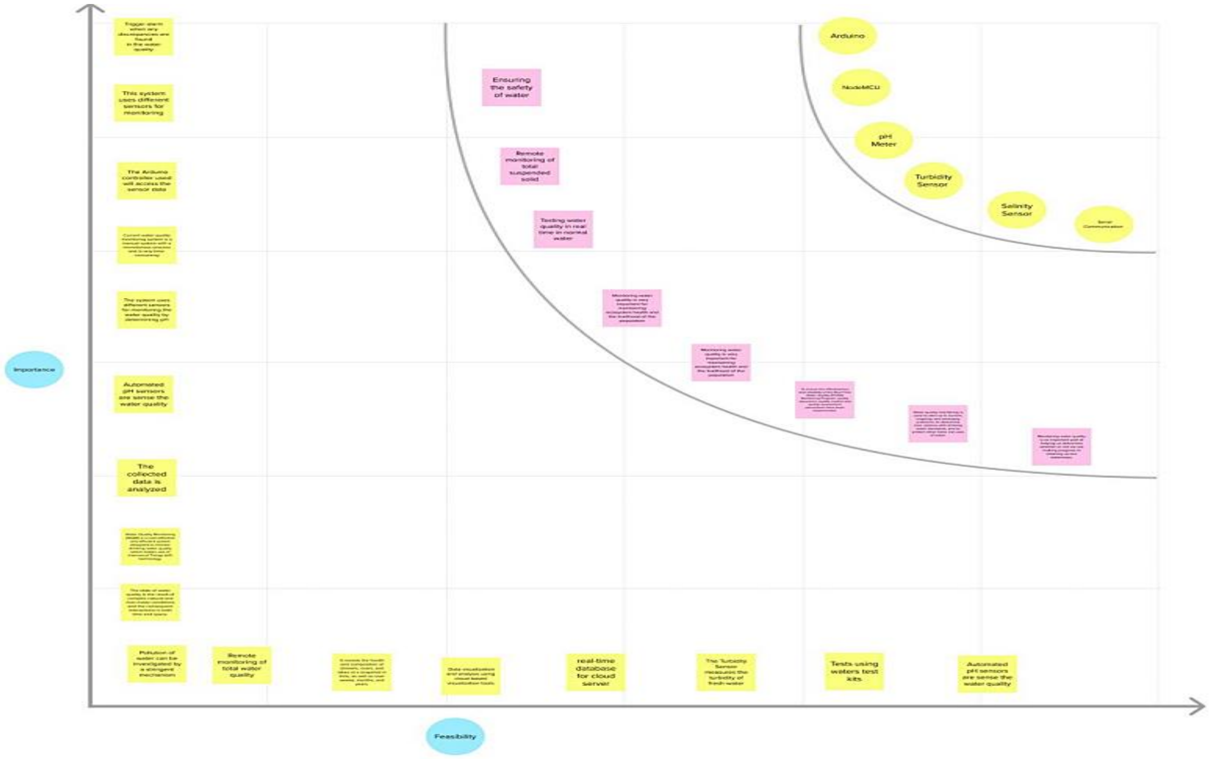
Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Water-quality monitoring is used to alert us to current, ongoing, and emerging problems; to determine con- pliance with drinking water standards, and to protect other bene cial uses of water.

Monitoring water quality is an important part of helping us determine whether or not we are making progress in cleaning up our waterways.

It reveals the health and composition of streams, rivers, and lakes at a snapshot in time, as well as over weeks, months, and years.



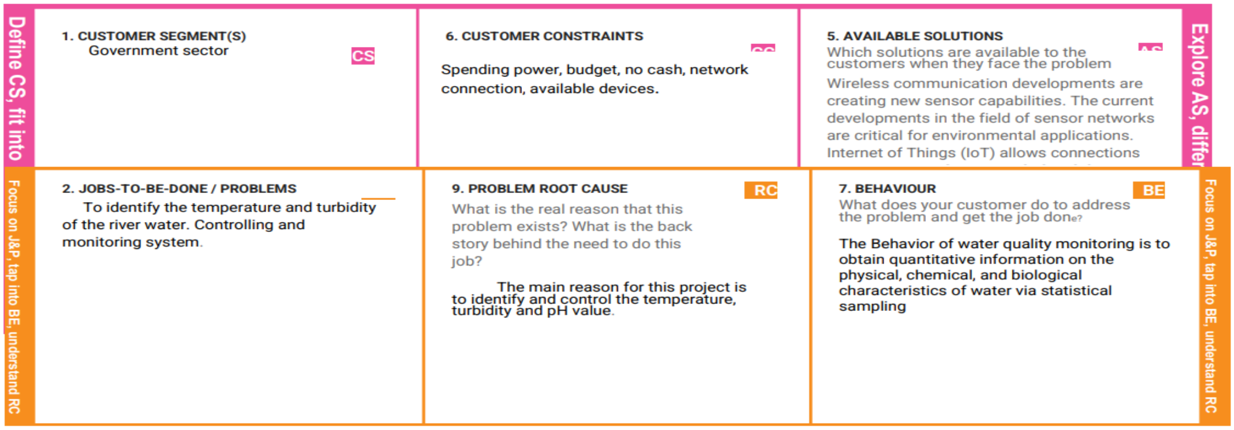
Trigger alarm when any discrepancies are found in the water quality
This system uses different sensors for monitoring
The Arduino controller used will access the sensor data
The system uses different sensors for monitoring the water quality by determining pH
Importance
Automated pH sensors are sense the water quality
The collected data is analyzed
Pollution of water can be investigated by a stringent mechanism
Remote monitoring of total water quality
Data visualization and analysis using cloud based visualization tools
real-time database for cloud server
The Turbidity Sensor measures the turbidity of fresh water
Tests using waters test kits
Automated pH sensors are sense the water quality
Feasibility
Ensuring the safety of water
Remote monitoring of total suspended solid
Testing water quality in real time in normal water
Arduino
NodeMCU
pH Meter
Turbidity Sensor
Salinity Sensor
Serial Communication

PROPOSED SOLUTION:

S.No.	***Parameter***	***Description***
1.	*Problem Statement (Problem to besolved)*	*Water is a finite resource that is necessary foragriculture, industry and the survival of all livingthings on the planet, including humans.* *Many people are unaware of the need of drinking adequate amounts of water on a daily basis.* *Many unregulated methods waste more water. Poor water allocation, inefficient consumption,lack of competent and integrated water management are all factors that contribute to this problem. Therefore, efficient use and watermonitoring are potential constraint for homeor office watermanagement system*
2.	*Idea / Solution description*	***The proposed WQM system consists of sensors, Field Programmable Gate Array(FPGA), Zigbee wireless communication protocol and personal computer****. The system is designed for monitoring water quality such as water temperature, water level, water pH, turbidity of water and Carbon dioxide on the surface of water.*
3.	*Novelty / Uniqueness*	*Using real-time monitoring, instant data* ***allows pre-cursors to potential issues (such as corrosion) to be flaggedup and immediately be addressed beforemajor issues occur****. The ability to make real-time decisions during critical moments can be vital in preventing expensive repairs and breakdown.*

4.	Social Impact/ Customer Satisfaction	● *It gives the accurate measure* ● *The rapid development of WSNtechnology provides a novel approach to real-time data acquisition, transmission, and processing. The clients can get ongoing water quality information from far away.* ***forest fire and earlyearthquake, reduce air population, monitor snow level, preventlandslide, and avalanche*** *etc*
5.	Business Model(Revenue Model)	*The section presents the system deployment strategy and focuses on the sensor probes, the calibration process, and the cloud-based web portal design used for reporting and analysing the data obtained from the deployment environment.*
6.	Scalability of the Solution	● *Well monitoring system with accurateindication.* ● *Easy maintenance.* ● *Reasonablecost*

PROBLEM SOLUTION FIT:


Define CS, fit into
Explore AS, differ
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, understand RC
CS
RC
BE

1. CUSTOMER SEGMENT(S)
Government sector

6. CUSTOMER CONSTRAINTS

Spending power, budget, no cash, network connection, available devices.

5. AVAILABLE SOLUTIONS
Which solutions are available to the customers when they face the problem

Wireless communication developments are creating new sensor capabilities. The current developments in the field of sensor networks are critical for environmental applications. Internet of Things (IoT) allows connections

2. JOBS-TO-BE-DONE / PROBLEMS
To identify the temperature and turbidity of the river water. Controlling and monitoring system.

9. PROBLEM ROOT CAUSE
What is the real reason that this problem exists? What is the back story behind the need to do this job?

The main reason for this project is to identify and control the temperature, turbidity and pH value.

7. BEHAVIOUR
What does your customer do to address the problem and get the job done?

The Behavior of water quality monitoring is to obtain quantitative information on the physical, chemical, and biological characteristics of water via statistical sampling

3. TRIGGERS TR	10. YOUR SOLUTION SL	8. CHANNELS of BEHAVIOUR CH
Energy consumption, Water distribution system, Genetic algorithm, Operating cost.	Using real-time monitoring, instant data allows pre-cursors to potential issues (such as corrosion) to be flagged up and immediately be addressed before major issues occur. The ability to make real-time decisions during critical moments can be vital in preventing expensive repairs and breakdown.	8.1 ONLINE What kind of actions do customers take online? The temperature and turbidity of the water can be measured by the customers in online. 8.2 OFFLINE What kind of actions do customers take offline? The level of the water can be intimated through the offline mode.

Identif

4. EMOTIONS: BEFORE / AFTER EM		
BEFORE: Trouble in identify the turbidity of the river water. Rural peoples affected by the unpurified water. AFTER: Using real-time monitoring, instant data allows pre-cursors to potential issues (such as corrosion) to be flagged up and immediately be addressed before major issues occur. The ability to make real-time decisions during critical moments can be vital in preventing expensive repairs and breakdown.		

REQUIREMENT ANALYSIS:

Functional Requirements:

Following are the functional requirements of the proposed solution.

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

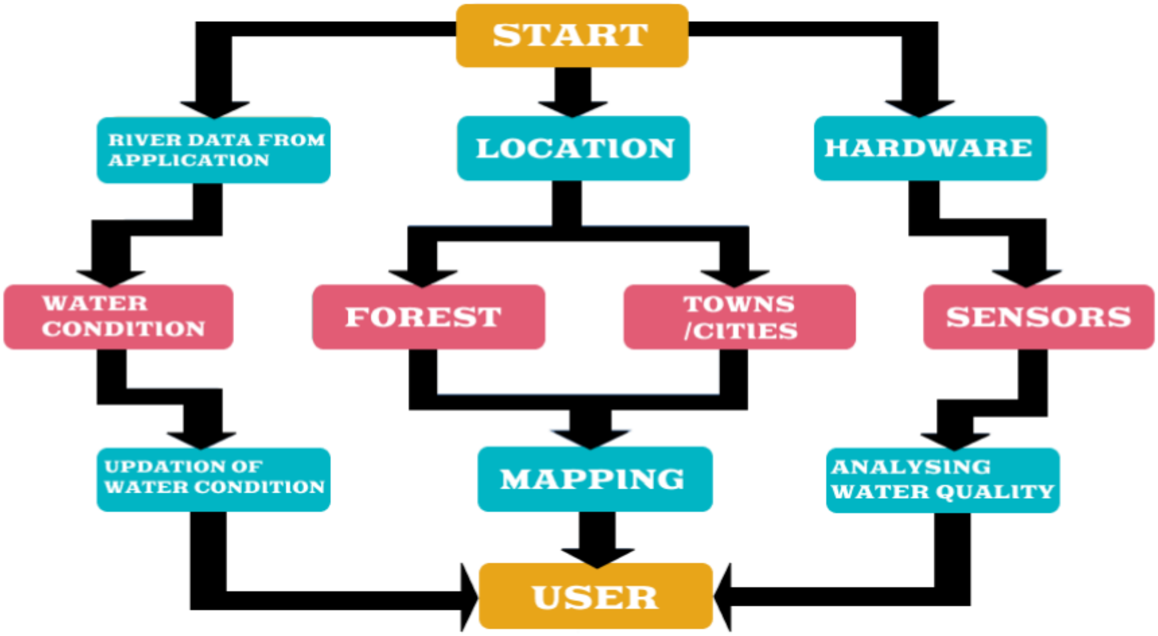
Non-functional Requirements:

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

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 2 seconds. Login info verified within 10 seconds.
NFR-5	**Availability**	Maximum down time will be about 4 hours
NFR-6	**Scalability**	System can handle about 1000 users at any given time

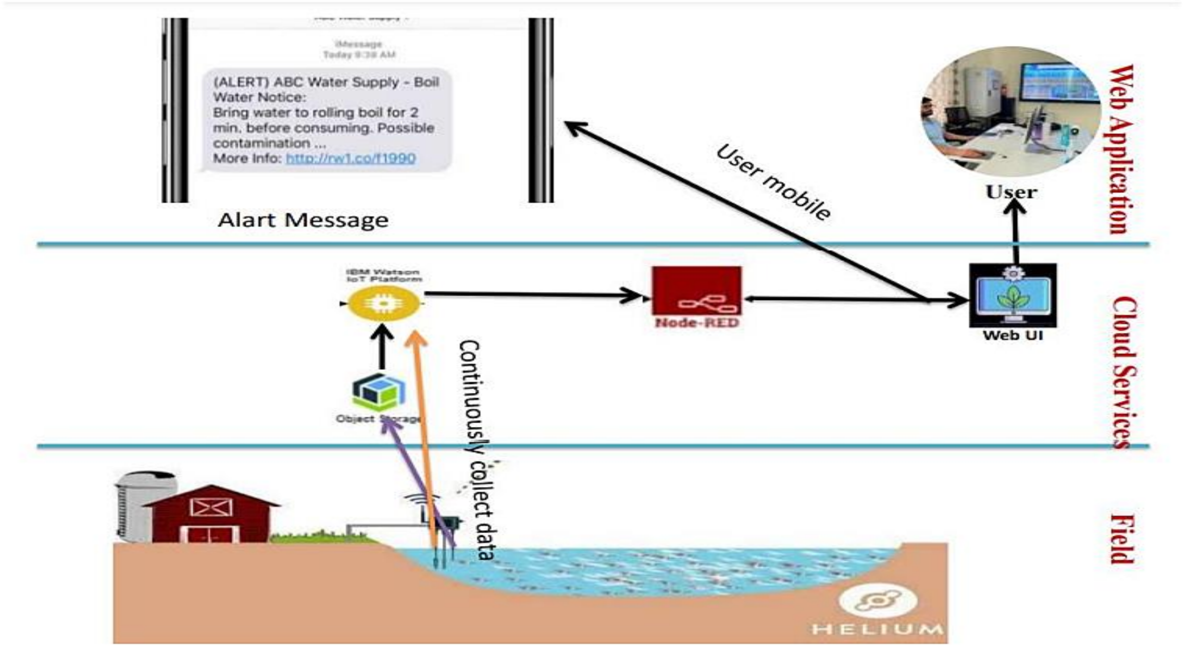
PROJECT DESIGN:

DATA FLOW DIAGRAM:

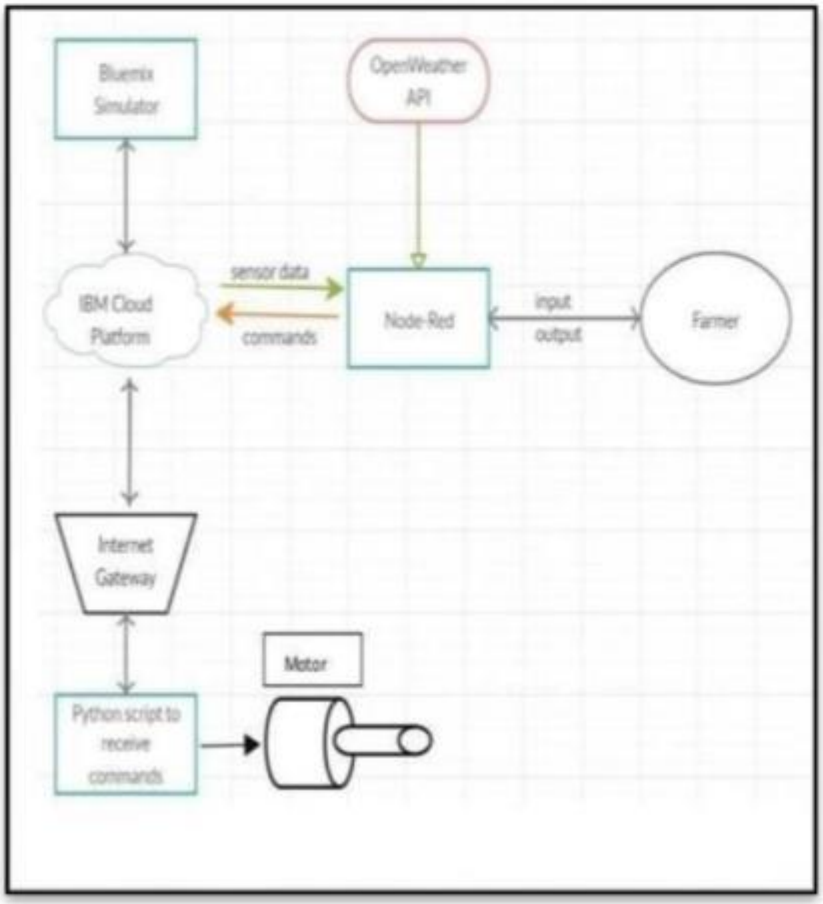


SOLUTION AND TECHNICAL ARCHITECTURE:

Solution Architecture Diagram:


(ALERT) ABC Water Supply - Boil Water Notice:
Bring water to rolling boil for 2 min. before consuming. Possible contamination ...
More Info: http://rw1.co/f1990

TECHNICAL ARCHITECTURE



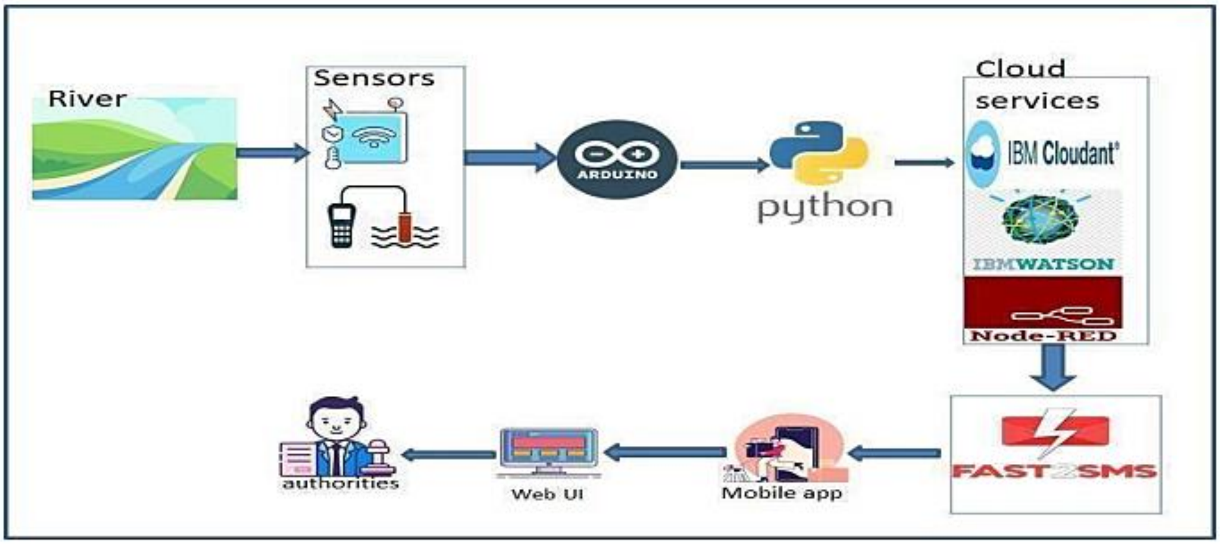


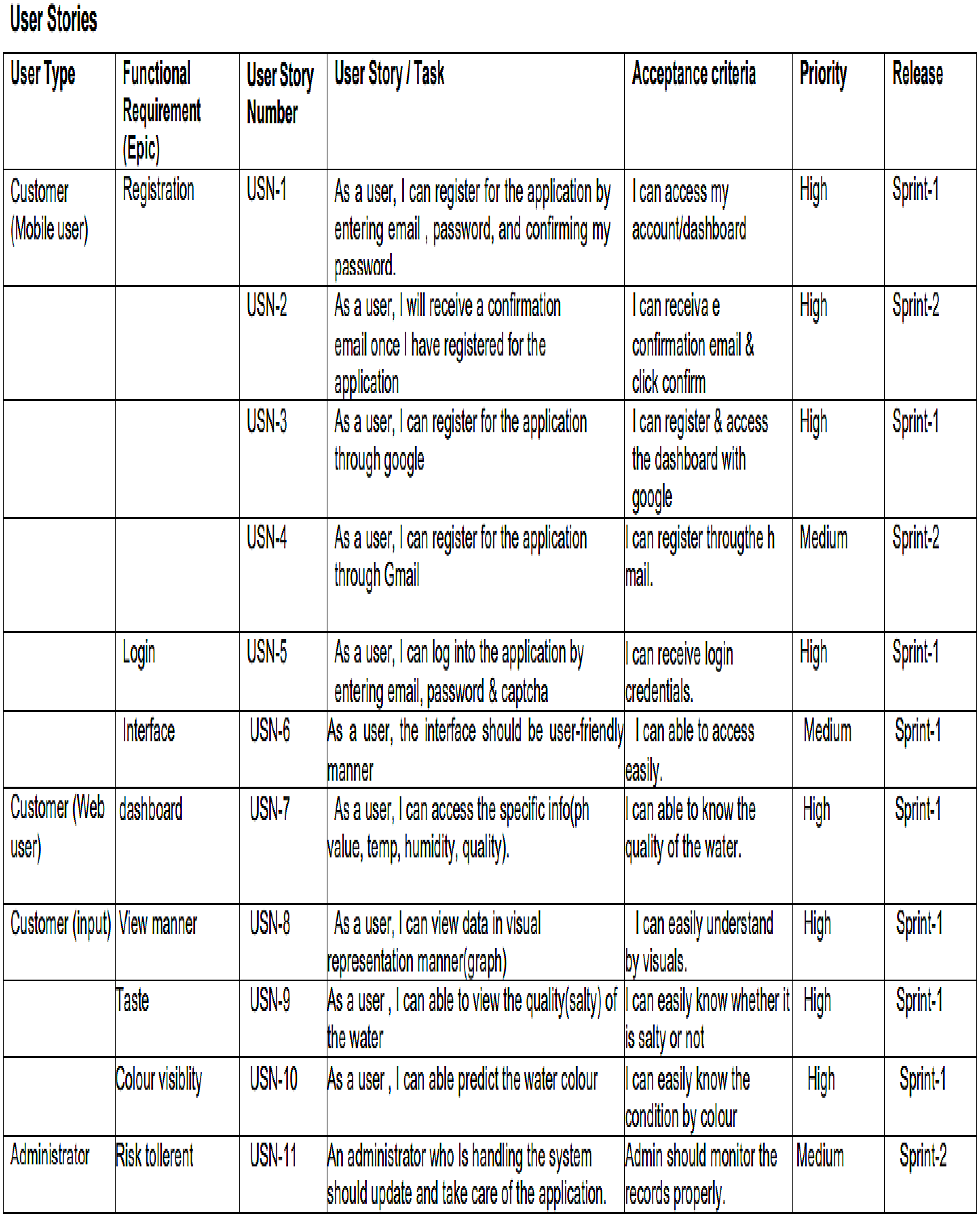
Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application	HTML, CSS, Node-Red ,Cloud,etc
2.	Application Logic-1	Logic for a process in the application	JAVA/PYTHON
3.	Application Logic-2	Logic for a process in the application	IBM WATSON STT services
4.	Application Logic-3	Logic for a process in the application	BM WATSON Assistant
5.	Database	Data Type, Configurations etc	MySQL,PostgresSQL
6.	Cloud Database	Database Service on Cloud	IBM DB2,IBM Cloudant etc
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc
9.	External API-2	Purpose of External API used in the application	Aadhar API, etc
10.	Machine Learning Model	Purpose of External API used in the application	Object Recognition Model, etc..
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of application	Technology used
5.	Performance	Design consideration for the performance of the application	Technology used

USER STORIES:



User Stories

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 email , password, and confirming my password.	I can access my account/dashboard	High	Sprint-1
		USN-2	As a user, I will receive a confirmation email once I have registered for the application	I can receiva e confirmation email & click confirm	High	Sprint-2
		USN-3	As a user, I can register for the application through google	I can register & access the dashboard with google	High	Sprint-1
		USN-4	As a user, I can register for the application through Gmail	I can register througthe h mail.	Medium	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email, password & captcha	I can receive login credentials.	High	Sprint-1
	Interface	USN-6	As a user, the interface should be user-friendly manner	I can able to access easily.	Medium	Sprint-1
Customer (Web user)	dashboard	USN-7	As a user, I can access the specific info(ph value, temp, humidity, quality).	I can able to know the quality of the water.	High	Sprint-1
Customer (input)	View manner	USN-8	As a user, I can view data in visual representation manner(graph)	I can easily understand by visuals.	High	Sprint-1
	Taste	USN-9	As a user , I can able to view the quality(salty) of the water	I can easily know whether it is salty or not	High	Sprint-1
	Colour visiblity	USN-10	As a user , I can able predict the water colour	I can easily know the condition by colour	High	Sprint-1
Administrator	Risk tollerent	USN-11	An administrator who Is handling the system should update and take care of the application.	Admin should monitor the records properly.	Medium	Sprint-2

PROJECT PLANNING AND SCHEDULING:

Sprint	***Functional Requirement (Epic)***	***UserStory Number***	***User Story / Task***	***Story Points***	***Priority***	***Team Members***
Sprint-1	Registration	USN-1	Asa user, I can register for the applicationby entering my email,password, and confirming my password.	2	High	*Absarkhan. A* *Aslam Basha. S* *BalaMurugan. M*
	Registration via facebook	USN-3	As a user, I can register for the application through Facebook	2	Low	
	Registration via MailID	USN-4	Asa user, I can register for the application through Gmail	2	Medium	
Sprint-2	*Confirmation*	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	
	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	
	IBM Cloud serviceaccess		Get accessto IBM cloudservices.	2	High	

Sprint-3	Create the IBM Watson IoT and device Settings	*USN-6*	*To createthe IBM WatsonIoT Platform andintegrate the microcontroller with it, to send the sensed dataon cloud*	*2*	*High*	*Absarkhan Aslambasha*
	Create a node red service	*USN-7*	*To createa node red service to integrate theIBM Watson along with the WebUI*	*2*	*Medium*	*Absarkhan Balamurugan*
	Create a Web UI	*USN-8*	*To createa Web UI,to access the data from the cloud and display all parameters.*	*2*	*Medium*	*Aslam Basha Bala murugan*
	To develop a Python code	*USN-9*	*Create a pythoncode to sensethe physical quantity and store data.*	*2*	*Medium*	*Absarkhan Aslam Basha*

	Publish Data to cloud.	*USN-10*	*Publish Datathat is sensedby the microcontr*	*3*	*High*	*BalaM*

			oller tothe Cloud			*u* *r* *u* *g* *a* *n*
Sprint-4	*Fast-SMS Service*	*USN-11*	*Use Fast SMS to send alert messages once the parameters like pH, Turbidity and temperature goes beyond the threshold*	*3*	*High*	*Absar khan*
	Testing	*USN-12*	*Testing of project and final deliverables*	*3*	*Medium*	

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

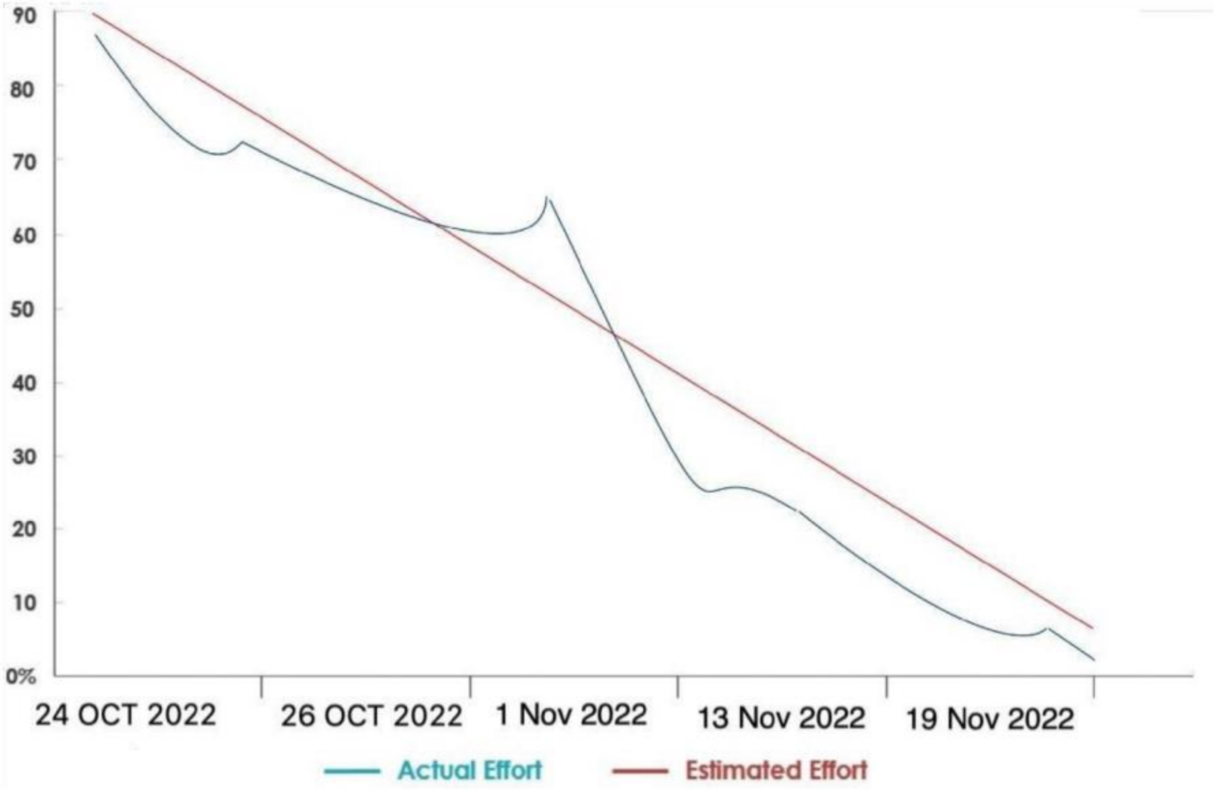
Sprint	***Total Story Points***	***Duration***	***Sprint Start Date***	***SprintEnd Date (Planned)***	***Story Points Completed (as on Planned End Date)***	***SprintRelease Date (Actual)***
Sprint-1	20	2Days	24Oct 2022	26Oct 2022	20	29Oct 2022
Sprint-2	20	4Days	26Oct 2022	30Oct 2022	*40*	
Sprint-3	20	12 Days	1 Nov2022	12 Nov 2022	*60*	
Sprint-4	20	6Days	13Nov 2022	19Nov 2022	*80*	*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) periteration unit (story pointsper day)

Burndown Chart:

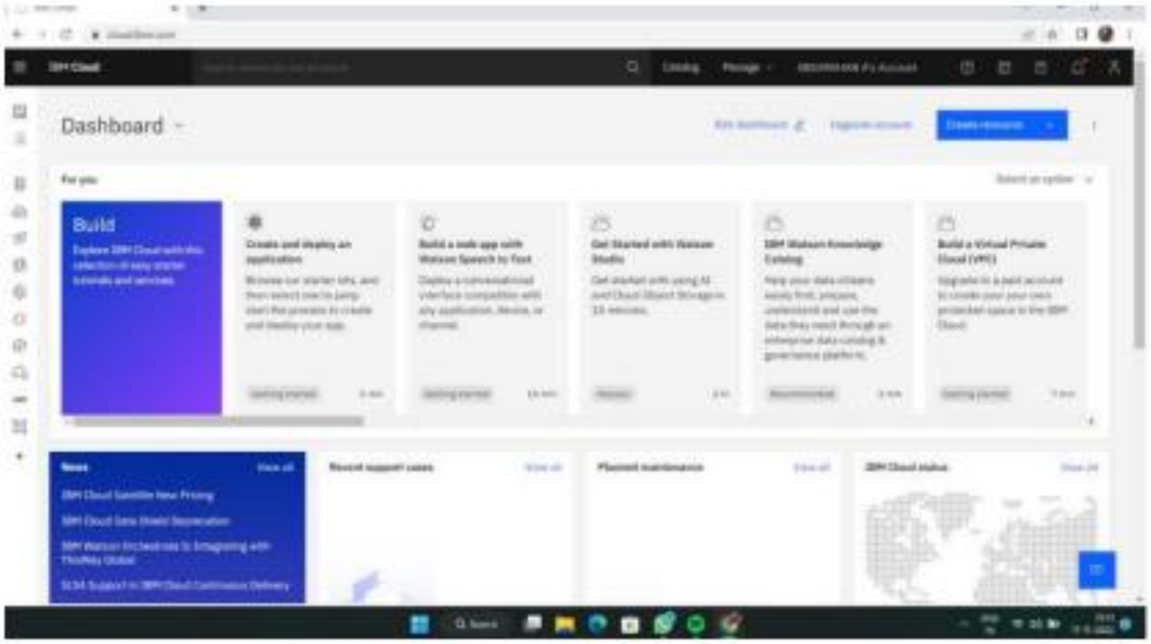
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies suchas Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



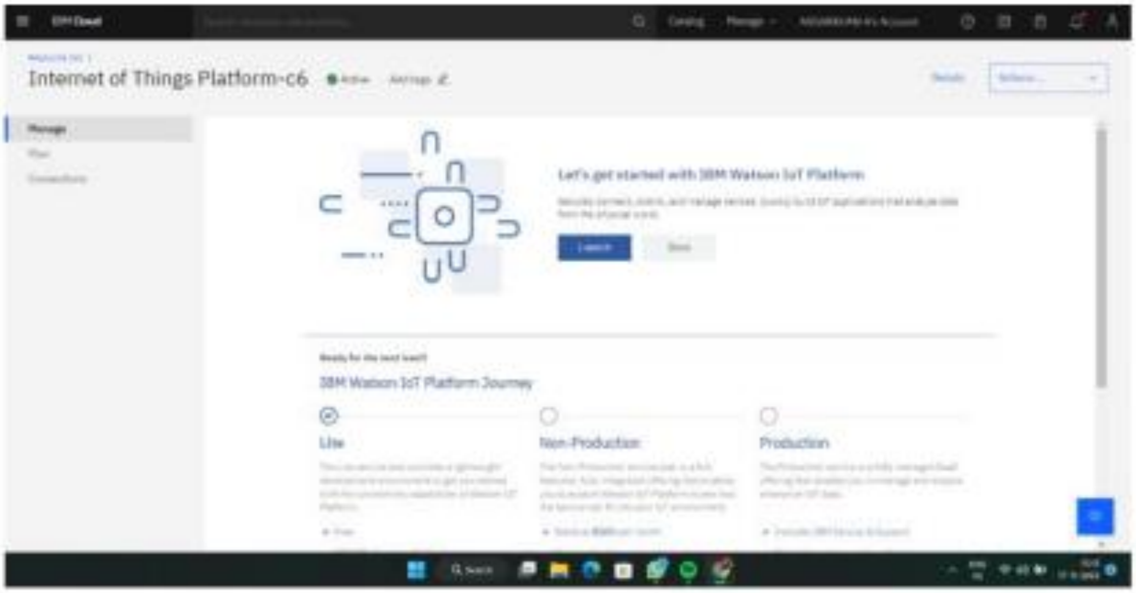
Sprint Delivery Schedule:

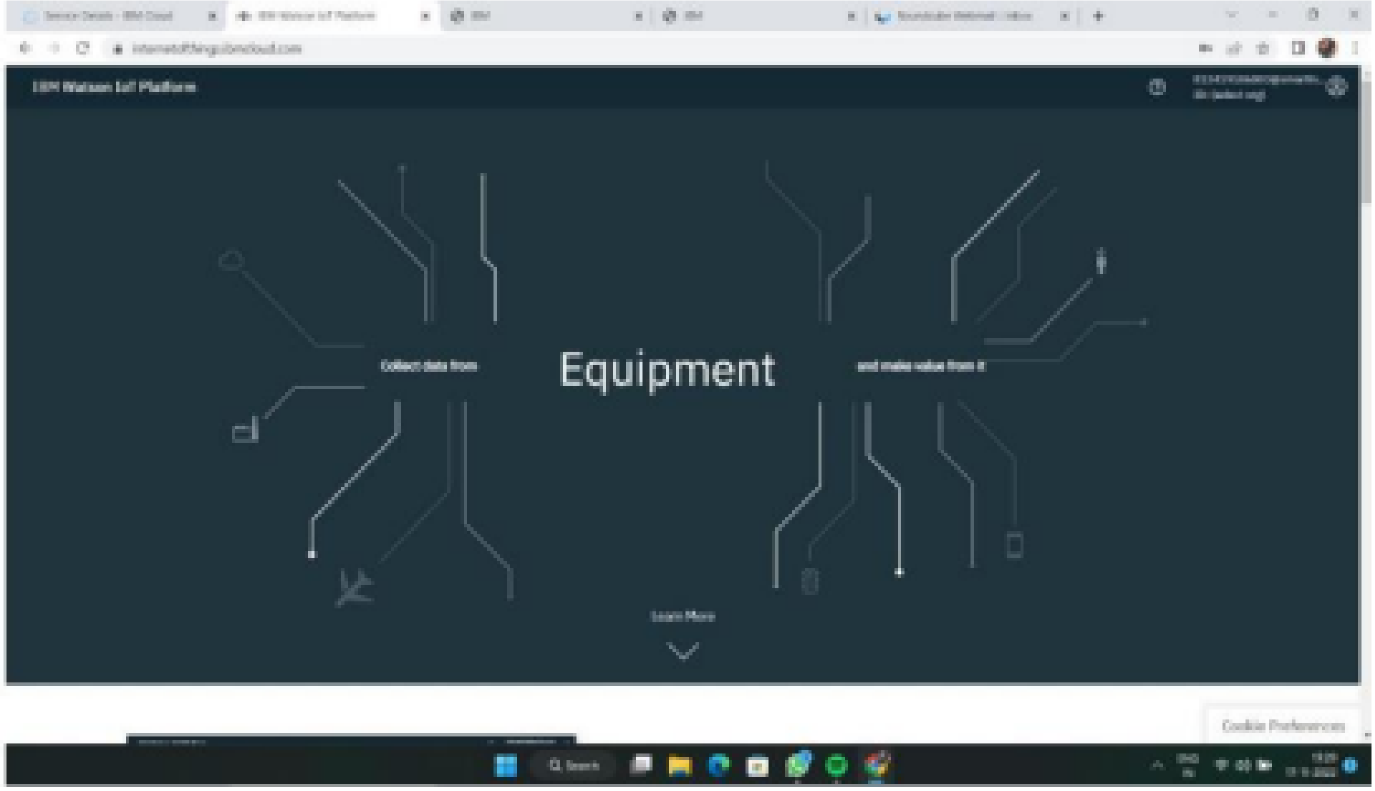
SPRINT 1:

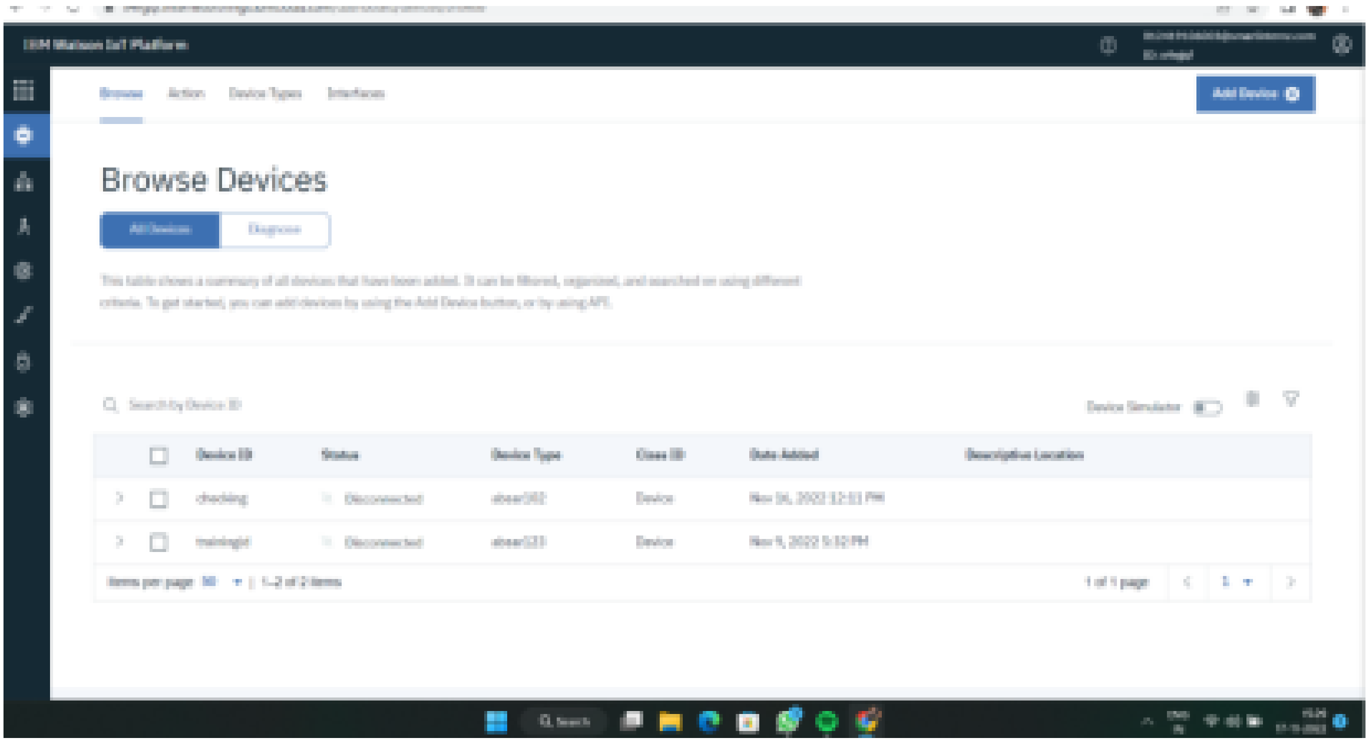
1) IBM CLOUD ACCOUNT


IBM Cloud
Catalog
Manage
Dashboard
For you
Build
Create and deploy an application
Build a web app with Watson Speech to Text
Get Started with Watson Studio
IBM Watson Knowledge Catalog
Build a Virtual Private Cloud (VPC)
News
View all
Recent support cases
Planned maintenance
IBM Cloud status

2) IBM WATSON PLATFORM


IBM Cloud
Catalog
Manage
Internet of Things Platform-c6
Active
Manage
Plan
Connections
Let's get started with IBM Watson IoT Platform
Launch
Docs
Ready for the next level?
IBM Watson IoT Platform Journey
Lite
Non-Production
Production





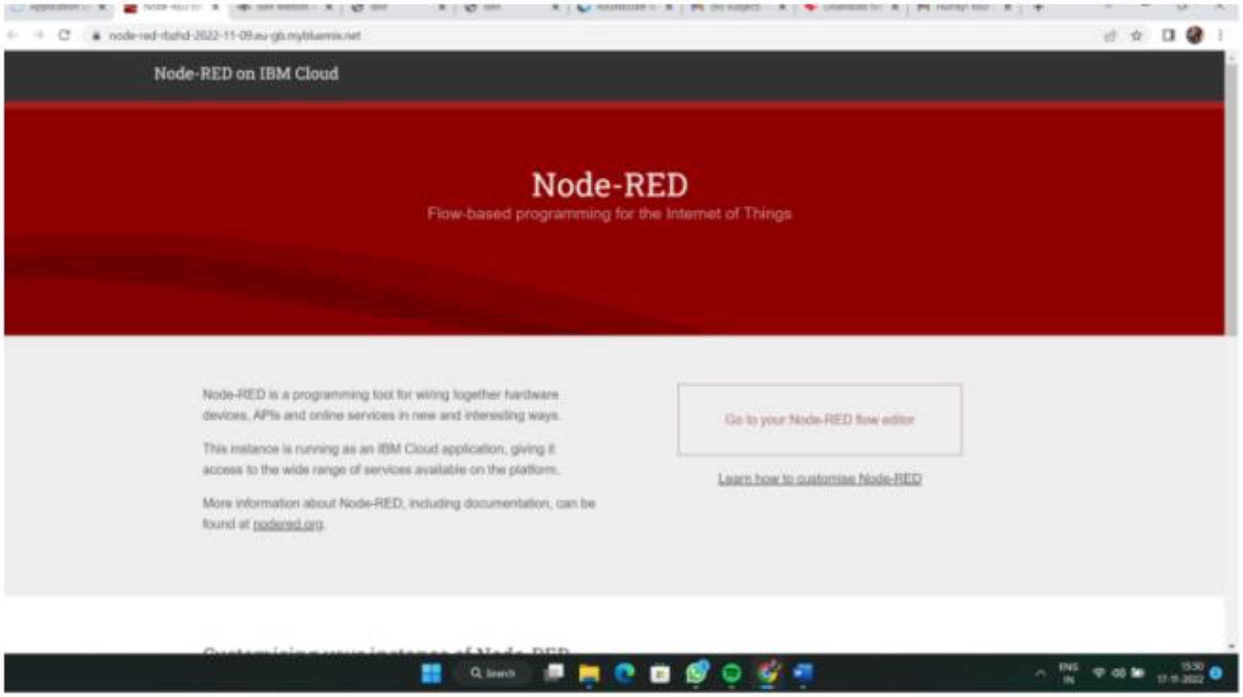
Browse Devices

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.

	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
>	checking	Disconnected	abear102	Device	Nov 16, 2022 12:11 PM	
>	trainingid	Disconnected	abear123	Device	Nov 9, 2022 5:32 PM	

Items per page 50 | 1–2 of 2 items   1 of 1 page   1

3) NODE-RED



node-red-rbzhd-2022-11-09.eu-gb.mybluemix.net

Node-RED on IBM Cloud

Node-RED

Flow-based programming for the Internet of Things

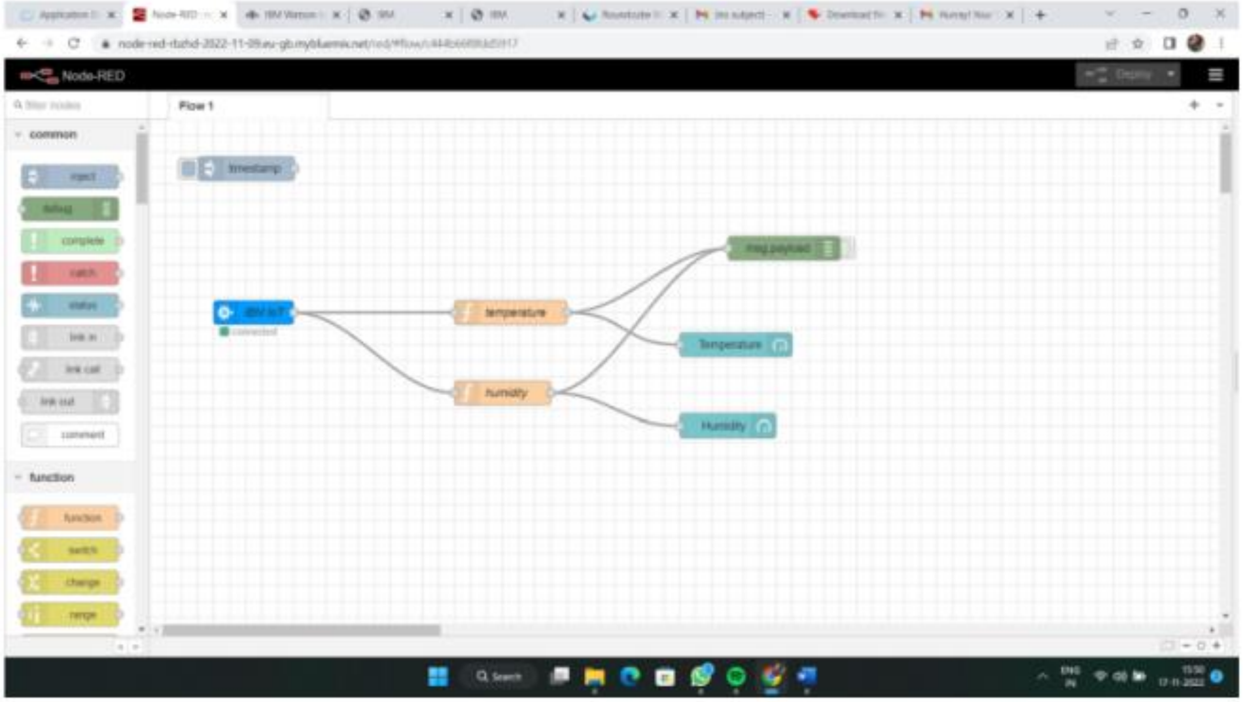
Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

This instance is running as an IBM Cloud application, giving it access to the wide range of services available on the platform.

More information about Node-RED, including documentation, can be found at nodered.org.

Go to your Node-RED flow editor

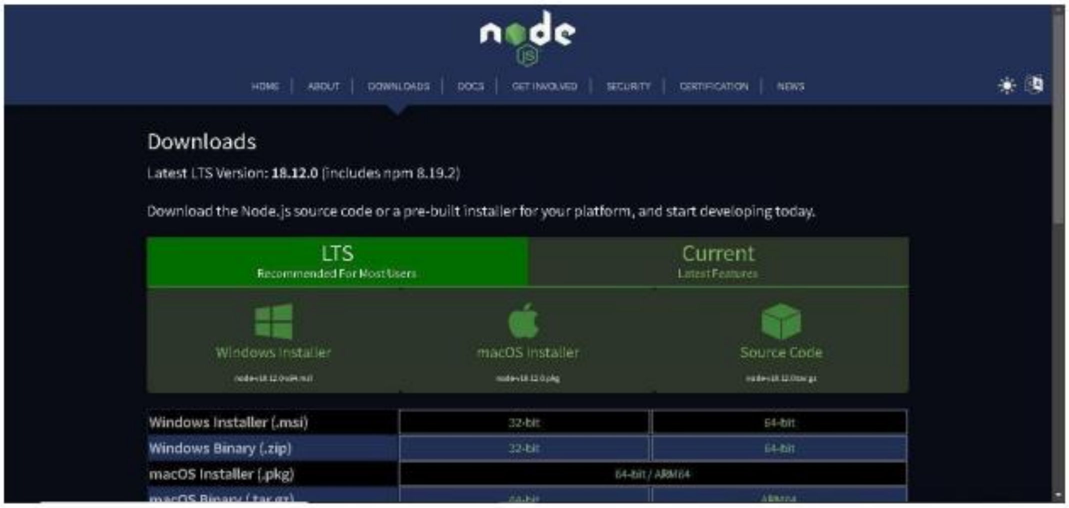
Learn how to customise Node-RED



SPRINT 2:

The following steps are involved:

STEP 1: Download and Install node.js.


node js
HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | CERTIFICATION | NEWS

Downloads

Latest LTS Version: **18.12.0** (includes npm 8.19.2)

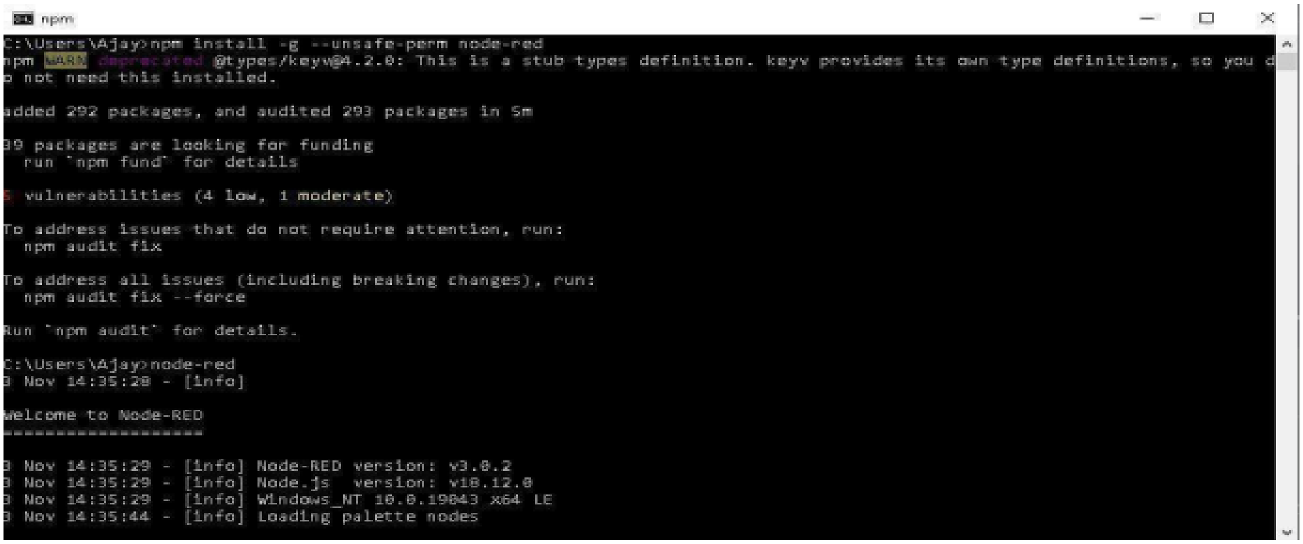
Download the Node.js source code or a pre-built installer for your platform, and start developing today.

LTS Recommended For Most Users	Current Latest Features

Windows Installer node-v18.12.0-x64.msi	macOS Installer node-v18.12.0.pkg	Source Code node-v18.12.0.tar.gz

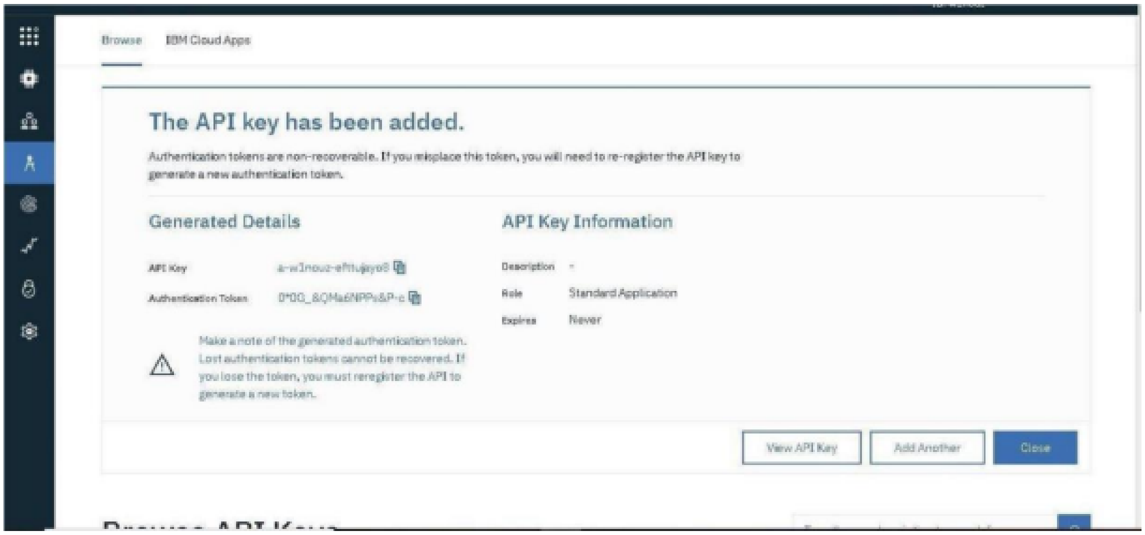
Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.zip)	32-bit	64-bit
macOS Installer (.pkg)	64-bit / ARM64	
macOS Binary (.tar.gz)	64-bit	ARM64

STEP 2: Setup node.js and configure command prompt for error check. Open node-red from the generated link.



```
C:\Users\Ajay>npm install -g --unsafe-perm node-red
npm WARN deprecated @types/keyv@4.2.0: This is a stub types definition. keyv provides its own type definitions, so you d
o not need this installed.

added 292 packages, and audited 293 packages in 5m

39 packages are looking for funding
  run `npm fund` for details

5 vulnerabilities (4 low, 1 moderate)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

C:\Users\Ajay>node-red
3 Nov 14:35:28 - [info]

Welcome to Node-RED
===================

3 Nov 14:35:29 - [info] Node-RED version: v3.0.2
3 Nov 14:35:29 - [info] Node.js  version: v18.12.0
3 Nov 14:35:29 - [info] Windows_NT 10.0.19043 x64 LE
3 Nov 14:35:44 - [info] Loading palette nodes
```

STEP 3: Generating API key and Authentication token.



The API key has been added.

Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the API key to generate a new authentication token.

Generated Details

API Key a-w1nouz-efttujsyo8

Authentication Token 0*OG_&QMa6NPPs&P-c

Make a note of the generated authentication token. Lost authentication tokens cannot be recovered. If you lose the token, you must reregister the API to generate a new token.

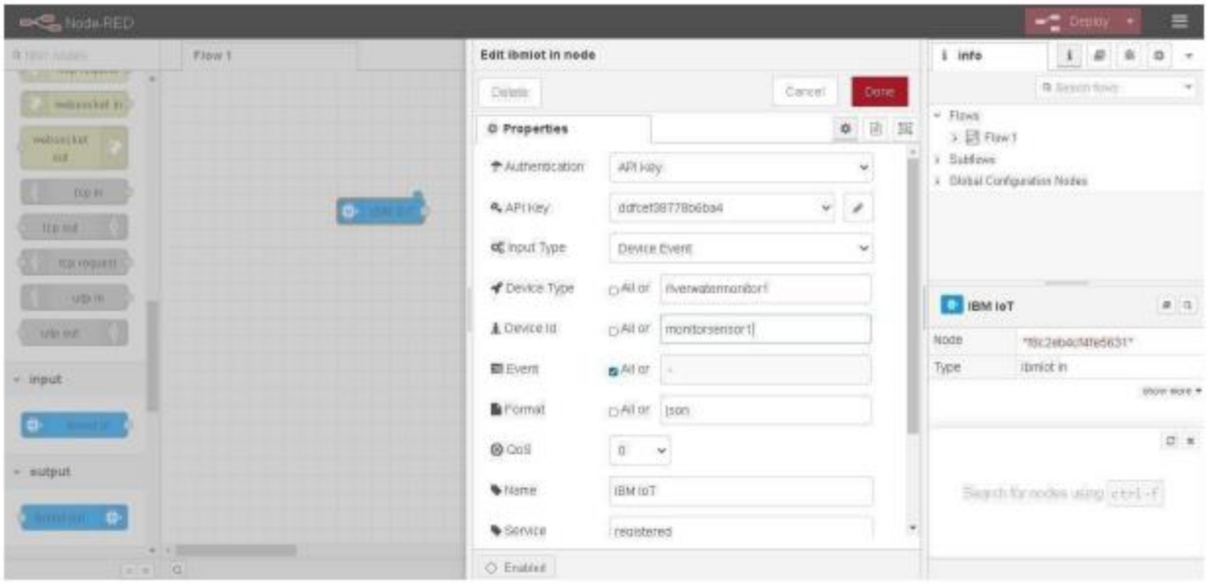
API Key Information

Description -

Role Standard Application

Expires Never

STEP 4: Edit Ibmiot in node.



Edit ibmiot in node

Authentication: API Key

API Key: ddfcef38778b6ba4

Input Type: Device Event

Device Type: □ All or riverwatermonitor1

Device Id: □ All or monitorsensor1

Event: ☑ All or -

Format: □ All or json

QoS: 0

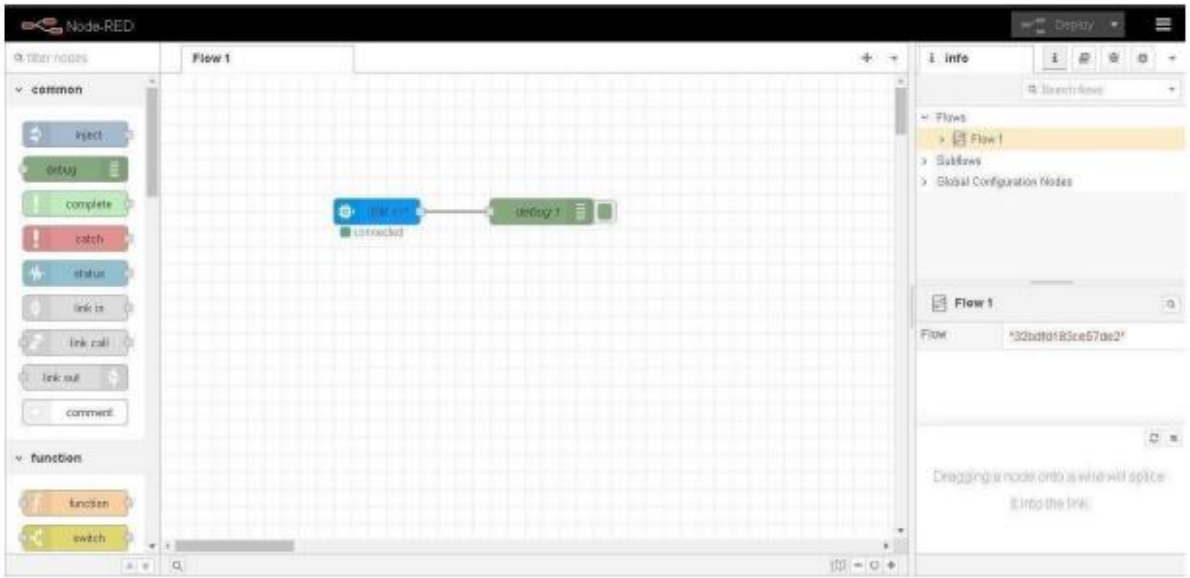
Name: IBM IoT

Service: registered

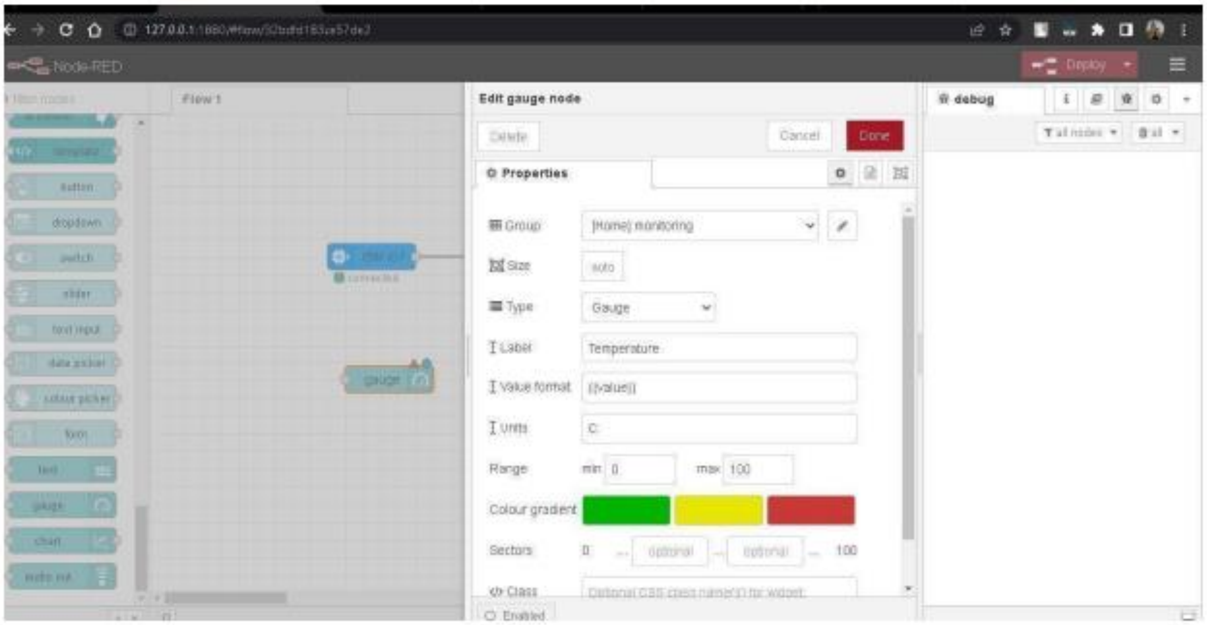
IBM IoT

Node	"f8c2eba0f4fe5631"
Type	ibmiot in

STEP 5: Connect Ibmiot in and debug 1 and deploy.



STEP 6: Edit gauge node (here the gauge nodes are named as Temperature, pH and Turbidity).



Edit gauge node

Group: [Home] monitoring

Size: auto

Type: Gauge

Label: Temperature

Value format: {{value}}

Units: C

Range: min 0 max 100

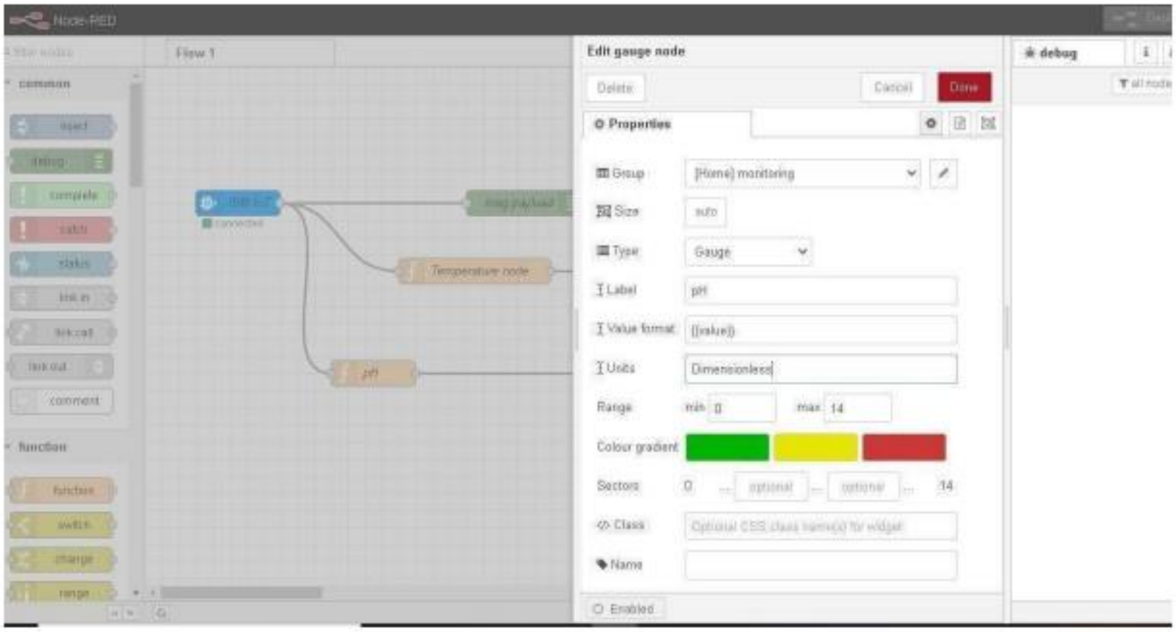
Colour gradient: 🟩 🟨 🟥

Sectors: 0 ... optional ... optional ... 100

Class: Optional CSS class name(s) for widget

○ Enabled

Fig 1



Edit gauge node

Group: [Home] monitoring

Size: auto

Type: Gauge

Label: pH

Value format: {{value}}

Units: Dimensionless

Range: min 0 max 14

Colour gradient: 🟩 🟨 🟥

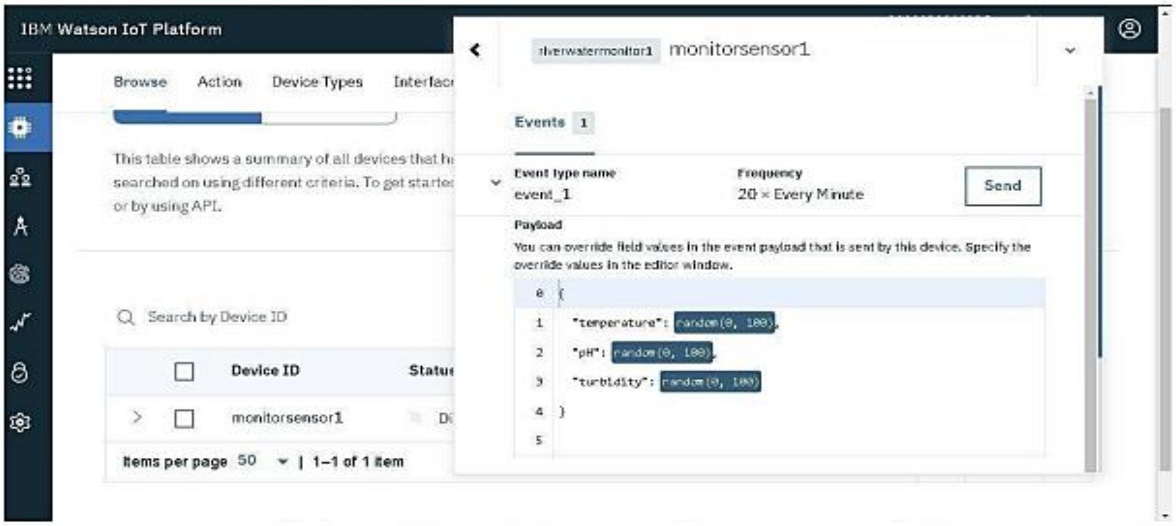
Sectors: 0 ... optional ... optional ... 14

Class: Optional CSS class name(s) for widget

Name:

Fig 2

STEP 7: Simulated program to get the random values.



This table shows a summary of all devices that h… searched on using different criteria. To get starte… or by using API.

□	Device ID	Status
□	monitorsensor1	Di

Items per page 50 | 1–1 of 1 item

riverwatermonitor1 monitorsensor1

Event type name: event_1

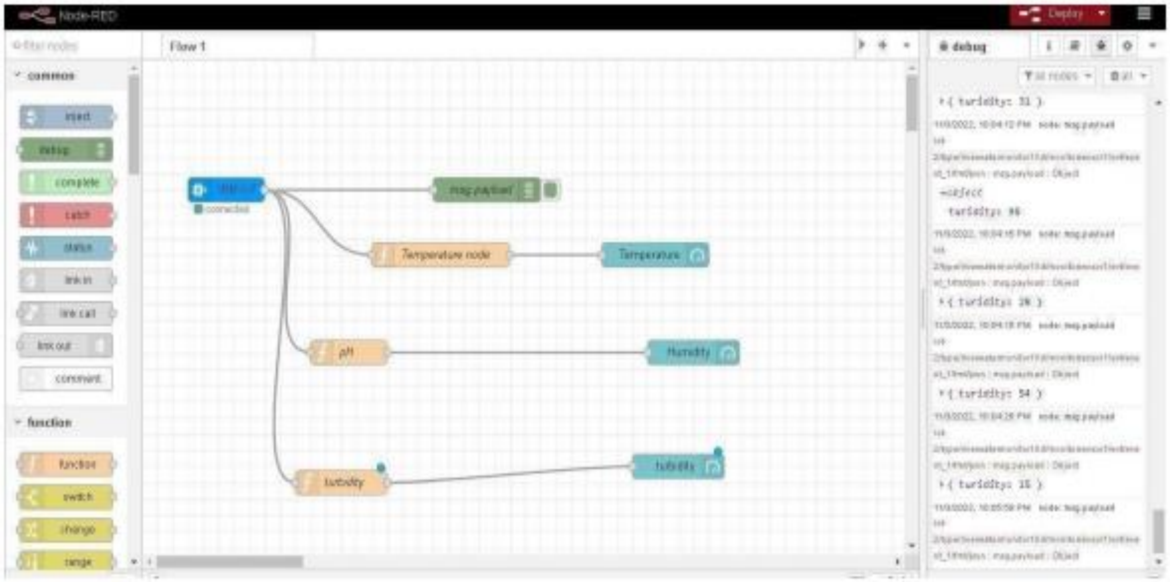
Frequency: 20 × Every Minute

Payload

You can override field values in the event payload that is sent by this device. Specify the override values in the editor window.

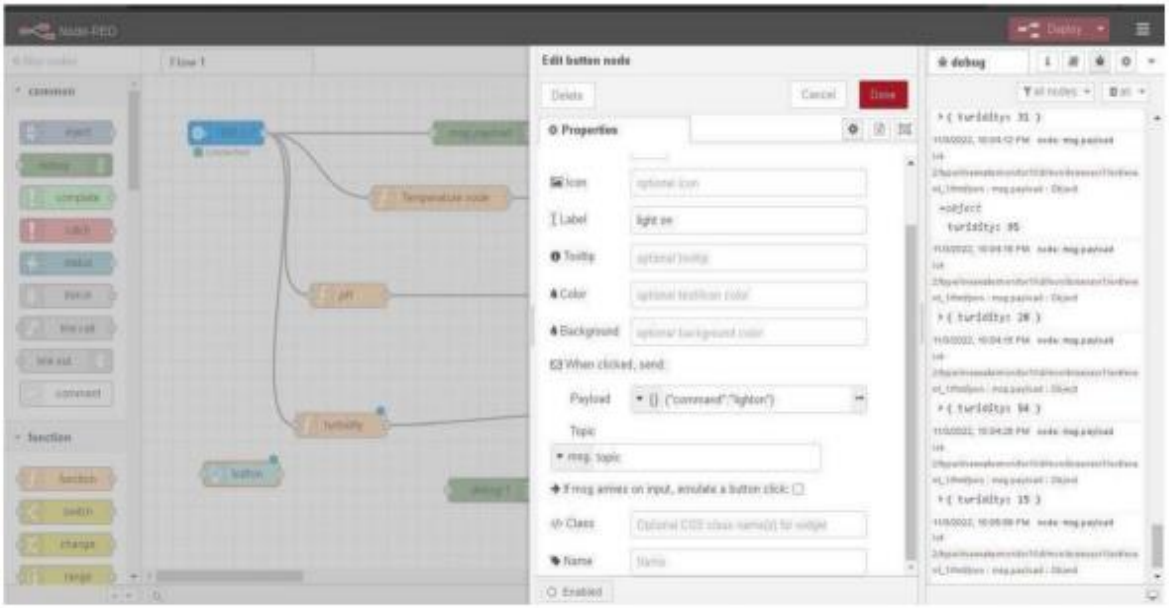
```
{
  "temperature": random(0, 100),
  "pH": random(0, 100),
  "turbidity": random(0, 100)
}
```

STEP 8: Generate debug message from IBM Watson IoT Platform and connect the nodes.



```
▸{ turiddty: 31 }
11/5/2022, 10:04:12 PM node: msg.payload
iot-
[illegible]
[illegible] : msg.payload : Object
▾object
  turiddty: 86
11/5/2022, 10:04:15 PM node: msg.payload
iot-
[illegible]
[illegible] : msg.payload : Object
▸{ turiddty: 28 }
11/5/2022, 10:04:18 PM node: msg.payload
iot-
[illegible]
[illegible] : msg.payload : Object
▸{ turiddty: 54 }
11/5/2022, 10:04:26 PM node: msg.payload
iot-
[illegible]
[illegible] : msg.payload : Object
▸{ turiddty: 15 }
11/5/2022, 10:05:59 PM node: msg.payload
iot-
[illegible]
[illegible] : msg.payload : Object
```

STEP 9: Edit button mode [light ON and light OFF].



Edit button node

Icon: optional icon

Label: light on

Tooltip: optional tooltip

Color: optional text/icon color

Background: optional background color

When clicked, send:

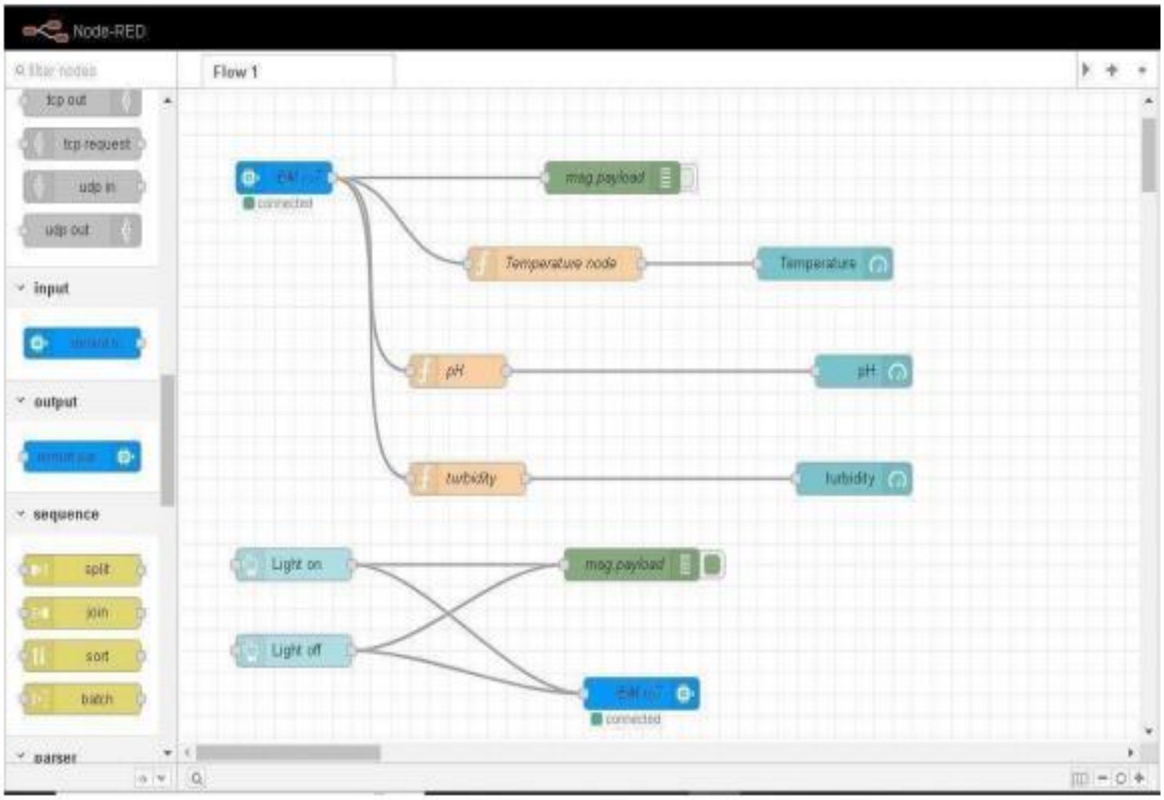
Payload: {} {"command":"lighton"}

Topic: msg. topic

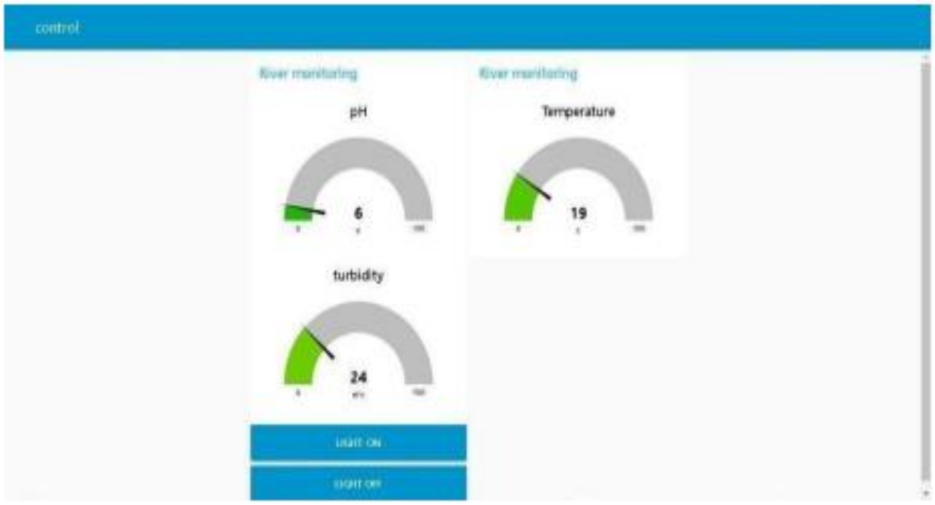
If msg arrives on input, emulate a button click: □

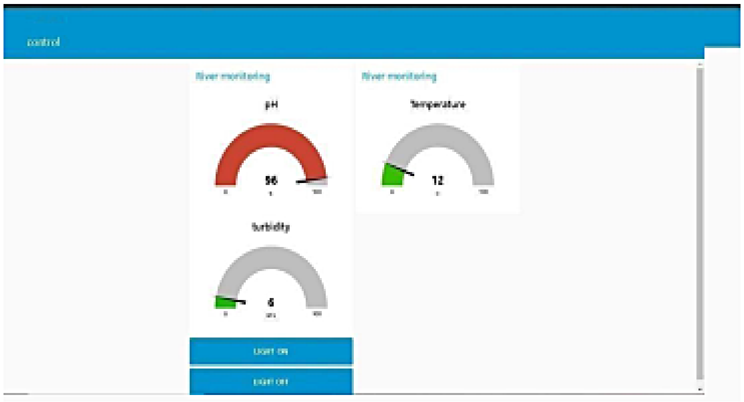
Class: Optional CSS class name(s) for widget

Name: Name

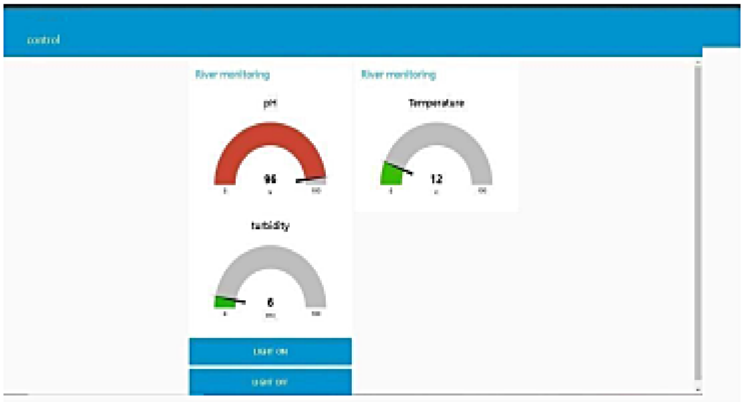


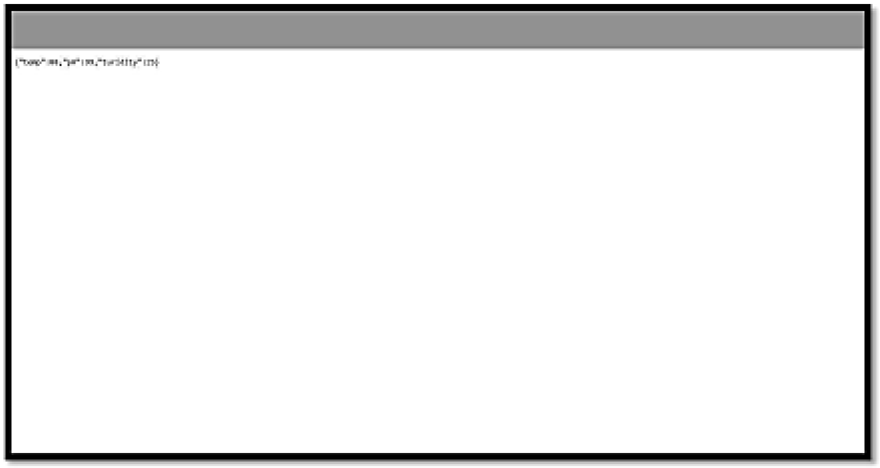
STEP 11: Generate the output from recent events.





STEP 12: Implementing url in the function node to generate output.





```
{"temp":[illegible],"ph":[illegible],"turbidity":[illegible]}
```

Step 13: MIT app inverter to design the app.

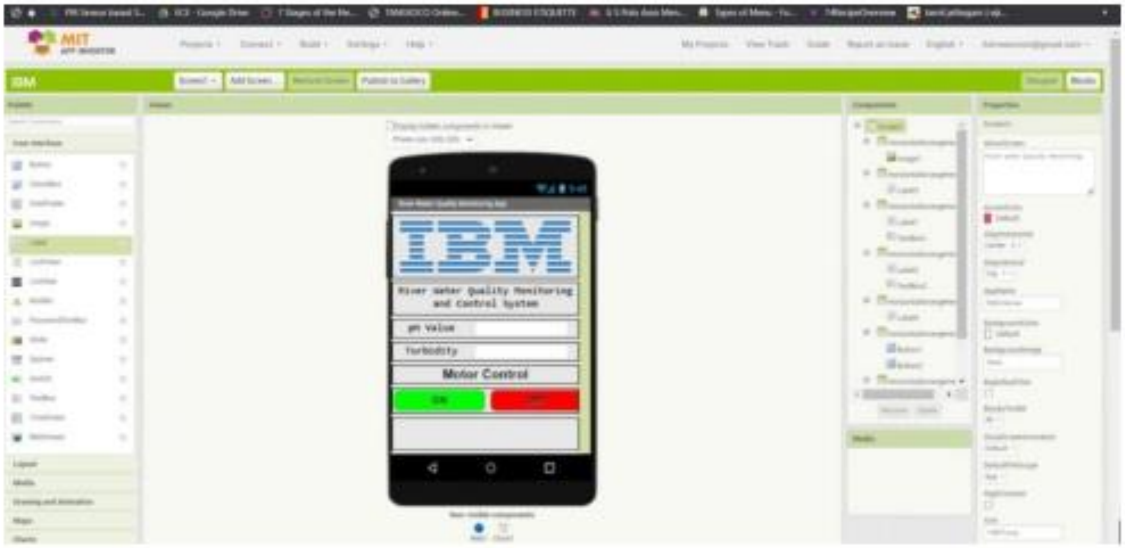


Fig 1

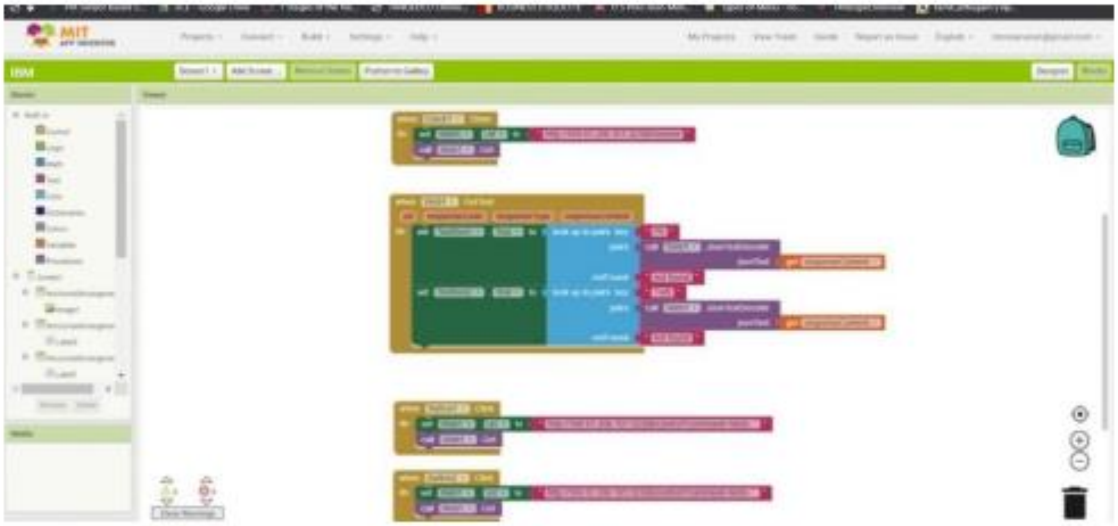
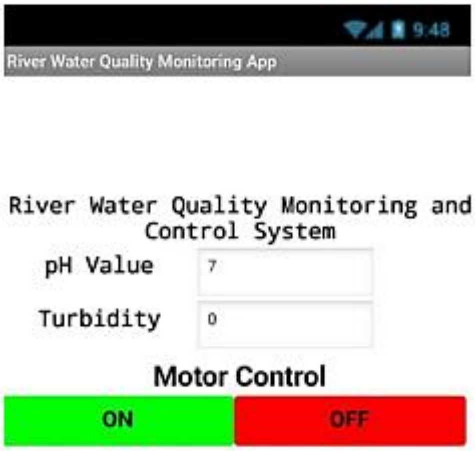


Fig 2



River Water Quality Monitoring and Control System

pH Value: 7

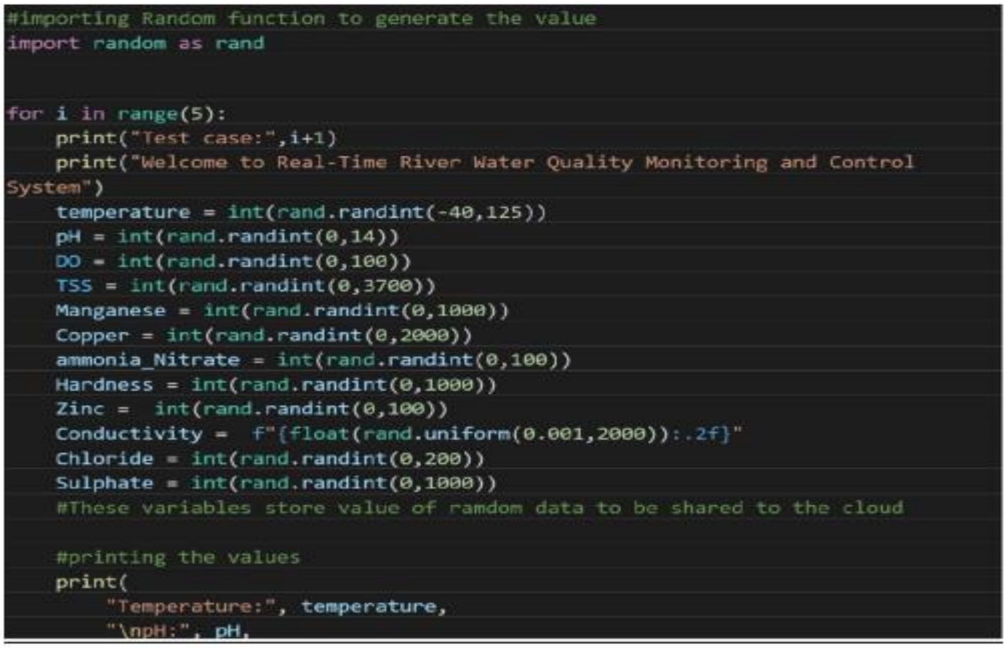
Turbidity: 0

Motor Control

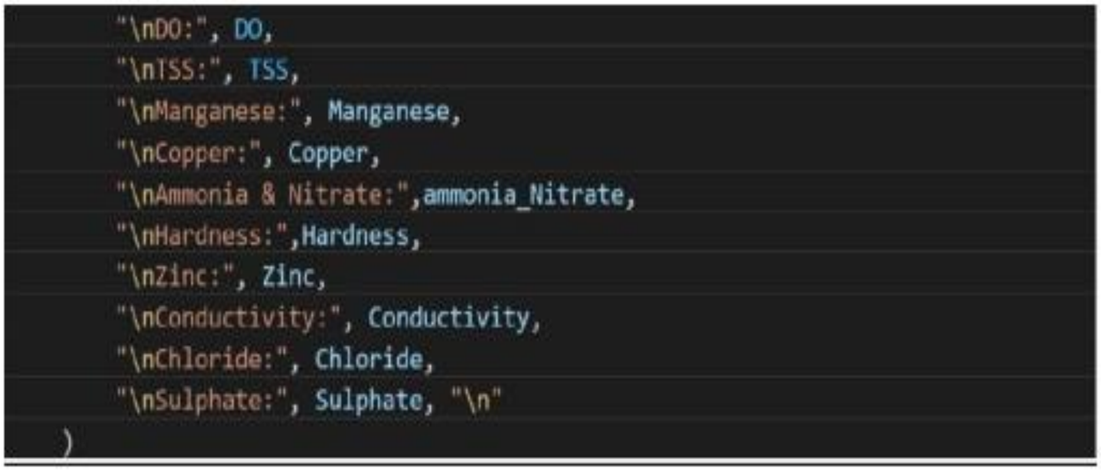
Fig 3

SPRINT 3:

PYTHON CODE:

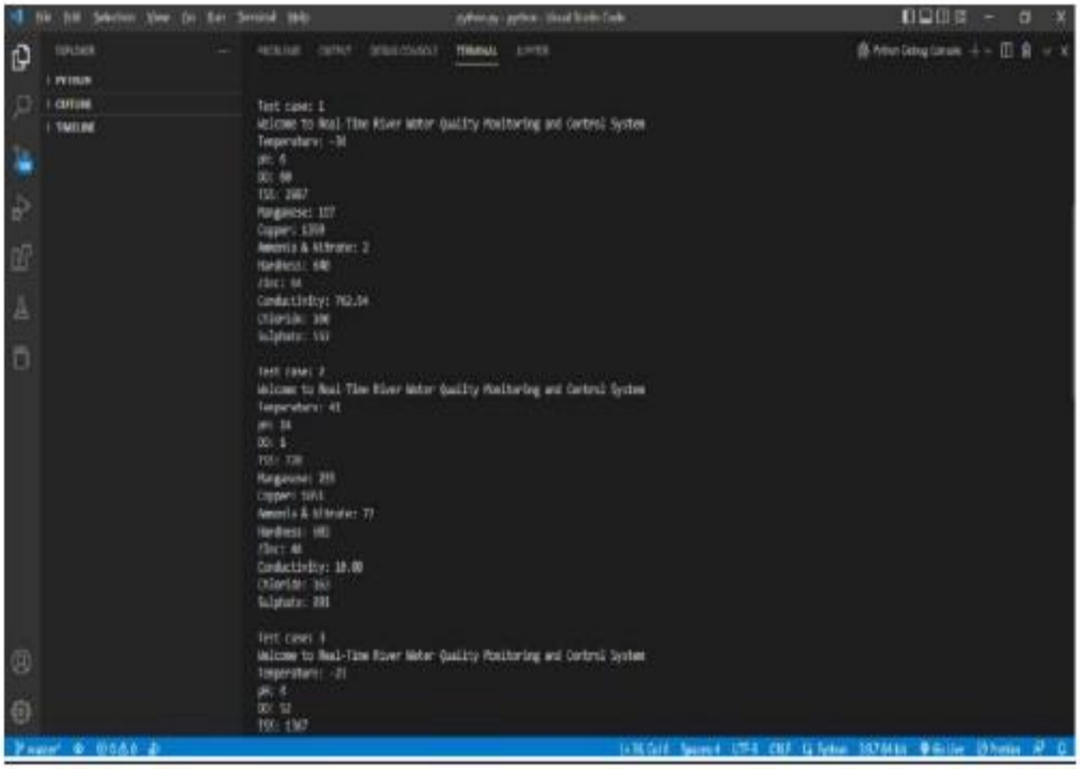


```python
#importing Random function to generate the value
import random as rand


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,
```

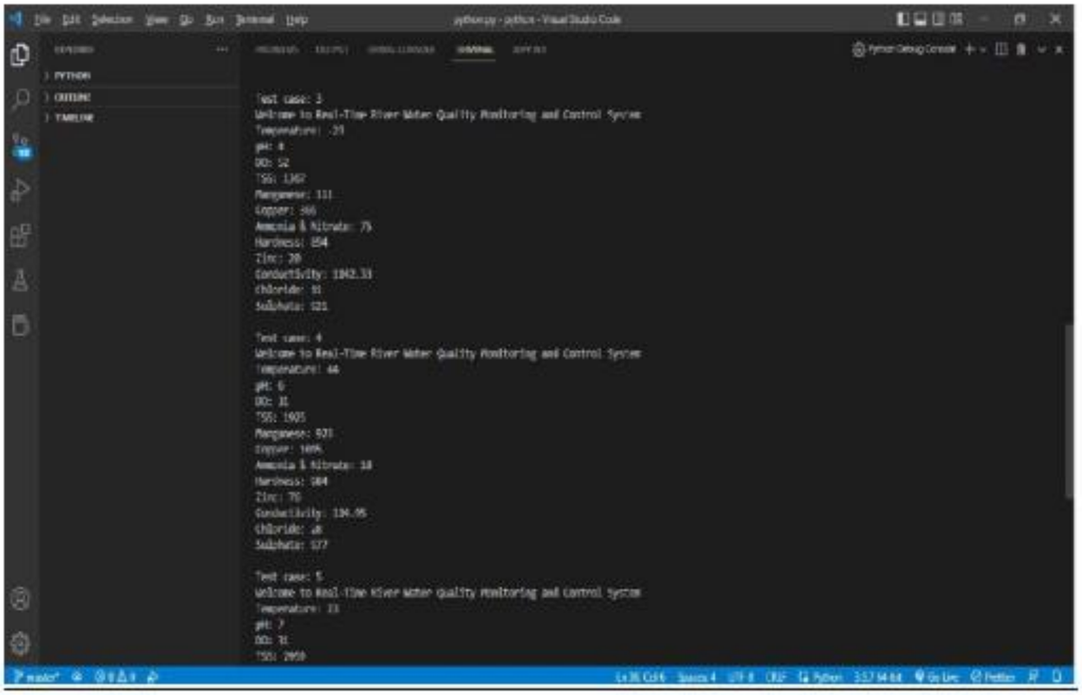


```
        "\nDO:", DO,
        "\nTSS:", TSS,
        "\nManganese:", Manganese,
        "\nCopper:", Copper,
        "\nAmmonia & Nitrate:",ammonia_Nitrate,
        "\nHardness:",Hardness,
        "\nZinc:", Zinc,
        "\nConductivity:", Conductivity,
        "\nChloride:", Chloride,
        "\nSulphate:",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

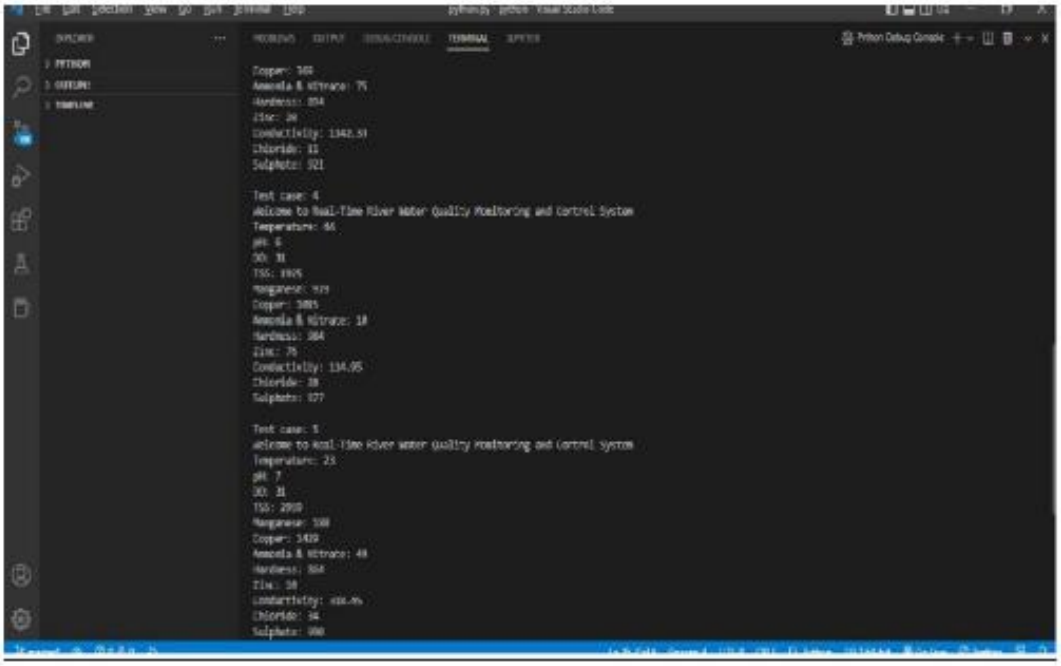
PYTHON OUTPUT:


File Edit Selection View Go Run Terminal Help
python.py - python - Visual Studio Code
EXPLORER
PYTHON
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python Debug Console

```
Test case: 1
Welcome to Real Time River Water Quality Monitoring and Control System
Temperature: -30
pH: 6
DO: 60
TSS: 2667
Manganese: 137
Copper: 1358
Ammonia & Nitrate: 2
Hardness: 640
Zinc: 64
Conductivity: 762.54
Chloride: 306
Sulphate: 553

Test case: 2
Welcome to Real Time River Water Quality Monitoring and Control System
Temperature: 41
pH: 14
DO: 1
TSS: 720
Manganese: 255
Copper: 1661
Ammonia & Nitrate: 77
Hardness: 681
Zinc: 46
Conductivity: 18.00
Chloride: 162
Sulphate: 391

Test case: 3
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -21
pH: 4
DO: 52
TSS: 1367
```

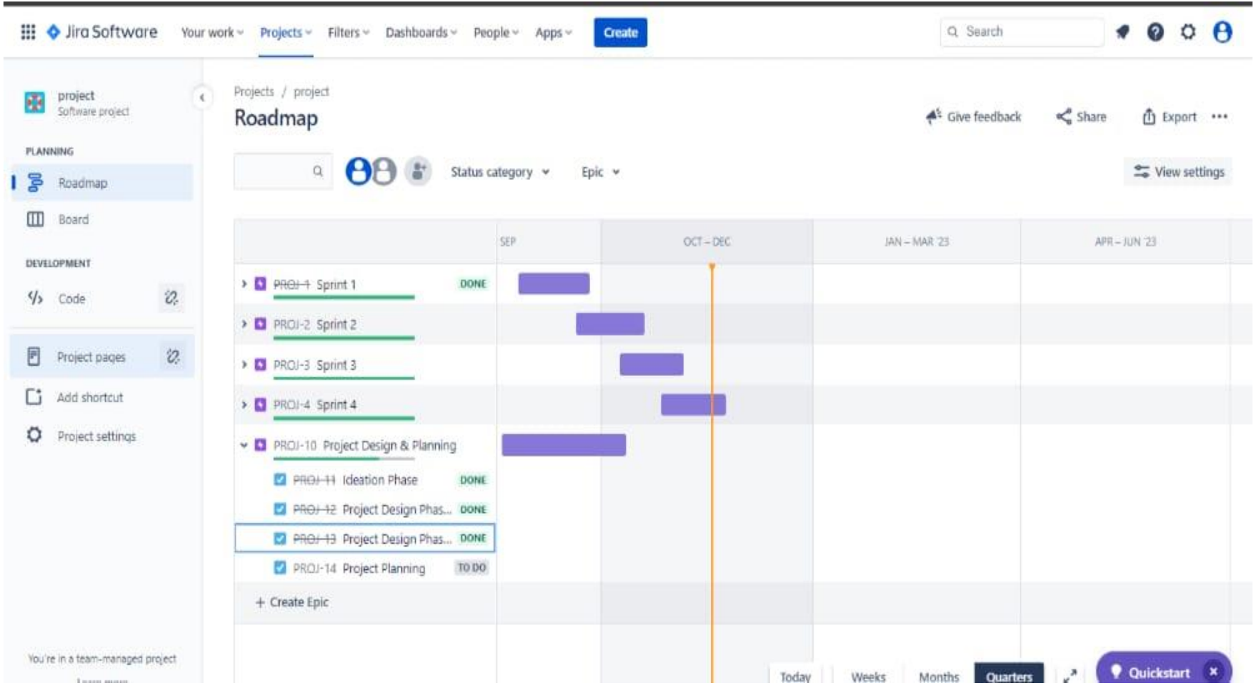


```
Test case: 3
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: -23
pH: 8
DO: 52
TSS: 1367
Manganese: 111
Copper: 566
Ammonia & Nitrate: 75
Hardness: 854
Zinc: 20
Conductivity: 1842.33
Chloride: 31
Sulphate: 521

Test case: 4
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 44
pH: 6
DO: 31
TSS: 1905
Manganese: 929
Copper: 1695
Ammonia & Nitrate: 18
Hardness: 384
Zinc: 76
Conductivity: 134.95
Chloride: 28
Sulphate: 577

Test case: 5
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 23
pH: 7
DO: 31
TSS: 2959
```



```
Copper: 568
Ammonia & Nitrate: 75
Hardness: 854
Zinc: 20
Conductivity: 1842.33
Chloride: 31
Sulphate: 521

Test case: 4
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 46
pH: 6
DO: 31
TSS: 1905
Manganese: 929
Copper: 1095
Ammonia & Nitrate: 18
Hardness: 384
Zinc: 76
Conductivity: 134.95
Chloride: 28
Sulphate: 577

Test case: 5
Welcome to Real-Time River Water Quality Monitoring and Control System
Temperature: 23
pH: 7
DO: 31
TSS: 2959
Manganese: 538
Copper: 1429
Ammonia & Nitrate: 48
Hardness: 364
Zinc: 38
Conductivity: [illegible]
Chloride: 34
Sulphate: 890
```

Report from JIRA:



~~PROJ-1~~ Sprint 1 DONE

PROJ-2 Sprint 2

PROJ-3 Sprint 3

PROJ-4 Sprint 4

PROJ-10 Project Design & Planning

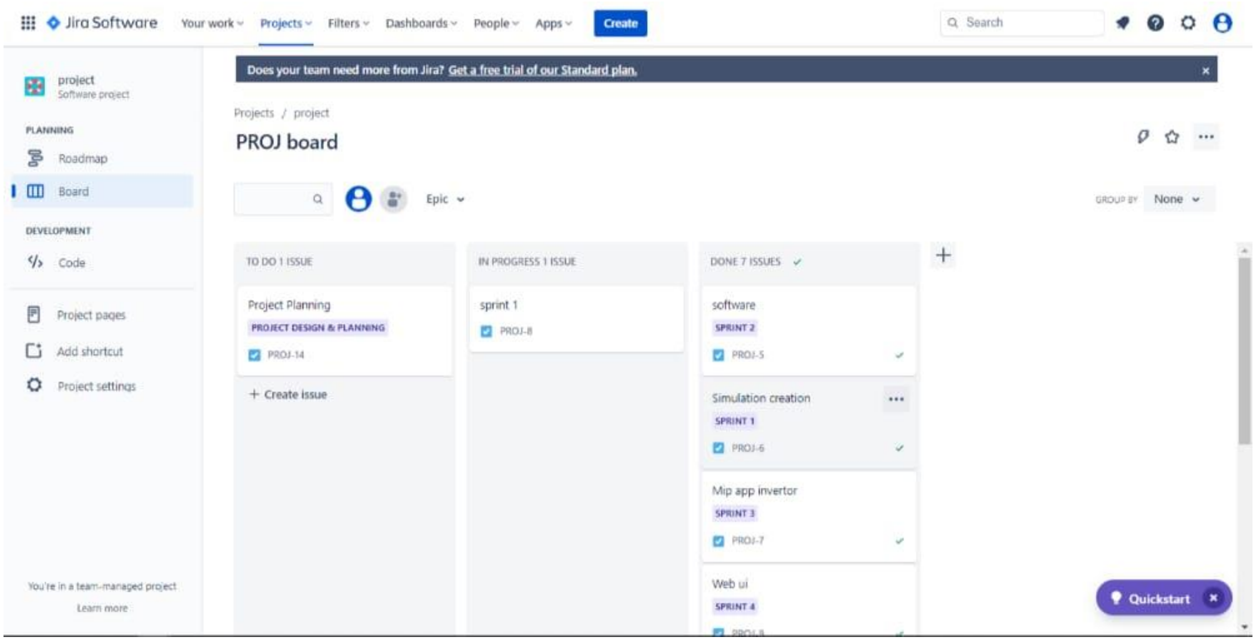
~~PROJ-11~~ Ideation Phase DONE

~~PROJ-12~~ Project Design Phas... DONE

~~PROJ-13~~ Project Design Phas... DONE

PROJ-14 Project Planning TO DO

+ Create Epic



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / project

PROJ board

TO DO 1 ISSUE

Project Planning
PROJECT DESIGN & PLANNING
PROJ-14

+ Create issue

IN PROGRESS 1 ISSUE

sprint 1
PROJ-8

DONE 7 ISSUES ✓

software
SPRINT 2
PROJ-5 ✓

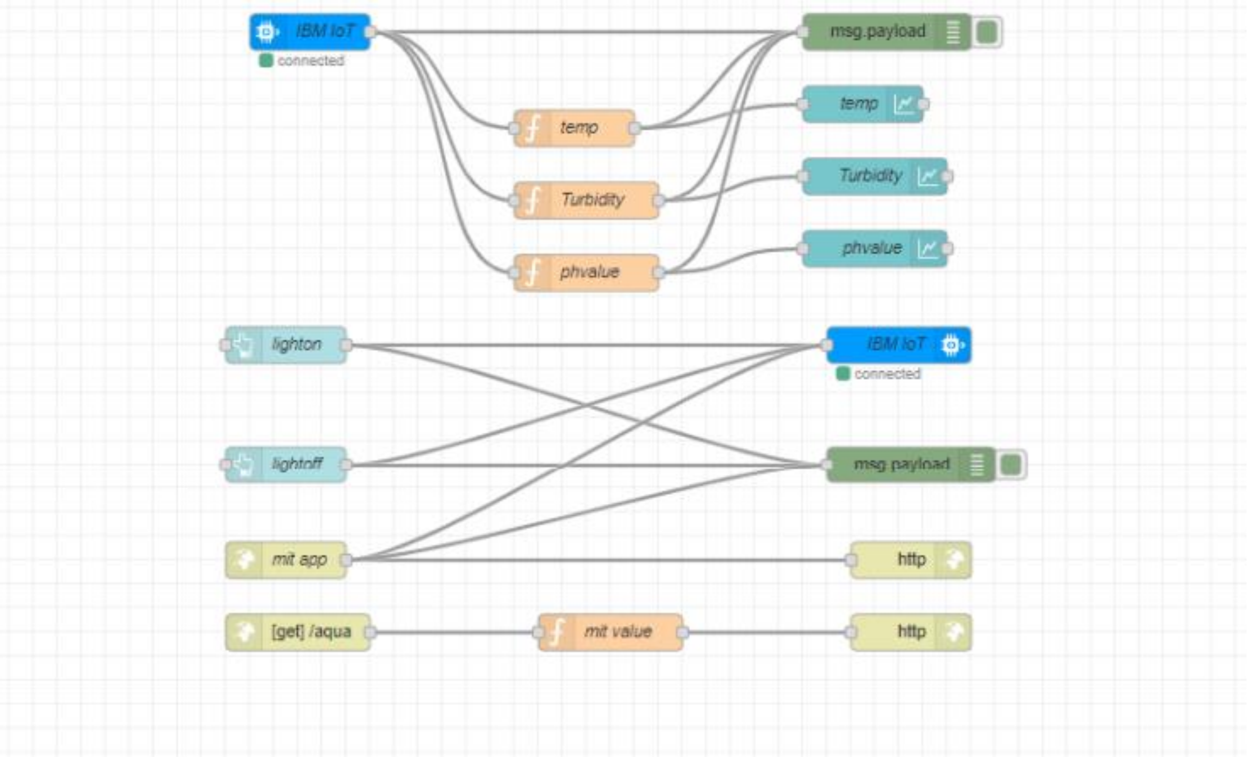
Simulation creation
SPRINT 1
PROJ-6 ✓

Mip app invertor
SPRINT 3
PROJ-7 ✓

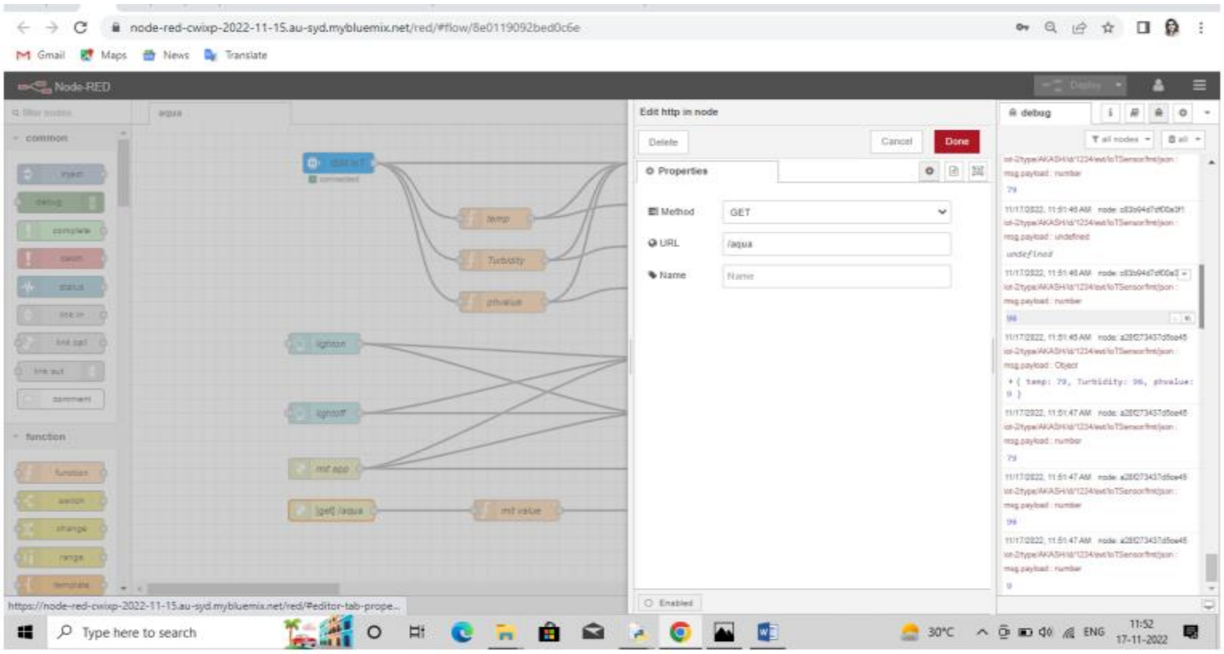
Web ui
SPRINT 4

CODING & SOLUTIONING:

NODE RED SERVICE ASSOCIATED WITH IBM CLOUD:



Node red Outputs:



Edit http in node

Method GET

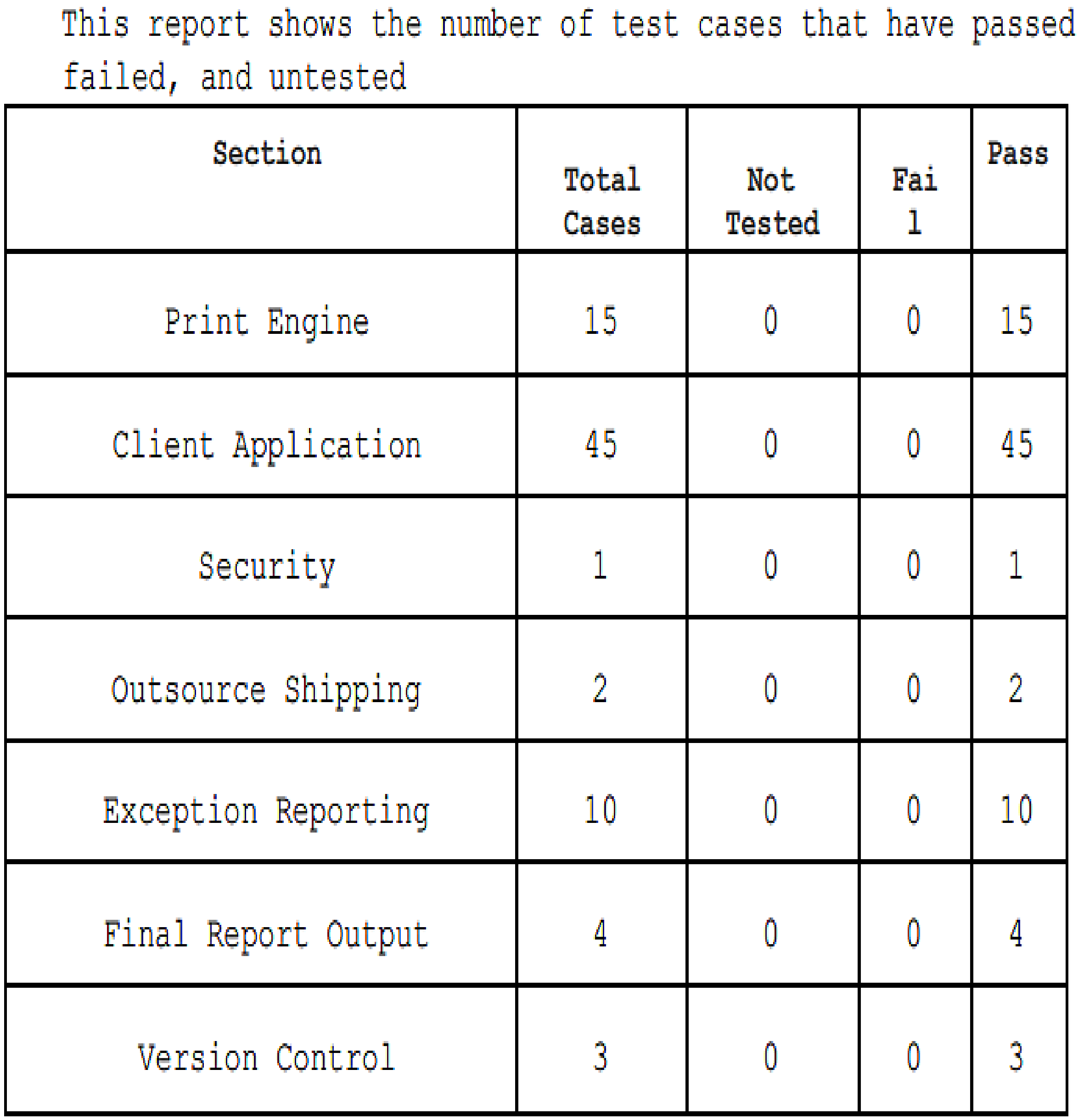
URL /aqua

Name Name

```
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
79
11/17/2022, 11:51:46 AM  node: c83b94d7df00a3f1
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : undefined
undefined
11/17/2022, 11:51:46 AM  node: c83b94d7df00a3
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
96
11/17/2022, 11:51:46 AM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : Object
▸{ temp: 79, Turbidity: 96, phvalue:
9 }
11/17/2022, 11:51:47 AM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
79
11/17/2022, 11:51:47 AM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
96
11/17/2022, 11:51:47 AM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
9
```

TESTING :

Test Case Analysis:



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

Section	Total Cases	Not Tested	Fai l	Pass
Print Engine	15	0	0	15
Client Application	45	0	0	45
Security	1	0	0	1
Outsource Shipping	2	0	0	2
Exception Reporting	10	0	0	10
Final Report Output	4	0	0	4
Version Control	3	0	0	3

USER ACCEPTANCE TESTING:

Test case id	Feature	Component	Test Scenario	Steps to Execute	Test Data	Actual Result	Status
Login page	Functional	Home page	Verify user is able to see the Given app	1.Download the given APK File 2.Click on download button 3.Verify login popup displayed or not"	APK File	Working as expected	Pass
Login page	Functional	Home page	Verify user is able to see the Login/Signup popup when user open the Aqua Meter	1. Download the given APK File 2.Click on download button 3.Verify login popup displayed or not"	APK File	Working as expected	Pass
Login page	Functional	Home page	Verify the UI elements in Login/Signup popup	1. Download the given APK File 2.Click on download button 3.Verify login popup with below UI elements: A .Username text box A .password text box B .Submit button	APK File	Working as Expected	Pass
Login Page	Functional	Home page	Verify user is able to log into application with Valid credentials	"1 Download the given APK File 2.Click on download button 3.Enter Valid "Given " username in Username text box 4.Enter valid password in password text box 5.Click on Submit button"	Username: Username Password: Password	Working as Expected	Pass
Login Page	Functional	Home page	Verify user is able to see the output	1.output displayed	APKFile	Working as Expected	Pass

RESULTS:

Performance Metrics:

This process is used to protect, restore, and enhance environmental quality towards good public health, environmental integrity, and economic viability.

Advantages :

Water quality standards also protect iconic, locally grown products such as wild rice and walleye. Protecting human health — Some pollutants pose risks to human health. Water quality standards protect human health and avoid the costs related to medical care, productivity loss, and even loss of life.

DISADVANTAGES:

Drawback is that this system not reliable for long distance can only apply to single source of water. The pollution of water resource becomes a common human problem. Traditional method of water quality monitor contains the manual collection of water sample from different locations.

conclusion:

Real-time monitoring of water quality by using IoT integrated Big Data Analytics 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 integrated big data analytics is appeared to be a better solution as reliability, scalability, speed, and persistence can be provided. During the project development phase an intense comparative analysis of real-time analytics technologies such as Spark streaming analysis through Spark MLlib, Deep learning neural network models, and Belief Rule Based (BRB) system will be conducted .

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.

APPENDIX:

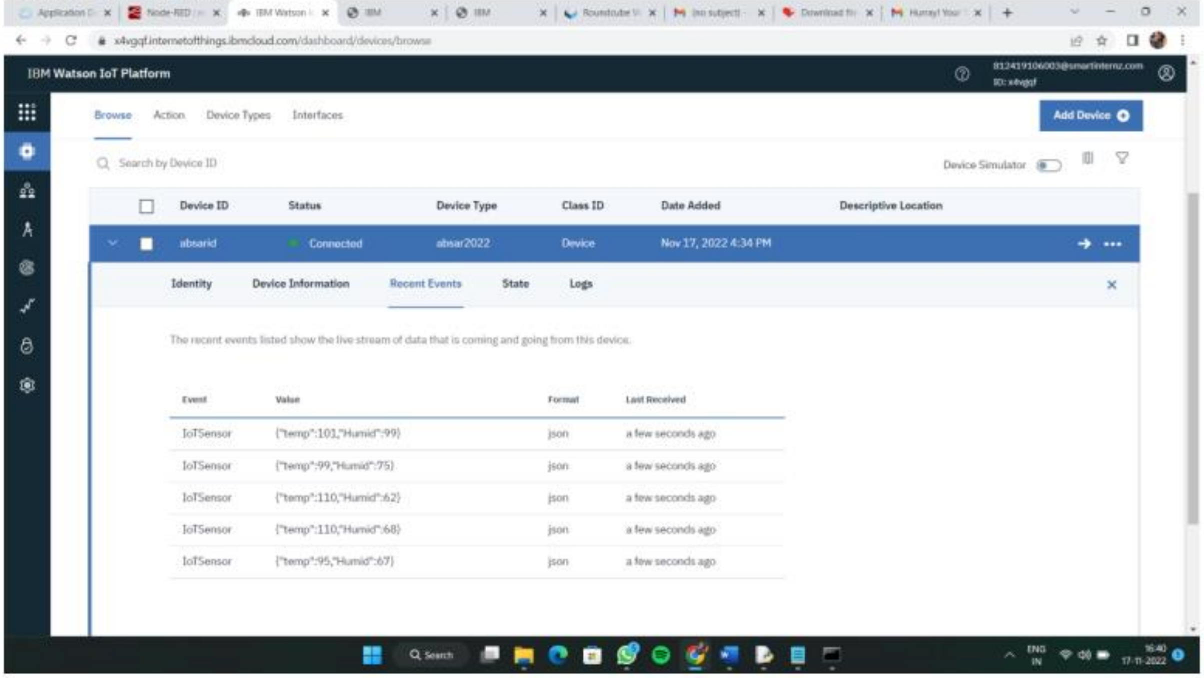
Source Code:

Python Code:

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

#Provide your IBM Watson Device Credentials
organization = "x4vgqf"
deviceType = "absar2022"
deviceId =
"absarid"authMethod =
"token"
authToken = "7YPz)vgxfokT83mkf3"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print("led is on")
    elif status == "lightoff":
        print("led is off")
    else :
        print ("please send proper command")

try:
deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method":
authMethod, "auth-token": authToken}
deviceCli = ibmiotf.device.Client(deviceOptions)
#..............................................
except Exception as e:
print("Caught exception connecting device: %s" % str(e))
sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event oftype
"greeting" 10 times
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11

        temp=random.randint(90,110)
        Humid=random.randint(60,100)
        data = { 'temp' : temp, 'Humid': Humid }
        #printdata
        def myOnPublishCallback():

            print ("Published Temperature = %s C" % temp, "Humidity = %s %%"
%Humid, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

OUTPUT:

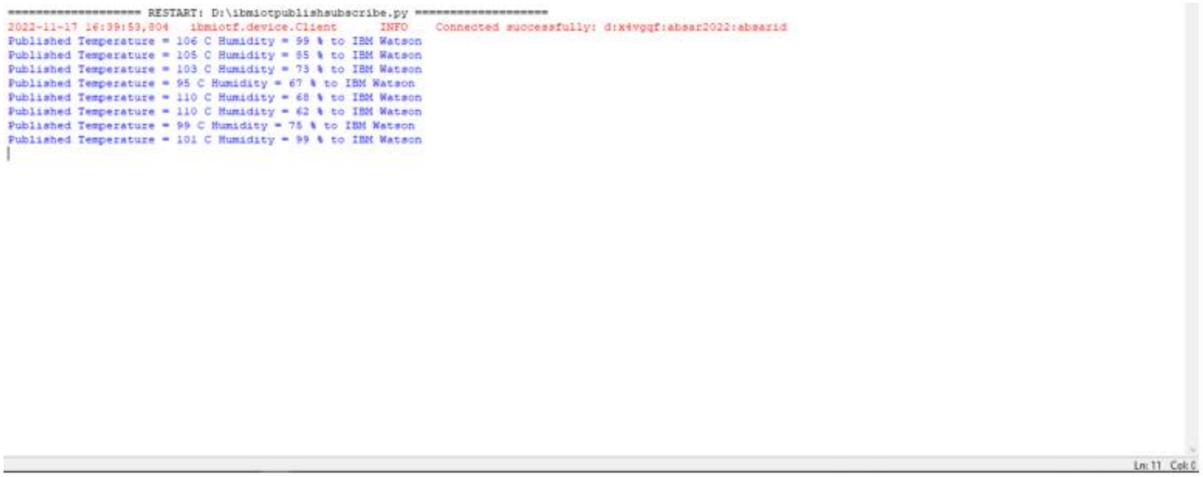
IBM Cloud Data:



Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
absarid	Connected	absar2022	Device	Nov 17, 2022 4:34 PM	

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensor	{"temp":101,"Humid":99}	json	a few seconds ago
IoTSensor	{"temp":99,"Humid":75}	json	a few seconds ago
IoTSensor	{"temp":110,"Humid":62}	json	a few seconds ago
IoTSensor	{"temp":110,"Humid":68}	json	a few seconds ago
IoTSensor	{"temp":95,"Humid":67}	json	a few seconds ago



```
=================== RESTART: D:\ibmiotpublishsubscribe.py ===================
2022-11-17 16:39:53,804   ibmiotf.device.Client     INFO    Connected successfully: d:x4vgqf:absar2022:absarid
Published Temperature = 106 C Humidity = 99 % to IBM Watson
Published Temperature = 105 C Humidity = 85 % to IBM Watson
Published Temperature = 103 C Humidity = 73 % to IBM Watson
Published Temperature = 95 C Humidity = 67 % to IBM Watson
Published Temperature = 110 C Humidity = 68 % to IBM Watson
Published Temperature = 110 C Humidity = 62 % to IBM Watson
Published Temperature = 99 C Humidity = 75 % to IBM Watson
Published Temperature = 101 C Humidity = 99 % to IBM Watson
```

GitHub link:

GitHub Link

Demo link:

DEMO LINK

WOKWI LINK