GUDLAVALLERU ENGINEERING COLLEGE



**ELECTRONICS AND COMMUNICATION ENGINEERING**

**PROJECT NAME: RIVER WATER QUALITY MONITORING SYSTEM**

SUBMITTED BY: A12 BATCH

D. Guru Rajesh(18481A0449)

M.Ananya(19485A0408)

N.L.V.Susmitha (19485A0410)

P.Komali(19485A0411)

**INDEX**

**TITLE PAGE NO**

1. INTRODUCTION

1.1 Overview 3

1.2 Purpose 3

2. Literature Survey

2.1 Existing problem 4

2.2 Proposed solution 4

3. Theoretical Analysis

3.1 Block diagram 5

3.2 Software designing 5

4. Experimental Investigations 6

5. Flowchart 8

6. Result 9

7. Advantages & Disadvantages 10

8. Applications 10

9. Conclusion 11

10. Future Scope 11

11. Bibliography 11

12. Appendix

12.1 Source code 12

12.2 UI output Screenshot 13

**1.INTRODUCTION**

* 1. **Overview:**

Current River 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 while we are doing project in Hardware ,but as we are doing in software we use the IBM platform to complete this project in an effective manner.

Real-time data access can be done by using remote monitoring and Internet of Things (IoT) technology. 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 watermonitoring system with high frequency, high mobility,and low powered. In our project we find ph value and water quality of river water and if the values of these parameters is exceeded then a message will be send to the authorities by using the fast sms .

* 1. **Purpose**

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 compliance with drinking water standards, and to protect other benecial uses of water.

As we know when any floods , cyclones etc takes place then water parameters like ph value , turbidity nothing but water quality are changed . if these parameters are not in range then peoples will effect with health problems ,so to avoid these effects we implemented the river water quality monitoring system using IBM Watson. By using this system we can identify if any changes takes place in water and also we can alert the authorities for taking care of people ,by announcing “don’t drink river water, water is not pure “.

**2.LITERATURE SURVEY**

**Existing Problem:**

The quality of any body of surface or ground water is a function of either both natural influences and human influences. Globally, the most prevalent water quality problem is eutrophication, a result of high-nutrient loads (mainly phosphorus and nitrogen), which substantially impairs beneficial uses of water.

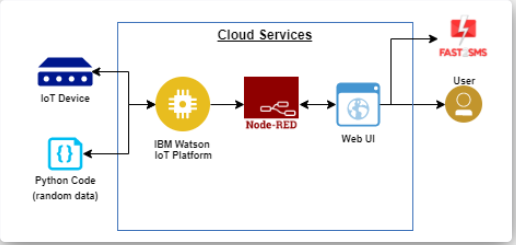
Poor water quality can pose a health risk for people. Too many nutrients in the water can cause excess growth of algae, which can smother corals and seagrass. Pollutants such as metals, oils, pesticides, and fertilizers run off from land into the waters, causing excess algae growth and other harmful impacts.

**2.2 Proposed Solution:**

A controller forms the central part of the IOT enabled water quality monitoring system.The sensor parameters such as turbidity, pH are measured by placing the sensor into different solutions of water.by using this system we can monitor the water parameters continuesly and we can alert the authorities if any change in water quality.

**3.THEORETICAL ANALYSIS**

**3.1 Block Diagram:**



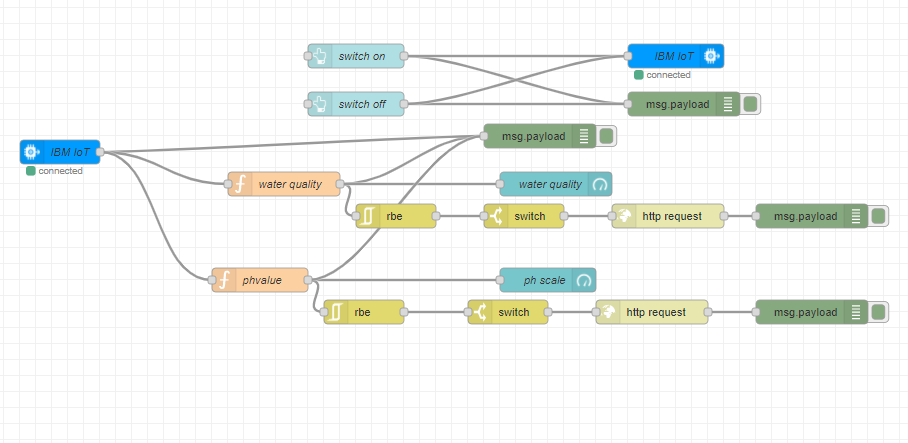
**3.2 Software Designing:**

**Tools used:**

* NODE-RED
* IBM Watson
* Python IDLE

**4. EXPERIMENTAL INVESTIGATIONS**

**NOTE RED FLOW:**

****

**CODE:**

from ibm\_watson import TextToSpeechV1

from ibm\_cloud\_sdk\_core.authenticators import IAMAuthenticator

from playsound import playsound

import wiotp.sdk.device

import time

import random

import os

myConfig = {

"identity": {

"orgId": "aeqag9",

"typeId": "iotdevice",

"deviceId":"1234"

},

"auth": {

"token": "1234567890"

}

}

authenticator = IAMAuthenticator('8oF1v86EliXvM143bGIGFHjzzWOxvlhjdb15k\_FAP3LA')

text\_to\_speech = TextToSpeechV1(

authenticator=authenticator

)

text\_to\_speech.set\_service\_url('https://api.us-south.text-to-speech.watson.cloud.ibm.com/instances/8f6159d7-7bc0-4e7b-89ef-2eadd048b9bd')

def myCommandCallback(cmd):

print("Message received from IBM IoT Platform: %s" % cmd.data['command'])

m=cmd.data['command']

def phvalue():

with open('new.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'ph value of water is exceeded ,dont drink river water',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new.mp3')

time.sleep(1)

os.remove('new.mp3')

return

def waterquality():

with open('new1.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'water quality is not good ,dont drink river water',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new1.mp3')

time.sleep(1)

os.remove('new1.mp3')

return

def both():

with open('new2.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'water is not suitable for drinking',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new2.mp3')

time.sleep(1)

os.remove('new2.mp3')

return

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect()

while True:

wq=random.randint(0,100)

ph=float("{:.1f}".format(random.uniform(1,14)))

myData={'waterquality':wq, 'phvalue':ph}

if ph>=8.5 and wq<=70:

both()

elif wq<=70:

waterquality()

elif ph>=8.5:

phvalue()

time.sleep(2)

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)

print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

client.disconnect()

**5 . FLOW CHART**

START

RUN the python code

Deploy the note red flow

Observe the water parameter changes in web ui

If wq<=70

If ph >=8.5

yes

yes

Message is send through fast sms as “ don’t drink river water,water quality is not good”

Message is send through fast sms as “ don’t drink water,ph value is exceeded”

The message we send is also converted into speech at python output

**pH sensor:**

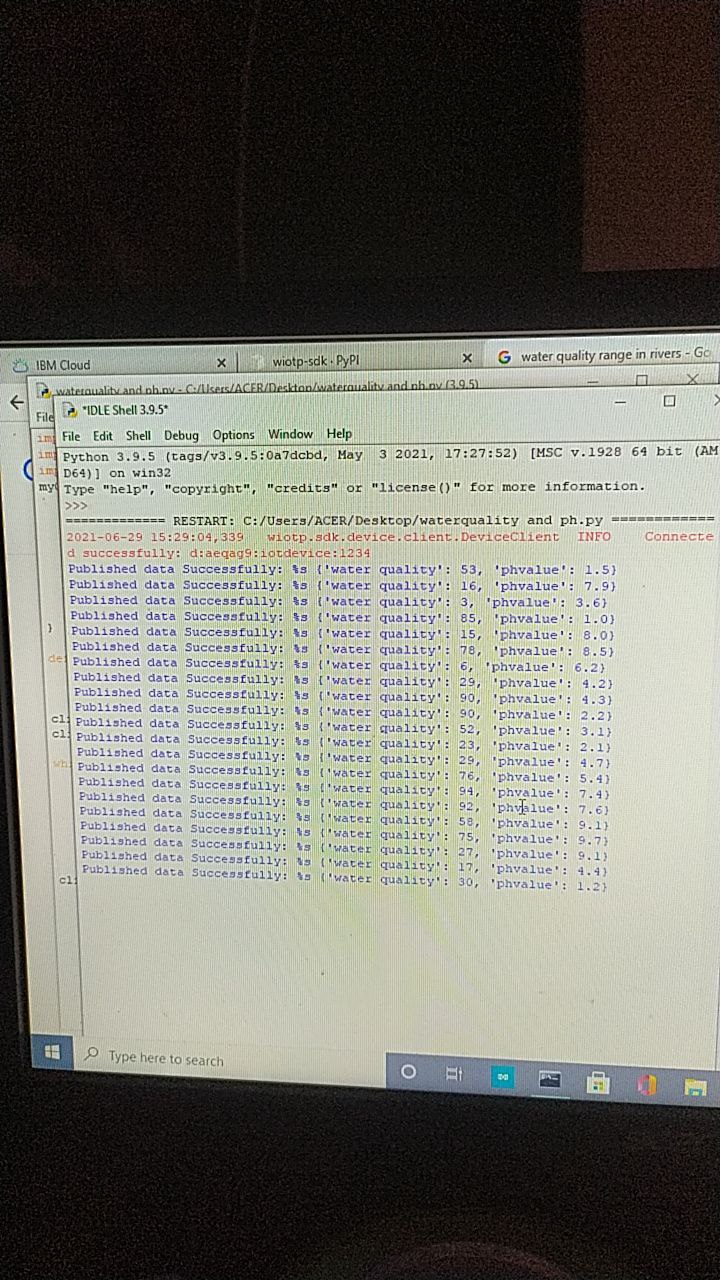
The pH of a solution is the measure of the acidity or alkalinity of that solution. The pH scale is a logarithmic scale whose range is from 0-14 with a neutral point being 7. Values above 7 indicate a basic or alkaline solution and values below 7 would indicate an acidic solution. It operates on 5V power supply and it is easy to interface with arduino.The normal range of pH is 6 to 8.5.

**Turbidity sensor:**

Turbidity is a measure of the cloudiness of water. Turbidity has indicated the degree at which the water loses its transparency. It is considered as a good measure of the quality of water. Turbidity blocks out the light needed by submerged aquatic vegetation. It also can raise surface water temperatures above normal because suspended particles near the surface facilitate the absorption of heat from sunlight.

**6.RESULT**

**PYTHON SHELL OUTPUT:**



**7.ADVANTAGES &DISADVANTAGES**

**ADVANTAGES:**

* Irrigation and soil fertility
* Allows life sustenance (e.g. feed on fish and access to fresh water)
* Entertainment. People can swim and play, or go fishing, kayaking etc.
* Can be used as natural borders

**DISADVANTAGES:**

* Rivers can dry out and their course change over time too
* Water is not always safe to drink
* In some tropical rivers, some animals could spoil your fun. E.g. crocodiles, piranhas or anacondas
* Rivers constitute natural obstacles to one’s mobility on the territory. You might have to build a bridge to get across or go around altogether

**APPLICATIONS:**

* Navigation
* water power production
* Fish culture
* water supply
* Habitation
* Recreation industry
* Commerce and transport
* Mining
* waste water disposal
* International borders and harbors.

**9.CONCLUSION**

we are displaying the resulting sensed pH, temperature. It continuously senses the values of pH, temperature are displayed in nodered. . 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. 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.

**10.FUTURE SCOPE**

Monitoring of Turbidity, PH & Temperature of Water makes use of water detection sensor with unique advantage and existing GSM network. The system can monitor water quality automatically, and it is low in cost and does not require people on duty. So the water quality testing is likely to be more economical, convenient and fast. The system has good flexibility. Only by replacing the corresponding sensors and changing the relevant software programs, this system can be used to monitor other water quality parameters. The operation is simple. The system can be expanded to monitor hydrologic, air pollution, industrial and agricultural production and so on. It has widespread application and extension value.

In future we use IOT concept in this project

• Detecting the more parameters for most secure purpose

• Increase the parameters by addition of multiple sensors

• By interfacing relay we controls the supply of water

**11.BIBLIOGRAPHY**

* <https://www.electronicsforu.com/electronics-projects/sms-based-smart-notice-board>
* <https://www.ijert.org/iot-based-smart-notice-board>
* <https://how2electronics.com/iot-web-controlled-notice-board-esp8266/>

**12.APPENDIX**

**12.1 Source Code:**

from ibm\_watson import TextToSpeechV1

from ibm\_cloud\_sdk\_core.authenticators import IAMAuthenticator

from playsound import playsound

import wiotp.sdk.device

import time

import random

import os

myConfig = {

"identity": {

"orgId": "9dump4",

"typeId": "iotdevice",

"deviceId":"1001"

},

"auth": {

"token": "1234567890"

}

}

authenticator = IAMAuthenticator('V3xba4kA3nL555d7KMtjb\_oJmn47yhVtFAY49v2j8bau')

text\_to\_speech = TextToSpeechV1(

authenticator=authenticator

)

text\_to\_speech.set\_service\_url('https://api.eu-gb.text-to-speech.watson.cloud.ibm.com/instances/135c7563-991a-4990-ace3-58214bc40491')

def myCommandCallback(cmd):

print("Message received from IBM IoT Platform: %s" % cmd.data['command'])

m=cmd.data['command']

def phvalue():

with open('new.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'ph value of water is exceeded ,dont drink river water',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new.mp3')

time.sleep(1)

os.remove('new.mp3')

return

def waterquality():

with open('new1.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'water quality is not good ,dont drink river water',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new1.mp3')

time.sleep(1)

os.remove('new1.mp3')

return

def both():

with open('new2.mp3', 'wb') as audio\_file:

audio\_file.write(

text\_to\_speech.synthesize(

'water is not suitable for drinking',

voice='en-US\_AllisonV3Voice',

accept='audio/mp3'

).get\_result().content)

playsound('new2.mp3')

time.sleep(1)

os.remove('new2.mp3')

return

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect()

while True:

wq=random.randint(0,100)

ph=float("{:.1f}".format(random.uniform(1,14)))

myData={'waterquality':wq, 'phvalue':ph}

if ph>=8.5 and wq<=70:

both()

elif wq<=70:

waterquality()

elif ph>=8.5:

phvalue()

time.sleep(2)

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)

print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

client.disconnect()

**2.2 UI Output Screenshot:**

