Discover Applied Sciences

Research

A wireless sensor network IoT platform for consumption and quality monitoring of drinking water

Christodoulos Axiotidis[1] · Evangelia Konstantopoulou[2] · Nicolas Sklavos[2]

Received: 9 May 2024 / Accepted: 25 November 2024
Published online: 18 December 2024

Abstract

Generally, municipal water supply companies use manual collection and laboratory analysis for water quality testing. However, these methods have limitations such as lack of real-time information, inability to sample the entire water supply, and high costs. Therefore, continuous, real-time water quality monitoring is crucial for public health protection and ensuring that the whole water supply network is monitored. This paper proposes an Internet of Things (IoT) platform for the measurement of consumption and the quality of drinking water in rural or semi-rural environments. Data collected through temperature, flow, potential of hydrogen (pH), turbidity, and Oxidation Reduction Potential (ORP) sensors is exchanged with a database through a long-range wireless communication protocol. Two mobile applications and one desktop application were also developed, with the purpose of being used by simple users, technicians, and network administrators respectively. The presented implementation process includes the design of the hardware surrounding the ESP32 microcontroller and its mounted peripherals, as well as the software run by the microcontroller and the mobile devices. A prototype system was built and tested under controlled conditions, successfully recognizing an increase in water turbidity and its unsuitability when contaminated with different agents. This method may prove to be a financially advantageous solution for rural, semi-rural, and even urban environments when used with groups of data collection nodes, helping significantly in the upkeep and surveillance of the water supply network.

Highlights

- The IoT platform is equipped with sensors that measure water consumption as well as turbidity, temperature, and pH.
- Tested under controlled conditions, detecting different signs of water contamination.
- A low-cost option for continuous water monitoring, particularly in resource-limited environments.

Keywords ESP32 · Water quality · Flow control · Real-time systems · IoT · Wireless sensor network

✉ Christodoulos Axiotidis, chr.axiotidis@gmail.com; Evangelia Konstantopoulou, e.konstanto@ac.upatras.gr; Nicolas Sklavos, nsklavos@upatras.gr | [1]School of Science and Technology, Hellenic Open University, Hellas, Greece. [2]SCYTALE Group, Computer Engineering and Informatics Department, University of Patras, Hellas, Greece.

Discover

1 Introduction

The complexity, diversity, and number of Internet of Things (IoT) platforms have increased tremendously in the past few years, thus enabling IoT to improve efficiency and convenience in a wide range of deployment contexts [1]. In recent years, cities all over the developed world are undergoing radical functional changes because of this. In these new models, known as Smart Cities, data sensing and collection are crucial tasks for a variety of information-driven services [2].

At the same time, technologies for wireless communication and the Wireless Sensor Network (WSN) have been increasingly aiding people with daily work in their personal as well as professional lives. Wi-Fi, Bluetooth, Zigbee, General Packet Radio Service (GPRS), RFID, and cellular technologies are only a few of the wireless communication standards that the WSN system uses to enable users to monitor and control the connected devices from the base station [3]. Through a wireless network that is built using one of those protocols, users can monitor the data remotely. Low power usage, multiple sources of data, quick network setup, extensive coverage, high precision, and low duty cycle are among the benefits of WSN.

One popular application domain is environmental monitoring [4], specifically, access to clean drinking water. Nowadays, water is being polluted due to industrialization, intensive use of fertilizers and pesticides, urban development, etc., which greatly affects the citizens' everyday lives.

Drinking water quality standards are set by the World Health Organization. These parameters, divided into four broad categories, must be monitored, and controlled regularly to protect the health of consumers and ensure that water is potable and clean [5]:

- Microbiological factors, for ensuring that the water is not contaminated with human or animal waste.
- Chemical factors, to prevent prolonged consumption of chemically contaminated water over many years.
- Radiological agents, for detecting radioactive substances that could prove harmful to human health.
- Aesthetic factors, for ensuring that drinking water has an acceptable appearance, taste, and odor, qualities that are important to maintain consumer trust.

The water quality testing methods followed by most municipal water supply companies include a manual collection of water samples from various locations, followed by laboratory analytical techniques to evaluate water quality. However, such approaches are no longer effective. Although this methodology allows for a thorough analysis, including chemical and biological factors, it has several disadvantages. These are:

- The lack of real-time information for critical decisions, to protect public health.
- The inability to carry out an extensive sampling of the entire water supply.
- The methodology's relatively high cost.

According to research conducted by the US Environmental Protection Agency (USEPA) [6], pollutants have distinct effects on water parameters, which can be identified with the right water quality sensors. The products that are already on the market for tracking the parameters of water quality are often large and rather costly. It is important to note that analyzers and panels bear most of the expense, rather than the sensing probes [7]. These systems are only able to frequently collect water quality samples from a very small number of sites. Therefore, there is a clear need for low-cost, continuous, and real-time monitoring of water quality throughout the entire water supply network of a region.

The aim of this paper is to design and develop a reliable system for monitoring water consumption and quality, using appropriate sensors and transmitting the information collected to a database. The solution proposed is roughly divided into two parts. Firstly, an IoT platform was designed to which the Turbidity, Temperature, Oxidation Reduction Potential (ORP), potential of hydrogen (pH), and flow sensors were connected. The platform, using a long-range wireless communication protocol, transfers the information to the database. There is a noticeable absence of an easy-to-use mobile application that presents information on the consumption and quality of the water reaching the consumer simply and understandably. Therefore, for the second part of our implementation, three applications were designed, for the three types of system users: consumers, water system technicians, and water system operators.

The rest of the paper is organized as follows. A literature review on water quality monitoring is presented next. Section 3 provides the implementation details of our solution and describes the system architecture. Evaluation and testing under controlled conditions are presented in Sect. 4. Lastly, Sect. 5 concludes the paper, with a short discussion and possible directions for future research.

2 Literature review

There are very limited relevant papers in the literature that deal with the monitoring of microbiological and radiological factors that determine water quality outside of a laboratory environment. This is due to the lack of low-cost and commercially available sensors that measure reliably, do not need supervision, and do not require recalibration over long periods of time. Moreover, most of the research focuses solely on the monitoring of chemical factors. When it comes to the aesthetic factors, the only concern is the turbidity of the water.

When researching related articles, an emphasis was placed on drinking water quality measurement through a hardware node platform. This is due to one of the main objectives of this comparison being choosing appropriate sensors for this work, that would be compatible with well-known microcontrollers in the market. This significantly narrows down the research area. Some of the works that were positively evaluated refer to drinking water, while others refer to the water quality of aquatic ecosystems.

The work in [7] describes a low-cost technology that continuously monitors qualitative water parameters. The system consists of optical sensors, networked embedded systems, event detection algorithms, and experimental performance evaluation in tainted drinking water. Turbidity, ORP, pH, Electrical Conductivity (EC), and temperature were measured, while the study focuses on the pollution caused by arsenic and E. Coli. In [8] authors present a low-cost wireless multi-sensor network for real-time monitoring of flow, temperature, conductivity, ORP, and pH sensors, while not including turbidity. Algorithms detect potential contamination, alerting users of normal or abnormal water quality. A notification node consists of a microcontroller, LCD, and buzzer; therefore, a history of readings cannot be offered. In [9], a reconfigurable smart water quality monitoring (WQM) system consisting of a Field Programmable Gate Array (FPGA) design board, sensors, a Zigbee-based wireless communication module, and a personal computer (PC), is presented. The proposed WQM system acquires levels of water, pH, turbidity, carbon dioxide (CO2) on the surface of water, and water temperature in real-time, not including ORP. In [10], an autonomous real-time device for domestic use using an Arduino Atmega 2560 Microcontroller is developed, measuring just pH, temperature, and turbidity. The device generates alarms and sends alerts via Zigbee to authenticated users about abnormalities in quality parameters or overheaded tanks. The data is manipulated via the microcontroller, transmitted via Zigbee, and displayed on an LCD. In these papers [9, 10], an application displaying the records for either the maintenance team or the user is missing. A low-cost water quality monitoring system and an in-pipe contamination detection system are presented in [11]. Fuzzy logic is used to examine the real-time data before it is uploaded online. When pollution is found in the water, the system uses a solenoid valve to stop the flow of water through the polluted area of the pipe and notifies the users by SMS of the water quality metrics. An IoT and WSN-based water quality monitoring system in [12] monitors the pH, turbidity, temperature, and dissolved oxygen of water samples. Based on a LoRaWan protocol, a receiver obtains sensor values from a LoRa sender, that then sends live data to a ThingSpeak IoT platform for visualization. In [13], a similar system that can log bulk data and transfer it to remote locations is proposed. Water is categorized using contamination detection algorithms. Customers and local agencies access the data through SMS notifications and a single mobile app. The water quality measuring system proposed in [14] uses sensors to measure water quality in real-time, detecting pollutants based on pH, conductivity, temperature, and turbidity, while missing ORP. The data collected is sent to a microcontroller via Wi-Fi, and then to a smartphone or PC for further analysis. Finally, authors in [15] present a reconfigurable smart sensor interface for industrial WSN in IoT environments, designed using IEEE1451 protocol. It includes a software client that sends control signals through a serial port, that collects data, and displays it in the serial debug terminal.

Table 1 summarizes the papers [7] through [15] by listing the most important findings about the monitored parameters as well as the technological solutions adopted to ensure water quality.

In papers [7–14], the wireless communication protocol that was predominantly used was ZigBee. One thing to note is that the visualization of the accumulated information was displayed on a relatively close node, as the ZigBee protocol has a range of no more than 100 m. When it comes to the software, a user interface that is either located locally or online presents the measurement results. However, in none of the works is the implemented mobile application different for the user-consumer and different for the water network maintenance team.

The latest advances for sensors for the drinking water parameters monitored in this work, namely pH, temperature, ORP, turbidity, and flow, were also reviewed. In [16, 17] an RGB light source and light detector-based, and a miniaturized

Table 1 Summarization of related works

Paper	Monitored parameter	Hardware node platform	Wireless communication protocol
[7]	pH Temperature Conductivity Turbidity Oxidation reduction potential	PIC32 MCU	ZigBee
[8]	pH Temperature Conductivity Flow Oxidation reduction potential	PIC32MX220F032B	ZigBee
[9]	pH Temperature Turbidity Level Carbon dioxide (CO2) on the surface of water	Altera DE1-SoC board	ZigBee
[10]	pH Temperature Turbidity	ATMega2560	ZigBee
[11]	pH Temperature Conductivity Turbidity Oxidation reduction potential	Arduino uno	ZigBee
[12]	pH Temperature Turbidity Performs calculations of the dissolved oxygen	ATMega328	LoRaWAN
[13]	pH Temperature Conductivity Turbidity Oxidation reduction potential	ATMega328	Zigbee
[14]	pH Temperature Conductivity Turbidity	Arduino uno + ESP8266	GPRS/GSM
[15]	Temperature Turbidity Carbon dioxide (CO2) Light intensity	XC2C256 CLPD	ZigBee

on-tube turbidity sensor were proposed respectively. Three colorimetric sensors were developed by the authors in [18] for the quantitative immediate detection of pH, lead (Pb 2 +), and fluoride (F -) levels in drinking water. To enable the development of a point-of-care testing (POCT) device, the fluctuations in the color-intensities of the paper sensors with the variation in the levels of lead (Pb 2 +), fluoride (F -), and pH were converted into electrical signals. Graphene-Based Chemiresistor Sensors in [19] is another emerging technology for measuring pH in drinking water. An enhanced Water Temperature Model (WTM +) was developed in [20], which uses an extra insulation layer of soil around the water pipe to account for the drinking water temperature that affects drinking water temperature. Finally, regarding flow, an IoT-based water flow meter using the ThingSpeak cloud is presented in [21].

A variety of types of sensors were used in the related works. In [7], an in-pipe turbidity sensor is built from the ground up using the authors' prior research [22], while Sensorex Corp® provided the other sensor probes. An RTD sensor, which is utilized for temperature detection and temperature correction of pH and EC readings, is included in the pH sensor. A venturi-tube flow meter, a thermistor temperature sensor, a two-electrode method conductivity sensor, a pH electrode, and an ORP electrode were used in [8]. Authors in [9] used a MaxBotix LV-Max-Sonar-EZ 1 ultrasonic sensor to measure the water level, a DS18B20 digital thermometer sensor, an Atlas Scientific pH kit, an SKU: SEN0189 turbidity sensor, and an Analogue Infrared SKU: SEN0219 CO_2 Sensor. In [12], a DS18B20 temperature sensor, a pH Sensor Module E201-C-9, and a Liquid Crystal Display and LED Setup for turbidity sensing were used. Lastly, in [15] a DS18B20 temperature sensor, an MG811 Co2 sensor, a GY-30 light intensity sensor, a KIE-TS-300B turbidity sensor, an MPX5999D pressure sensor, and a YBK10-WQ201 sensor were utilized. No specific sensor types were mentioned in the rest of the articles [10, 11, 13, 14], as more emphasis was given to the interface and communication between the sensors and the microcontroller.

3 Materials and methods

3.1 System architecture

3.1.1 Node architecture

The system consists of nodes, all of which are installed near the water supply of the customer-consumer and collect information from the sensors. This information is then uploaded to the database (remote database) that is implemented as a Firebase. The user and technician applications have been developed as hybrid mobile applications in Android Studio using Android WebView. The system administrator's application has been developed as a desktop application. The HTML and JavaScript files of the above applications are located on GitHub, which in our case also works as an application server. The structural diagram of the system architecture is depicted in Fig. 1.

Each node of the system consists of the ESP32 microcontroller, the water quality and consumption monitoring sensors (flow, temperature, turbidity, pH, and ORP), the node's display, showing the total water consumption, and the Global System for Mobile communication (GSM)/Global Navigation Satellite System (GNSS) module for communication with the database and the synchronization of the node's Real-Time Clock. The power supply of each node is based on a photovoltaic panel which in addition to providing the node with the necessary operating power, also charges the Lithium battery. The structural diagram of the architecture of each node is shown in Fig. 2.

Fig. 1 The structural diagram of the system architecture

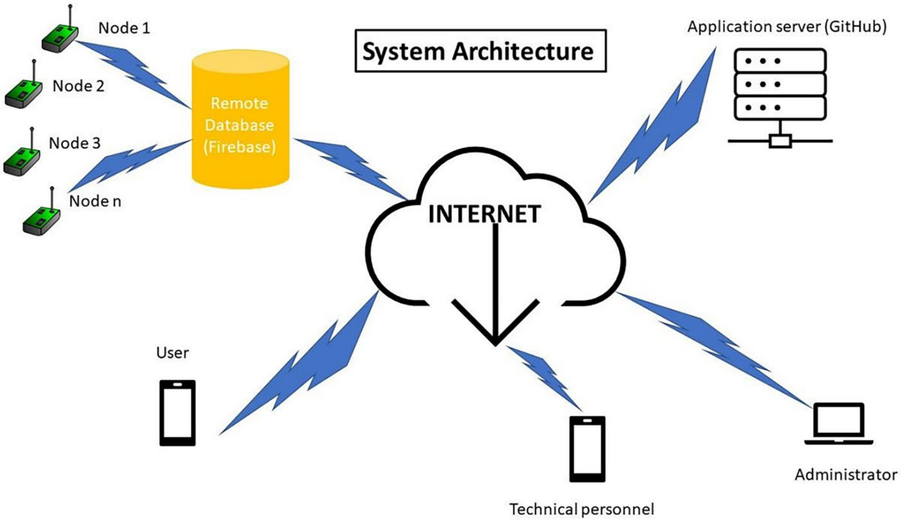
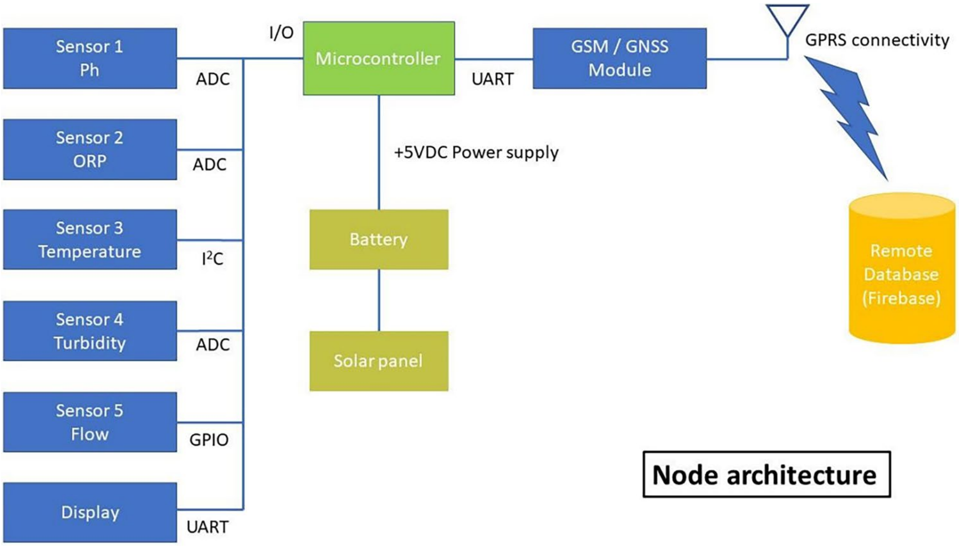


Fig. 2 The structural diagram of the architecture of each node



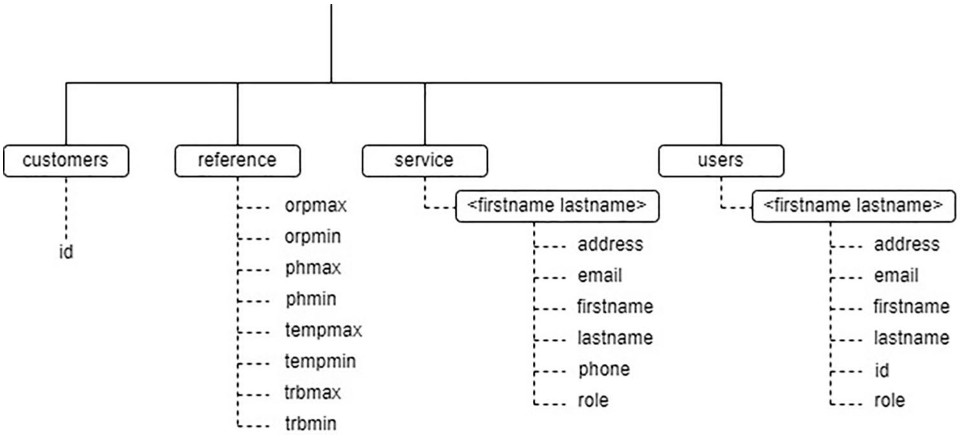
3.1.2 Database design

A more modern approach was taken on the back end, using Google Firebase to take advantage of the availability and reliability of Google's servers. Additionally, the Firebase inherently supports Android applications and provides REpresentational State Transfer Application Programming Interfaces (REST APIs) for various languages such as JavaScript, which was used to develop the front-end application. A tree structure was used to develop the Firebase, with its highest level being divided into the *Customers, Reference, Service,* and *Users* branches.

The ID of each customer is listed under the *Customers* key, which is also the number of their hydrometer. The IMEI number of the GSM modem was used for easy obtainment. Each ID has a corresponding *consumption, location, sensors,* and *user* key. Under the *consumption* key, the consumption history is stored by year, month, day, and time, as well as the total consumption. Under the *location* key, the longitude and latitude of the supply of the consumer are stored. The most recent measured values are stored under the *sensors* key. Finally, the details of the user-consumer who has registered in the application are stored under the *users* key.

The *Reference* key holds the reference values of the drinking water parameters, against which the software makes the comparison to decide the water's quality. The users' account details that support the operation of the system are listed under the *Service* key. Lastly, the consumers' account details are displayed under the *Users* key. A schematic of the hierarchy of the database is illustrated in Fig. 3.

Fig. 3 The hierarchy of the Firebase keys



3.2 System requirements

Identification of water quality is usually done based on the turbidity, the temperature, the ORP, the pH, and the flow of water that reaches the consumer [23]. In this work, these standards were monitored with the aid of sensors. Something that should be taken into consideration is that water quality parameters that require frequent maintenance and calibration to maintain accurate readings over extended periods of time, such as nitrate levels, free chlorine concentration, and dissolved oxygen [7] were chosen not to be monitored. For a long-term, real-time water quality monitoring system like the one proposed in this work, this would not be achievable. The measurements of temperature, pH, and ORP can be used to approximate the Free Chlorine content (HOCl). However, free chlorine monitoring is expensive due to it being highly sensitive to changes when sampling pH, temperature, flow, and pressure. Despite being a crucial factor for human health, nitrates are also not chosen for assessment as methods are either prohibitively priced (UV spectrophotometric approach) or are prone to malfunctions (Ion-Selective electrodes) [7].

Turbidity is a parameter that falls under the aesthetic factors and has to do with the appearance of drinking water, with acceptable values ranging between 0 and 10 NTU (Nephelometric Turbidity Unit) [4]. The temperature of the water indirectly affects other parameters. Water with a pH of 11 or higher has the potential to irritate mucous membranes, skin, and eyes. Since acidic water has a corrosive quality (pH 4 and below), it can also irritate skin [4]. In this paper, the lowest temperature limit is set at 4°C so that there is no possibility of freezing water in the network pipes. The upper limit is set at 25°C so that the drinking water has an acceptable temperature for consumption. The ORP is a measurement that indicates whether a solution tends to gain or lose electrons. A positive ORP reading indicates that water is an oxidizing agent and a negative reading indicates that it is a reducing agent. Under normal conditions, drinking water has a value of ORP between 200 and 400 mV [7]. Lastly, the pH of water is one of the most important factors when testing water quality, as it measures how acidic or basic the water is [7]. Drinking water normally has a pH value between 6.5 and 8.5. These parameters are also presented in Table 2.

Before designing the nodes' IoT platform, suitable sensors that would surround them had to be selected. The criteria for sensor selection in this paper were:

- Satisfactory accuracy in the area of the corresponding measured value.
- Satisfactory resolution.
- Small response time.
- Satisfactory temperature range.
- Long lifetime.
- Long enough time between recalibrations.
- Outputs that are compatible with the inputs of well-known microcontrollers in the market.
- Availability in the European market.
- Reasonable price.

Moreover, in the proposed implementation, the application that was developed displays understandably whether the water that is reaching the user's supply is suitable for consumption or not. Another application addressed to the maintenance team of the water supply network clearly shows on the map of the network any possible problems with the water supply, as well as their cause. Another desktop application aimed at the network manager was developed, which has general supervision of the system and its users.

Finally, each node independently uploads its measurements to a Firebase database using REST API. Therefore, a wireless protocol was used that provides a greater range and meets the requirements for wide geographical coverage, low power consumption, and the usage of commercially available transceivers.

Table 2 Water parameters that were considered in this paper

Parameter	Units	Acceptable values
Turbidity	NTU	0–10 [4]
Temperature	°C	4–25 [4]
ORP	mV	max 800 [7]
pH	pH	6.5–8.5 [7]

3.3 System hardware

In this section, the hardware parts that were used for the implementation of the monitoring system as well as some of their basic characteristics are presented.

3.3.1 Sensors

3.3.1.1 Temperature sensor The temperature sensor that was chosen was the Resistance Temperature Detector (RTD) type platinum sensor PT1000 from Atlas Scientific [24]. It shows an extremely linear response compared to similar sensors made of nickel or copper. To connect the PT1000 sensor to the microcontroller of the IoT platform the EZO™ RTD Temperature Adapter Circuit from Atlas Scientific [25] was used.

3.3.1.2 ORP sensor For the measurement of ORP the Consumer Grade ORP Probe of Atlas Scientific [26] was chosen. ORP measurements can be either positive or negative, and represent how strongly electrons are transferred to or from different substances in a liquid. To connect the ORP sensor to the microcontroller of the IoT platform, the Gravity™ Analog ORP Sensor/Meter adapter circuit of Atlas Scientific [27] was used.

3.3.1.3 pH sensor The Consumer Grade pH Probe from Atlas Scientific [28] was chosen for pH measurement. The pH probe measures the activity of hydrogen ions in a liquid. On the head of the pH sensor, there is a glass membrane, which allows the hydrogen ions from the examined liquid to diffuse into the outer layer of the glass, while larger ions remain in the solution. The difference in the concentration of hydrogen ions creates a very small current which is proportional to the concentration of hydrogen ions in the liquid that is being measured. To connect the pH sensor to the microcontroller of the IoT platform, the Gravity™ Analog pH Sensor / Meter adapter circuit of Atlas Scientific [29] was used.

3.3.1.4 Gravity water flow sensor The Gravity Water Flow YF-S201 [30] sensor was chosen for the monitoring of water consumption, which measures the rate at which a liquid flows through it. The flow sensor consists of a plastic body, a magnetic rotor, and a hall sensor. When the water flows, the magnetic rotor rotates accordingly, and then the hall sensor generates pulses according to the flow rate. According to the manufacturer, 480 pulses correspond to 1 L of water. The average water flow rate for a European household, as reported by the European Commission [31], typically ranges between 4 and 12 L per minute for taps/faucets. Therefore, in the context of building a prototype system for this work, a flow sensor that measures up to 30 L per minute was deemed satisfactory.

3.3.1.5 Turbidity sensor For the measurement of water turbidity, an OEM [32] sensor of type Amphenol TSD-10 was chosen. The turbidity sensor can detect suspended particles in water by measuring the propagation of light and the scattering rate which changes with the amount of Total Suspended Solids (TSS) in the water. As the TSS increases, the level of turbidity of the liquid also increases. TSS is another common metric for measuring turbidity in water and it has a linear relationship with NTU [32]. The documentation includes a reference chart for the mapping from the output voltage to the NTU according to different temperatures. e.g. In pure water, the output of the sensor is NTU < 0.5, it should output "4.1 ± 0.3V" when the temperature is 10 ~ 50°C [32].

Some of the basic characteristics of the sensors and the adapter circuits that were mentioned are summarized in Tables 3 and 4 accordingly.

3.3.2 Microcontroller

For the implementation of the IoT platform in this work, the ESP32 microcontroller of the company Espressif, and specifically the NodeMCU-32S module [33] developed by Anxinke Technology was chosen. The ESP32 contains two Central Processing Unit (CPU) cores that can be controlled separately. The clock frequency ranges from 80 to 240 MHz, supports the operating system FreeRTOS, and includes onboard Wi-Fi, Bluetooth, and Bluetooth Low Energy (BLE). The onboard Wi-Fi and Bluetooth peripherals of the ESP32 were not used in this work. Its micro Universal Serial Bus

Table 3 Sensor characteristics

	PT1000 temperature sensor [24]	Consumer Grade ORP Probe [26]	Consumer Grade pH Probe [28]	Gravity Water Flow YF-S201 sensor [30]	OEM Amphenol TSD-10 sensor [32]
Range	–200°C to 850°C	± 1100 mV	2–13	3.5 to 12 V	0–4.5V (Analog output)
Accuracy	± (0.15 + (0.002*t))	± 1.1 mV	± 0.1	± 5% (2 – 30 L/min)	–
Response time	90% in 13 s	95% in 1 s	95% in 4 s	–	< 500 ms
Temperature range	–	1°C to 60°C	1°C to 60°C	< 120°C	5°C to 90°C
Time before recalibration	–	1 year	3 months	–	–
Expected lifetime	15 years	1.5 year	12–18 months	–	–

Table 4 Adapter circuit characteristics

	EZO™ RTD Temperature Adapter Circuit [25]	Gravity™ Analog ORP Sensor/Meter Adapter Circuit [27]	Gravity™ Analog pH Sensor/Meter Adapter Circuit [29]
Range	– 126.000°C to 1254°C	– 1500mV to 1500mV	0.1–14.0
Accuracy	± (0.1 + 0.0017 x °C)	+ /– 1mV	+ /–0.2
Response time	1 reading per sec	Continuous analog output	Continuous analog output
Power supply	3.3 V/14.3 mA to 5 V/16 mA	3.3V/3mA to 5V/3mA	3.3V/3mA to 5V/3mA
Data protocol	UART & I2C	Analog voltage 0 to 3 V	Analog voltage 2.7 to 0.2 V

(USB) port was used to download the firmware that is run by the IoT platform to the processor, as well as power the platform from the Solar Power Manager.

3.3.3 Display and user interface design

For the user to have direct communication with the system, and also to simulate the system with a water meter familiar to the user, it was considered necessary to install a display showing the water level and water consumption. Moreover, basic information about the system such as the intensity of the GSM and GNSS signal strength should also be displayed. The display chosen is model NX3224T024, a 2.4-inch Liquid Crystal Display (LCD) type from Nextion [34].

The Nextion editor tool was used to design the User Interface of the LCD screen. The screen mainly shows the total user consumption and secondarily some basic technical information such as the GSM signal level and the GNSS signal level. There are two "pages" of the UI displayed, one at the launch of the node, and the other at the initialization of the node's system. The information on the screen is updated through serial communication between the microcontroller and the Universal Asynchronous Receiver/Transmitter (UART). When making the User Interface (UI) pages, the design was compiled in the editor to produce a Thin Film Transistor (TFT) file. That file was then stored in a micro Secure Digital (SD) card and placed in the corresponding slot of the display. The TFT file is stored in the screen's memory once powered by 5 V, making the UI fully operable. The power supply was then shut off and the card was removed.

3.3.4 Power supply

As each node should operate autonomously and indiscriminately, they are each powered by a 5 V/10 W max photovoltaic panel [35]. To achieve an uninterruptible operation, even during the night, the photovoltaic panel is also charged by a 3.7 V / 5000 mAh Lithium battery. Particular attention should be paid to the placement of the photovoltaic panel so that it is located in a place that is illuminated by the sun for a long period of time and with a proper inclination towards it. A Solar Power Manager 5 V module from DFRobot [36] was used to charge the battery and power the node circuits. The module has connectors for the photovoltaic panel (SOLAR IN), the battery (BAT IN), and a USB port (USB OUT) through which it can uninterruptedly provide 5 V voltage for the operation of the node. The maximum output current is 1 A and exceeds the needs of the node, which, according to the measurements made during the experimental operation, consumes an average of 175 mA, while the maximum power consumption comes to 250 mA.

3.3.5 PCB layout design

A board for the hosting of the node components was designed, to enable the connection of the individual circuits to the ESP32 microcontroller. In addition, some elements were added to improve the node's consumption. The power supply of some subsystems does not necessarily have to be continuous, therefore we can interrupt the power supply of the sensor adaptation modules during the time they are inactive. To save up on energy consumption, measurements are taken every hour, so that all sensors (except for the flow sensor) remain inactive in the interval between measurements. Through one of the GPIO pins, the microcontroller activates or deactivates a micro-relay of two contacts. One contact of the relay powers the sensors operating at 3.3 V while the other powers the sensors operating at 5 V. The board was designed using Autodesk's EAGLE software. Figure 4 illustrates the placement of the materials on the board (left) and the assembled board of a node (right). A schematic of the developed system is depicted in Fig. 5.

3.4 Communication protocol

For an internet connection, the SIM7070G Cat-M/NB-IoT/GPRS HAT modem of Waveshare [37] was chosen. This modem is based on SIMCom's SIM7070G and includes a GNSS geolocation receiver, a function that was useful for the synchronization of the Real-Time Clock of the IoT platform. An important criterion for the selection of this modem was the ability to support Hypertext Transfer Protocol Secure (HTTPS) traffic and Secure Sockets Layer and Transport Layer Security (SSL/TLS) 1.2, a requirement for communication with a Firebase. The modem communicates with the microcontroller of the IoT platform via a UART. Although the modem is designed as an expansion board for Raspberry pi it can also work with most microcontrollers on the market by using AT commands. It has a power supply of 5V and a current consumption of about 41mA in idle mode [37].

3.5 System firmware

For the design of the firmware, the Integrated Development Environment (IDE) Eclipse was used. The IoT platform was designed as a Finite State Machine (FSM). The states to which the FSM transitions are:

- System initialization.
- Water flow calculation and LCD screen update.
- Get signal strength.
- Reading sensor values.
- Get GNSS time.
- Update Firebase.

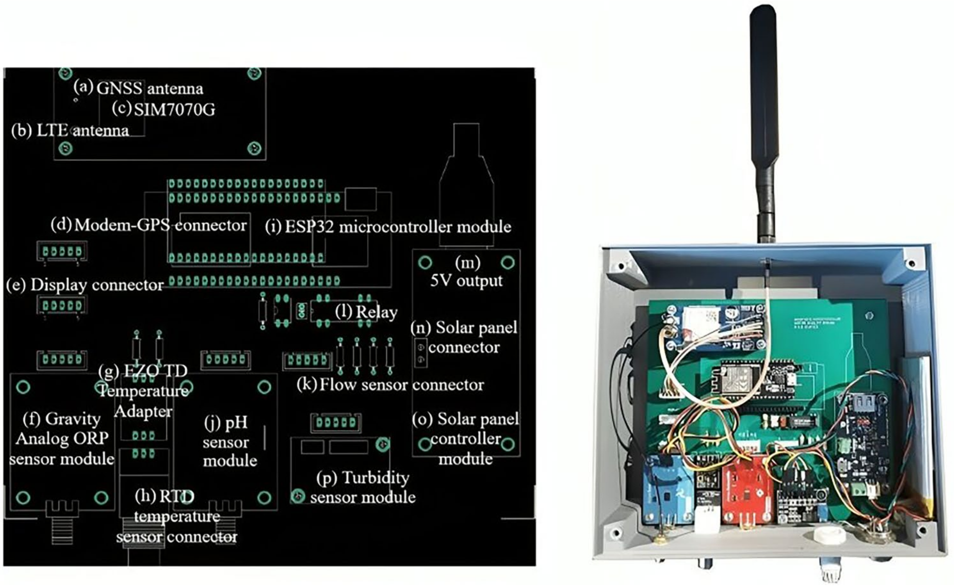

(a) GNSS antenna
(c) SIM7070G
(b) LTE antenna
(d) Modem-GPS connector
(i) ESP32 microcontroller module
(e) Display connector
(m) 5V output
(l) Relay
(n) Solar panel connector
(g) EZO TD Temperature Adapter
(k) Flow sensor connector
(f) Gravity Analog ORP sensor module
(j) pH sensor module
(o) Solar panel controller module
(h) RTD temperature sensor connector
(p) Turbidity sensor module

Fig. 4 Left: The placement of the system's hardware modules on the board. Right: The assembled board of a singular node

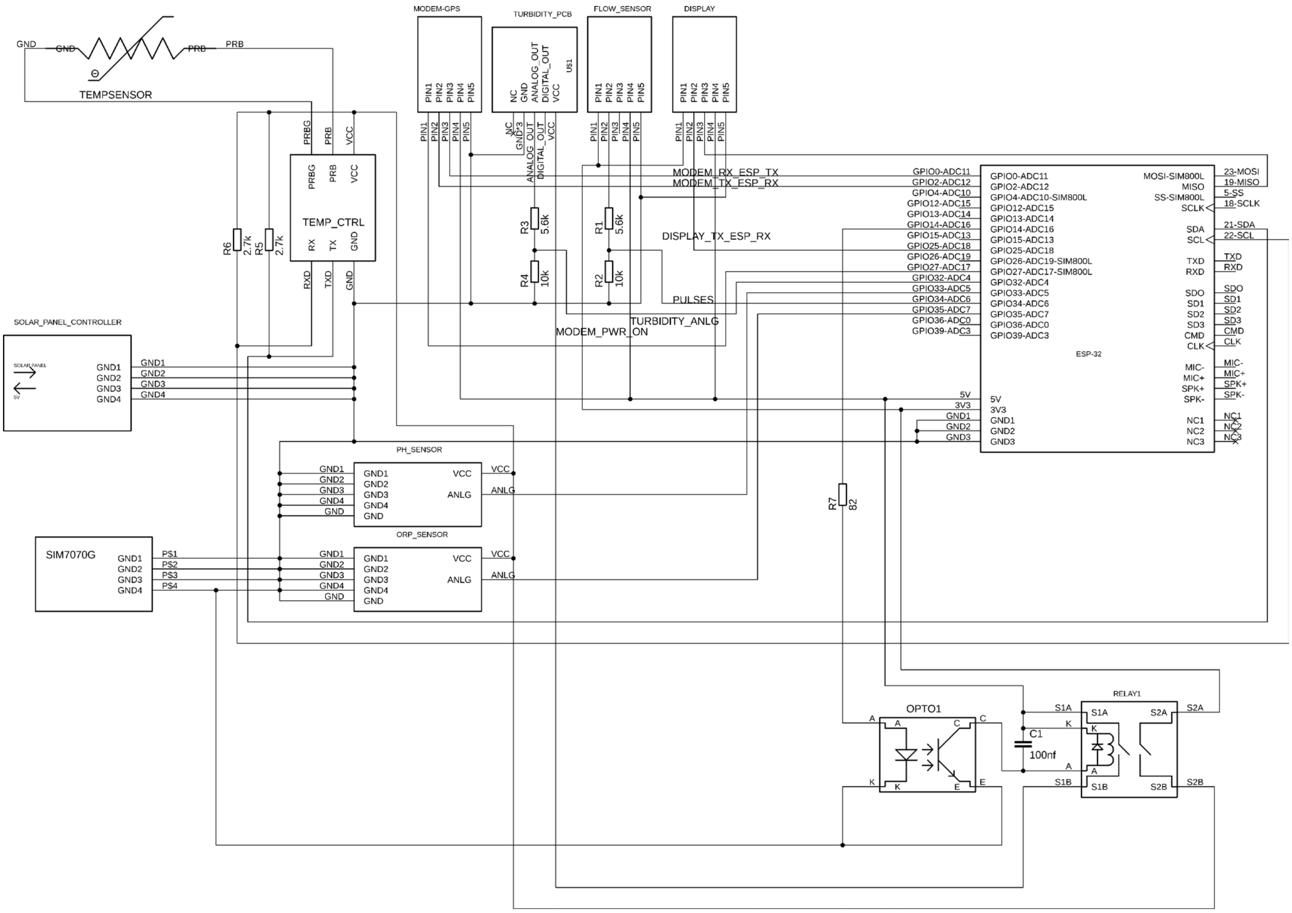


Fig. 5 The schematic of the developed device

Figure 6 shows the structure of the FSM. A hardware timer that uses interrupts acts as a time base for the FSM to transition from one state to another. Every 60 s, a timer interrupt is generated. Then, every 5 min the software checks the GSM signal level, and every 60 min, the software updates the database. If the signal level is too low, indicating that the signal is too weak, the modem response will be ignored and the software will keep checking the level until it reaches a satisfactory level. The following section provides the operation of the FSM implementation code that is executed by the microcontroller.

When in the 'System initialization' state, the role of the General-Purpose Input/Output (GPIO) pins is defined for switching the power supply to the sensors, for sending and receiving data in serial communication with the modem, and for serial communication with the LCD screen. The Analog-to-Digital Converter (ADC) is also configured, defining the channel to which the ORP, pH, and turbidity sensors are to be connected. The initialization of the PCNT pulse counter which is used to calculate the water consumption is followed, as well as the UART initialization. The initialization of the Inter-Integrated Circuit (I2C) bus, which reads the values of the water temperature sensor, is then performed. After all the peripherals have been initialized, a task is created that continuously monitors the data received from the modem. The modem is then turned on and the International Mobile Equipment Identity (IMEI) of the modem is read, as well as the name of the provider whose network it is connected to. The GPRS connection is then attempted. If the connection is successful, a timer interrupt is triggered to drive the FSM to subsequent states, otherwise, the user screen displays an error message. The next step is connecting to the Firebase with an HTTP GET request to read the stored total consumption since the microcontroller has no built-in memory for the permanent storage of user data. After the stored information is read, the appropriate information on the LCD is updated. The reading of the node's position follows, using the GNSS receiver embedded in the SIM7070G module. A task dedicated to measuring the water consumption is then created and the initialization of the system is completed.

When in the 'Water flow calculation and LCD screen update' state, the microcontroller manages three tasks "in parallel": the modem data receiving task, the volume measuring task, and the FSM state transition task. Through the latter task,

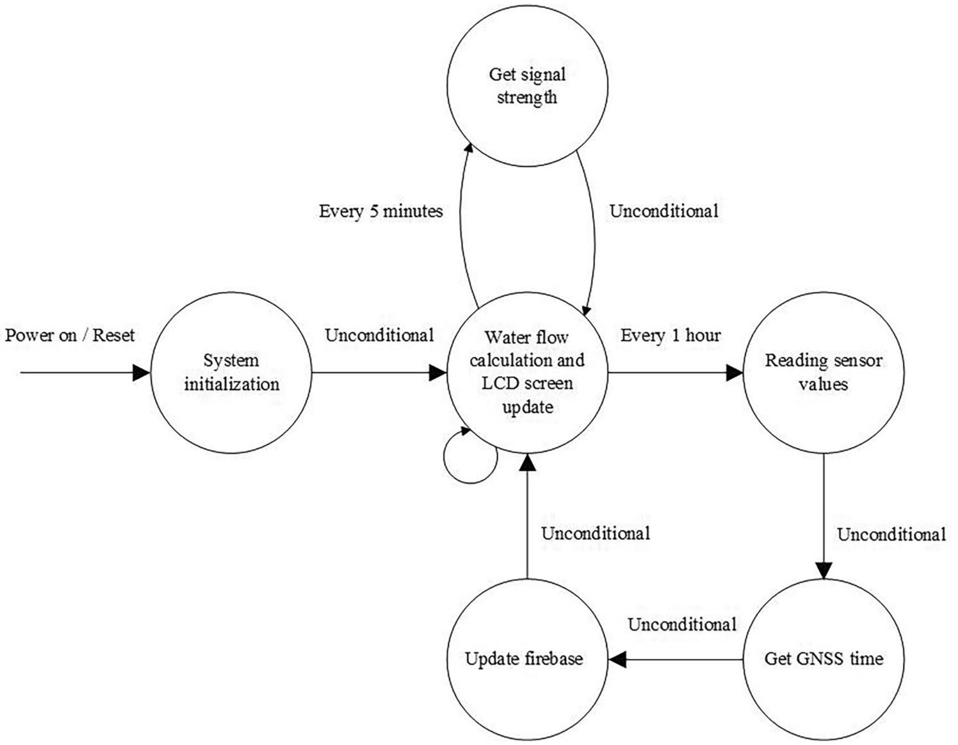


Fig. 6 The structure of the finite state machine

the FSM transitions to the 'Get signal strength' state every five minutes. The current GSM signal level is read and the result is displayed on the LCD screen. Every sixty minutes, the FSM transitions successively to the 'Reading sensor values', 'Get GNSS time', and 'Update Firebase' states. Lastly, the FSM returns to the 'Water flow calculation and LCD screen update' state and remains there until it is triggered to one of the states.

3.6 GUI application

The user application was designed in HTML and JavaScript. It was then "wrapped" using Android WebView in the integrated application development environment Android Studio. The development of the application in Android Studio was done in Java. At the launch of the application, the user is prompted to create an account if they do not already have one. After successfully connecting to the application, the user is taken to the "Summary screen" in which the user can see whether the quality of the water consumed is acceptable, and the total water consumption up to the last update of the application. A different screen is displayed for the cases of acceptable or unacceptable water quality. The user can see the personal details they have entered by selecting "Settings" from the drop-down menu, and make corrections. A view of their consumption for the current month and a detailed bar graph of their consumption per day is available by selecting the 'Consumption analysis' option from the drop-down menu. They can also select another month or even a different year to see the corresponding consumption. Figure 7 displays some indicative pages of the user application.

The technician's application was designed using the same tools. However, this account must already be registered in the database and have at least "support" role rights, otherwise, this account cannot access the application and is automatically disconnected. After a successful connection to the application, the user automatically goes to the "Problems in the network" screen, where they see in list form any problems that occur in the consumers of the water supply network and where these problems are caused. The consumer's details are also presented so that it is easy to identify and communicate with them. By selecting the "Network Map" option from the drop-down menu the user can see visually if there are any identified problems in the water supply network, in the area in which they are located. Their location is identified on the map by a blue pin, the problematic supplies are identified by a red pin, and the non-problematic supplies are marked with a green pin. The map is implemented with the help of the open-source JavaScript library Leaflet [38]. These pages, along with the map, are displayed in Fig. 8.

The administrator application is not a mobile application but a desktop one, designed in HTML and JavaScript. Just like the technician's application, the account must already be registered in the database and have an "admin" role,

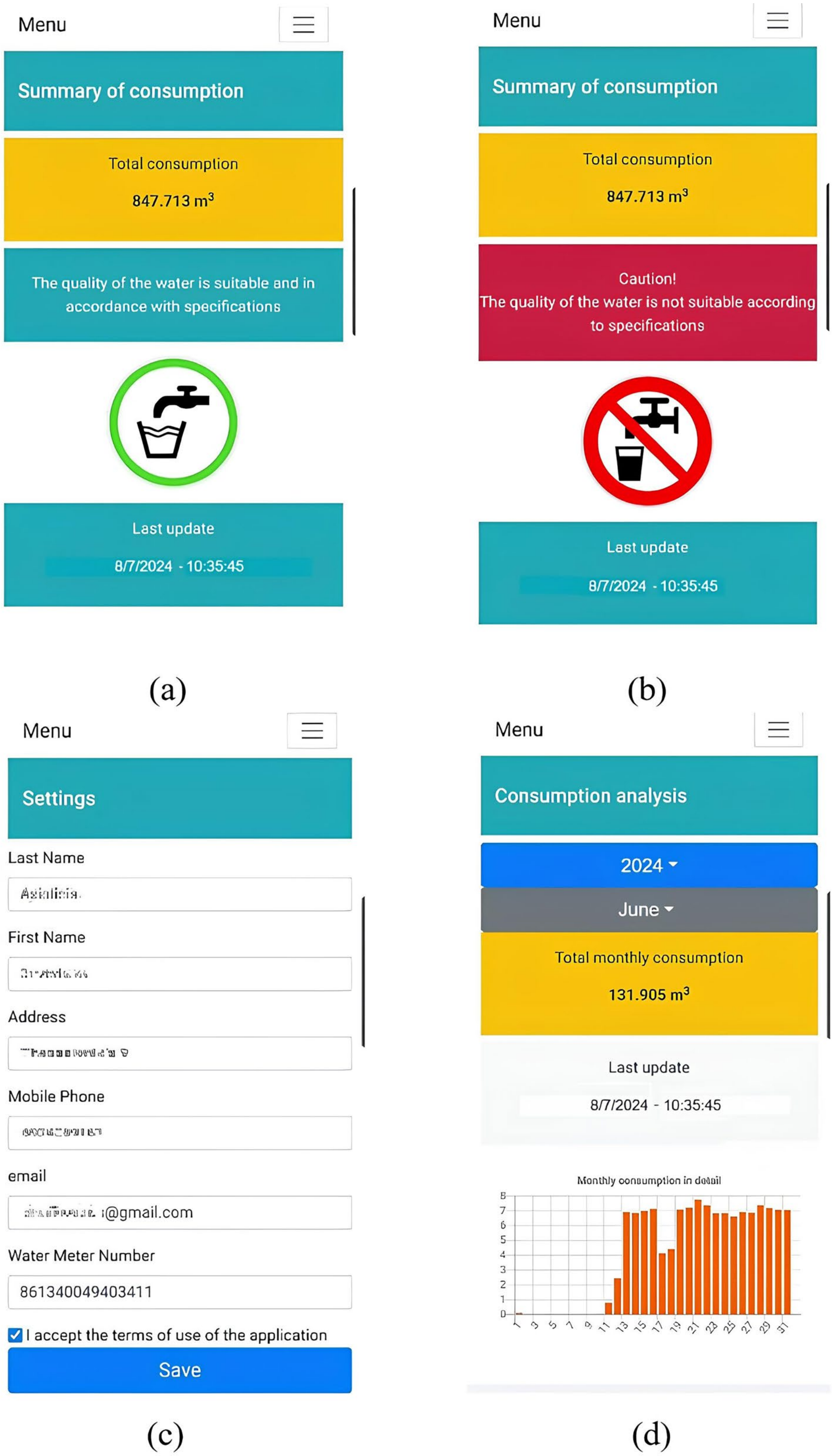


Fig. 7 The user's mobile application. **a** The menu when the water quality is acceptable, **b** The menu when the water quality is unacceptable, **c** The settings menu, **d** The consumption analysis menu for June of 2024

(a)

Menu

Summary of consumption

Total consumption

847.713 m^3

The quality of the water is suitable and in accordance with specifications

Last update

8/7/2024 - 10:35:45

(b)

Menu

Summary of consumption

Total consumption

847.713 m^3

Caution!
The quality of the water is not suitable according to specifications

Last update

8/7/2024 - 10:35:45

(c)

Menu

Settings

Last Name: [redacted]

First Name: [redacted]

Address: [redacted]

Mobile Phone: [redacted]

email: [redacted]@gmail.com

Water Meter Number: 861340049403411

☑ I accept the terms of use of the application

Save

(d)

Menu

Consumption analysis

2024

June

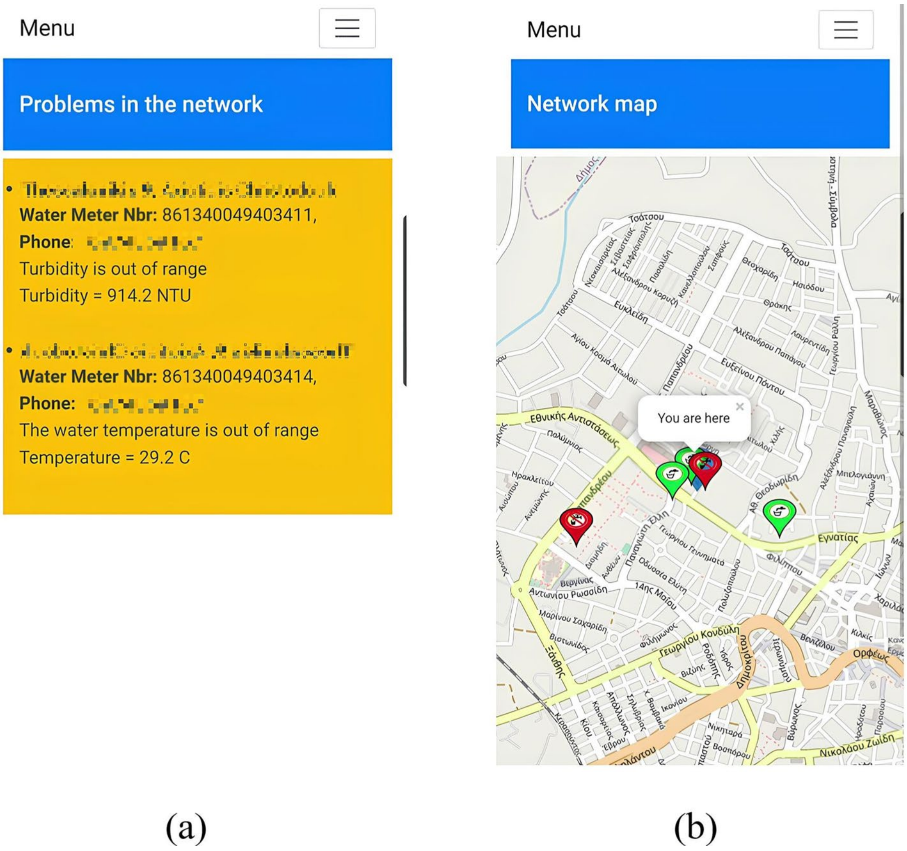
Total monthly consumption

131.905 m^3

Last update

8/7/2024 - 10:35:45

otherwise, this account cannot have access to the application and is automatically disconnected. After successfully connecting to the application, the user is automatically redirected to the main administration screen, where they can select the desired action. By selecting the "Manage users" icon, the administrator can enter users and assign them roles (consumer, technician, or administrator). If the user selects the "Problems in the network" icon then they see



Menu

Problems in the network

- [redacted]
  Water Meter Nbr: 861340049403411,
  Phone: [redacted]
  Turbidity is out of range
  Turbidity = 914.2 NTU
- [redacted]
  Water Meter Nbr: 861340049403414,
  Phone: [redacted]
  The water temperature is out of range
  Temperature = 29.2 C

Fig. 8 The technician's mobile application. **a** The menu that lists network problems, **b** The map that indicates the status of the network

in list form any problems that the consumers of the water supply network are facing and where these problems are caused, along with the consumers' details. If the user selects the "Network Map" icon, they can see visually if there are any identified problems in the water supply network. Problematic supplies are identified by a red pin and non-problematic supplies by a green pin. These pages, along with the map, are displayed in Fig. 9.

4 System evaluation

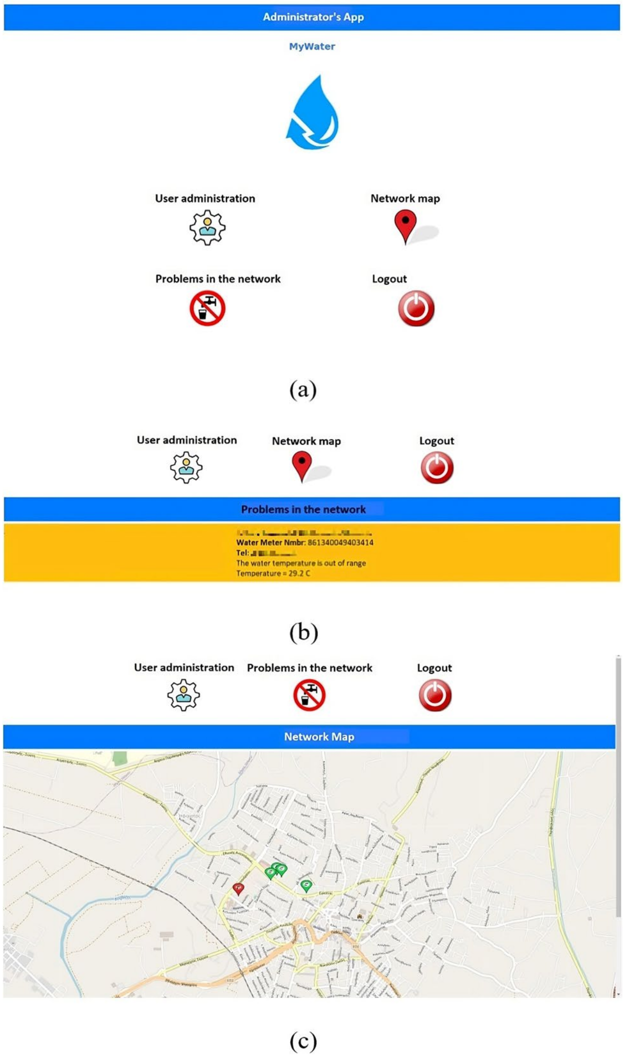
4.1 Testbench construction

To be able to test the operation of the system, a test model was constructed, where, under conditions simulating real ones, all subsystems were tested.

First, the box that houses the board of the node to collect data from the sensors was designed and printed on a 3D printer. The design of the box was done with Autodesk's Inventor software which is available for free for academic use. Using the same procedure, the two stands of the photovoltaic panel were designed and printed, giving the panel an inclination of 32°.

For the installation of the "in pipe" turbidity sensor in the hydraulic circuit, a freely available adapter with a Creative Commons license was used, which was printed on a 3D printer [39]. The turbidity sensor was then adapted to it using an o-ring and a plastic tire up. The temperature, ORP, and pH sensors were mounted with T-type connectors, supplied as an accessory for these sensors by their manufacturer, Atlas Scientific.

The flow sensor was placed in the water intake, which is taken care of by a 12 V submersible pump. The operation of the pump is not continuous but is turned on and off at random times with the help of an Arduino uno circuit. To avoid wastage of water when testing the system, a closed hydraulic circuit is used, which recycles the water present in a container. The completed and fully functional test model is shown in Fig. 10.



[redacted]
Water Meter Nmbr: 861340049403414
Tel: [redacted]
The water temperature is out of range
Temperature = 29.2 C

Fig. 9 The network administrator's desktop application. **a** The main menu, **b** The menu that lists network problems, **c** The map that indicates the status of the network

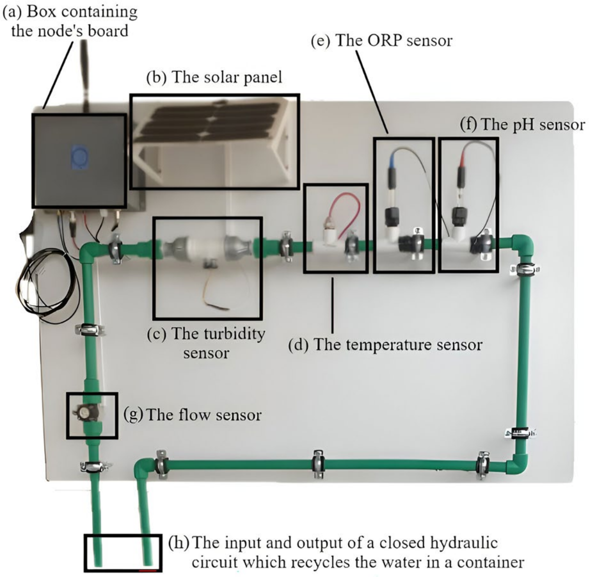
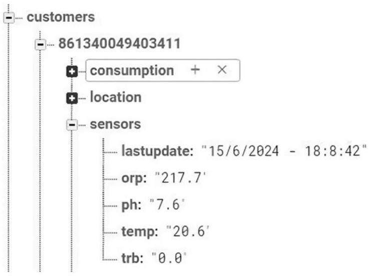
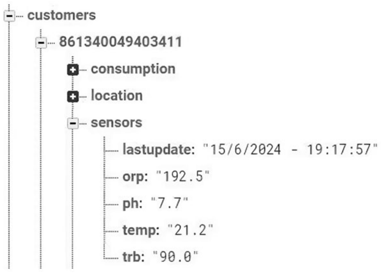

(a) Box containing the node's board
(e) The ORP sensor
(b) The solar panel
(f) The pH sensor
(c) The turbidity sensor
(d) The temperature sensor
(g) The flow sensor
(h) The input and output of a closed hydraulic circuit which recycles the water in a container

Fig. 10 The fully operational test model of the monitoring system



```
customers
  861340049403411
    consumption
    location
    sensors
      lastupdate: "15/6/2024 - 18:8:42"
      orp: "217.7"
      ph: "7.6"
      temp: "20.6"
      trb: "0.0"
```

Fig. 11 Sensor records in the database for uncontaminated drinking water, from a customer's water supply



```
customers
  861340049403411
    consumption
    location
    sensors
      lastupdate: "15/6/2024 - 19:17:57"
      orp: "192.5"
      ph: "7.7"
      temp: "21.2"
      trb: "90.0"
```

Fig. 12 Sensor records in the database for soil-contaminated water, from a customer's water supply

4.2 Calibration

Before the measurements and testing of the system can begin, the sensors must be calibrated.

The calibration of the temperature sensor, as described in the manufacturer's data sheets for the PT1000 temperature probe and the EZO-RTD embedded temperature circuit, was done via the I2C interface by sending the command "Cal,0" with the temperature sensor immersed in a container of water and ice. To achieve the calibration, the microcontroller executes a special program specifically for this purpose. The manufacturer recommends that the calibration is repeated every three years.

The calibration of the pH sensor and the ORP sensor is performed using buffers with specific values, which are included in the sensor kit. Calibration is achieved when running the normal microcontroller program by dipping the sensors into the container with the specific buffers, making repeated measurements, and taking the average by reading the values recorded in the database. Finally, based on the measurements, the offset variables in the microcontroller program are set accordingly. The manufacturer suggests that the probes need calibration once a year for the first two years of their life and then every six months.

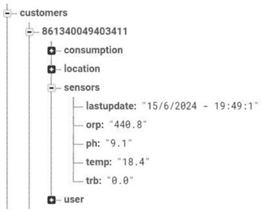
4.3 Testing

After the calibration of the sensors was completed, the overall testing of the system was performed. The users were defined from the management environment, and the user and technician applications were installed on an Android smartphone. Initially, the values recorded by the sensors and the readings presented by the user and technician applications were checked when clean drinking water flows in the hydraulic circuit. These readings are shown in Fig. 11.

The water was then "contaminated" by pouring soil into the container, to mimic a water pipe breakage situation, and observed the change in the water supply of the sensor readings (Fig. 12). The biggest change that was observed was in the turbidity sensor, as expected.

The water was also "contaminated" with a chemical agent by adding a few drops of sodium hypochlorite NaClO—the well-known bleach—to the clean drinking water of the container, to simulate the conditions of a chemical accident. In this case, a large increase in the pH sensor reading was observed, which, puts this indicator out of range, giving the water an alkaline behavior (Fig. 13). Moreover, a large increase in the ORP sensor reading was observed, yet keeping the ORP value within specifications.

Fig. 13 Sensor records in the database for water containing NaClO, from a customer's water supply



```
customers
  861340049403411
    consumption
    location
    sensors
      lastupdate: "15/6/2024 - 19:49:1"
      orp: "440.8"
      ph: "9.1"
      temp: "18.4"
      trb: "0.0"
    user
```

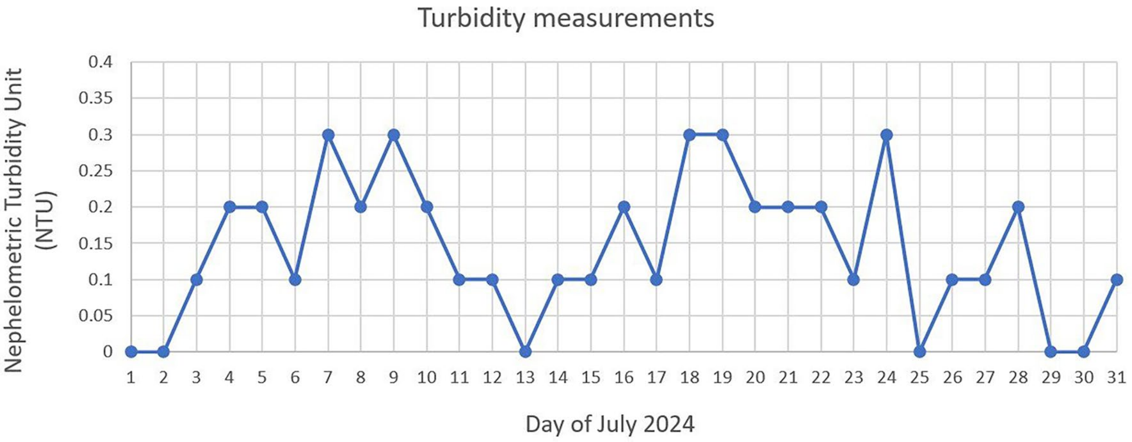


Fig. 14 Recorded values of water turbidity over the course of the month of July 2024

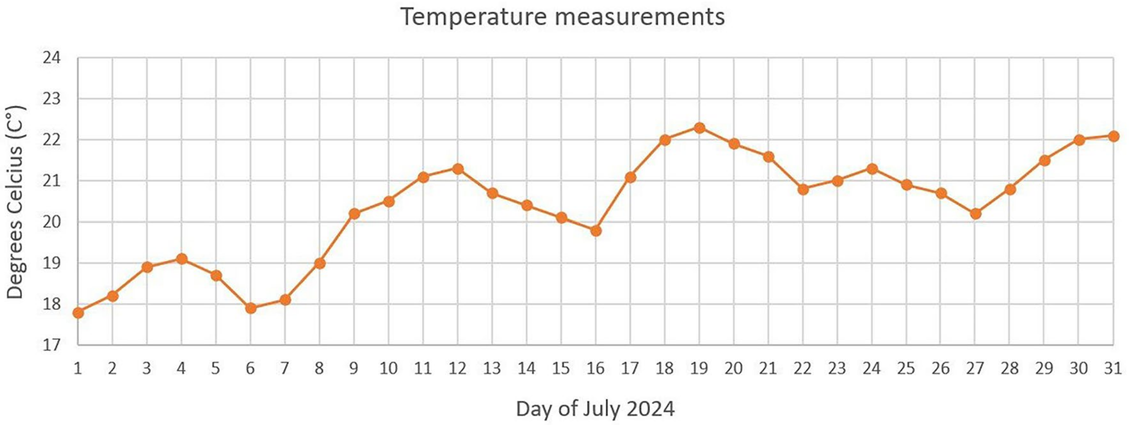


Fig. 15 Recorded values of water temperature over the course of the month of July 2024

Finally, Figs. 14, 15, 16 and 17 present the readings recorded over a month by the sensors for turbidity, temperature, ORP, and pH, respectively. The samples were subjected to a reductionist approximation, showing the average daily value of each metric, for the month of July of 2024.

5 Conclusion

In this paper, the steps for the implementation of a monitoring system for drinking water quality and consumption were outlined. The system consists of two parts, the hardware part, and the software part. The hardware part consists of the board where the ESP32 microcontroller with its peripherals is mounted, while the software consists of the mobile applications for consumers and technicians, and the desktop application of the network administrator.

This work is complementary to similar works in the literature and offers an integrated solution for monitoring the quality and consumption of drinking water, including not only the data collection nodes but also developing

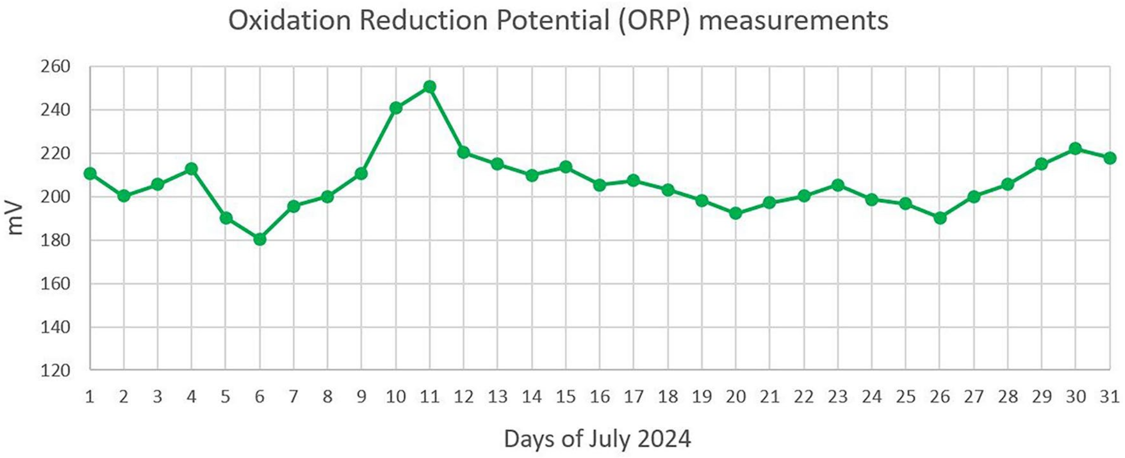


Fig. 16 Recorded values of water ORP over the course of the month of July 2024

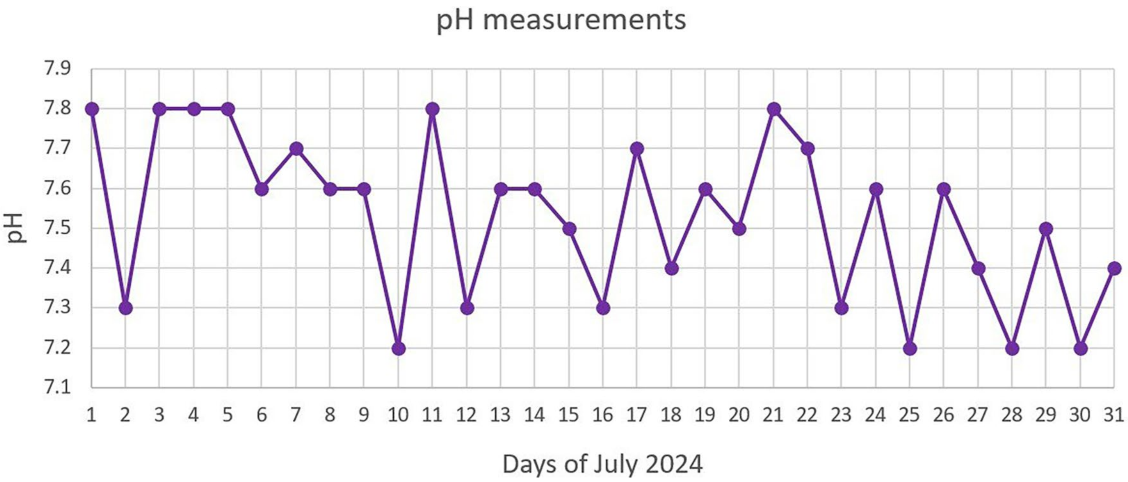


Fig. 17 Recorded values of water pH over the course of the month of July 2024

the related applications. By grouping the nodes and using a set of water quality detection sensors per group, this system could eventually be a cost-effective solution that is highly beneficial for the monitoring and maintenance of the water supply network.

As it is concluded from the readings, the system worked well, recognizing the increase in water turbidity when the water became "contaminated" by adding soil and recognizing the unsuitability of the water when it was "contaminated" at a second time with a chemical agent.

The system is initially intended to work in rural or semi-rural areas, where usually each building is a separate property and has a drinking water supply with an associated water meter. When it comes to the cost of using GPRS for data exchange every time the system updates the database, it varies depending on factors such as location and service provider. In Europe, GSM messages generally do not exceed €0.02 per MB (megabyte) [40]. The cost declines further when using bundles or special plans for business communications. This system could also be extended to the environment of an urban area by changing the communication protocol of the nodes with the internet. Instead of having each node communicate autonomously with GPRS, a group of nodes (e.g. the users of an apartment building) could communicate utilizing a short-range wireless protocol (e.g. ZigBee). A sink node takes the role of a gateway for the communication of the nodes with the database and updates the Real-Time Clock and the location of the nodes.

Such an extension should consider the uninterrupted power supply to the individual nodes. Therefore, for each group of nodes, a photovoltaic panel with a corresponding battery should be installed, in order to satisfy the energy requirements of the nodes.

The board design could remain the same by removing the GSM modem and the photovoltaic panel controller module. In place of the GSM modem, a Zigbee module would be installed, using the existing serial port. Several changes would have to be made to the software of the nodes to adapt to the new communication philosophy, however, they are feasible and the cost of this change is tolerable.

Finally, a cost-effective change would be to install the sensors—apart from the flow sensor—per group of nodes, in the central supply of an apartment building. Therefore, the cost of the system per group of nodes could be reduced.

Author contributions Christodoulos Axiotidis (Original draft preparation, Investigation, Validation and Software). Evangelia Konstantopoulou (Writing, Submission, Review, Data curation, Editing). Nicolas Sklavos (Conceptualization, Supervision, Editing, Review and Revise). All authors wrote the manuscript text and prepared figures and tables. All authors reviewed the manuscript.

Funding No funding was received for conducting this study.

Data availability The data used in this study is private. However, upon request, the authors may consider sharing the data for further research purposes. Interested researchers can contact the corresponding author for inquiries regarding data availability.

Declarations

Competing interests The authors declare no competing interests.

References

1. Norris M, Celik ZB, Venkatesh P, Zhao S, McDaniel P, Sivasubramaniam A, Tan G. IoTRepair: flexible fault handling in diverse IoT deployments. ACM Trans Internet Things. 2022;3(3):1–33. https://doi.org/10.1145/3532194.
2. Zhu H, Chau SCK, Guarddin G, Liang W. Integrating IoT-sensing and crowdsensing with privacy: privacy-preserving hybrid sensing for smart cities. ACM Trans Internet Things. 2022;3(4):1–30. https://doi.org/10.1145/3549550.
3. Sharma H, Sharma S. A review of sensor networks: technologies and applications. Recent Adv Eng Comput Sci (RAECS). 2014. https://doi.org/10.1109/RAECS.2014.6799579.
4. Hamed N, Gaglione A, Gluhak A, Rana O, Perera C. Query interface for smart city internet of things data marketplaces: a case study. ACM Trans Internet Things. 2023;4(3):1–39. https://doi.org/10.1145/3609336.
5. World Health Organization. Guidelines for drinking-water quality. 4th ed. 2017. ISBN 978-92-4–154995-0
6. Hall J, et al. On-line water quality parameters as indicators of distribution system contamination. J Amer Water Works Assoc. 2007;99(1):66–77. https://doi.org/10.1002/j.1551-8833.2007.tb07847.x.
7. Lambrou TP, Anastasiou CC, Panayiotou CG, Polycarpou MM. A low-cost sensor network for real-time monitoring and contamination detection in drinking water distribution systems. IEEE Sens J. 2014;14(8):2765–72. https://doi.org/10.1109/JSEN.2014.2316414.
8. Cloete NA, Malekian R, Nair L. Design of smart sensors for real-time water quality monitoring. IEEE Access. 2016;4:3975–90. https://doi.org/10.1109/ACCESS.2016.2592958.
9. Myint CZ, Gopal L, Aung YL. WSN-based reconfigurable water quality monitoring system in IoT environment. In: Proceedings of the 14th international conference on electrical engineering/electronics, computer, telecommunications and information technology (ECTI-CON) 2017. https://doi.org/10.1109/ECTICon.2017.8096345
10. Neha N. Wireless sensor network based system design for chemical parameter monitoring in water. Int J Electr Commun Soft Comput Sci Eng. 2015;3(6):56.
11. Priya SK, Shenbagalakshmi G, Revathi T. IoT based automation of real time in-pipe contamination detection system in drinking water. In: Proceedings of the 2018 international conference on communication and signal processing (ICCSP) 2018. https://doi.org/10.1109/ICCSP.2018.8524255
12. Simitha K, Raj MS. IoT and WSN based water quality monitoring system. In: Proceedings of the 2019 3rd international conference on electronics, communication and aerospace technology (ICECA), 2019. https://doi.org/10.1109/ICECA.2019.8821859.
13. Priya SK, Shenbagalakshmi G, Revathi T. Design of smart sensors for real time drinking water quality monitoring and contamination detection in water distributed mains. Int J Eng Technol. 2017;7(1.1):47–51. https://doi.org/10.14419/ijet.v7i1.1.8921.
14. Kumar SS, Rao BVS, Prasad JR. Design and development of a water quality monitoring system by using IoT. Int J Emerg Trends Eng Res. 2020;8(9):6441–7. https://doi.org/10.30534/ijeter/2020/245892020.

15. Chi Q, Yan H, Zhang C, Pang Z, Xu LD. A reconfigurable smart sensor interface for industrial WSN in IOT environment. IEEE Trans Industr Inf. 2014;10(2):1417–25. https://doi.org/10.1109/tii.2014.2306798.
16. Parra L, Ahmad A, Sendra S, Lloret J, Lorenz P. Combination of machine learning and RGB sensors to quantify and classify water turbidity. Chemosensors. 2024;12(3):34. https://doi.org/10.3390/chemosensors12030034.
17. Lee J, Kwon J, Kim J, Koo J, Lee S. In-situ turbidity sensor system for residential water quality monitoring. Int J Nanotechnol. 2024;21(3):157–64. https://doi.org/10.1504/IJNT.2024.137414.
18. Mandal N, Mitra S, Bandyopadhyay D. Paper-sensors for point-of-care monitoring of drinking water quality. IEEE Sens J. 2019;19(18):7936–41. https://doi.org/10.1109/JSEN.2019.2919269.
19. McGarrity M, Zhao F. Graphene-based chemiresistor sensors for drinking water quality monitoring. Sensors. 2023;23(24):9828. https://doi.org/10.3390/s23249828.
20. Blokker M, Van Summeren J, Van Laarhoven K. Measuring drinking water temperature changes in a distribution network. 2024. Editorial Universitat Politècnica de València. https://doi.org/10.4995/WDSA-CCWI2022.2022.14116.
21. Ganesan P, Hameed T, Maruthakutti M. IOT based water flow meter using ThingSpeak. Int J Sci Res Arch. 2024;11(02):1956–62. https://doi.org/10.30574/ijsra.2024.11.2.0717.
22. Lambrou TP, Anastasiou CC, Panayiotou CG. A nephelometric turbidity system for monitoring residential drinking water quality. In: Sensor Networks Applications Experimentation and Logistics, New York, USA; Springer-Verlag. 2009; https://doi.org/10.1007/978-3-642-11870-8_4.
23. Kaur E, Oza A. Blockchain-based multi-organization taxonomy for smart cities. SN Appl Sci. 2020. https://doi.org/10.1007/s42452-020-2187-4.
24. Atlas Scientific. PT-1000 Temperature Probe Datasheet v. 2.1. https://atlas-scientific.com/probes/pt-1000-temperature-probe/. Accessed Oct. 2024.
25. Atlas Scientific. Revised Oct. 2021. EZO™ RTD Circuit Datasheet v. 3.6. https://atlas-scientific.com/embedded-solutions/ezo-rtd-temperature-circuit/. Accessed Oct. 2024.
26. Atlas Scientific. Consumer Grade ORP Probe Datasheet v. 2.6. https://atlas-scientific.com/probes/consumer-grade-orp-probe/. Accessed Oct. 2024.
27. Atlas Scientific. Gravity™ Analog ORP Sensor / Meter Datasheet v. 1.2. https://atlas-scientific.com/embedded-solutions/orp-sensor-meter/. Accessed Oct. 2024.
28. Atlas Scientific. Consumer Grade pH Probe Datasheet v. 2.7. https://atlas-scientific.com/probes/consumer-grade-ph-probe/. Accessed Oct. 2024.
29. Atlas Scientific. Revised Dec. 2020. Gravity™ Analog pH Sensor / Meter Datasheet v. 2.4. https://atlas-scientific.com/embedded-solutions/gravity-analog-ph-sensor-meter/. Accessed Oct. 2024.
30. Gravity: Water Flow Sensor (1/2") For Arduino datasheet. 2023. https://www.farnell.com/datasheets/3176135.pdf. Accessed Oct. 2024.
31. European Commision (DG ENV). 2009. Study on water efficiency standards. https://ec.europa.eu/programmes/erasmus-plus/project-result-content/f3abca0c-bd78-42d3-8c78-c42d9dfce5ac/AQUAVET_H-P3_Background%20information%20on%20this%20topic.pdf. Accessed: Oct, 2024.
32. Gravity: Analog Turbidity Sensor For Arduino datasheet. 2023. https://wiki.dfrobot.com/Turbidity_sensor_SKU__SEN0189. Accessed Oct. 2024.
33. Shenzhen Ai-Thinker Technology Co. NodeMCU-32 Specification v. 1.3. 2020. https://www.waveshare.com/wiki/NodeMCU-32S. Accessed Oct. 2024.
34. NEXTION. NX3224T024 Specifications. https://nextion.tech/datasheets/nx3224t024/#3. Accessed Oct. 2024.
35. DFRobot. Semi Flexible Solar Panel Specifications. https://core-electronics.com.au/semi-flexible-solar-panel-5v-2a.html. Accessed Oct. 2024.
36. DFRobot. Solar Power Manager 5V Specifications. https://www.dfrobot.com/product-1712.html. Accessed Oct. 2024.
37. Waveshare. SIM7070G NB-IoT / Cat-M / GPRS / GNSS HAT for Raspberry Pi. https://www.waveshare.com/sim7070g-cat-m-nb-iot-gprs-hat.htm. Accessed Oct. 2024.
38. Leaflet. Leaflet API reference. https://leafletjs.com/. Accessed Oct. 2024.
39. Thingiverse. Case for turbidity sensor in pipe. 2028. https://www.thingiverse.com/thing:3059233. Accessed Oct. 2024.
40. Publications Office of the European Union. Mobile broadband prices in Europe 2018: Final report and executive summary. 2018. ISBN 978–92–7–981308–5; https://doi.org/10.2759/137481.