IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Real-Time Water Quality Monitoring using IoT

[1] A. Praveena, [2]Renugha.S.S, [3]Sabari Vasan.P, [4]Santhosh.R, [5]Ragul Gandhi

[1]Assistant Professor, [2,3,4,5]Final year CSE students

[1,2,3,4,5]Department of Computer Science and Engineering,

[1,2,3,4,5]Jansons Institute of Technology, Coimbatore, India

Abstract: There are numerous ways to consume water. People use municipal water, borewells, and many other resources. But is it safe to drink straight from the ground or to filter it in any way? It might have dangerous substances in it, making it unfit for drinking. Drinking such water could have a number of negative health effects. Several things can contaminate water. Water pollution has an impact on both people and flora. We can determine its pollution level using IoT. We can keep an eye on the water's quality for drinking, agricultural, and domestic use. Knowing the water's quality allows us to change the filtering method. This water quality monitoring system aims to determine the water quality, or how clean the water is. This water quality monitoring system uses various sensors and microcontrollers to find water quality, i.e., how the water's pH content and dissolved oxygen contents vary, and send the data to the users. The user can check the previous data and can know the pattern of the varied data, if any. This whole system fits in any water point. The values from the sensors are checked with the standard water qualities and the data is displayed in the web application from the cloud server.

Keywords: Water Quality Monitoring System, sensors, pH, Wi-Fi module.

I. INTRODUCTION

Water impurities are those regions with contaminated water, such as lakes, gutters, abysses, aquifers that store additional groundwater, etc. The quality of the water needs to be determined in order to provide pure water for people and other living things. Calculate the amount of oxygen and pollutants in the water in preparation for water sanctification. The design aims to detect water that requires filtration, or a secondary compass region. Likewise, an expense-full approach initially. Therefore, when a wireless oxygen detector is employed, this can easily apply. Additionally, by adjusting other factors including turbidity, dissolved ions, and saltiness. We can also use messaging technology, which is used to send dispatches to the appropriate authorities, in addition to continuous monitoring. Changing the water's quality is important since it affects people's health, underwater creatures' situations, and a variety of agricultural fields. to prolong the existence of aquatic bodies. The goal of this work is to identify water satisfaction parameters like pH. foreign molecules, oxygen in water Additionally, provide the user with water data.

II. EXISTING SYSTEM AND ITS DRAWBACKS

Advanced mobile water monitoring systems are appropriate for remote estimation of swimming pools and portable water monitoring. It contains independent hubs that connect to the cloud to continuously control the flow of water. The key benefit of this technology is that it can cover a large region and is extremely accurate. This model's disadvantage is its price.

Several information procurement systems are embracing water nature following (WOM). The system is more dependable, independent, and adaptable. A low capture rate is discovered, and the cost is an issue.

The sensor-based water nature verification framework will be the basis of the alternative model. Information-following nodes, base stations, and other components make up the structural engineering framework. The water-checking framework for halter shelter frequency will be acquired in this phase. secondarily mobile and weakly powered. The model's price and lack of user-friendliness are its drawbacks.

III. PROPOSED SYSTEM

The suggested system will use the Internet of Things to educate people about the cleanliness of drinking water. This technology is quickly developing. Everyone will be informed if the water they are drinking is polluted or not via our proposed system. It protects against a variety of diseases that are spread by drinking contaminated water. Therefore, we may maintain adequate hygiene, which is next to godliness, by employing IoT technology in a wiser way. An Arduino microcontroller and a collection of sensors for measuring water quality characteristics were used to create the system. The pH sensor, dissolved oxygen sensor, temperature sensor, and turbidity sensor are the sensors employed in the system. The microcontroller gathers the sensor data, which is then transmitted via the Wi-Fi module to the cloud server. The Thingspeak cloud is used to implement the cloud server. Machine learning methods are used to analyse the data obtained from the microcontroller in order to find abnormalities and forecast trends in water quality. Based on the parameters that were measured, the server sends instructions back to the microcontroller to adjust water quality. HTML was used to construct the web application.

IV.MODULE SPLIT-UP

Module 1: Connect sensors to the circuit. In this module, we are doing the sensor set-up. The sensors are connected and Arduino is launched. Module 2: Integrating the IoT platform and cloud. In this module, the data collected is collected in the IBM Watson IoT platform and then send to the local cloud server. For this python program is used. Module 3: Developing web application. In this module, the web application was developed using HTML.

4.1 Framework

The iterative model is a particular implementation of a programming improvement life cycle that focuses on examining an initial, rearranged implementation, followed by progressive additions that are only the tip of the iceberg multifaceted nature and a more extensive characteristic placed until the final framework may be finished. The phrase "incremental improvement"—which describes the gradual changes made during the planning and implementation of each new cycle—will also commonly be used liberally and interchangeably when discussing the iterative technique.

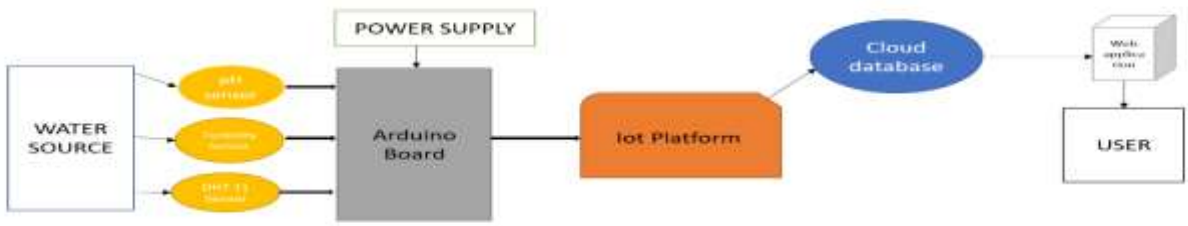

Figure: 4.1 Block diagram of the system

The Arduino software (download) can be used to programme the Arduino Uno. Depending on the type of microcontroller on your board, choose "Arduino Uno" from the Tools Board menu.

This configuration has additional effects. The Uno resets each time a connection is made to it from software (through USB) when it is attached to a computer running Mac OS X or Linux. The bootloader is active on the Uno for the next half-second or so. The initial few bytes of data transmitted to the board after a connection is established will be intercepted, despite the fact that it is configured to disregard invalid data (i.e., anything other than an upload of new code). To turn off the auto-reset, a trace on the Uno can be severed. To re-enable the trace, solder the pads on either side of it together. It has "RESET-EN" written on it. By connecting a 110 ohm resistor from 5V to the reset line, you might also be able to turn off the auto-reset; for more information, see this forum discussion.

STEP 1: Building of sensor network

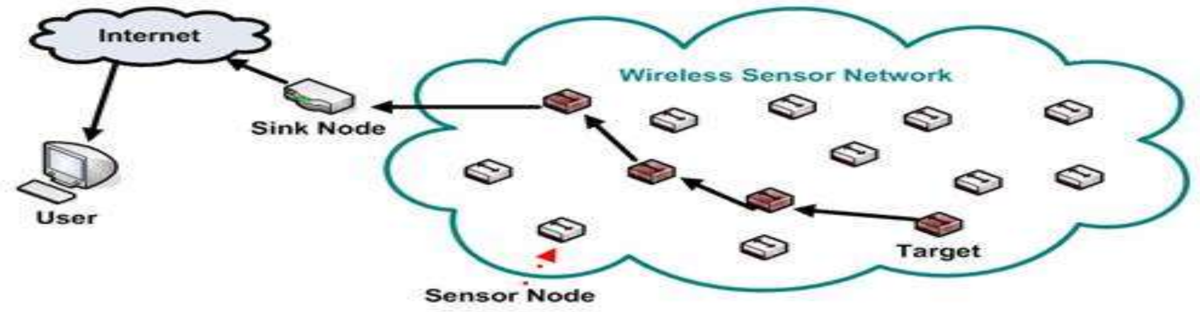

Figure 4.2: Conventional Sensor Network

A typical sensor network consists of radio sensor nodes that can sense physical parameters, store collected data, perform basic processing on the data, and transmit the data through radio interface. Such a network's goal is to push data to a sink node so that it can be forwarded to a server (or cloud).

STEP 2: Connection of sensor network to the cloud.

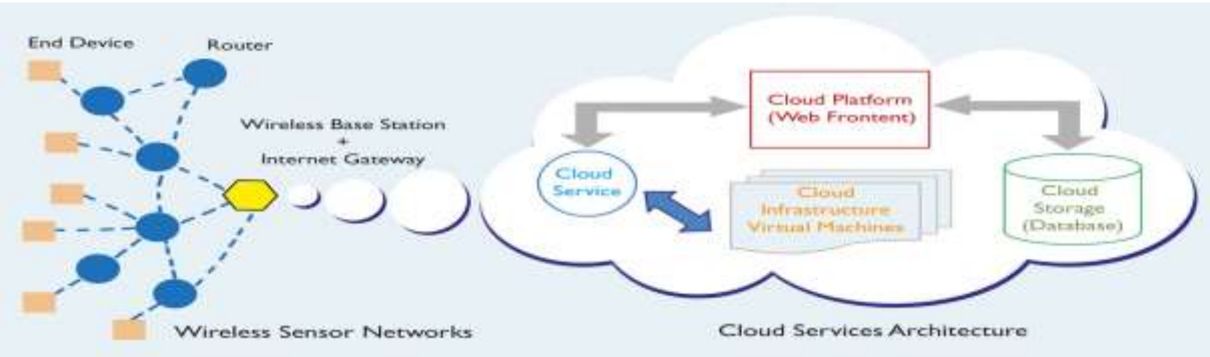

Figure 4.3: Sensor Network over cloud

The link between the cloud and sensor network is depicted in Figure 4.3. But a lot of real-time applications use sensors dispersed over large areas. They are handled as independent networks as a result. The Internet of Things is a brand-new paradigm for linking microcontrollers and other smart items to the cloud. We can now immediately link sensors to the internet via IoT services. Cluster-based approaches, in which clusters are formed by groups of nodes, are a frequent architecture for sensor networks. They are additionally known as coordinator nodes. These nodes collect information from all of their nearby nodes. Theoretically, the sensor network can be infinitely grown if these nodes can connect to the internet using their individual IP addresses.

STEP 3: Interfacing with the embedded system.

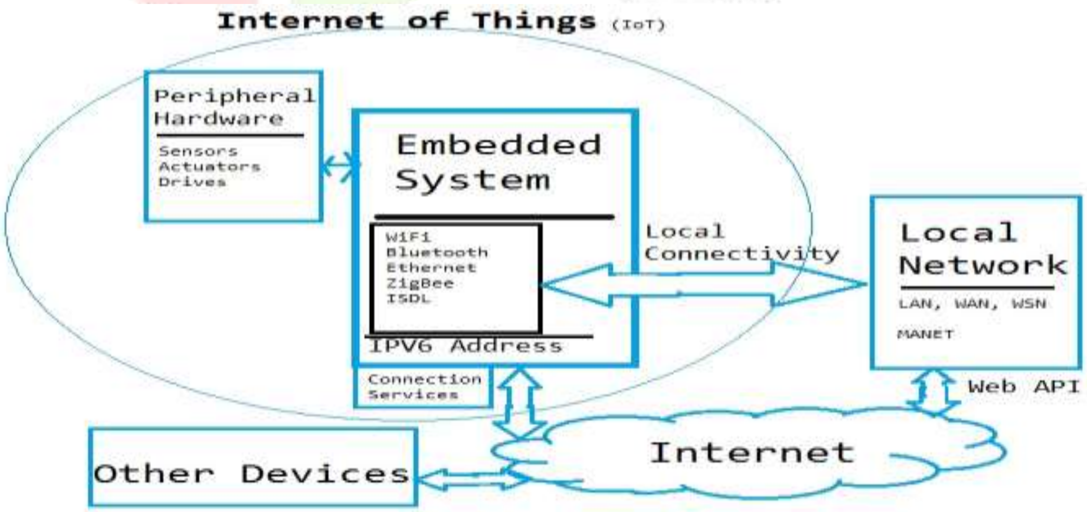

Figure4.4: Structure Of IOT

Above is a diagram showing the IoT's fundamental structure. The following cutting-edge research enhancements to the current WSN and IoT framework would be part of our effort.

In the beginning, we'll concentrate solely on sensors, namely coordinator nodes in peripheral hardware. Our approach would consider the entire isolated sensor network as a single peripheral and connect that to the cloud using IoT, rather than concentrating on integrating separate gear over the cloud. Given that ZigBee is the most widely used WSN standard, that is where we would mostly concentrate our attention. Our techniques would offer services for data collection and analysis in addition to communication. To validate the designed principles, we would combine both simulation and real-time test environments.

STEP 4: Possible outcome of the IoT:

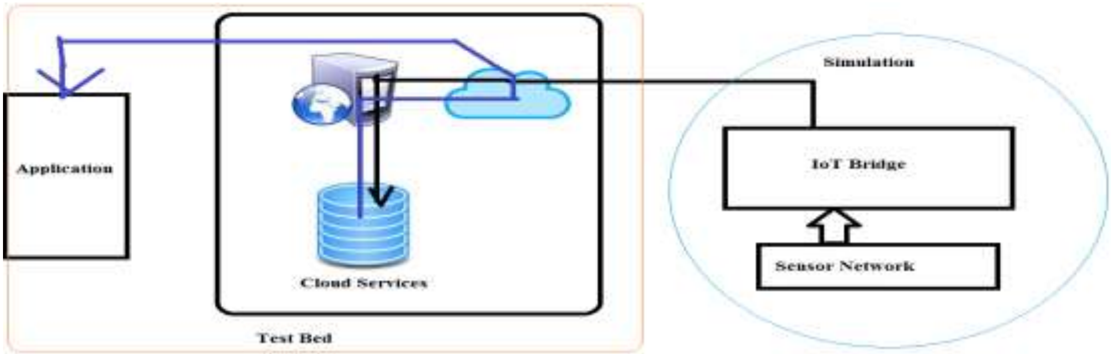

Figure 4.5: possible outcome of project

The major goal of the project work is to construct a more sophisticated, cutting-edge cloud extension of WSN using IoT. First, the research will concentrate on improving each of the state-of-the-art building blocks, such as the coordinator protocol, data analysis in sensor networks, cloud services, IoT protocols, and sensor networks, among others. The creation of a special framework to link current WSN to the cloud is one of the first outcomes anticipated. The system then has to show how much data analysis services have improved in order to demonstrate the benefit of such an extension. Results should demonstrate that mesh sensor networks can be built using IoT, and that sensor networks can be connected to other control systems using improved bandwidth.

The cloud should receive all of the data provided by the sensor network via javascriptnode.js. Here, the cloud is a private programmed that the general public can use, and it is simple for any user to install. This application will assist the user in gathering data from the sensor network as well as in analyzing and visualizing the data. The built-in Matlab visualization feature offered by these programs can be used to accomplish this. The user can take action based on the needs based on the data provided by the apps.

STEP 5: Sending data to THING SPEAK APPS:

Access to a wide variety of embedded devices and web services is made possible by the Internet of Things. Thing Speak is an open data platform and API for the Internet of Things that allows us to gather, store, analyze, visualize, and take action on data from sensors or actuators, like Arduino®, and other hardware. For instance, one may program their home thermostat to regulate itself based on their location using the sensor-logging, location-tracking, and social network of things capabilities of Thing Speak. The channel, which includes data fields, location fields, and a status field, is the main component of Thing Speak activity. You can write data to a Thing Speak channel after creating it, use MATLAB® code to process and view the data and respond to the data with tweets and other notifications.

V. TESTING AND RESULTS

Table 5.1 Sensors Test result

Sensors Testing						
Pre-Conditions for test: The water should be polluted to the sensor to be detected						
Sensor name	Stage 0	Result	Stage 1	Expected output	Actual Result	Test Status
pH sensor	pH is not detected	Error message (failed to read from the sensor)	pH is detected	Arduino shows the value in the LCD	Arduino shows the value in the LCD	Pass
Temperature sensor	Temperature is not detected	Error message (failed to read from the sensor)	Temperature is detected	Arduino shows the value in the LCD	Arduino shows the value in the LCD	Pass
Turbidity sensor	Turbidity is not detected	Error message (failed to read from the sensor)	Turbidity is detected	Arduino shows the value in the LCD	Arduino shows the value in the LCD	Pass

VI. OUTPUT

Figure 6.1: Result on computer after connected to the cloud

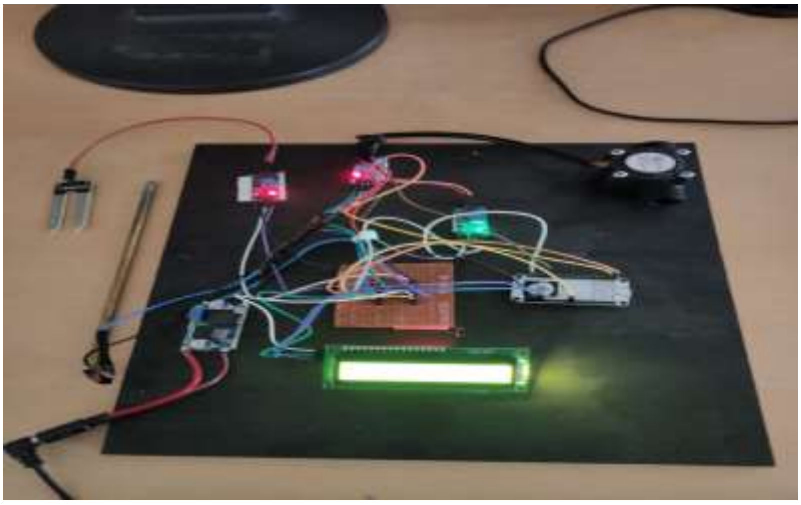

Figure 6.2: Hardware connection and result in the LCD

Figure 6.3: Web page where the user can view the data

VII. CONCLUSION AND FUTURE SCOPE

The system under consideration is built using a variety of sensors, a Nodemcu as the controller, and the cloud to store data from the Controller and transmit instructions to an Arduino to measure water quality and water level. Everywhere throughout the city, a web interface can be used to see the generated data. The system's advantage is that it gives every residence, business, and other entity access to a sufficient supply of high-quality water. The suggested model can be used to build a smart city.

Checking the pH reading from the sensor related to water usage. These frameworks could monitor water usage automatically, transmit notifications to the person in charge, and don't call for physical or other involvement from family members. Those water nature attempts around need to be more efficient, practical, and swift. These frameworks need to be flexible enough to swap out the relevant sensors and advance the relevant initiatives. Other water characteristics including turbidity, temperature, and quantities of broken-down oxygen can be screened using this framework. The scope of this study will include determining the water's temperature, turbidity (a measure of how clean the water is), and pH levels. In this manner, the framework examines each of these components before sending the data or information. SMS can also be used to notify those designated people. This can be produced with customization in future works based on the requirements of the user. In addition, we can create a tailored mobile app that consumers may use to check the water quality wherever they are.

VIII. REFERENCES

[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 low-cost autonomous water quality monitoring system", 2013 IEEE, pp. 14-19.

[9] 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, pp 1-16.