© January 2026 | IJIRT | Volume 12 Issue 8 | ISSN: 2349-6002

AquaShield: IoT-Based Smart Water System

Shubhra Zope[1], Simantini Deo[2], Rutuja Waghchaure[3], Siddhi Jadhav[4]
Professor. Sarika Dhawade[5]
[1,2,3,4,5]*Department of Computer Engineering, JSPM's Rajarshi Shahu College of Engineering, Polytechnic, Tathawade*
doi.org/10.64643/IJIRTV12I8-190290-459

***Abstract*—Abstract—Rapid urbanization, population growth, and increasing stress on freshwater resources have made efficient water management a global necessity. In residential and institutional environments, conventional overhead water tank systems largely depend on manual supervision, often leading to water overflow, pump dry-run conditions, excessive power consumption, and undetected leakages. These issues result in significant water wastage, equipment damage, and increased operational costs.**

This paper presents AquaShield, an Internet of Things (IoT) based smart water tank monitoring and leak detection system designed to provide real-time monitoring, intelligent motor control, and early fault detection. The proposed system integrates multiple sensors to monitor critical parameters, including water level using an ultrasonic sensor, water quality using a Total Dissolved Solids (TDS) sensor, and leakage detection using a moisture sensor placed beneath the tank and pipelines. An ESP32 microcontroller acts as the central processing unit, handling sensor data acquisition, local processing, and wireless communication.

Sensor data is transmitted over Wi-Fi to a Java Spring Boot backend through RESTful APIs, where it is processed and stored in a relational database. A web-based dashboard developed using HTML and CSS provides real-time visualization of water level, water quality, leakage status, and motor operation. The system supports both automatic and manual motor control through the dashboard interface. Experimental evaluation demonstrates stable performance, accurate monitoring, and effective automation, making AquaShield a practical, low-cost, and scalable solution for smart water management.

***Index Terms*—ESP32 Microcontroller, IoT-Based Automation, Pump Automation, Smart Water Management, Water Quality Monitoring**

I. INTRODUCTION

Water is one of the most vital natural resources for human survival, economic development, and environmental sustainability. Despite its importance, inefficient water storage and management practices continue to cause significant wastage, particularly in residential, commercial, and institutional environments. Overhead water tanks are commonly used for water storage; however, their operation is largely dependent on manual monitoring and control. Such conventional systems frequently result in water overflow, pump dry-run conditions, excessive electricity consumption, and undetected water leakages.

Water overflow not only leads to wastage of a limited natural resource but also causes structural damage to buildings, damp walls, and increased maintenance costs. Pump dry-running occurs when a water pump operates without adequate water supply, leading to overheating, mechanical damage, reduced pump lifespan, and unnecessary energy consumption. Another major issue is water leakage from tanks, joints, or pipelines. Even small leakages, if unnoticed, can result in continuous water loss, higher utility bills, and potential safety hazards.

Recent advancements in the Internet of Things (IoT) have enabled the integration of sensors, microcontrollers, communication modules, and control mechanisms into physical systems. IoT-based solutions allow real-time monitoring, automated decision-making, and remote access to system data. These capabilities make IoT a promising approach for modern water management. However, many existing smart water tank systems primarily focus on water level monitoring and pump automation, often ignoring water quality monitoring and early leakage detection. To overcome these limitations, this paper proposes AquaShield, an IoT-based smart water tank monitoring and leak detection system that integrates water level monitoring, water quality assessment using TDS sensing, moisture-based leakage detection, and intelligent pump control using an ESP32 microcontroller and a web-based dashboard.

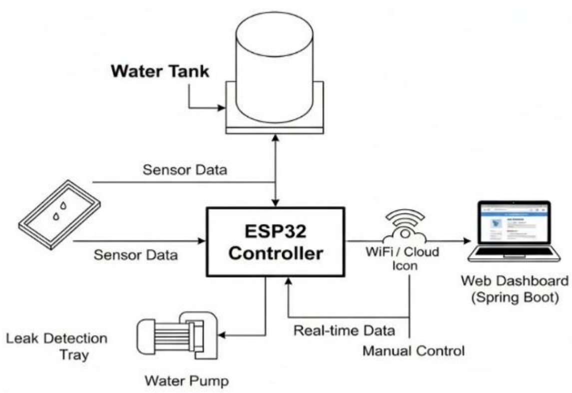

Fig. 1. Conceptual Diagram of AquaShield Smart Water Tank System

II. LITERATURE SURVEY

The literature survey examines existing research and technological developments related to water tank monitoring, automation, water quality assessment, and leakage detection. The objective is to analyze current approaches, identify their shortcomings, and highlight the need for an integrated system such as AquaShield.

A. Traditional and Automated Water Tank Monitoring Systems

Traditional water tank monitoring systems rely heavily on manual supervision, where users physically check water levels and manually operate water pumps. Although simple and inexpensive, such methods are highly inefficient and prone to human error. Overflow and pump dry-run conditions are common due to delayed user response.

Basic automation using mechanical float valves and level switches has been adopted in some systems. While these solutions reduce manual effort, they lack intelligence, flexibility, and real-time feedback. Additionally, they do not provide any remote monitoring or data logging capabilities.

With the advancement of embedded systems, microcontroller-based automation using platforms such as Arduino UNO and 8051 microcontrollers gained popularity. These systems typically use ultrasonic sensors or float sensors to measure water levels and automatically control pumps based on predefined thresholds. While effective in preventing overflow, these systems are usually limited to local operation and do not support remote access, data storage, or system scalability.

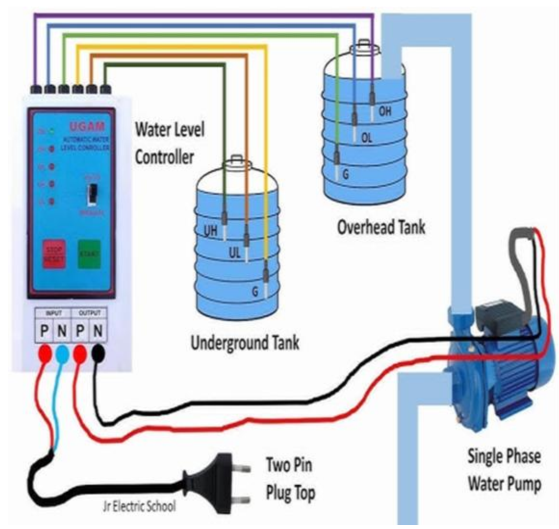

Fig. 2. Conventional Automatic Water Level Control System Using Overhead and Underground Tanks

B.IoT-Based Water Monitoring and Control Systems

IoT-based water monitoring systems enhance traditional automation by enabling real-time data transmission and remote access. Wi-Fi-enabled microcontrollers transmit sensor data to cloud platforms or backend servers, allowing users to monitor water levels and control pumps through web or mobile applications.

Although these systems improve convenience and efficiency, many implementations depend on proprietary cloud services, increasing operational costs and limiting customization. Furthermore, most IoT-based systems focus primarily on water level monitoring and pump control, often neglecting important aspects such as water quality monitoring and early leakage detection.

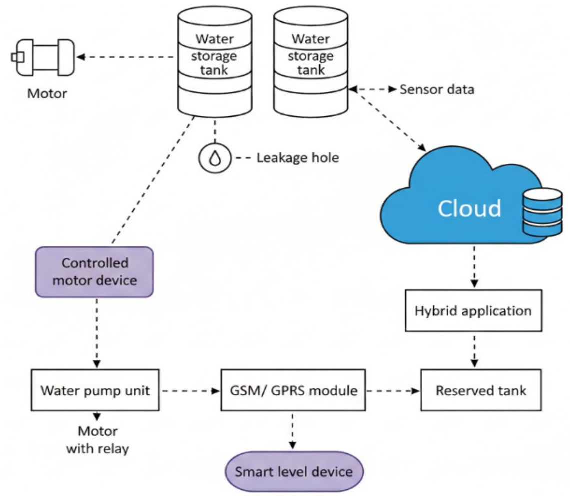

Fig. 3. IoT-Based Smart Water Tank Monitoring and Pump Control System

C. Water Quality Monitoring Using TDS Sensors

Water quality monitoring is essential for ensuring safeand healthy water usage. Total Dissolved Solids (TDS) sensors are commonly used to estimate water purity by measuring dissolved salts and minerals in parts per million (ppm). While TDS monitoring is widely used in water purification systems, it is rarely integrated into domestic water tank automation systems, limiting its usefulness for everyday monitoring.

D. Leakage Detection Approaches for Residential Water Systems

Leakage detection is often overlooked in residential water systems. Conventional methods such as flow-based, pressure-based, or acoustic sensing techniques are complex and costly. Recent research highlights moisture-based sensors as a simple, low-cost, and effective solution for detecting leakages in small-scale applications.

E. Research Gap and Motivation

Most existing systems address individual water management problems in isolation. There is a lack of an integrated, low-cost, and beginner- friendly solution that combines water level monitoring, water quality assessment, leakage detection, and real-time visualization. This research gap motivates the development of the AquaShield system.

III. PROPOSED SYSTEM

The proposed AquaShield system is an integrated IoT-based solution designed for smart water tank monitoring, water quality assessment, leakage detection, and intelligent pump control. The system follows a modular and layered architecture to ensure scalability, reliability, and ease of maintenance.

A. System Overview

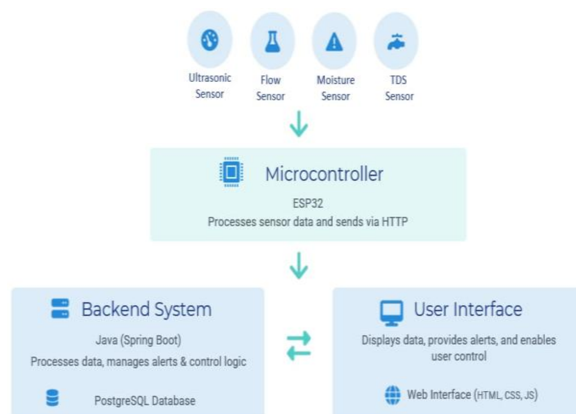

Fig. 4. Overall System Overview of AquaSheild

The hardware layer consists of an ESP32 microcontroller connected to an ultrasonic sensor, TDS sensor, moisture sensor, and a relay module controlling the water pump. The ESP32 processes sensor data locally and transmits it to the backend server via Wi-Fi using RESTful APIs. The backend processes and stores the data, while the frontend dashboard displays real-time system status.

B. System Architecture

The system architecture illustrates the interaction between hardware components, communication protocols, backend services, and the user interface. The ESP32 acts as the central processing unit, the Spring Boot backend handles data processing and storage, and the web dashboard provides visualization and control.

C. Functional Modules and Objectives

The AquaShield system consists of the following functional modules:

- Water level monitoring module
- Water quality monitoring module
- Leakage detection module

- Pump control module
- Communication and data management module

The primary objective is to minimize water wastage, prevent pump damage, detect leakages early, and provide real-time monitoring and control.

IV. HARDWARE DESIGN

The hardware design of AquaShield emphasizes affordability, reliability, safety, and ease of implementation. All components operate at low voltage levels and are suitable for residential use.

A. Hardware Components Used

Table 1. Hardware Components of AquaShield

Sr. No.	Component	Description
1	ESP32 DevKit V1	Central microcontroller with built-in WiFi
2	Ultrasonic Sensor (HC-SR04)	Measures water level in the tank
3	TDS Sensor with Probe	Measures water quality in ppm
4	Moisture Sensor (FC-37 / YL-83)	Detects water leakage
5	Relay Module (5V)	Controls pump operation
6	Mini Submersible Pump (5V)	Pumps water into the tank
8	Breadboard & Jumper Wires	Prototyping and connections
9	5V 2A Power Adapter	External power supply

B. Water Level Measurement Using Ultrasonic Sensor

The ultrasonic sensor calculates the distance between the sensor and the water surface. This non-contact method ensures durability, accuracy, and minimal maintenance.

C. Water Quality Monitoring Using TDS Sensor

The TDS sensor measures dissolved solids in water. The ESP32 converts analog readings into approximate ppm values, providing an estimation of water purity.

D. Leakage Detection and Pump Control

Leakage detection is implemented using a moisture sensor placed below the tank or pipelines. When leakage is detected, the ESP32 updates the backend and reflects the alert on the dashboard. Pump control is achieved using a relay module supporting both automatic and manual operation.

E. Physical Placement of Components

The physical placement of components in the AquaShield system is designed to ensure accurate sensing and reliable operation. The ultrasonic sensor is mounted at the top of the water tank to measure the distance to the water surface and calculate the water level. The TDS sensor probe is submerged inside the tank to monitor water quality continuously. The moisture sensor is placed in a leakage detection tray positioned beneath the tank or pipelines to detect leaked water. The submersible pump is installed in the water source and connected to the tank through piping, with its operation controlled using a relay module.

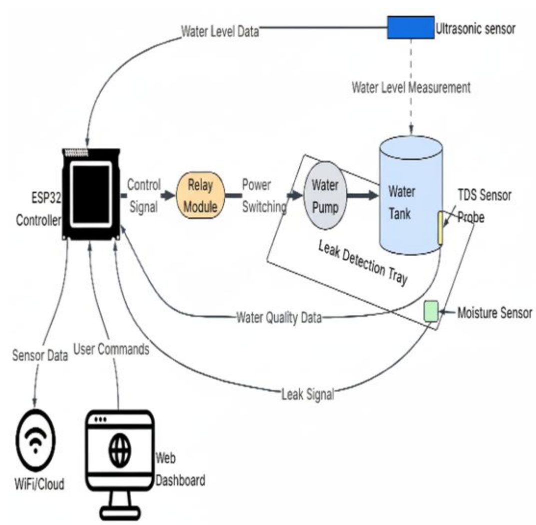

Fig. 5. Physical Arrangement of Components

V. SOFTWARE DESIGN

The software design of the AquaShield system plays a crucial role in enabling real-time monitoring, data processing, intelligent control, and user interaction.

The software architecture is divided into three main layers: ESP32 firmware, backend application, and frontend dashboard. This layered design ensures modularity, scalability, and ease of maintenance. Each layer performs a specific function and communicates with other layers using standardized interfaces, allowing smooth data flow and reliable system operation.

A. Overall Software Architecture

The ESP32 firmware handles sensor data acquisition and actuator control. It collects data from the ultrasonic, TDS, and moisture sensors and controls the pump through a relay module. The backend processes and stores sensor data while managing business logic and system states. The frontend provides real-time visualization of water level, water quality, leakage status, and pump operation. Communication between these layers is achieved using RESTful APIs over HTTP, enabling reliable and real-time data exchange.

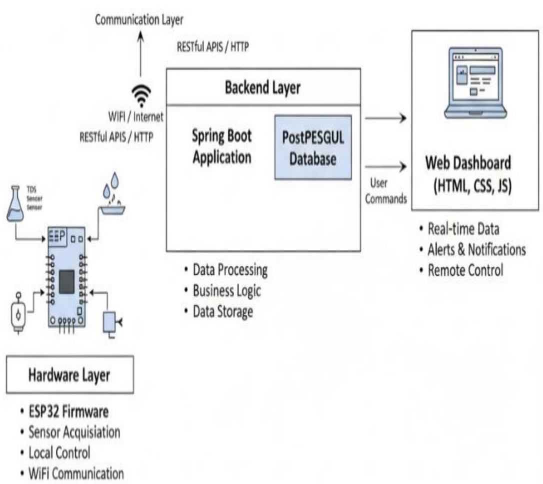

Fig. 6. Overall System Architecture

B. ESP32 Firmware Design

The ESP32 firmware is developed using the Arduino framework, which offers simplicity and compatibility with the required sensors and libraries. The firmware initializes all connected sensors and the relay module and establishes WiFi connectivity for backend communication. It periodically acquires sensor data and performs local decision-making for pump control and leakage detection. The ultrasonic sensor calculates water level, the TDS sensor provides approximate ppm values, and the moisture sensor detects leakage. When leakage is detected, the ESP32 updates the backend server, and pump operation is controlled automatically or manually through backend commands.

C. Backend, Database and Frontend Design

The backend is implemented using Java Spring Boot and follows a layered architecture consisting of controller, service, repository, and model layers. Sensor data is validated, processed, and stored in a PostgreSQL database. System states such as motor status and leakage status are maintained to ensure consistency. The frontend dashboard, developed using HTML and CSS, displays real-time sensor values, leakage alerts, and pump status, and provides manual pump control options to the user.

D. Communication Protocol

Communication between the ESP32, backend, and frontend is handled using HTTP-based RESTful APIs. The ESP32 sends sensor data to the backend over WiFi, and the frontend retrieves updated information from the backend. Control commands are transmitted from the frontend to the backend and then forwarded to the ESP32. This communication mechanism enables reliable two-way data exchange and real-time system updates.

VI. WORKING METHODOLOGY

The working methodology of the AquaShield system explains the sequential flow of operations starting from sensor data acquisition to system response and user interaction. The system operates continuously in real time, ensuring efficient monitoring, intelligent decision-making, and timely alerts. The overall workflow is illustrated in the working methodology diagram and is described step by step below.

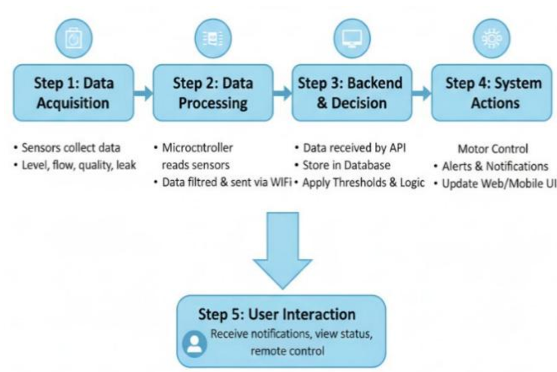

Fig. 7. Working Methodology

Step 1: Data Collection
The ESP32 microcontroller continuously collects real-time data from all connected sensors. The ultrasonic sensor measures the distance to the water surface to determine the current water level in the tank. The TDS sensor measures dissolved impurities to estimate water quality in parts per million. The moisture sensor detects the presence of leaked water in the leakage detection tray placed beneath the tank or pipeline. This step ensures continuous monitoring of critical system parameters.

Step 2: Data Processing and Transmission
The ESP32 processes the raw sensor data locally. Ultrasonic readings are converted into water level values, and the analog output from the TDS sensor is converted into approximate ppm values. The moisture sensor provides a digital leakage indication. After processing, the ESP32 transmits the formatted sensor data to the backend server using HTTP- based RESTful APIs over WiFi, ensuring timely system updates.

Step 3: Backend Processing
The Java Spring Boot backend receives sensor data through REST APIs and performs validation and processing. The backend stores readings in a PostgreSQL database, monitors water level and leakage thresholds, and manages pump control logic. It maintains the latest system state, including motor and leakage status, ensuring consistency across the application.

Step 4: System Action
Based on predefined control logic, the system automatically switches the pump ON or OFF using a relay module to prevent overflow and dry-run conditions. Leakage detection updates system status and displays alerts on the web-based dashboard. All system actions and sensor updates are recorded in the database for monitoring purposes.

Step 5: User Interaction and Continuous Monitoring Cycle
Users monitor system status through a web-based dashboard displaying water level, TDS values, leakage alerts, and pump state. Manual pump control commands are sent to the backend and forwarded to the ESP32. After completing all steps, the system returns to data collection, ensuring continuous monitoring and efficient water management.

VII. RESULTS AND OBSERVATIONS

The AquaShield system was implemented and tested under controlled conditions to evaluate its accuracy, responsiveness, reliability, and overall performance. The prototype was tested by varying water levels, simulating leakage scenarios, and observing system behavior through the web-based dashboard.

A. Experimental Setup

The system was tested using a prototype water tank setup consisting of an overhead container, a submersible pump, and the integrated sensor arrangement. The ESP32 was connected to a stable WiFi network to enable uninterrupted communication with the Spring Boot backend. Sensor readings were observed in real time through the web-based dashboard. Leakage conditions were simulated by allowing small amounts of water to collect in the leakage detection tray.

B. Observed Performance Metrics

Table 2. Performance Evaluation of AquaShield

Parameter	Observed Result
Water level measurement accuracy	±2 cm
Leakage detection response	Detected within 2–3 water drops
TDS measurement	Approximate ppm values (calibration dependent)
Pump response time	Less than 1 second
Data transmission	Reliable HTTP communication
Dashboard update delay	Near real-time

C Overall System Behaviour:
The system demonstrated stable and continuous operation throughout testing. Automatic pump control successfully prevented overflow and dry-run conditions, while manual control commands were executed promptly through the dashboard. The

integration of monitoring and control functionalities proved effective for domestic applications.

D. Key Observations
- The system successfully integrated multiple monitoring functionalities into a single platform
- Leakage detection provided quick alerts with minimal water loss
- Pump automation reduced the need for manual intervention
- The web dashboard enabled clear and real-time system visualization

VIII. ADVANTAGES

- Prevents water overflow and pump dry-run, reducing water wastage
- Enables early detection of water leakage using moisture sensors
- Provides real-time monitoring of water level, quality, and system status
- Cost-effective implementation using ESP32 and low-cost sensors
- Supports remote monitoring and manual pump control through a web dashboard
- Modular and scalable architecture for future feature expansion

IX. APPLICATIONS
- Residential overhead water tanks
- Apartment complexes and housing societies
- Educational institutions and laboratories
- Small commercial buildings
- Smart homes and smart buildings
- Water conservation and sustainability initiatives

X. FUTURE SCOPE

- Integration of a mobile application for enhanced user accessibility
- Cloud-based data analytics for long-term water usage analysis
- Support for multiple water tanks within a single system
- Advanced water quality monitoring using additional sensors
- Predictive maintenance and anomaly detection using machine learning
- Integration with renewable energy sources such as solar power
- SMS or push notification-based alert mechanism

XI. CONCLUSION

This paper presented AquaShield, an IoT-based smart water tank monitoring and leak detection system that integrates water level sensing, TDS- based water quality assessment, leakage detection, and intelligent pump control. The system employs an ESP32 microcontroller, a Java Spring Boot backend, and a web-based dashboard to enable real-time monitoring, reliable communication, and user-friendly control. The experimental evaluation demonstrates stable operation, accurate monitoring, effective leakage detection, and responsive pump automation, making the system suitable for residential and academic applications.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the Department of Computer Engineering for providing valuable guidance, continuous support, and constructive feedback throughout the development of the "AquaShield: IoT-Based Smart Water Tank Monitoring and Leak Detection System." The academic assistance and mentorship received during this work played a significant role in shaping the direction and successful completion of this research.

REFERENCES

[1] IEEE, "IoT-Based Water Management Systems: Smart Monitoring and Control Approaches," IEEE Journals on Internet of Things, vol. 5, no. 3, pp. xx–xx, 2021.
[2] Spring Framework Team, Spring Boot Official Documentation, Version 3.0, Spring.io, 2022.
[3] K. Sharma, A. Patel, and S. Singh, "TDS-based water quality monitoring using IoT sensors," International Journal of Advanced Water Research, vol. 12, no. 4, pp. 255–265, 2021.
[4] R. Gupta and P. Verma, "Leakage detection techniques in smart water distribution systems,"

in Proc. IEEE Int. Conf. on Smart Infrastructure
and Utilities, 2019, pp. xx–xx.