Water Quality Monitoring System Based on the Internet of Things

Khawla A. Alnajjar, Mahmoud M. A. Fahmy, Yousef M. Ali, and Sam Ansari
Department of Electrical Engineering, University of Sharjah, Sharjah, United Arab Emirates
{kalnajjar, u18103149, u18100948, samansari}@sharjah.ac.ae

***Abstract*—People's lives always necessitate prioritizing having the best possible and accessible water quality. Since pollution sources continuously expand, water quality monitoring systems have become essential. However, such monitoring systems remain too expensive. Fortunately, with the recent growth of the Internet of Things (IoT) technology and the ongoing development of smart monitoring systems, a low-cost smart water quality monitoring system can be implemented. This paper presents a novel methodology for water quality monitoring systems, allowing smart acquisition of water data, real-time monitoring, and the maintenance of the optimal water quality required for the intended audience. This work provides the necessary monitoring options in a reliable and cost-efficient manner. The system design consisting of hardware and software architectures is presented and investigated.**

***Index Terms*—Automation, data acquisition, Internet of Things, real-time monitoring, water quality monitoring.**

I. Introduction

Nowadays, many water quality vendors use traditional water quality monitoring systems, which include many manual procedures that could be automated. Current water quality monitoring systems are vulnerable to human error, loss of documents, and unreliability. In addition, such monitoring systems cost vendors more money than they should when automated [1]. Due to the growth of IoT technology, the water quality monitoring process can be automated [2]. The following section discusses the methodology of automating the monitoring system. The water quality monitoring system is divided into two main parts: software and hardware designs. For low-cost projects, using a microcontroller is found to be the best option to have. The microcontroller serves as the central processing unit of the system.

Moreover, certain parameters should be measured to collect pertinent data about the quality of a water sample [3]. The parameters mentioned below are the foremost parameters to be evaluated:

- The potential of hydrogen (pH),
- The turbidity of water,
- Total Dissolved Solids (TDS),
- The water Temperature.

Accordingly, four sensors are exploited to determine the value of each parameter in a tested water sample. Most related sensors use analog measurement to serve the need [4]. Finally, a user interface, which is a mobile application, is designed to complete the process of monitoring and matching the needs of the system. Mobile Application consists of three main sections: live monitoring system's data, display list of alerts and events, and controlling system components such as relays and sensors. The mobile application is designed utilizing one programming language that is suitable for a certain application developing tool. Smart monitoring systems need to follow several criteria. The hardware design criteria the system complies with are as follows: ease of configuration, cost, and portability. Additionally, software design follows some criteria: user-friendly, well-organized, and compatible with updates. However, the scope of the proposed monitoring system should be able to

- Monitor and report alerts in real-time.
- Act immediately upon certain readings and alarms.
- Provide current system status, including water parameters and its real-time reading through the user interface or short message service (SMS).
- Control specific components through the user interface.
- View and monitor previous system alerts and readings through the user interface.

The rest of this paper is organized as follows. Related studies and a literature review are highlighted in Section II. Section III presents the methodology of this work, which includes hardware components, system model, hardware design, and software design. Section IV provides the simulation analysis and results. Finally, Section V contains a summary and conclusion.

II. Literature Review

To better understand the context of our work, it is essential to examine the relevant studies in the field of water quality monitoring. This section reviews related work in order to study their challenges and develop the proposed system in this paper.

Gokulanathan et al. [5] designed a water quality monitoring system based on GSM communication. The proposed system uses pH, conductivity (TDS), and Temperature sensors as parameters to measure water quality for their system. The sensors are attached to an Arduino Uno, which reads sensor data and sends it to a GSM module connected to the Arduino. The GSM module then sends sensor readings to a user's smartphone or PC via SMS for display and monitoring purposes.

Another related work was done by Rao et al. [3], who designed a low-cost autonomous water quality monitoring system using Arduino Mega as a microcontroller (server node). The parameter sensors used in this system included pH, light, electrical conductivity (TDS), dissolved oxygen (DO), and oxidation-reduction potential (ORP). A data acquisition cycle is used to read sensor data in steps; where Arduino starts by reading light, pH, and temperature first, second acquire TDS, third acquire DO, and finally acquires ORP. In the case of failure in any step, the Arduino resets to the first step again and repeats until it finishes acquiring all required parameters. The Arduino is powered through a PC USB port and uses serial communication to send sensor data and upload it to the MySQL database.

Moreover, Doni et al. [6] presented a water quality monitoring system using IoT where they used pH, turbidity, and temperature sensors as parameters connected to the microcontroller. The system used a GPRS module to send sensor data to a workstation and later into the cloud. The process of receiving the data on a user interface is done by displaying sensor data on a webpage after establishing a connection between the server storing the data and the user's device.

A. Limitations and Drawbacks

While all related works mentioned above successfully implemented a smart water quality monitoring system, their work exhibited certain limitations that present opportunities for refinement and innovation in our system that we shall discuss.

1) Simplicity and Accessibility : Simplicity and accessibility are key considerations in the proposed design, as it caters to both experienced and inexperienced users in domestic and industrial environments. However, previous designs have suffered from a lack of user-friendly data display options, with some even lacking any dedicated user interface for accessing data. In contrast, the proposed system addresses this limitation by incorporating a mobile application as a user interface software. This software offers an intuitive and effective means of displaying all system data, ensuring a more user-friendly experience for the end-users. Moreover, previous work has been in continuous connection with PC, which has led to a lack of portability and outdoor operations. The proposed design doesn't require any connection to external devices and can operate in an environment that has an internet connection.

2) System Capabilities : In contrast to previous works that solely focused on data readings, The proposed system goes beyond passive monitoring by incorporating intelligent functionality to initiate actions based on specific readings and alerts. The system can automatically respond to critical water quality conditions by leveraging algorithms and decision-making capabilities. For instance, it can notify relevant users when predefined thresholds are exceeded and take action upon unwanted readings. This proactive approach ensures timely intervention and enables prompt mitigation of potential water quality issues. By extending the capabilities beyond mere data acquisition, the proposed system sets a new standard in water quality monitoring, transforming it into a dynamic and responsive solution that actively safeguards the water environment.

3) User Interface Features: The proposed design addresses the limitations of previous designs in terms of user-friendly data display and incorporates a comprehensive user interface. In addition to displaying sensor data, the mobile application offers a range of additional functionalities. It provides real-time monitoring of sensor readings, allowing users to stay informed about the water quality status at all times. Moreover, the interface includes alert mechanisms to promptly notify users about critical events or abnormal readings, ensuring proactive management of water quality. Additionally, the user interface allows for system control, enabling users to remotely adjust settings and perform necessary actions for optimal water quality management. These enhanced features go beyond simple data display, providing users with a powerful and versatile tool for efficiently monitoring and controlling the water quality monitoring system.

III. Methodology

A. Hardware Components

The hardware components in the proposed system are responsible for reading data taken from water samples to control the system upon particular measurements. In addition, components communicate with different clients through cloud databases. Each component employed in the proposed monitoring system and its purpose is thoroughly discussed.

- NodeMCU ESP8266: low-cost Wi-Fi microchip with integrated TCP/IP networking software and a microcontroller. It offers sufficient onboard processing and storage capability to integrate it with sensors and other application-specific devices via its GPIOs with minimal development and load during runtime [7].
- pH Sensor PH0-14: can detect alkalinity and acidity levels in water and other liquids.
- Turbidity Sensor: Calculate the amount of light reflected by the water's suspended particulates through two emitting and receiving light detectors. The turbidity level of water increases together with the total amount of suspended solids (TSS) in the water.
- TDS Sensor: stands for Total Dissolved Solids, which affect water conductivity. The sensor consists of two terminals, a voltage waveform is transmitted from one terminal to deliver a digital representation after the second terminal receives the waveform signal.
- Temperature Sensor DS18B20: measures water temperature using one wire mechanism.
- ADC ADS1115: Analog to Digital Converter used to digitalize analog signals received from the used sensors.
- GSM Module SIM800L: contains a micro-SIM Card slot and a mounted antenna that can be used to send and receive SMS from/to the system.
- Relay Module: electromagnetic switch that is controlled by small open and close signals.

- DC Water Pump: 2.5-6 VDC Motor that can control water flow from and to a water tank at a flow rate of 80-120L/H.
- DC Step Down Converter LM2596: is a step-down switching regulator that has good line and load regulation and can drive a 3-A load. It can take input from 3-40V and step down to 1.5-35V.
- Local LCD screen: 16×2 LCD screen used to display certain characters upon instruction provided.

B. System Model

In the proposed system, hardware and software designs are implemented to provide a faster, more reliable, and more convenient method for monitoring water quality, which solves many issues related to the traditional methods currently in use. Having a software user interface to such monitoring systems helps to automate the process and prevent human errors more easily. Moreover, the proposed system can take actions upon normal and up-normal inputs. The system includes multiple alert interfaces through the mobile application, SMS alerts, and a local LCD screen.

The ESP8266 microcontroller serves as the core of the system. The microcontroller is leveraged to control the monitoring process based on inputs taken from water parameter sensors. The ESP8266 uploads sensor data to a specific cloud database that is used to provide service to the mobile application. A DC water pump controls the water flow from the measured water sample to a destination water tank. The water pump is controlled using a relay module receiving open and close commands from the microcontroller based on input data measured by sensors. The user interface, i.e., mobile application, is capable of performing real-time live monitoring for sensor data, providing users with daily reports about the system's results, controlling specific components through the application, and allowing the user to receive alerts reported by the system. In addition to the mobile application warnings, the GSM Module SIM800L is exploited to send SMS alerts to authorized clients. Based on certain water parameters, the microcontroller sends commands to the GSM module in order to send SMS warnings to specified cell numbers.

C. Hardware Design

As illustrated in Fig. 1, the proposed system's hardware design consists of four water parameters sensors as input devices that directly connect to the ADC ADS1115 data pins. The ADS1115 communicates with the ESP8266 through I2C ports (SCL and SDA), allowing the ESP microcontroller to read and process analog sensors data as per the required standards. Consequently, the ESP microcontroller performs the following actions in case of sensors readings match pure water standards:

1) Send close command to the relay module allowing the DC water pump to flow the water from the tested tank to a desired tank for pure water only.

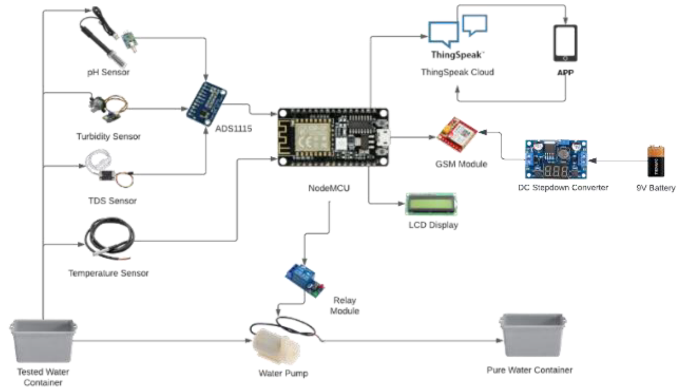


Figure 1: Hardware design.

2) Send commands to the local LCD screen indicating that the tested water is pure.

However, in case of undesired water parameter readings received by the sensors, the following actions will be performed:

1) Send open command to the relay module in order to immediately stop the water flow through the DC water pump.
2) Send commands to the local LCD screen indicating that the tested water is impure and water flow has stopped.
3) Send commands to the GSM module in order to send SMS alerts to authorized administrators.

In both cases, the ESP microcontroller uploads sensor data to an application programming interface (API) cloud database called Thingspeak in order to provide the service needed for the mobile application.

D. Software Design

The software design of the system consists of the cloud database, where the sensor data are uploaded by the microcontroller, and the system's user interface, which is the mobile application. The cloud database leveraged is called Thingspeak Cloud. Thingspeak is an IoT cloud platform that allows uploading sensor data to its storage and performing analytics and visualizations to view data locally from the cloud interface. Moreover, Thingspeak uses REST API and JSON files to store uploaded data. The ESP microcontroller is programmed to communicate with the Thingspeak client after every measurement in order to upload its local sensor data on the system-specified cloud storage on Thingspeak. A mobile application is designed to possess five main sections that provide the service needed for the monitoring system. Fig. 2 depicts the proposed mobile application flowchart.

Every application section is demonstrated on a separate page as follows:

a) Live Monitoring Page: designed to extract sensor readings from the Thingspeak database and display them on screen. The live monitoring page depicts the following for each sensor:

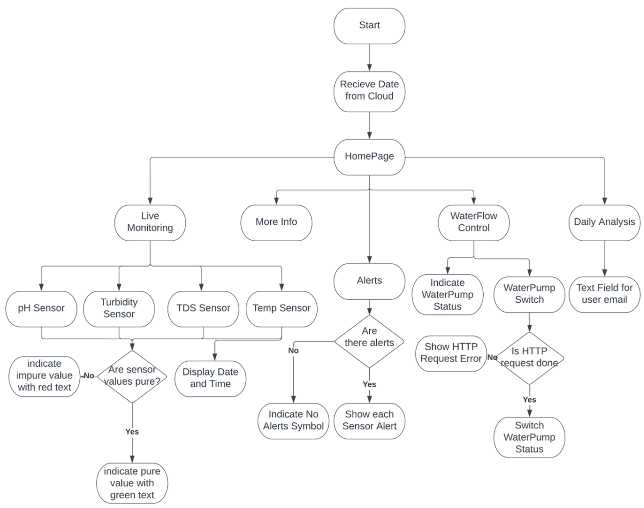


Figure 2: Mobile application flowchart.

- Green/Red text indicating the latest reading measured.
- Green/Red text indicating the latest reading measured status (pure/impure).
- Indication of the date and time of the latest reading measured.

b) Daily Analysis Page: prompts the user to enter an email address to receive an analysis sheet of the last 100 readings measured by the system's sensors.

c) Waterflow Control Page: designed to let the user control the DC water pump manually if required. In addition, it contains a live indication of the water pump status (ON/OFF).

d) Alerts Page: designed to allow the user to access previous and current system alerts, which indicate alert details, date, and time of the alert. Ideally, the mobile application pushes a notification that directs the user to this page, however; the user can access this page from the top app bar.

e) More Info Page: designed to hold extra information about the system, which are standard pure water parameters, administration contacts, and application time zone.

IV. Results and Discussion

The following section discusses the system results in different situations. The section include results, i.e., outputs of hardware and software parts, the measure of accuracy, and feedback about the overall outcomes.

A. Results in Pure Samples

This subsection demonstrates the system results when tested in pure water samples. The tested water sample is mineral water (drinking water).

1) Hardware Results: System sensors are able to give accurate results about the pure water tested. Table I illustrates the accepted range, accuracy, and average reading of every

Table I: Sensor Results

Sensor	Measured Value	Accepted Value/Range	Accuracy Status %
pH Sensor	7.43	6.5 – 8.5	0.93%
Turbidity Sensor	6.1%	0 – 8%	Accepted
TDS Sensor	107.45ppm	50 – 150ppm	Accepted
Temperature Sensor	25 °C	10 – 30 °C	Accepted

sensor obtained within 10 seconds (one measurement per second).

According to mineral water parameters (tested water sample) with a pH of 7.5, Turbidity less than 8%, and TDS of 50-150ppm, the system results were accepted since they are accurate and in the accepted standard range. However, sensor readings are not permanently fixed and may fluctuate by 0.5V every 5 – 10 seconds, which is an acceptable fluctuation since analog sensors can easily be affected by external noise. Figs. 3a through 3c demonstrate the fluctuation that happened during the 10 seconds measurements. It can be concluded from the sensor readings charts provided that all sensors have acceptable fluctuations. TDS and turbidity sensors showed more accurate results since the small voltage changes do not heavily affect its result. On the other side, pH had more fluctuations due to its high impedance output which can be easily affected by external noise, bias current, and solder flux residue.

2) Software Results: In the following, software results during pure water testing are discussed. Software results consist of a cloud database test and a mobile application test.

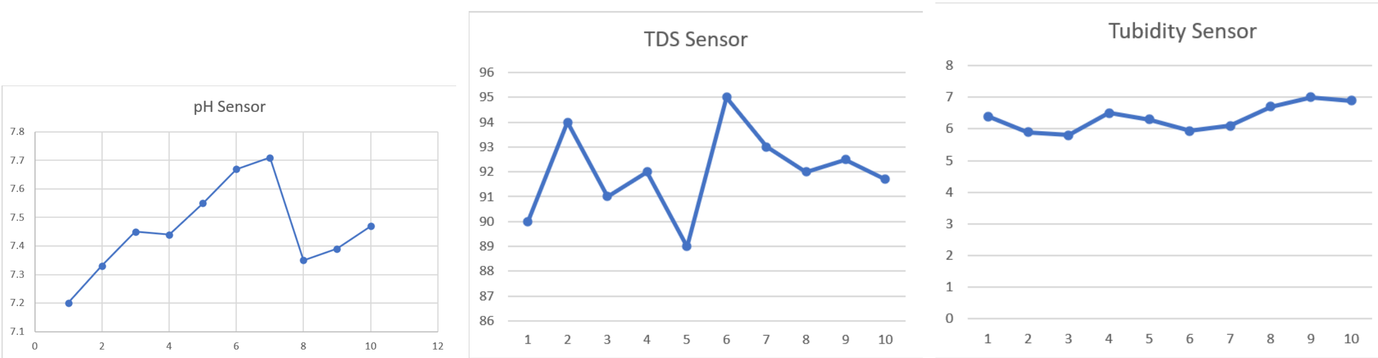
a) Thingspeak Database Test: Once the system's hardware measures the testing water samples, Thingspeak should be able to receive sensor data uploaded by the ESP microcontroller and store them in the cloud database. By viewing the Thingspeak web cloud interface, we can confirm that Thingspeak is able to receive sensor data and update it every five seconds. Thingspeak web interface is shown in Fig. 5.

b) Mobile Application Test: The mobile application is able to take the latest sensor data from the Thingspeak database and display it on the screen for the user. The mobile application can process the received values to provide details for each sensor reading (readings status, date, and time). Fig. 6 shows an example of the mobile application's live monitoring screen during pure water testing.

B. Results in Polluted Samples

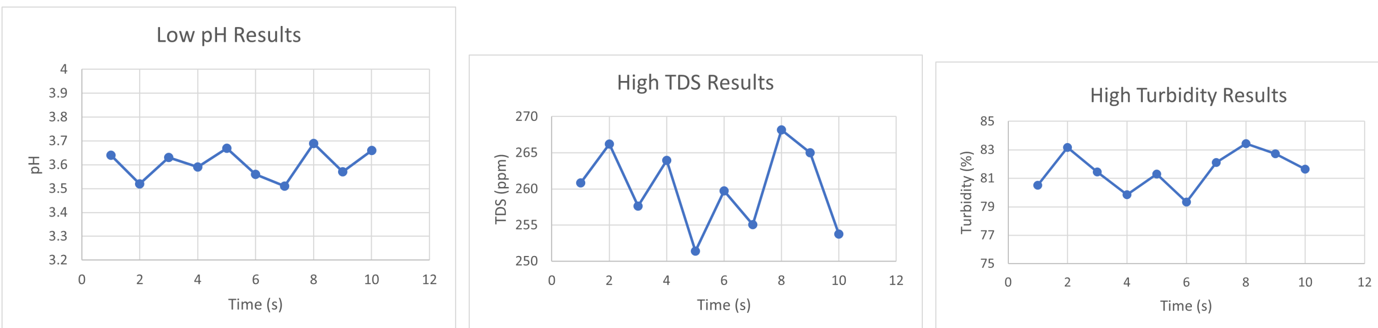
This subsection discusses the system hardware and software results while measuring different polluted water samples. Five samples test five different scenarios that the system could face. The scenarios include Vinegar (Acid – low pH), Clorox (alkaline – high pH), tap water (high TDS), dark solution (high turbidity percentage), and boiled water (high temperature).

1) Hardware Results: The system's microcontroller is able to send open commands to the relay module connected to the DC water pump in order to stop the water flow as expected.



(a) (b) (c)

Figure 3: Fluctuation in measurements, (a) The pH sensor readings, (b) The TDS sensor readings, (c) The turbidity sensor readings.



(a) (b) (c)

Figure 4: Measurement results, (a) Low pH results, (b) High TDS results, (c) High turbidity results.

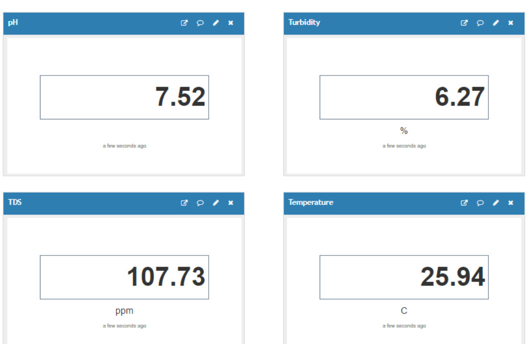


Figure 5: The Thingspeak web interface.

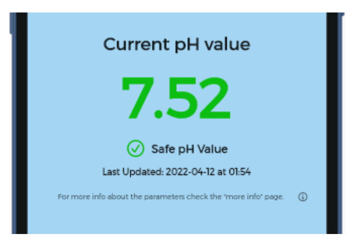


Figure 6: Illustration of pH reading during pure water testing.

In addition, the GSM module is able to send SMS alerts to the provided administrator number. Figs. 4a through 4c show sensor results in the abovementioned fluctuations that occur due to external noise.

2) Software Results: During the polluted samples test, the mobile application is able to detect that sensor values taken from the Thingspeak database are impure. Accordingly, the application provided alert notifications as well as details about the impure measurements on the live monitoring page, as demonstrated in Figs. 7 and 8a.

C. Other Features Results

In addition to the results acquired during the testing phase of pure and impure samples, some other system features that the system provides were tested as follows:

- Waterflow control through user interface: The water flow control page can send open and close commands via HTTP requests to the microcontroller in order to control the DC water pump, which controls the water flow from the tested tank to the pure water tank. Additionally, the page is able to indicate the water pump status, as shown in Fig. 8b.

- Daily Analysis: the mobile application is able to send a link containing a daily analysis sheet that includes the last 100 readings the system has registered. The user can type the desired email and receive the sheet in approximately one minute, as illustrated in Fig. 9.

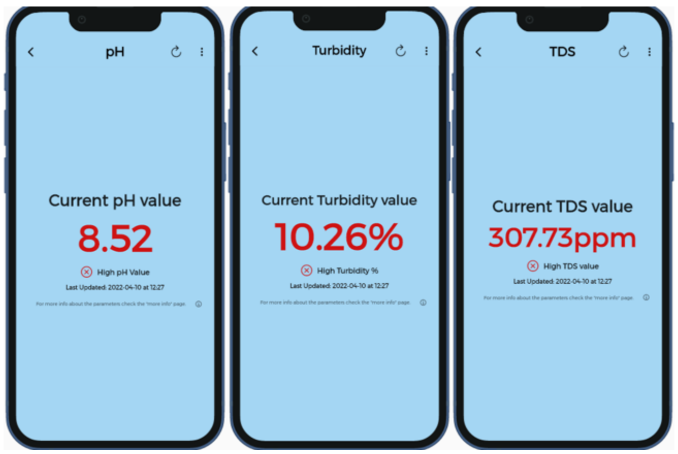
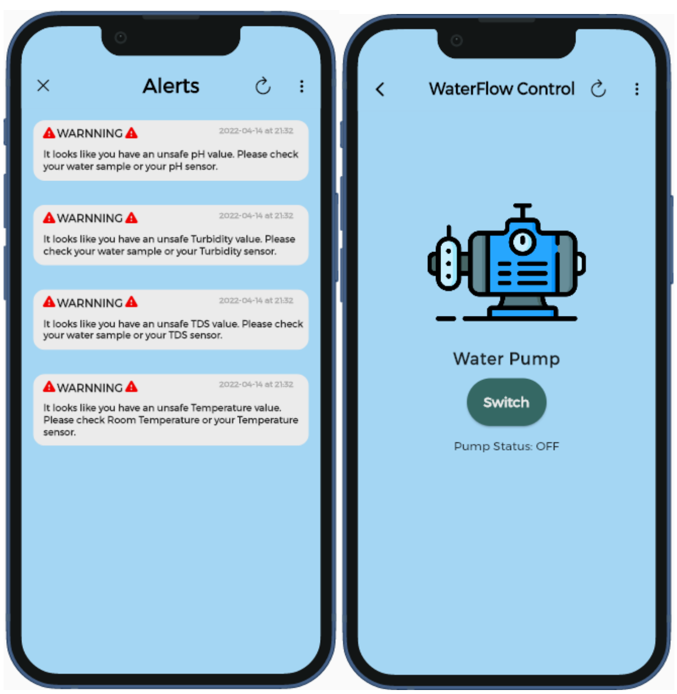


Figure 7: Live monitoring page during impure samples test.



(a) (b)

Figure 8: An illustration of displays, (a) Alerts page during impure samples test, (b) Water flow application page.

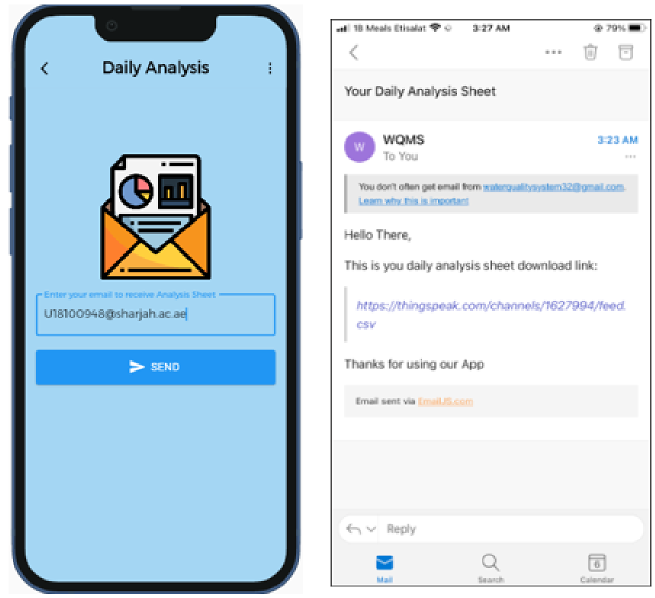


Figure 9: Daily analysis test result.

V. Conclusion

This study proposes a low-cost intelligent water quality monitoring system based on IoT equipment and software, which improves the currently used monitoring systems. The entire proposed system operation is controlled by a small microchip capable of performing hardware, software, and communication tasks. Along with the mobile application design, a complete monitoring system is able to read and identify alerts more easily and effectively.

References

[1] Z. Rasin and M. R. Abdullah, "Water quality monitoring system using zigbee based wireless sensor network," *International Journal of Engineering & Technology*, vol. 9, no. 10, pp. 24–28, 2009.

[2] S. Balakrishna, M. Thirumaran, and V. K. Solanki, "A framework for iot sensor data acquisition and analysis," *EAI Endorsed Transactions on Internet of Things*, vol. 4, no. 16, pp. e4–e4, 2018.

[3] A. S. Rao, S. Marshall, J. Gubbi, M. Palaniswami, R. Sinnott, and V. Pettigrovet, "Design of low-cost autonomous water quality monitoring system," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2013, pp. 14–19.

[4] V. Lakshmikantha, A. Hiriyannagowda, A. Manjunath, A. Patted, J. Basavaiah, and A. Arlene a, "Iot based smart water quality monitoring system," *Global Transitions Proceedings*, vol. 2, 08 2021.

[5] S. Gokulanathan, P. Manivasagam, N. Prabu, and T. Venkatesh, "A gsm based water quality monitoring system using arduino," *Shanlax International Journal of Arts, Science and Humanities*, vol. 6, no. 4, pp. 22–26, 2019.

[6] A. Doni, C. Murthy, and M. Kurian, "Survey on multi sensor based air and water quality monitoring using iot," *Indian J. Sci. Res*, vol. 17, no. 2, pp. 147–153, 2018.

[7] Esp8266 introduction. https://fabacademy.org/archives/2015/doc/networking-esp8266.html.