ICIT Journal (Innovative Creative and Information Technology) | Print ISSN: 2356-5195 | Online ISSN: 2654-8704 | Vol. xx No. x, August 2026 | DOI Prefix: https://doi.org/10.33050/icit.v12i2.4474

Design of A Water Level Sensor System for Water Tanks in Internet of Things-Based Companies

Yunita Wulansari[*1], Mochamad Al Ghoniyu Oka[2], Farell Havids Abdillah[3]
[1,2]University of Raharja, Tangerang, Indonesia
[3]PLN Institute of Technology, Jakarta, Indonesia
*Corresponding author: yunita@raharja.info

Article Information:
Received: 06 July 2026
Revised: 24 July 2026
Accepted: 03 August 2026
DOI: 10.33050/icit.v12i2.4474

ABSTRACT
Water reservoir management in company facilities requires water level monitoring that can provide fast information and help prevent empty or overflow conditions. This study designs and simulates an Internet of Things (IoT)-based water reservoir monitoring and filling system using NodeMCU ESP32, HC-SR04 ultrasonic sensor, LCD, LED, buzzer, relay, and Blynk. Different from studies that report the implementation of physical devices, this study is limited to Wokwi simulations so that the results cannot be considered as field validation. The experimental design uses five reference distance points, namely 200, 150, 100, 50, and 2 cm, with 200 cm as the empty reference condition and 2 cm as the full reference condition. Evaluation is carried out using absolute difference, mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE), as well as Black Box functional testing of connections, sensor readings, indicators, pump control, and data transmission to Blynk. The simulation results show an MAE of 0.056 cm, an RMSE of 0.066 cm, and a MAPE of 0.337%. The system also demonstrates monitoring and control functions according to the simulation scenario. The contribution of this research is not in the novelty of the components, but rather in the integration of the monitoring-control flow tailored to the company's reservoir monitoring needs, the definition of measurement reference points, and quantitative evaluation during the simulation stage. Subsequent validation needs to be carried out using a physical prototype, repeated measurements, latency testing, packet loss, and operational resilience.

Keywords— Internet of Things, ESP32, HC-SR04, Blynk, water level monitoring, Wokwi

1. INTRODUCTION

Water tanks are used to support various company facility needs, such as sanitation, toilets, watering, and other utility needs. When water levels are not monitored continuously, empty or overflowing conditions can be detected late. Manual monitoring also relies on human inspection, potentially causing information delays and reading errors. IoT provides an approach to connecting sensors, microcontrollers, networks, and monitoring applications so that data can be monitored remotely and the filling process can be controlled automatically [1]–[5]. ESP32 is suitable for this scenario because it provides Wi-Fi connectivity and peripheral interfaces that support IoT applications [6], while Blynk provides a dashboard for remote device monitoring and control [7].

However, the use of ESP32, ultrasonic sensors, and IoT platforms for water monitoring is not a new approach. Previous studies have demonstrated the implementation of automatic monitoring and filling using ESP32/ESP8266, ultrasonic sensors, LCDs, LEDs, buzzers, relays, and IoT platforms. While more recent research has shown that field validation can include metrics such as measurement error, latency, and packet loss [8]. Thus, the research gap in this study is not the absence of similar systems, but the need for a design that explicitly maps the operational needs of the company's reservoirs into the sensor-control-monitoring flow, while documenting the limits of simulation implementation and quantitatively evaluating measurement accuracy.

Based on this gap, this study aims to design and simulate a tank level monitoring and filling system using ESP32, HC-SR04, relay, local indicator, and Blynk. The research novelty is stated in limited terms as an engineering and methodological contribution, namely: (1) integration of local and remote monitoring with filling control in one system flow; (2) determination of 200 cm and 2 cm reference points for mapping empty-full conditions; (3) quantitative evaluation using MAE, RMSE, and MAPE on simulation data; and (4) Black Box functional testing. This study does not claim novelty in the use of ESP32, HC-SR04, or Blynk individually.

2. LITERATURE REVIEW

Research on monitoring the height and filling of reservoirs based on the Internet of Things (IoT) has been widely developed with various combinations of microcontrollers, sensors, monitoring platforms, and actuators. Ma'ruf et al. [1] and Eriyanto et al. [2] developed an IoT-based automatic water monitoring and filling system, while Ramiz et al. [3] implemented reservoir water level monitoring using the Blynk application. Wijoseno et al. [4] developed an ESP32-based automatic water filling system with MQTT, WhatsApp notifications, and data logging, while Azhar et al. [5] used ESP8266 for capacity monitoring and automatic water filling. The use of ESP32 for reservoir monitoring has also been implemented by Fernandes et al. [6], while Estu et al. [7] and Diono et al. [8] demonstrated the application of ESP32 and ultrasonic sensors for water level monitoring.

In terms of development methods, Khairullah et al. [9] applied the Waterfall method with systematic and sequential stages. ESP32 has also been used for IoT-based monitoring and control systems in the research of Basir et al. [10]. Meanwhile, Jaya et al. [11] and Aziz et al. [12] demonstrated the use of ultrasonic sensors for measuring the height or distance of water surfaces. Alamsyah et al. [13] developed a real-time IoT-based water level monitoring system, while Faturochman et al. [14] used ultrasonic sensors in an early warning system for water conditions. The use of actuators for automatic control was also demonstrated by Sinaga et al. [15], while Blynk has been implemented as an IoT-based monitoring medium in the research of Syamsul et al. [16].

Based on previous research [1]–[16], the use of ESP32, HC-SR04, and Blynk is not new at the component level because it has been widely used in water monitoring and IoT systems. The research gap in this study lies in the integration of local monitoring, remote monitoring, and filling control in one system flow that is tailored to the needs of the company's reservoir. This study also sets the reference points for empty and full conditions at 200 cm and 2 cm and conducts quantitative evaluations using MAE, RMSE, and MAPE at the Wokwi simulation stage. Thus, the contribution of this research is more directed at system integration, determination of measurement references, and quantitative evaluation, rather than on the novelty of using individual components.

3. METHODOLOGY

3.1 Research Design and Scope

The study used a simulation-based experimental design to evaluate the relationship between the sensor's reference distance and the system's read distance, while also examining the control and IoT communication functions. Wokwi was used as a hardware simulation environment so that the ESP32, HC-SR04, LCD, LED, buzzer, and relay could be tested without a physical prototype. Because there were no direct measurements on a real reservoir, this study does not claim field validation or device reliability under real-world conditions.

3.2 Data collection

System requirements data was obtained through interviews with the building engineering team to understand the monitoring requirements for the reservoir and filling flow. Experimental data was derived from the distance values set in the Wokwi simulation and the values received/displayed by the system. The five measurement points available in the initial study were retained to avoid adding data to experiments that were not actually conducted.

3.3 System Development Methods

The Waterfall model is used as an engineering development framework, not as the sole scientific method. Its stages include requirements analysis, design, simulation implementation, testing, and maintenance [9]. Research validity is strengthened through experimental design, calibration procedures, error metrics, and Black Box functional testing. With this separation, Waterfall explains how the system is built, while experimental design explains how the system's performance is evaluated.

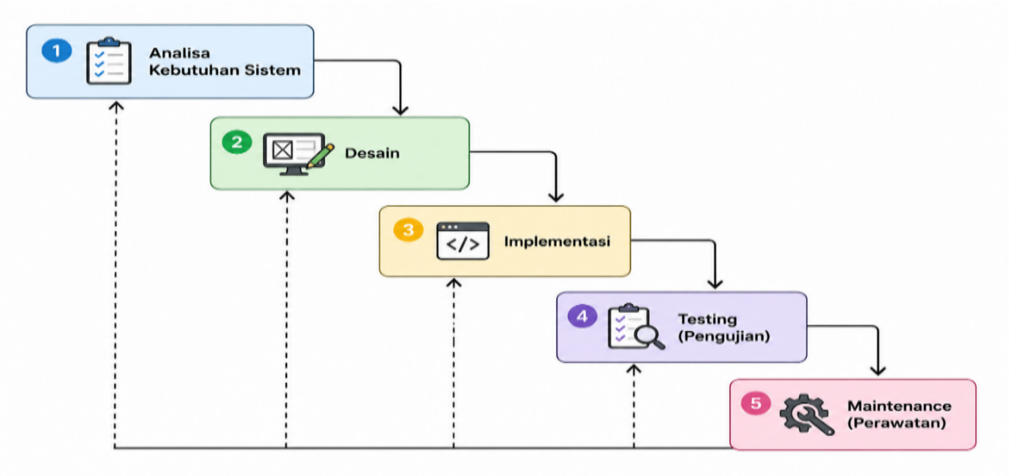

Figure 1. Waterfall Model Stages

3.4 System Development Methods

The system components consist of ESP32 as a network processor and connector, HC-SR04 as a proximity sensor, LCD as a local display, LED and buzzer as indicators, relay as a load controller, and Blynk as a monitoring medium. ESP32 is used because it supports Wi-Fi connectivity for IoT applications, while Blynk is used as a monitoring and control platform for devices through a dashboard. In the simulation, a yellow LED is used as a visual representation of the solenoid valve state; this component should not be interpreted as a physical solenoid valve that has been tested.

Table 1. Components and Functions

Component	Function
NodeMCU ESP32	Processes sensor readings and provides Wi-Fi connectivity [10].
HC-SR04	Measuring the distance of the sensor to the water surface [11].
LCD	Displays water status locally [12].
LED	System condition indicator [13].
Buzzer	Voice warning on specified conditions [14].
Relay	Controlling the pump based on system logic [15].
Blynk	Monitoring and control via IoT dashboard [16].

ICIT Journal (Innovative Creative and Information Technology) | Print ISSN: 2356-5195 | Online ISSN: 2654-8704 | Vol. xx No. x, August 2026 | DOI Prefix: https://doi.org/10.33050/icit.v12i2.4474

3.5 Calibration Procedure System Development Method and Operational Limits

Calibration in the simulation phase was performed by setting two reference points: D_empty = 200 cm as the sensor distance to the water surface when empty and D_full = 2 cm as the full condition. The water level percentage was calculated using the equation: Level(%) = ((D_empty − D_measured)/(D_empty − D_full)) × 100%. Thus, 200 cm was mapped to 0% and 2 cm to 100%. The values of 150, 100, and 50 cm were used as accuracy test points, not as proof that the system has three classes Low–Normal–High. Based on the actual available data, the validated limits are the empty reference conditions at 200 cm and full at 2 cm; the normal condition thresholds need to be established through additional testing on a physical prototype before use in the field. The distance measurement characteristics of the HC-SR04 have also been discussed in previous research. In actual implementation, the sensor distance should be calibrated against the actual tank dimensions. For distance-based systems, the water level value is not the same as the sensor's distance from the water surface; the smaller the measured distance, the higher the water level. Therefore, calibration must use the effective tank height and sensor installation location, not just simulated values.

3.6 Evaluation Metrics

Evaluation of measurement accuracy was performed using four metrics:

1. Absolute Error

$$e_i = |x_i - y_i|$$

2. Mean Absolute Error (MAE)

$$MAE = MAE = (1/n)\ \Sigma_{i=1}^{n} |x_i - y_i|$$

With test data:

$$MAE = (0.08 + 0.08 + 0.09 + 0.00 + 0.03) / 5$$
$$MAE = 0.056 \text{ cm}$$

3. Root Mean Square Error (RMSE)

$$RMSE = \sqrt{[(1/n)\ \Sigma_{i=1}^{n} (x_i - y_i)^2]}$$

Test data substitution:

$$RMSE = \sqrt{[((0.08)^2 + (0.08)^2 + (0.09)^2 + (0.00)^2 + (0.03)^2) / 5]}$$
$$RMSE \approx 0.066 \text{ cm}$$

4. Mean Absolute Percentage Error (MAPE)

$$MAPE = (100\%/n)\ \Sigma_{i=1}^{n} |(x_i - y_i) / x_i|$$

Data substitution:

$$MAPE = (100\%/5)\ [(0.08/200) + (0.08/150) + (0.09/100) + (0.00/50) + (0.03/2)]$$
$$MAPE \approx 0.337\%$$

Description: x_i = reference value; y_i = reading result value; n = number of observations.

3.7 Black Box Testing

Black box testing examines a system's output based on given inputs or conditions without assessing the program's internal structure. A PASS result in the following table means the function appears to be working according to the scenario in the Wokwi simulation; this status is not evidence of physical device performance.

Table 2. Black Box Testing

No.	Test Scenario	Input/Condition	Expected Output	Results
1	Booting ESP32	The system is turned on	ESP32 performs initialization	PASS
2	Wi-Fi Connection	Network available	ESP32 connected and continued the process	PASS
3	Empty/Low condition	Distance 200 cm	Green LED/yellow indicator on, pump ON, Low Water LCD	PASS
4	Full/High condition	Distance 2 cm	Red LED on, pump OFF, LCD Water Full	PASS
5	Blynk Monitoring	Sensor data sent	Distance and percentage values are displayed on the dashboard.	PASS
6	Buzzer	Warning conditions according to the program	Buzzer follows program logic	PASS
7	Relay	Low/High Condition	The relay changes according to the pump logic.	PASS

4. RESULTS AND DISCUSSION

4.1 System Flowchart

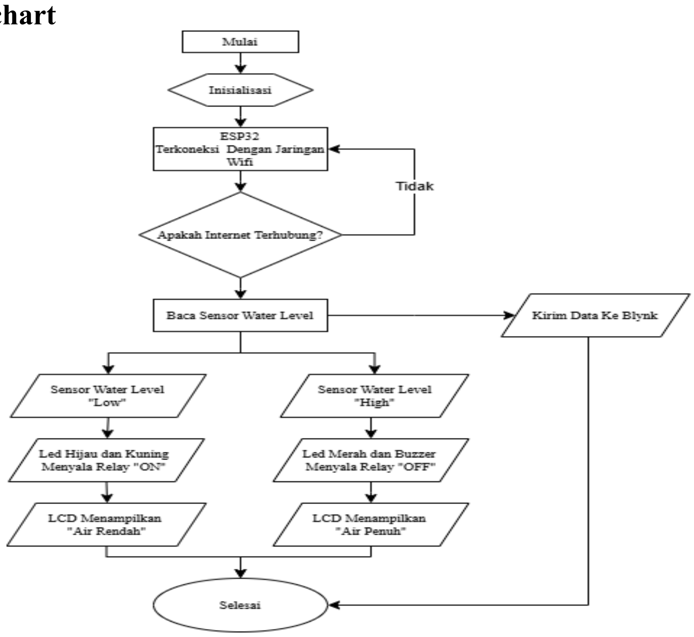

Figure 2. System Monitoring Flowchart

The system begins by initializing the ESP32 and supporting components, then attempting to connect to Wi-Fi. Once a connection is established, the sensor reads the water level and sends the data to Blynk. At 200 cm, the system treats the reservoir as empty/Low and activates the fill control. At 2 cm, the system treats the reservoir as full/High and stops the pump. In a real-world implementation, this cycle must repeat itself at the specified reading interval.

4.2 *S*ystem Design

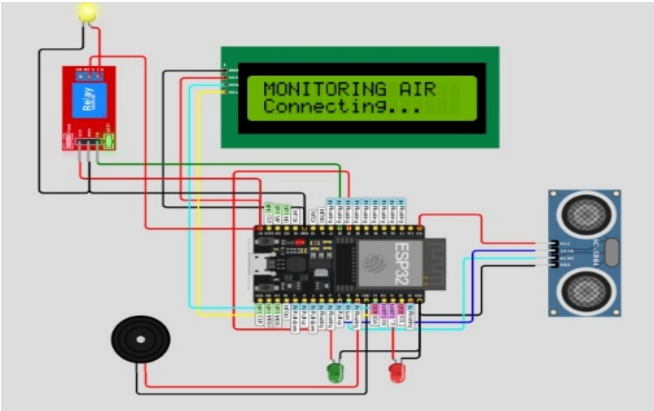

Figure 3. Initial conditions of the simulation: ESP32 boots and the LCD displays the connection status.

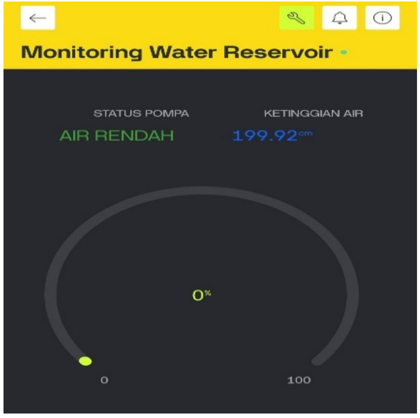

Figure 4. Blynk view in initial/low level condition

Figure 3 shows the initial state when the ESP32 boots and attempts to establish a connection. The yellow LED in the simulation serves as a visual indicator to represent the solenoid valve state, not as a physical solenoid valve. Therefore, referring to the yellow LED as a “solenoid valve” needs to be clearly distinguished: the LED only represents the actuator status on the Wokwi, whereas in a real device the LED can be replaced by a series of relays and a corresponding solenoid valve. Figure 4 shows the Blynk dashboard before monitoring data is received normally.

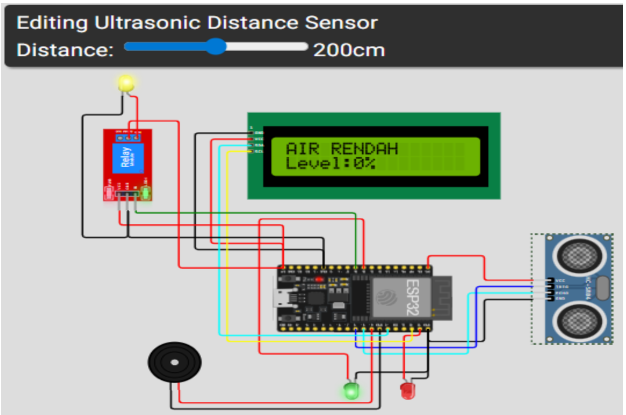

Figure 5. Simulation of distance conditions of 200

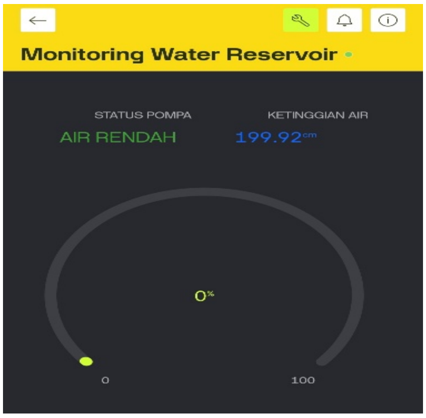

Figure 6. Blynk Dashboard at 199.92 cm and 0% level

In Figure 5, a reference distance of 200 cm is used to represent the empty/Low condition. The system activates a local indicator and relay to activate the filling process according to the program logic. The Blynk reading of 199.92 cm results in an absolute difference of 0.08 cm from the reference value. The 0% level percentage is the result of mapping the 200 cm and 2 cm reference points, not a direct measurement of water volume.

ICIT Journal (Innovative Creative and Information Technology) | Print ISSN: 2356-5195 | Online ISSN: 2654-8704 | Vol. xx No. x, August 2026 | DOI Prefix: https://doi.org/10.33050/icit.v12i2.4474

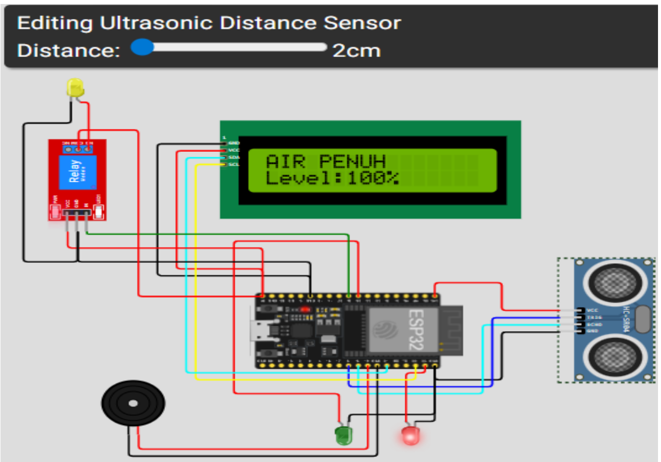

Figure 7. Simulation of 2 cm distance condition (full reference/100%)

Figure 8. Blynk Dashboard at 1.97 cm distance and 100% level

In Figure 7, 2 cm is used as the full reference condition. The system turns off the relay/pump and activates the full condition indicator as programmed. Blynk receives a reading of 1.97 cm, resulting in an absolute difference of 0.03 cm. A value of 100% indicates that the reading is at the full reference point established in the simulation.

Table 3. Simulation Accuracy Test Results

No.	Reference (cm)	Reading (cm)	Absolute Error (cm)	Error (%)	Mapped Level (%)
1	200.00	199.92	0.08	0.040	0.04
2	150.00	149.92	0.08	0.053	25.29
3	100.00	99.91	0.09	0.090	50.55
4	50.00	50.00	0.00	0.000	75.76
5	2.00	1.97	0.03	1,500	100.02

Five test points yielded absolute errors of 0.08; 0.08; 0.09; 0.00; and 0.03 cm. These data yielded MAE of 0.056 cm, RMSE of 0.066 cm, and MAPE of 0.337%. The largest percentage error occurred at the 2 cm point, which was 1.50%, because the denominator of the MAPE was relatively small. This value indicates that the simulation readings were very close to the reference value. However, these results should not be directly generalized as the accuracy of the HC-SR04 in the field because Wokwi does not represent all environmental factors, such as humidity, condensation, sensor position, water surface turbulence, reservoir shape, and reflection interference.

Table 3 also shows that five measurement points are not sufficient to conclude statistical stability. Repeating measurements at each point, for example 10–30 times per level, would allow for analysis of standard deviations, confidence intervals, error distributions, and repeatability. Because such repeat data were not collected in the original study, this revision does not add unavailable experimental numbers.

4.3 Discussion and Comparison with Previous Research

Simulation results show that the ESP32–HC-SR04–Blynk architecture can perform the basic monitoring and control functions as reported in various previous studies. Therefore, the main value of the system is not in the selection of new components, but rather in mapping the operational requirements of the reservoir into a single flow: the sensor reads the distance, the ESP32 processes and transmits the data, Blynk provides remote monitoring, and the relay represents the pump control. The use of LCD, LED, and buzzer provides local feedback to complement the monitoring via the network.

Compared to studies that only present monitoring, this design incorporates a charging control path. Compared to studies that have used a combination of ESP32, ultrasonic sensors, and IoT platforms, this study's contribution is more precisely positioned in documenting the company's reservoir requirements, defining empty-full reference points, and evaluating errors during the simulation phase. Other research has shown that similar systems can be evaluated using more comprehensive metrics, including measurement error, latency, and packet loss. Therefore, the results of this study serve as a baseline for simulation, while field validation will need to demonstrate whether these performance levels remain achievable on physical devices.

Another significant limitation is the use of the HC-SR04 and simulation components to represent the reservoir environment. In a real-world application, sensors must be placed with humidity and potential condensation in mind. Furthermore, the system needs a fail-safe mechanism to prevent sensor or network failures from causing the pump to run continuously. Blynk itself provides device monitoring and control capabilities from the dashboard, but communication quality still depends on the network and device configuration.

4.4 Research Limitations and Further Validation Plans

- Current research implementation is limited to Wokwi simulations; no physical prototype has yet been installed in the company's reservoirs.
- The accuracy data only consists of five measurement points and does not have 10–30 repetitions at each point.
- End-to-end latency measurements from the sensor to Blynk, packet loss, uptime, and reliability in long-term operation have not been performed.
- Calibration has not been carried out on the height and shape of the actual tank, water surface conditions, humidity, and sensor installation position.
- The yellow LED on the Wokwi is only a visual representation of the solenoid valve state; physical actuator testing has not been performed.
- The Low/High thresholds validated in this study only cover the 200 cm and 2 cm reference points. Normal and hysteresis thresholds to prevent frequent relay ON/OFFs need to be tested on a prototype.

For further research, the physical prototype should be tested at multiple water levels with a minimum of 10 repetitions per point, compared to manual reference measurements, and then calculated MAE, RMSE, MAPE, standard deviation, and confidence intervals. Communication testing should record timestamps of sensor readings and reception on a dashboard to obtain latency, as well as calculate packet loss and uptime. Endurance testing over several hours or days is also needed to assess the reliability of the sensors, ESP32, relays, and IoT connections.

5. CONCLUSION

This research successfully designed and simulated an IoT-based tank level monitoring and filling system using ESP32, HC-SR04, local indicators, relays, and Blynk. Five test points at Wokwi produced MAE of 0.056 cm, RMSE of 0.066 cm, and MAPE of 0.337%, while Black Box testing showed the main functions of the system ran according to the simulation scenario. This research does not claim novelty in the use of ESP32, HC-SR04, or Blynk because the combination has been widely used. The research contribution is more appropriately placed on the integration of monitoring and control tailored to the needs of the company's tank, the determination of empty-full reference points, and quantitative evaluation at the simulation stage.

The main limitations are the lack of a physical prototype, repeatable measurements, latency and packet loss analysis, and long-term reliability testing. Therefore, simulation results should be viewed as initial evidence of functional feasibility, not field validation. Further development will require implementing the physical device in a real reservoir, performing dimension-based calibration, adding hysteresis and fail-safe mechanisms, and conducting repeated statistical and communication testing.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the Building Engineering Team for their valuable assistance during the interview process, system requirement analysis, and technical discussions that significantly contributed to the development of this research. The authors also gratefully acknowledge PT Artajasa Pembayaran Elektronis for providing the opportunity to conduct observations and internship activities, which served as the primary basis for understanding the operational requirements of the company's water reservoir monitoring system. The insights and practical experience gained throughout the internship greatly supported the system design and simulation presented in this study. Finally, the authors appreciate all individuals and institutions who directly or indirectly contributed to the completion of this research.

CONFLICTS OF INTEREST

The authors declare no conflict of interest regarding the publication of this research. This study was conducted as part of an internship program, and all research activities, data collection, analysis, and report preparation were conducted objectively without any commercial, financial, or personal interests that could influence the results or conclusions of this study.

REFERENCES

[1]. D. Ma'ruf, K. Harianto, and L. Hartono, "*Engineering of IoT-based smart water reservoir detection device*," Journal of Applied Microcontroller and Autonomous System, vol. 5, 2025, doi: 10.71302/jamas.v5i2.87.

[2]. HD Eriyanto, M. Ismail, and EN Budisusila, "*Design and construction of an IoT (Internet of Things)-based automatic water filling monitoring device*," AVITEC, vol. 5, no. 2, 2023, doi: 10.28989/avitec.v5i2.1706.

[3]. RA Ramiz, LD Samsumar, A. Subki, and M. Zulpahmi, "*Design of an IoT-based water level monitoring system for home water tanks with the Blynk application,*" Journal of Data Analytics, Information and Computer Science, vol. 1, no. 4, 2024, doi: 10.70248/jdaics.v1i4.1309.

[4]. A. Wijoseno, HP Uranus, and J. Pangaribuan, "*Automatic water filling control using ESP32, MQTT, and Kodular with WhatsApp notification and data logging via Google Sheets,*" Jurnal Informatika dan Tekno Elektro Terapan, vol. 13, no. 3, 2025, doi: 10.23960/jitet.v13i3.7139.

[5]. AR Azhar, DA Setiawan, NA Yasmin, TA Putri, and GF Nama, "*IoT-based automatic water capacity monitoring and filling system using ESP8266 module,*" Journal of Informatics and Applied Electrical Engineering, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3966.

[6]. JAT Fernandes, GF Loureiro, and LB Giovanelli, "*Development of a Prototype for Water Reservoir Level Monitoring Using an ESP32 Board,*" International Journal of Geoscience, Engineering and Technology, vol. 10, no. 1, 2024.

[7]. DS Estu, M. Yantidewi, MB Adikuasa, BM Rusdi, and M. Khoiro, "*IoT-Based Sea Water Level Monitoring Device with NodeMCU ESP32 and HC-SR04,*" Jurnal Kolaboratif Sains, vol. 6, no. 7, 2023.

[8]. D. Diono, MS Gozali, and HT Guntoro, "*IoT implementation in water level monitoring system using ESP32 and Antares platform,*" Journal of Integration, vol. 18, no. 1, 2026, pp. 18–24, doi: 10.30871/ji.v18i1.12814.

[9]. M. Khairullah, M. Syamsu, and E. Sestri, "*Design and Construction of a Web-Based Waste Bank Information System Using the Waterfall Method (Case Study: Pamegarsari Village Waste Bank),*" Jurnal Sistem Informasi (JUSIN), vol. 5, no. 1, pp. 9–20, 2024, doi: 10.32546/jusin.v5i1.2452.

[10]. Y. Basir, MRA Pratama, and MW Aminullah, "*Design of a flood detection and mitigation system using IoT-based ESP32,*" Jurnal Ilmiah Giga, vol. 26, no. 1, pp. 11–22, 2023, doi: 10.47313/jig.v26i1.2127.

[11]. IPWPK Jaya, IGA Widagda, and IGA Kasmawan, "*Design and construction of a remote water level detection system using an HC-SR04 sensor and a SIM800L GSM module based on an ATMega 328 microcontroller,*" Physics Bulletin, vol. 24, no. 1, pp. 32–39, 2021, doi: 10.24843/BF.2023.v24.i01.p05.

[12]. A. Aziz, MS Aditya, S. Rahyadi, A. Apipah, ZA Munawarah, and D. Aribowo, "*Water level sensor based on Arduino UNO R3,*" Jupiter: Publication of Industrial Engineering, Electrical Engineering and Informatics, vol. 2, no. 6, pp. 93–102, 2024, doi: 10.61132/jupiter.v2i6.620.

[13]. H. Alamsyah, P. Sawaludin, and TU Kalsum, "*Monitoring and detection system of level and turbidity in embankment water in real time based on IoT,*" Internet of Things and Artificial Intelligence Journal, vol. 4, no. 3, 2024, doi: 10.31763/iota.v4i3.760.

[14]. W. Faturochman, P. Dauni, ET Arujisaputra, A. Husain, and I. Abdurrohim, "*Design of flood early warning monitoring system using a microcontroller ultrasonic sensor based on website and Telegram Chat Bot,*" JESII: Journal of Electronic Information Systems, vol. 1, no. 2, pp. 101–110, 2023, doi: 10.31848/jesii.v1i2.3358.

[15]. R. Sinaga, N. Novriyenni, and S. Syahputra, "*Design of an automatic water faucet system using the IoT based HC-SR04 sensor,*" Journal of Artificial Intelligence and Engineering Applications (JAIEA), vol. 3, no. 1, pp. 274–278, 2023, doi: 10.59934/jaiea.v3i1.308.

[16]. Syamsul, D. Hendryady, and N. Syam, "*Flood detection monitoring prototype using ESP32-based Blynk application,*" Journal of System Information And Computer, vol. 1, no. 1, pp. 20–34, 2023, doi: 10.63989/ammatoa.v1i1.4.