Project Report

1. INTRODUCTION

a. PROJECT OVERVIEW

Real-Time River Water Quality Monitoring and Control System

Water pollution is a foremost global problem which needs ongoing evaluation and adaptation of water resource directorial principle at the levels of international down to individual wells. It has been studied that water pollution is the leading cause of mortalities and diseases worldwide. The records show that more than 14,000 people die daily worldwide due to water pollution. In many developing countries, dirty or contaminated water is being used for drinking without any proper prior treatment. Current water quality monitoring system is a manualsystem with a monotonous process and is very time-consuming. This projectproposes 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.

b.PURPOSE

Inthe 21st century, there are lots of inventions, but at the same time were pollutions, global warming and so on are being formed, because of this there is no safe drinkingwater for the world's pollution. Nowadays, maintaining pure supply of water to the people is getting more challenging day by day. In India mainly is big cities the municipality corporation use lots of chemical to purify the river water then supply that to the people. And we reserved that water without any test. And we also don't know the water is either safe for drinking or not. And now a day's water quality monitoring in real time faces challenges because of global warming limited water resources, growing population, etc. Hence there is need of developing better methodologies to monitor the water quality parameters in real time. The water parameters pH measures the concentration of hydrogen ions. It shows the water is acidic or alkaline. Pure water has 7pH value, less than 7pH has acidic, more than 7pH has alkaline. The range of pH is 14pH. For drinking purpose it should be 6.5-8.5pH. Turbidity measures the large number of suspended particles in water that is invisible. Higher the turbidity higher the risk of diarrhea, cholera. Lower the turbidity then the water is clean. Temperature sensor measures how the water is, hot or cold. Here in this paper we tried to find the problem and then make a solution for it.

2.LITERATURE SURVEY

a. EXISTING SYSTEM

Central Water Commission (CWC) monitors water quality, by collecting samples from representative locations within the processing & distribution system. These samples are analyzed at the well-equipped laboratories. At these laboratories samples from raw water, filter water and treated water are taken for analysis. The estimation of water parameters like turbidity, pH, dissolved oxygen, etc. is done with the help of meters. So the disadvantages of this existing system are that; there is no continuous and remote monitoring, human resource is required, less reliable, no monitoring at the source of waters i.e. no on field monitoring and the frequency of testing is very low. Due to these disadvantages of the existing system it is required to develop a system that will allow real time and continuous monitoring of water quality.

b. REFERENCES

- Monira Mukta, Samia Islam, SurajitDasBarman and Ahmed Wasif Reza, “IOT based smart water quality monitoring system”- 2019 IEEE 4th International Conference on Computer and CommunicationSystems

- Mohammad Salah Uddin Chowdurya, Talha Bin Emranb, Subhasish Ghosha, Abhijit Pathaka, Mohd.Manjur Alama, Nurul Absara, Karl Anderssonc, Mohammad Shahadat Hossain, “IoT Based Real-time River Water QualityMonitoring System”- The 16th International Conference on Mobile Systems and Pervasive Computing (MobiSPC)August 19-21, 2019, Halifax, Canada

- Shabana Shafi Aijaz Ahmad Reshi, Dr. M Sridhar,Dr. Rajani S.M,“Real Time Sensor Web Enabled Water Quality Monitoring System Using ServiceOriented Architecture”- International Journal of Scientific & Engineering Research, Volume 3, Issue5, May-2012

- Anantha Naik G. D, Dr. GeethaV, “IoT based Real-Time Water Quality Monitoring System using smart Sensors”- International Research Journal of Engineering and Technology (IRJET), Volume 7, Issue9, Sep-2020

- S. Geetha and S. Gouthami, “Internet of things enabledreal time waterquality monitoring system”- Geetha and Gouthami Smart Water(2017)

- Prasanna Kumar S.C, AshwiniKotrappa, K. Jyothi, Soumya B.L.” Real-Time Water QualityMonitoring system”- International Dr Research Journal of Engineering and Technology (IRJET), Volume 5, Issue3, Sep-2019

- Mithila Barabde, Shruti Danve,” Real-Time Water Monitoring and control system”- International Journalof Innovative Research in Computer and

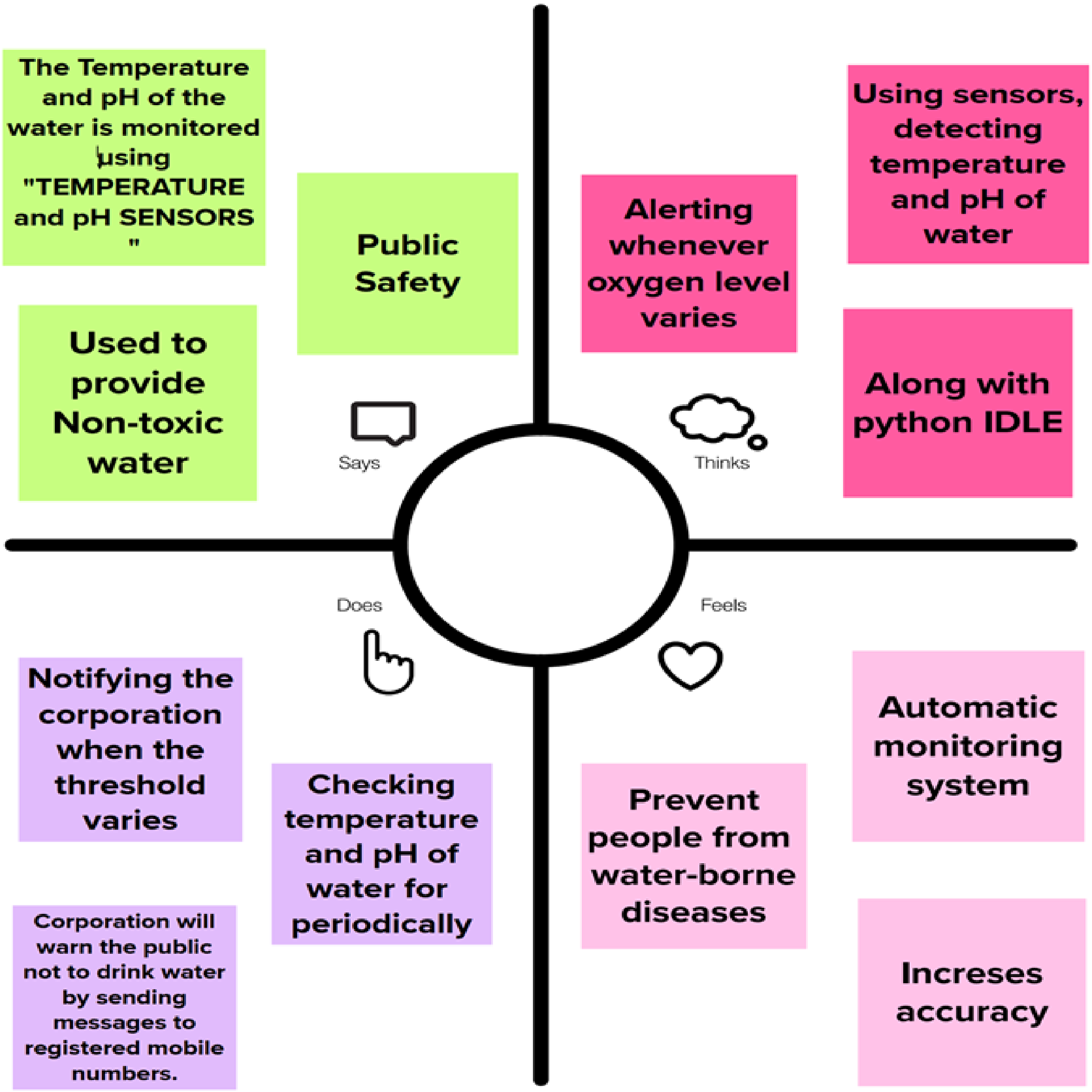
Communication Engineering, Volume 3, Issue 6, June-2015

c.PROBLEM STATEMENT DEFINITION

Due to the fast growing urbanization supply of safe drinking water is a challenge for the every city authority. Water can be polluted any time. So the water we reserved in the water tank at our roof top or basement in our society or apartment may not be safe. Still in India most of the people use simple water purifier that is not enough to get surety of pure water. Sometimes the water has dangerous particles or chemical mixed and general purpose water purifier cannot purify that. And it’s impossible to check the quality of water manually in every time. So an automatic real-time monitoring system is required to monitor the health of the water reserved in our water tank .

3.IDEATION & PROPOSED SOLUTION

a.Empathy Map Canvas



The Temperature and pH of the water is monitored using "TEMPERATURE and pH SENSORS "

Public Safety

Used to provide Non-toxic water

Alerting whenever oxygen level varies

Using sensors, detecting temperature and pH of water

Along with python IDLE

Notifying the corporation when the threshold varies

Checking temperature and pH of water for periodically

Corporation will warn the public not to drink water by sending messages to registered mobile numbers.

Prevent people from water-borne diseases

Automatic monitoring system

Increses accuracy

b. IDEATION & BRAINSTORMING

Ideation phase: The main aim is to develop a system for continuous monitoring of river water quality at remote places using wireless sensor networks with low power consumption, low-cost and high detection accuracy. pH, conductivity, turbidity level, etc. are the limits that are analysed to improve the water quality. Following are the aims of idea implementation. a) To measure water parameters such as pH, dissolved oxygen, turbidity, conductivity, etc. b) using available sensors at a remote place. c) To assemble data from various sensor nodes and send it to the base station by the d) wireless channel. e) To simulate and evaluate quality parameters for quality control. f) To send SMS to an authorized person routinely when water quality detected does not g) match the present standard so that, necessary actions can be taken.

Control surface: An Arduino mega is utilized as a core person. The Arduino victimized here is mega 2560 Because multiple analog sign sensors probe requisite to be conterminous with the Arduino inhabit. It has a set of registers that use as a solon use RAM. Specific intend to know registers for on-chip component resources are also mapped into the assemblage grapheme. The addressability of store varies depending on instrumentation series and all PIC devices someone several banking mechanisms to utilise addressing to additional faculty. Subsequent series of devices have move instructions which can covert move had to be achieved via the register. Thus the mechanism functions with the exploit of coding intrinsically in the Arduino UNO R3 skate. pH sensor: The pH of thing is a useful constant to display because graduate and low pH levels can hump large effects on the author. The pH of a statement can grasp from 1 to 14. A pH sensor is an instrumentation that measures the hydrogen-ion density in a bleach, indicating its tartness or alkalinity. It constitute varies from 0 to 14 pH. Uttermost pH values also process the solubility of elements and compounds making them cyanogenetic. Mathematically pH is referred as, pH = -log [H+]. Turbidity sensor: Turbidity train sensor is victimised to measure the clarity of element or muddiness utter in the water. The muddiness of the open cut food is ordinarily between 255 NTU. Irrigate is visibly at levels above 80 NTU. The standards for intemperance liquid is 130 NTU to 250 NTU. The turbidity device consists of soft sender and acquirer, the transmitter needs to transmit unsubtle bright, it is said to be turbid. The consequence of turbidity is a reduction in water clarify.

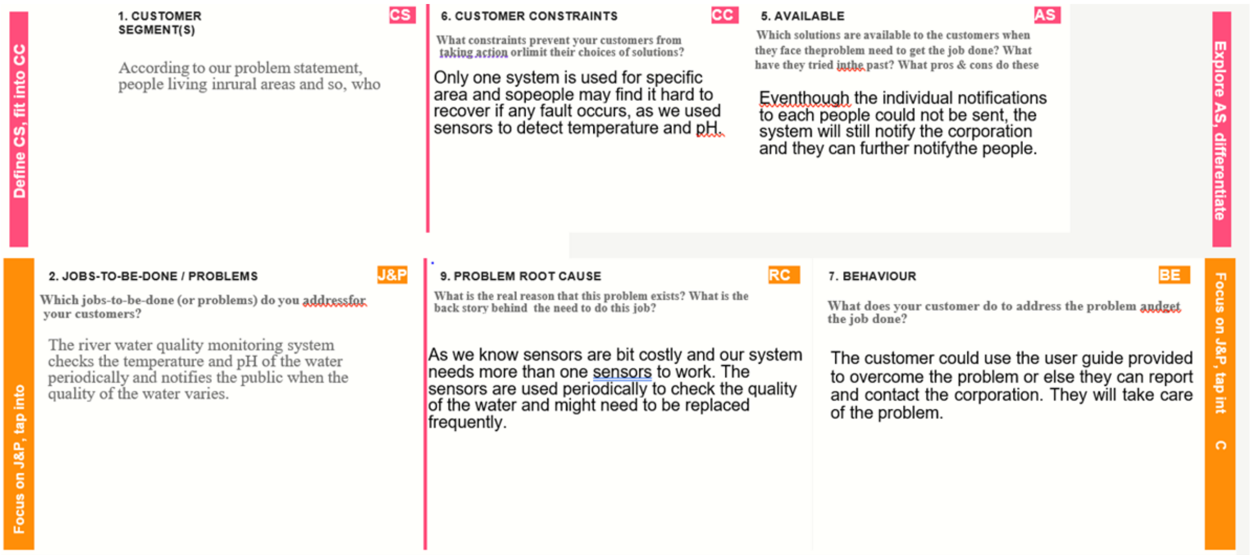
c. PROPOSED SOLUTION

S.NO	PARAMETERS	DESCRIPTIONS

1	Problem Statement (Problem to be solved)	This project represents an IoT (Internet of things) based smart water quality monitoring system that aids in continuous measurement of water condition based on physical parameters
2	ea / Solution description	Field teams deploy sensors strategically at designated points water area to monitor waters within defined measurement parameters
3	Novelty / Uniqueness	The uniqueness of our proposed project is to develop of novel small form factor, low cost sensing technologies.
4	Social Impact / Customer Satisfaction	It socially help water system managers identify threats to surface water earlier, make more fully informed decisions affecting the systems and the public they serve, and comply with ever-changing regulatory water quality monitoring requirements at federal, state and local levels.

5	Business Model (Revenue Model)	The River Water Management and Alert System built on this architecture enable access, control and management of river water pollution.
6	alability of Solution	The Sensor Web architecture for crisis management, described in this project, provides active monitoring of measuring parameters and timely responses in cases of environmental disasters

d. PROBLEM SOLUTION FIT



1. CUSTOMER SEGMENT(S) CS

According to our problem statement, people living inrural areas and so, who

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action orlimit their choices of solutions?

Only one system is used for specific area and sopeople may find it hard to recover if any fault occurs, as we used sensors to detect temperature and pH.

5. AVAILABLE AS

Which solutions are available to the customers when they face theproblem need to get the job done? What have they tried inthe past? What pros & cons do these

Eventhough the individual notifications to each people could not be sent, the system will still notify the corporation and they can further notifythe people.

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you addressfor your customers?

The river water quality monitoring system checks the temperature and pH of the water periodically and notifies the public when the quality of the water varies.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?

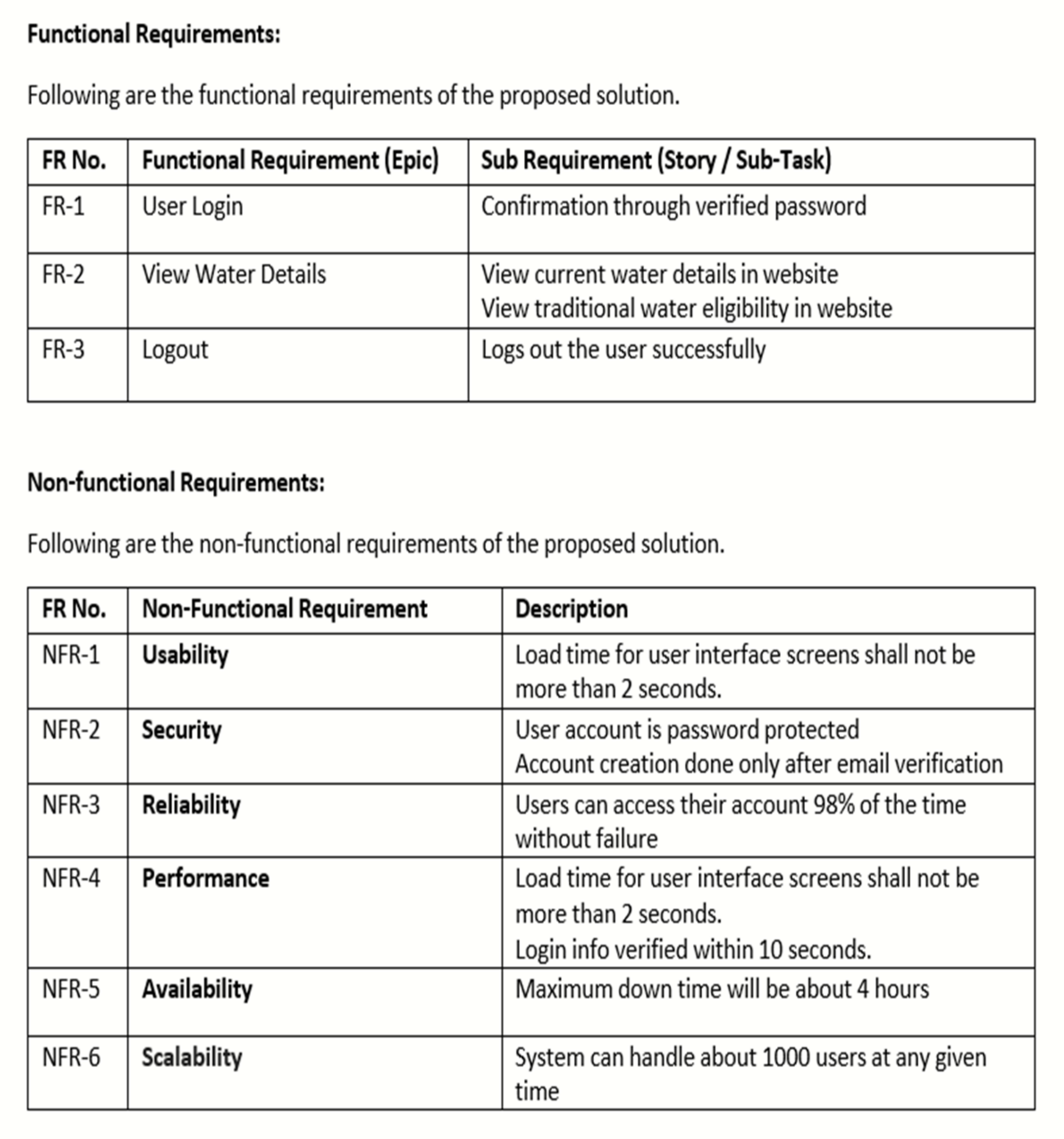
As we know sensors are bit costly and our system needs more than one sensors to work. The sensors are used periodically to check the quality of the water and might need to be replaced frequently.

7. BEHAVIOUR BE

What does your customer do to address the problem andget the job done?

The customer could use the user guide provided to overcome the problem or else they can report and contact the corporation. They will take care of the problem.

4.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

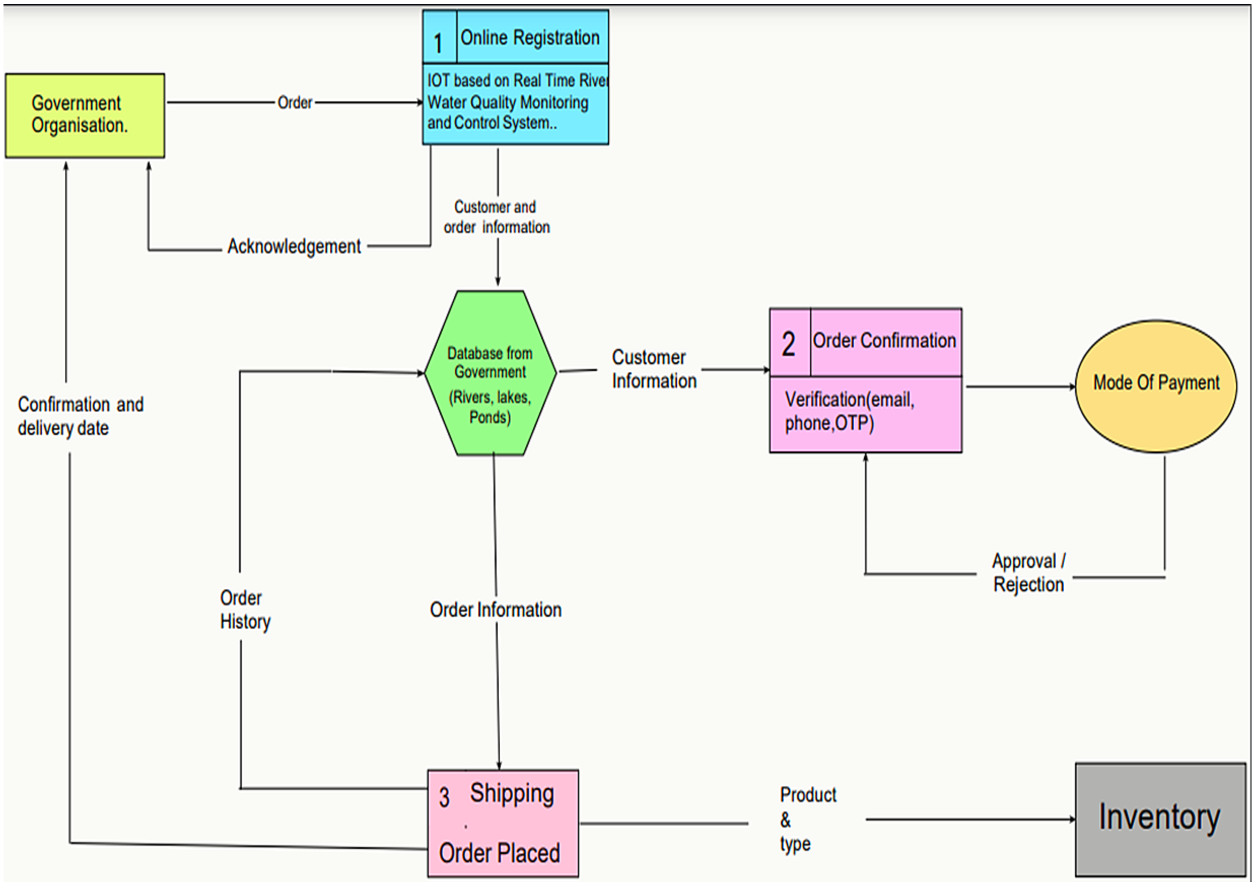
Non-functional Requirements:

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

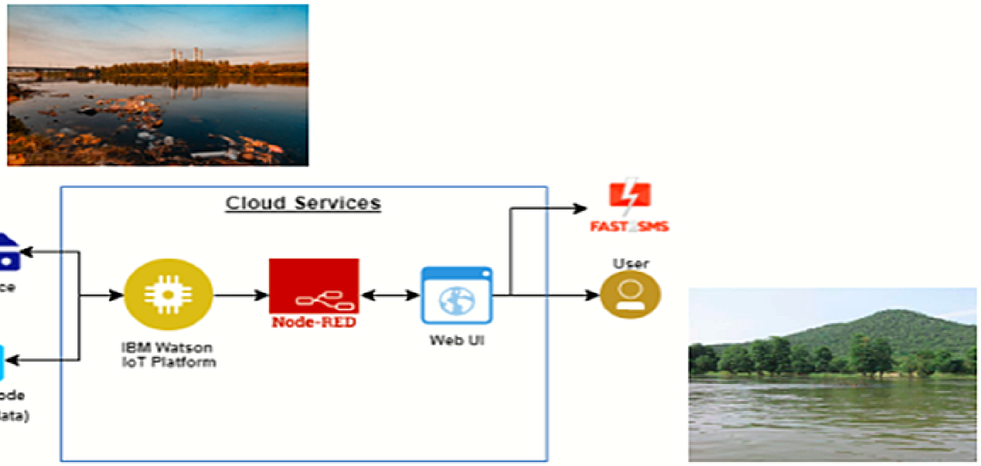
5.PROJECT DESIGN

a.Data Flow Diagrams



b.Solution & Technical Architecture

SOLUTION ARTCHITECTURE DIAGRAM :



IOT BASED REAL TIME RIVER
WATER
QUALITY MONITORING AND
CONTROL SYSTEM.

6. PROJECT PLANNING & SCHEDULING

a.Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story/ Task	Story Points	Priority	Team Members
Sprint-1	Procurement of Hardware requirements (ifneeded)	USN-1	Procurement of quality sensors and actuators, microcontroller that will be required to sense the physical parameters like pH, turbidity andTemperature.	2	High	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V
	Create IBMCloud Services	USN-2	Creation of an IBM Cloud account and registering a device.	2	High	
	Configure the IoT device in IBM Cloud.	USN-3	Creation and registering of a device	1	Medium	

Sprint-2	Development of Python code in IDLE,Install all required libraries.	USN-4	To develop the Python Code to generate random values of pH ,Temperature and turbidity values along withtheir units.	1	Medium	
	Create a IBM Watson IoT service and Publishthevalues generated by python code to Cloud.	USN-5	To create the IBM Watson IoT Platform and integrate themicrocontroller with it,to send the senseddata on cloud	1	High	
Sprint-3	Create a Node RedService	USN-6	To create a node red service to integrate theIBM Watson along with the WebUI	2	Medium	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V
	Create a Web UI	USN-7	To create a Web UI, to access the data from the cloudand display all parameters.	2	Medium	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V

	Generate a link to Interface the node redservice with the Web UI/Mobile app	USN-8	Generate Link to interface the services.	3	High	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V
Sprint-4	Design a Mobile App,to display pH, Temperature and turbidity values	USN-9	To design a Android Appusing MIT Appinventor,to display pH, Temperature and turbidity values.	2	High	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V
	Fast-SMS Service	USN-10	Use Fast SMS to send alert messages once the parameters likepH, Turbidity andtemperaturegoes beyondthe threshold	3	High	Gowsika G Dhatshineswaran B Adithyan P Guruprasad V
	Product Testing	USN-11	Testing of projectand final deliverables	3	Medium	

b.sprint delivery schedule

Sprint	Total Story Points	Duration	Sprint StartDate	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	15	6 Days	24 Oct 2022	31 Oct 2022	15	31 Oct 2022
Sprint-2	15	6 Days	1 Nov 2022	07 Nov 2022	15	07 Nov 2022
Sprint-3	10	6 Days	08 Nov 2022	13 Nov 2022	10	13 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	20 Nov 2022	10	20 Nov 2022

Velocity:

sprint-1 and Sprint-2

Sprint duration

AV =~~Velocity =~~ 15 = 2.14

Sprint-3 and Sprint-4

AV =Sprint duration

~~Velocity~~= 10 = 1.6

7.CODING AND SOLUTIONING

a.FEATURE 1

i. IOT device
ii. IBM Watson platform
iii. Node red
iv. Cloudant DB
v. Web UI
vi. Geofence
vii. MIT App
viii. Python code

b.FEATURE 2

1. Login page
2. OTP Verification
3. Recent value notable
4. Longer user can be monitor
5. NO Payment Charger
6. Instant value can be notified

8.**TESTING**

a.Test Cases

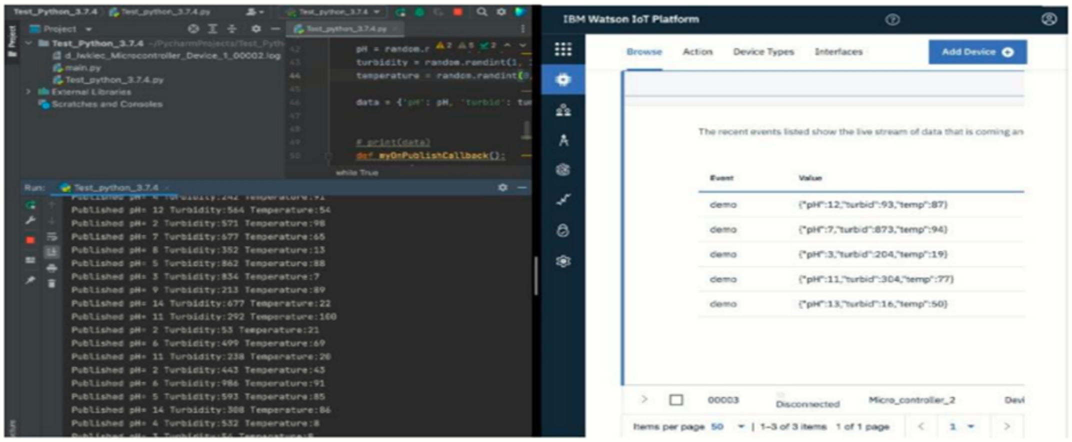
TSK 187242 FINAL_PYTHON_SCRIPT_IBM PYTHON SCRIPT

```
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
from twilio.rest import Client
import keys
Client = Client(keys.account_sid, keys.auth_token)
organization = "lwkiec"
deviceType = "Microcontroller_Device_1"
deviceId = "00002"
authMethod = "token"
authToken = "sushi@123"
pH = random.randint(1, 14)
turbidity = random.randint(1, 1000)
temperature = random.randint(0, 100)
def myCommandCallback(cmd):
 print("Command Received: %s" % cmd.data['command'])
 print(cmd)
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()
deviceCli.connect()
```

```
while True:
 pH = random.randint(1, 14)
 turbidity = random.randint(1, 1000)
 temperature = random.randint(0, 100)
 data = {'pH': pH, 'turbid': turbidity, 'temp': temperature}
 def SMS():
 message = Client.messages.create(
 body="ALERT!! THE WATER QUALITY IS DEGRADED",
 from_=keys.twilio_number,
 to = keys.target_number)
 print(message.body)
 if temperature>70 or pH<6 or turbidity>500:
 SMS()
 def myOnPublishCallback():
 print("Published pH= %s" % pH, "Turbidity:%s" % turbidity,
"Temperature:%s" %
temperature)
 success = deviceCli.publishEvent("demo", "json", data, qos=0,
on_publish=myOnPublishCallback)
 if not success:
 print("Not Connected to ibmiot")
 time.sleep(5)
 deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```

output

TSK 187343 PUBLISH DATA_IBM CLOUD



```
    pH = random.r
    turbidity = random.randint(1, 
    temperature = random.randint(0,

    data = {'pH': pH, 'turbid': tu


    # print(data)
    def myOnPublishCallback():
while True
```

```
Published pH= 4 Turbidity:242 Temperature:91
Published pH= 12 Turbidity:564 Temperature:54
Published pH= 2 Turbidity:571 Temperature:98
Published pH= 7 Turbidity:677 Temperature:65
Published pH= 8 Turbidity:352 Temperature:13
Published pH= 5 Turbidity:862 Temperature:88
Published pH= 3 Turbidity:834 Temperature:7
Published pH= 9 Turbidity:213 Temperature:89
Published pH= 14 Turbidity:677 Temperature:22
Published pH= 11 Turbidity:292 Temperature:100
Published pH= 2 Turbidity:53 Temperature:21
Published pH= 6 Turbidity:499 Temperature:69
Published pH= 11 Turbidity:238 Temperature:20
Published pH= 2 Turbidity:443 Temperature:43
Published pH= 6 Turbidity:986 Temperature:91
Published pH= 5 Turbidity:593 Temperature:85
Published pH= 14 Turbidity:308 Temperature:86
Published pH= 4 Turbidity:532 Temperature:8
Published pH= 3 Turbidity:54 Temperature:5
```

The recent events listed show the live stream of data that is coming an

Event	Value
demo	{"pH":12,"turbid":93,"temp":87}
demo	{"pH":7,"turbid":873,"temp":94}
demo	{"pH":3,"turbid":204,"temp":19}
demo	{"pH":11,"turbid":304,"temp":77}
demo	{"pH":13,"turbid":16,"temp":50}

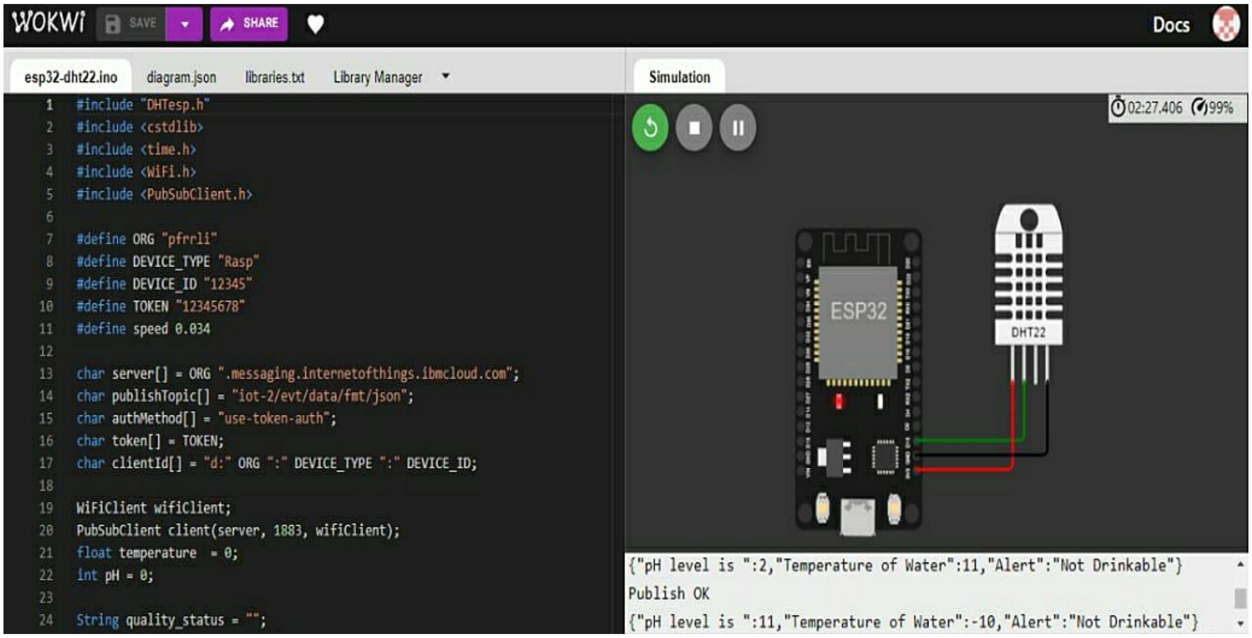
> □ 00003 Disconnected Micro_controller_2 Devi

Items per page 50 | 1–3 of 3 items 1 of 1 page < 1 >

9.RESULTS

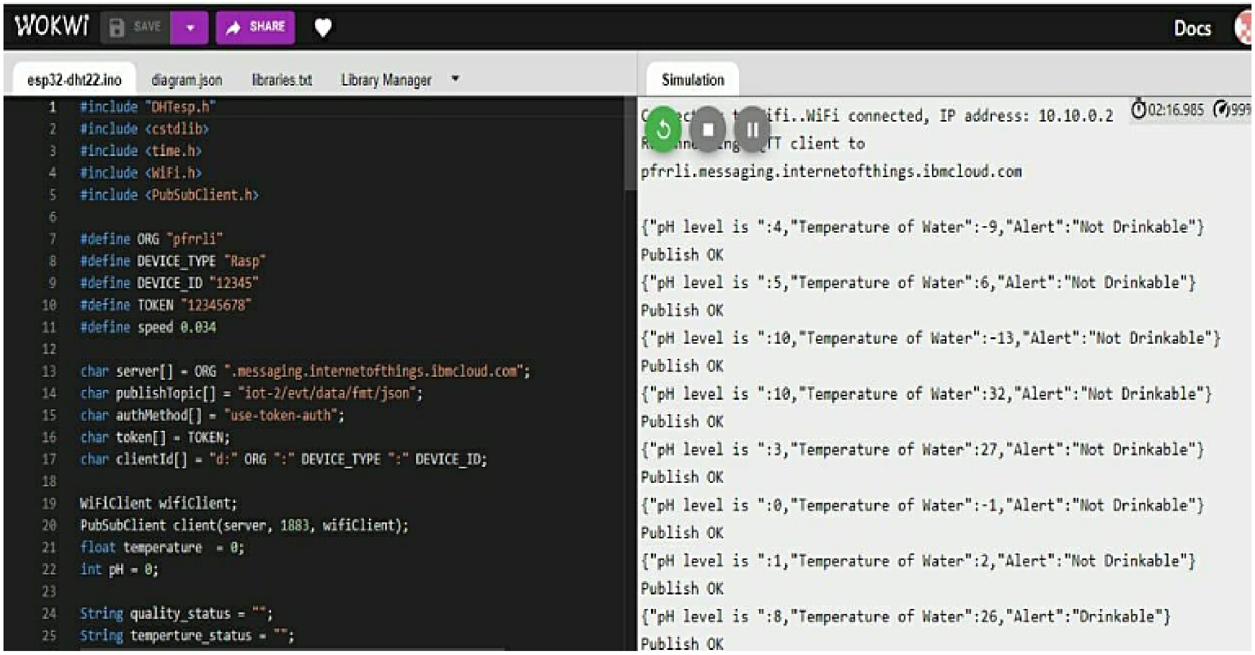
a.Performance Metrics

It continuously senses the values of pH, temp, turbidity, and ORP and the resulting values are displayed to the LCD, PC or mobile in real-time. If the acquired value is above the threshold value comments will be displayed as ‘BAD’. If the acquired value is lower than the threshold value comments will be displayed as ‘GOOD’. A bar/line graph will also be shown for perfect understanding.



```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

#define ORG "pfrrli"
#define DEVICE_TYPE "Rasp"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "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, wifiClient);
float temperature  = 0;
int pH = 0;

String quality_status = "";
```

```
{"pH level is ":2,"Temperature of Water":11,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":11,"Temperature of Water":-10,"Alert":"Not Drinkable"}
```



```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

#define ORG "pfrrli"
#define DEVICE_TYPE "Rasp"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "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, wifiClient);
float temperature  = 0;
int pH = 0;

String quality_status = "";
String temperture_status = "";
```

```
Connecting to Wifi..WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
pfrrli.messaging.internetofthings.ibmcloud.com

{"pH level is ":4,"Temperature of Water":-9,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":5,"Temperature of Water":6,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":10,"Temperature of Water":-13,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":10,"Temperature of Water":32,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":3,"Temperature of Water":27,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":0,"Temperature of Water":-1,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":1,"Temperature of Water":2,"Alert":"Not Drinkable"}
Publish OK
{"pH level is ":8,"Temperature of Water":26,"Alert":"Drinkable"}
Publish OK
```

10.ADVANTAGES & DISADVANTAGES

The recent advances in technologies enable the application of web-based data platforms for analysing real-time data for water quality management. Efficient and realtime monitoring of water quality as a key component of water quality management can predict future trends of water quality and enable rapid response to water quality events .

Drawback is high cost for smart sensors. In this system uses wireless sensors for monitoring quality of water parameters monitored are ph, turbidity, conductivity, temperature. A micro controller has the task of signal digitalizing, data transmission, network management.

11.CONCLUSION

Monitoring **provides the objective evidence necessary to make sound decisions on managing water quality today and in the future**. 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 beneficial uses of water.

12.FUTURE SCOPE

- Increase their personal connection with and appreciation for the natural world, especially the waterbodies they monitor.
- Produce water quality data needed to understand and protect our watersheds and aquatic resources.
- Become aware of water quality issues and aquatic resources and engaged in effective watershed stewardship, especially pollution prevention.
- Help to identify problem areas that need further investigation and reduce the risk of water quality degradation.
- Capture the excitement and attention of local communities and fellow citizens to water quality issues, thereby protecting drinking water quality and human health

13.APPENDIX

Source Code

```
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
from twilio.rest import Client
import keys
Client = Client(keys.account_sid, keys.auth_token)

organization = "lwkiec"
deviceType = "Microcontroller_Device_1"
deviceId = "00002"
authMethod = "token"
authToken = "sushi@123"


pH = random.randint(1, 14)
turbidity = random.randint(1, 1000)
temperature = random.randint(0, 100)


def myCommandCallback(cmd):
    print("Command Received: %s" % cmd.data['command'])
    print(cmd)


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()

deviceCli.connect()

while True:

    pH = random.randint(1, 14)
    turbidity = random.randint(1, 1000)
    temperature = random.randint(0, 100)

    data = {'pH': pH, 'turbid': turbidity, 'temp': temperature}
    def SMS():
        message = Client.messages.create(
            body="ALERT!! THE WATER QUALITY IS DEGRADED",
            from_=keys.twilio_number,
             to = keys.target_number)
        print(message.body)

    if temperature>70 or pH<6 or turbidity>500:
        SMS()

    def myOnPublishCallback():
        print("Published pH= %s" % pH, "Turbidity:%s" % turbidity, "Temperature:%s" % temperature)


    success = deviceCli.publishEvent("demo", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not Connected to ibmiot")
    time.sleep(5)
    deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
#Twilio Account Credentials
account_sid ='ACa0eb9bf43aa629b503bdd01d0962d465'
auth_token ='48c1a0ade0472038ab36d45e0d9fb6e7'
twilio_number ='+19804095xxx'
target_number ='+919940555xxx'
```

GitHub& Project Demo Link

https://github.com/IBM-EPBL/IBM-Project-22568-1659854024
https://youtu.be/c4RxoKj0Gn0

https://drive.google.com/file/d/1hbXYXGrQ799grUEJ22Em3Vjf7smdpegA/view?usp=drivesdk