

IoT-Based Smart Water Tank Monitoring System

Bernardo Lobo Marques, Bernardo Moreira Guimarães Gonçalves, Michel de Melo Guimarães, Thiago Neves Monteiro, Rigel P. Fernandes^{ID}, Talita V. Ribeiro^{ID} and Clayton J. A. Silva

Abstract—Manual inspection of domestic water tanks is inefficient and often misses critical drops in availability. We present an end-to-end IoT stack coupling an HC-SR04 ultrasonic sensor with an ESP32 controller that samples once per second, rejects transient multipath outliers, and batches 45 readings every 45 seconds into an authenticated Supabase Edge Function; a React dashboard adds historical analytics, configurable alerts, and remote calibration. Bench validation on an 8 cm prototype (1.0 L) gave a mean absolute error (MAE) of 0.63 cm, dominated by the test container’s non-flat geometry; a first-order calibration reduces it to 0.28 cm, confirming a deterministic rather than noise-driven residual. Under an untested size-independent error assumption, this MAE would project below 0.6 % relative error on 1,000 L tanks — a hypothesis we flag for full-scale validation. Ingestion latency was 320 ms median, dashboard refresh 1.6 s, and alert delivery 100 %, supporting residential viability.

Keywords—IoT, Water Level Monitoring, Ultrasonic Sensor, ESP32, Supabase, React.

I. INTRODUCTION

Brazil’s National Sanitation Information System (SNIS) reports that 40.1% of drinking water distributed in 2020 was lost or unaccounted for [1]. These losses arise in the utility network rather than in household storage, but they frame a scarcity context in which giving end-users tools to monitor their own reservoirs and detect anomalies early is a complementary, demand-side contribution. Ultrasonic metering offers non-contact sensing with no moving parts, and IoT connectivity extends it into continuous telemetry. Despite this maturity, many low-cost deployments rely on ad hoc stacks that offer limited security and omit historical analytics [2].

This paper presents an HC-SR04-based monitor coupled with an ESP32 edge node and a React web application. Contributions are: (i) a low-cost edge pipeline that acquires one ultrasonic sample per second, applies a range-gate filter to reject multipath outliers, compresses 45 readings into a validated JSON payload, and delivers it through an authenticated Supabase Edge Function; (ii) an analytics dashboard with trend history, alert management, and remote calibration; and (iii) a laboratory evaluation of signal accuracy and ingestion latency, together with a projective analysis of scalability to real-world tank volumes. Section II reviews related work; Section III

Bernardo L. Marques, Bernardo M. G. Gonçalves, Michel M. Guimarães, Thiago N. Monteiro, Rigel P. Fernandes, and Clayton J. A. Silva are affiliated with the undergraduate Tech programs at Centro Universitário Ibmecc, Rio de Janeiro, Brazil. Emails: {bernardolobomarques, bbernardo.goncalves, michelmg.dev145, thiagonmonteiro08, rigelfernandes}@gmail.com, clayton.silva@ibmecc.edu.br

Talita V. Ribeiro is affiliated with the undergraduate Tech programs at IBMEC-RJ and with the Computer Science Course at UERJ. Email: talita.ribeiro@ime.uerj.br.

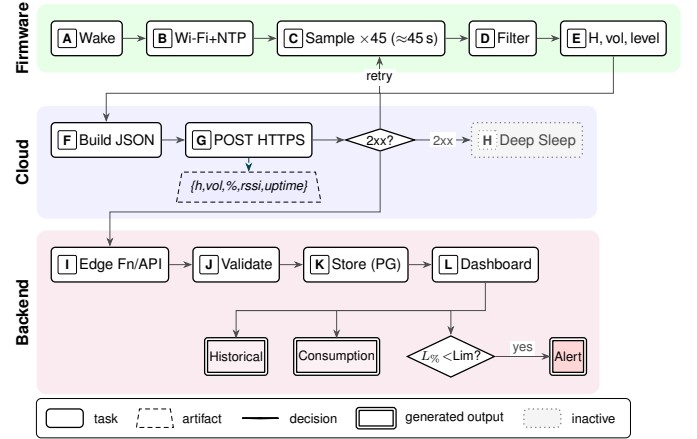


Figure 1. End-to-end architecture of the proposed system, spanning edge acquisition on the ESP32, cloud transmission, and backend storage and visualization. The deep-sleep step is drawn as inactive because it was disabled in the sandbox evaluation.

details the design; Section IV presents results; Section V concludes.

II. BACKGROUND AND RELATED WORK

The study in [3], though evaluating a low-cost *pressure* sensor, established that adequate accuracy (± 10 mm) is only reached after *per-device* calibration — a principle we adopt for our ultrasonic node, and one that aligns with our finding that the dominant bench residual is a deterministic, calibratable offset rather than sensor noise. Low-cost HC-SR04 stacks with cloud dashboards, such as [2], often neglect hardened APIs and statistical signal processing, while [4] formalized a TLS-protected IoT architecture for water infrastructure that informs our Supabase security design.

III. SYSTEM DESIGN

A. End-to-End Architecture

The solution follows a three-layer topology (Figure 1). The HC-SR04 measures the free surface distance; the ESP32 applies statistical filtering and aggregates samples; Supabase persists and serves readings to the React app.

The HC-SR04 (40 kHz) is a time-of-flight ranger widely used in low-cost tank monitoring [5]: it emits an ultrasonic burst and times the echo, so the reported distance depends on the speed of sound and on the geometry of the reflecting surface. Its $\sim 15^\circ$ beam cone means the returned echo integrates reflections over a finite footprint rather than a single point — a property that becomes relevant to the fill-dependent bias analysed in Section IV. The sensor is mounted at the tank top and triggered once per second. Averaging 45 validated samples

sets a ≈ 45 s reporting cycle, well below the timescale over which a residential tank level changes; each cycle POSTs one HTTP batch, with RSSI and uptime metadata, to the “iot-ingest” Edge Function (Algorithm 1).

Algorithm 1 Water Monitoring Firmware Loop

```

for i = 1 to 45 do                                ▷ 1 sample/s
    distance ← (echoWidth × 0.0343) / 2
    if distance ∈ [2, 400] cm then append to distanceBuffer
    end if
end for
avgDist ← mean(distanceBuffer) − SENSOR_OFFSET
h ← clamp(TANK_HEIGHT − avgDist, 0, TANK_HEIGHT)
vol ← ( $\pi r^2 \cdot h \cdot K_{\text{corr}}$ ) / 1000
payload ← JSON{h, vol, level%, rssi, uptime, timestamp}
if HTTP POST ≠ 200 then Log error; retry next cycle
end if
    
```

Presentation layer. The React app uses Supabase’s JavaScript SDK via the `useWaterData` hook, merging historical queries with real-time subscriptions feeding chart and alert components (Figure 2).



Figure 2. Web dashboard: real-time reservoir status, consumption trends, and alert management.

B. Signal Processing and Outlier Rejection

The range-gate filter addresses *transient* multipath echoes caused by free-surface perturbations (ripples, vibration), which generate spurious short-distance returns that would inflate the mean distance estimate. It is a lightweight rejection stage, not a model-based estimator: a Kalman filter would provide optimal state estimation under Gaussian noise [6], but the 45-sample integration window already attenuates zero-mean white noise by $\sqrt{45} \approx 6.7\times$, so mean filtering over validated samples yields comparable accuracy for slow-varying tank levels at negligible firmware cost. We stress that this filter targets *random* disturbances only; the *systematic*, fill-dependent bias observed in Section IV (an under-reading that trends upward with level) is determined by container geometry and is corrected by calibration (offset and scale).

C. Computation Model and Security

Volume is estimated as $V = \pi r^2 h K_{\text{corr}}$, where K_{corr} is an empirical correction absorbing the non-ideal base area of the bench container; the vessel’s full physical capacity (1.4 L over its 10 cm total height) versus the ideal-cylinder estimate

for the same dimensions yields $K_{\text{corr}} \approx 1.14$, which we then apply over the 8 cm useful calibration range (1.0 L nominal) reported in Section IV. The firmware assumes a fixed speed of sound ($0.0343 \text{ cm}/\mu\text{s}$); temperature compensation is not yet applied and is a known limitation for deployments with wide thermal swings. Device-to-cloud traffic runs over HTTPS; certificate pinning is not yet enabled on the device, so transport confidentiality currently relies on the server-side TLS endpoint. The Edge Function validates a per-device API key before any write — returning HTTP 401 on authentication failure, HTTP 400 on malformed or out-of-range payloads, and HTTP 429 under a per-device rate limit — broadly following the key-plus-transport model of [4].

D. Communication Protocol

Although MQTT achieves lower per-message latency in lightweight IoT scenarios [7], HTTP was chosen for native, broker-free integration with the Supabase serverless stack. The ~ 50 ms HTTP/TLS penalty represents less than 0.2% of the 45-second cycle, rendering it operationally negligible while a single HTTPS endpoint eliminates broker overhead. Wi-Fi sleep is disabled (`WiFi.setSleep(false)`) to guarantee consistent intervals.

IV. IMPLEMENTATION AND EVALUATION

A. Prototype Assembly

Figure 3 shows the bench prototype: a 1.0 L ($8 \text{ cm} \times 12.5 \text{ cm}$) container with HC-SR04 and ESP32, calibrated in 100 mL (10 %) increments. The same 2–400 cm range covers standard residential tanks (60–180 cm) without firmware changes.



Figure 3. Laboratory prototype: ESP32, HC-SR04, and 1.0 L reservoir.

B. Performance Metrics

Table I summarises the key indicators; Table II shows the per-level error breakdown for the 450-sample accuracy campaign (10 levels, 45 samples each). Latency was measured separately over 120 consecutive ingestion batches on a stable home Wi-Fi link, and alert reliability over all threshold-crossing events triggered during the campaign.

Table I. Key performance indicators.

Metric	Result
Mean absolute error	0.63 cm raw; 0.28 cm after affine calib.
Proj. relative error	<0.6 % at 1,000 L (projection, Table III)
Ingestion latency	320 ms median, Wi-Fi ($n=120$ batches)
Dashboard refresh	1.6 s avg., mid-range browser
Alert reliability	100 % for all thresholds

Table II. Detailed Prototype Error Analysis (HC-SR04)

Vol.(ml)	Level	Theor.(cm)	Meas.(cm)	Err.(cm)
100	10%	0.8	0.59	0.21
200	20%	1.6	1.77	0.17
300	30%	2.4	2.06	0.34
400	40%	3.2	2.95	0.25
500	50%	4.0	3.24	0.76
600	60%	4.8	3.83	0.97
700	70%	5.6	5.01	0.59
800	80%	6.4	5.30	1.10
900	90%	7.2	5.90	1.30
1000	100%	8.0	7.38	0.62
MAE:				0.631

C. Scalability and MAE Reframing

The 0.63 cm MAE is a conservative upper bound, not an intrinsic sensor limitation. The signed errors reveal a systematic structure: the measured height under-reads the true level in 9 of the 10 test points, and the deviation follows an approximately linear, increasing trend with fill (from -0.21 cm at 10% to -1.30 cm at 90%), for a mean signed bias of -0.60 cm. The trend is not strictly monotonic — the 20% point over-reads slightly and the 70% and 100% points fall below the local trend — yet an affine (offset + scale) fit to the ten bench points (height = $-1.146d + 12.96$, where d is the raw measured distance) reaches $R^2 = 0.98$, confirming that the residual is a deterministic geometry/calibration error rather than transient multipath noise. Such a bias is correctable by calibration and would not appear in a flat-bottom residential tank: applying the fit reduces the MAE from 0.63 cm to **0.28 cm**, meeting the sub-0.3 cm target without a model-based estimator. The deployed firmware absorbs this correction directly through a piecewise-linear distance-to-volume lookup table.

The fill-dependent bias arises from the interaction between the sensor’s beam cone and the container’s non-flat base. In the tapered lower portion of the bench vessel, the distance-to-volume mapping assumed for an ideal cylinder diverges from the true tapered volume, so the height inferred from a single distance reading under-reads the actual level, and the gap grows with fill. A flat-bottomed residential tank has constant cross-section and removes this source of divergence — which is why we treat the bench MAE as a conservative upper bound rather than a sensor limit.

Because the HC-SR04 measures distance independently of container size, we *hypothesize* that the absolute error remains approximately constant while the relative contribution decreases with tank height. Table III projects this headroom under that assumption: at 1,000 L the projected relative error is 0.53%, dropping to 0.29% at 10,000 L. We emphasise that these are projections, not measurements: factors absent from an 8 cm bench — the $\sim 15^\circ$ beam cone reflecting off tall walls and the temperature dependence of the speed of sound — may alter the absolute error at full scale. Experimental validation on a $\geq 1,000$ L flat-bottom tank is left as future work.

Table III. *Projected* relative error across tank sizes, under the (untested) assumption of a size-independent absolute error (MAE = 0.63 cm fixed; cylindrical geometry). Values are projections, not measurements.

Capacity (L)	Typ. Height (cm)	Rel. Error (%)
<i>Bench (1)</i>	8	7.90
500	90	0.70
1,000	120	0.53
5,000	180	0.35
10,000	220	0.29

V. CONCLUSION

We presented a low-cost IoT stack — HC-SR04 acquisition, ESP32 edge processing with range-gate rejection, API-key-authenticated Supabase ingestion, and a React dashboard — for residential water monitoring. The bench MAE of 0.63 cm is geometry-driven and drops to 0.28 cm under a first-order calibration; together with sub-second dashboard refresh and reliable threshold alerts, this meets our performance targets with headroom for deployment. Source code is available in [8]. Future work will validate the projected sub-0.6% relative error on a full-scale flat-bottom tank and characterise latency under reinforced-concrete radio conditions.

REFERENCES

- [1] Instituto Trata Brasil and GO Associados, “Perdas de água potável 2022 (snis ano base 2020): Desafios para disponibilidade hídrica e avanço da eficiência do saneamento básico no Brasil,” Instituto Trata Brasil, Tech. Rep., 2022, Índice de perdas na distribuição (IN049) de 40,1%, SNIS ano base 2020. Accessed: 31 Jul. 2026. [Online]. Available: <https://tratabrasil.org.br/apenas-uma-regiao-brasileira-demonstrou-breve-evolucao-no-indicador-de-perdas-de-agua-na-distribuicao/>
- [2] H. K. H. Singh, H. Kamaludin, I. R. A. Hamid, N. Z. M. Safar, and N. Abdullah, “IoT-based water tank level monitoring system using ultrasonic sensor,” in *The 2nd International Recent Trends in Engineering, Advanced Computing and Technology Conference (RETREAT) 2021*, ser. AIP Conference Proceedings, vol. 2608, no. 1. AIP Publishing, 2023, p. 020009. [Online]. Available: <https://doi.org/10.1063/5.0128168>
- [3] N. Ding, Q. Zhu, F. Cherqui, N. Walcker, J.-L. Bertrand-Krajewski, and P. Hamel, “Laboratory performance assessment of low-cost water level sensor for field monitoring in the tropics,” *Water Research X*, vol. 27, p. 100298, 2025. [Online]. Available: <https://doi.org/10.1016/j.wroa.2024.100298>
- [4] M. Al-Shareeda, A. Mohammed Ali, M. A. Hammoud, Z. H. M. Kazem, and M. A. Hussein, “Secure IoT-based real-time water level monitoring system using ESP32 for critical infrastructure,” *Journal of Cyber Security and Risk Auditing*, vol. 2025, no. 2, pp. 44–52, 2025. [Online]. Available: <https://doi.org/10.63180/jcsra.thestap.2025.2.4>
- [5] M. R. Hidayat, S. Sambasri, F. Fitriansyah, A. Charisma, and H. R. Iskandar, “Soft water tank level monitoring system using ultrasonic HC-SR04 sensor based on ATmega 328 microcontroller,” in *2019 IEEE 5th International Conference on Wireless and Telematics (ICWT)*, 2019, pp. 1–4. [Online]. Available: <https://doi.org/10.1109/ICWT47785.2019.8978229>
- [6] U. Pratiwi, I. Fadli, W. T. Cahyanto, and Hartono, “Implementation of Kalman filter algorithm to optimize the calculation of ultrasonic sensor distance value in Hooke law props system,” *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 5 (127), pp. 48–60, 2024. [Online]. Available: <https://doi.org/10.15587/1729-4061.2024.296667>
- [7] B. Wukkadada, K. Wankhede, R. Nambiar, and A. Nair, “Comparison with HTTP and MQTT in internet of things (IoT),” in *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2018, pp. 249–253. [Online]. Available: <https://doi.org/10.1109/ICIRCA.2018.8597401>
- [8] B. L. Marques, B. M. G. Gonçalves, M. de Melo Guimarães, T. N. Monteiro, R. P. Fernandes, T. V. Ribeiro, and C. J. A. S. Rocha, “Simblissima embarcada,” 2026, accessed: 26 May. 2026. [Online]. Available: <https://github.com/bernardolobomarques/SimblissimaEmbarcada>