

# Pengembangan Sistem IoT Terintegrasi untuk Deteksi Ketinggian Air dan Pengiriman Peringatan Berbasis di Telegram

1<sup>st</sup> Daffa Fathir Fajri  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

[daffafathir@student.telkomuniversity.ac.id](mailto:daffafathir@student.telkomuniversity.ac.id)  
[c.id](https://www.telkomuniversity.ac.id)

2<sup>nd</sup> Meta Kallista  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

[metakallista@telkomuniversity.ac.id](mailto:metakallista@telkomuniversity.ac.id)

3<sup>rd</sup> Hasbi Ash Shiddieqy  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

[hasbisiddiq@telkomuniversity.ac.id](mailto:hasbisiddiq@telkomuniversity.ac.id)

**Abstrak** — Banjir merupakan bencana yang kerap terjadi di berbagai wilayah, terutama di sekitar aliran sungai. Peningkatan volume air saat musim hujan sering kali menyebabkan kerugian material dan mengganggu aktivitas masyarakat. Salah satu kendala dalam mitigasi risiko banjir adalah belum tersedianya sistem peringatan dini yang cepat dan mudah diakses. Penelitian ini merancang sistem pemantauan ketinggian air berbasis Internet of Things (IoT) dengan fitur peringatan melalui aplikasi Telegram. Sistem menggunakan sensor ultrasonik untuk mendeteksi tinggi muka air secara real-time, lalu mengirimkan data ke platform Supabase sebagai media penyimpanan. Telegram digunakan sebagai antarmuka utama untuk menyampaikan peringatan otomatis saat ketinggian air melewati ambang batas, serta menyediakan perintah sederhana bagi pengguna untuk memeriksa status dan riwayat data. Berdasarkan hasil pengujian, sistem mampu memantau perubahan permukaan air dengan akurat dan memberikan respons notifikasi yang cepat. Sistem ini diharapkan dapat membantu meningkatkan kesiapsiagaan masyarakat dalam menghadapi potensi banjir secara praktis dan efisien.

**Kata kunci**— Banjir, Internet of Things, Peringatan Dini, Sensor Ultrasonik, Telegram, Supabase

## I. PENDAHULUAN

Banjir merupakan salah satu bencana hidrometeorologi yang paling sering terjadi di Indonesia dan berdampak signifikan terhadap kehidupan sosial dan ekonomi masyarakat [1]. Salah satu penyebab utama banjir adalah meningkatnya debit air sungai akibat curah hujan tinggi dan buruknya sistem drainase. Untuk mengurangi risiko dan dampak bencana, sistem peringatan dini sangat dibutuhkan sebagai upaya mitigasi yang efektif. Dengan kemajuan teknologi, Internet of Things (IoT) telah banyak diterapkan dalam sistem pemantauan lingkungan, termasuk monitoring tinggi muka air secara real-time [2]. Berbagai penelitian telah mengembangkan sistem berbasis sensor ultrasonik untuk mengukur ketinggian air dan menyampaikan data melalui jaringan nirkabel [3]. Namun, sebagian besar sistem yang ada masih bergantung pada dashboard berbasis web, yang kurang

efisien dalam kondisi darurat. Permasalahan utama adalah belum tersedianya sistem pemantauan yang mampu mengirimkan peringatan secara langsung dan cepat melalui

media komunikasi populer seperti aplikasi pesan instan. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem pemantauan ketinggian air berbasis IoT yang terintegrasi dengan platform Supabase sebagai penyimpanan data dan Telegram sebagai media notifikasi otomatis dan perintah interaktif. Sistem ini diharapkan dapat meningkatkan efektivitas peringatan dini banjir dan memperkuat kesiapsiagaan masyarakat.

## II. KAJIAN TEORI

### A. Internet of Things

Internet of Things (IoT) adalah konsep jaringan di mana berbagai perangkat fisik saling terhubung dan mampu bertukar data melalui jaringan internet tanpa intervensi manusia secara langsung. IoT memungkinkan otomatisasi sistem dan pemantauan real-time terhadap objek fisik, menjadikannya sangat relevan dalam bidang lingkungan, kesehatan, industri, dan mitigasi bencana [4]. Dalam konteks penelitian ini, IoT digunakan untuk menghubungkan sensor pemantau tinggi muka air dengan layanan cloud dan sistem notifikasi secara otomatis.

### B. Mikrokontroler ESP32

ESP32 adalah modul mikrokontroler berbasis SoC (System on Chip) yang dilengkapi dengan konektivitas Wi-Fi dan Bluetooth. Perangkat ini memiliki performa tinggi, konsumsi daya rendah, serta mendukung banyak antarmuka komunikasi seperti UART, SPI, dan I2C, menjadikannya cocok untuk aplikasi IoT [5]. Dalam sistem ini, ESP32 berperan sebagai unit pemroses utama yang membaca data dari sensor, menghubungkannya ke internet, dan mengirimkan data ke cloud serta Telegram Bot.

### C. Sensor Ultrasonik

Sensor ultrasonik bekerja dengan memancarkan gelombang suara berfrekuensi tinggi dan mengukur waktu pantulan gelombang tersebut dari permukaan objek untuk menghitung jarak. Dalam sistem pemantauan tinggi air, sensor ini digunakan untuk mengukur jarak antara sensor dan permukaan air secara akurat dan non-kontak [6]. Sensor ini banyak digunakan karena hemat energi, biaya rendah, dan mudah dipasang di lapangan.

#### D. Supabase

Supabase adalah platform backend open-source yang menyediakan layanan database real-time, autentikasi, serta API berbasis PostgreSQL. Dalam sistem ini, Supabase digunakan sebagai penyimpanan data cloud untuk menyimpan hasil pembacaan ketinggian air secara terstruktur dan dapat diakses kapan saja [7]. Kelebihannya meliputi kemudahan integrasi dengan REST API, dukungan data real-time, dan dokumentasi yang baik bagi pengembang IoT.

#### E. Telegram

Telegram merupakan aplikasi pesan instan berbasis cloud yang mendukung pengembangan bot otomatis melalui Telegram Bot API. Bot ini memungkinkan pengiriman pesan otomatis, serta menerima perintah teks dari pengguna. Dalam penelitian ini, Telegram digunakan sebagai sarana utama notifikasi peringatan dini kepada pengguna ketika ketinggian air melebihi ambang batas, sekaligus menyediakan perintah seperti status dan histori [8].

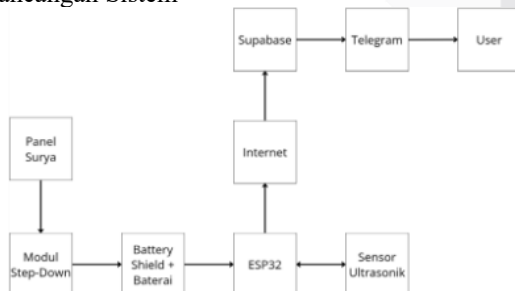
#### F. Sistem Pemantauan Dini Banjir

Sistem pemantauan banjir dini merupakan gabungan dari teknologi pengukuran, pengolahan data, dan penyampaian informasi yang bertujuan untuk mendeteksi potensi banjir sebelum terjadi. Sistem ini harus mampu memberikan informasi secara cepat, akurat, dan mudah diakses oleh masyarakat maupun instansi terkait [9]. Penggunaan teknologi berbasis IoT mempercepat pengiriman data dari lapangan ke pengguna akhir, sehingga langkah mitigasi dapat dilakukan lebih dini.

### III. METODE

Bab ini menjelaskan metode yang digunakan dalam perancangan dan implementasi sistem pemantauan ketinggian air berbasis Internet of Things (IoT) untuk peringatan dini banjir. Penjelasan mencakup tahapan penelitian secara sistematis, mulai dari perancangan perangkat keras dan lunak, proses integrasi sistem, hingga pengujian fungsionalitas dan keandalan sistem. Selain itu, bab ini juga memuat informasi mengenai tempat dan waktu pelaksanaan penelitian, sumber serta cara perolehan data, dan metode pengolahan data untuk mendukung validitas hasil yang diperoleh.

#### A. Rancangan Sistem



Sistem dirancang berbasis Internet of Things (IoT) dengan memanfaatkan mikrokontroler ESP32 sebagai pusat pemrosesan, sensor ultrasonik A02YYUW untuk membaca tinggi permukaan air, serta platform cloud Supabase untuk menyimpan data secara daring. Telegram digunakan sebagai media komunikasi untuk mengirimkan notifikasi otomatis.



Selain itu, Telegram juga bisa menerima perintah-perintah sederhana dari pengguna.



#### B. Prosedur Penelitian

Langkah-langkah penelitian ini meliputi :

1. Studi literatur dan perancangan sistem perangkat keras dan lunak.
2. Perakitan dan konfigurasi perangkat sensor serta mikrokontroler ESP32.
3. Pemrograman mikrokontroler menggunakan Arduino IDE.
4. Integrasi sistem dengan Supabase melalui API.
5. Pengembangan dan pengujian Telegram Bot untuk notifikasi dan perintah sederhana.
6. Pengujian fungsionalitas dan keandalan sistem secara simulatif.
7. Evaluasi performa sistem berdasarkan akurasi data, respons waktu notifikasi, dan kestabilan koneksi.

#### C. Sumber Data dan Cara Perolehan Data

Data utama diperoleh secara otomatis dari pembacaan sensor ultrasonik yang mendeteksi ketinggian permukaan air. Setiap hasil pembacaan dikirim dan disimpan ke Supabase dalam interval waktu tertentu. Selain itu, respons sistem terhadap perintah Telegram juga dikumpulkan untuk mengevaluasi kecepatan dan akurasi komunikasi.

#### D. Pengolahan Data

Pada bagian pengolahan data ketinggian air, digunakan perhitungan sederhana di ESP32 seperti berikut :

$$\text{water level} = 450 - \text{distance}$$

Untuk pengiriman data ke Supabase, mikrokontroler ESP32 terlebih dahulu menyimpan hasil pembacaan sensor ke dalam array sementara setiap satu menit sekali. Setelah array berisi sepuluh data, sistem akan menghitung nilai rata-rata dari keseluruhan data tersebut, kemudian mengirimkan hasil rata-rata tersebut ke platform Supabase secara berkala.

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \dots + x_{10}}{10}$$

Nilai ( $\bar{x}$ ) inilah yang kemudian akan dikirim ke Supabase.

#### E. Metode Pengujian

Pengujian dilakukan dalam dua tahap, yaitu:

1. **Pengujian fungsional**, untuk memastikan sistem dapat bekerja sesuai yang dirancang: pembacaan sensor, pengiriman data, dan respon Telegram.
2. **Pengujian simulasi kondisi banjir**, dengan memanipulasi jarak antara sensor dan permukaan air untuk menguji apakah notifikasi dikirimkan tepat waktu saat melewati ambang batas yang telah ditentukan.

#### F. Tempat dan Waktu Penelitian

Penelitian dilakukan melalui uji coba lapangan di sekitar aliran Sungai Sukapura pada rentang waktu awal Juni hingga akhir Juli 2025, dengan tujuan menguji performa sistem dalam kondisi nyata. Pengujian difokuskan pada aspek akurasi sensor, stabilitas sistem, serta efektivitas notifikasi Telegram dalam menyampaikan informasi kepada pengguna.

### IV. HASIL DAN PEMBAHASAN

Bab ini memaparkan hasil implementasi dan pengujian sistem pemantauan ketinggian air berbasis IoT yang telah dikembangkan. Penelitian ini bertujuan untuk mengetahui sejauh mana sistem dapat bekerja sesuai perancangan, mulai dari akurasi pembacaan sensor, kecepatan pengiriman data ke Supabase, efektivitas notifikasi Telegram, hingga stabilitas sistem secara keseluruhan. Data diperoleh melalui serangkaian pengujian di lapangan dengan pendekatan simulasi kondisi riil, serta dianalisis untuk memperoleh kesimpulan atas kinerja sistem secara objektif.

#### A. Hasil Implementasi Sistem

Bagian ini akan menunjukkan implementasi kajian teori yang digunakan, yang dimana sistem mampu mendeteksi variasi ketinggian air secara real-time melalui sensor ultrasonik, kemudian mengolah data tersebut menggunakan mikrokontroler ESP32. Hasil pembacaan dikirim secara berkala setiap 10 menit ke database Supabase dengan performa yang stabil.



GAMBAR 1  
(A) Lokasi Pemasangan Alat

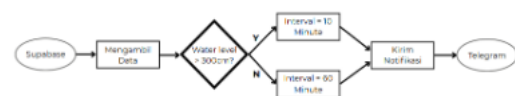


GAMBAR 2  
(B) Pengujian Sensor

Selanjutnya, informasi ini digunakan untuk menampilkan status ketinggian air serta level peringatan melalui Telegram sebagai media notifikasi.



GAMBAR 3  
(C) Aliran Sistem Otomatisasi Bot Telegram



GAMBAR 4  
(D) Detail Aliran Sistem Otomatisasi Bot Telegram

Melalui sistem ini, Bot Telegram dapat mengirimkan notifikasi otomatis dalam bentuk :

1. **Laporan rutin ketinggian air**, mengirim *update* mengenai ketinggian air terkini, status siaga terkini, dan lokasi setiap 60 menit.
2. **Notifikasi perubahan status siaga**, mengirim perubahan status siaga (kenaikan atau penurunan) serta detail ketinggian air terkini.
3. **Peringatan jika air melebihi threshold (> 300 cm)**, mengirim notifikasi secara intens setiap 10 menit ketika ketinggian air melebihi 300 cm yang berisi ketinggian air terkini serta himbauan atau tindakan yang harus dilakukan.

#### B. Data Baca Sensor Ultrasonik

Berikut adalah data baca dari pengujian yang menggunakan sensor ultrasonik :

TABEL 1  
(A) Hasil Pengujian Sensor Ultrasonik

| No. Uji | Meteran Taraf (cm) | Output Sensor (cm) | Selisih | Error (%) |
|---------|--------------------|--------------------|---------|-----------|
| 1       | 50                 | 51                 | 1       | 2.00      |
| 2       | 100                | 98                 | 2       | 2.00      |
| 3       | 150                | 148                | 2       | 1.33      |
| 4       | 200                | 202                | 2       | 1.00      |

|           |             |        |      |      |
|-----------|-------------|--------|------|------|
| 5         | <b>*250</b> | 255    | 5    | 2.00 |
| 6         | <b>*300</b> | 294    | 6    | 2.00 |
| 7         | <b>*350</b> | 347    | 3    | 0.86 |
| 8         | <b>*400</b> | 409    | 9    | 2.25 |
| 9         | <b>*450</b> | 445    | 5    | 1.11 |
| Rata-rata | 250         | 249.89 | 4.22 | 1.62 |

**Catatan :**

Tanda (\*) menunjukkan bahwa data simulasi digunakan untuk memenuhi kebutuhan uji coba.

Tabel menunjukkan bahwa pada sebagian besar titik pengujian, nilai pembacaan jarak oleh sensor tidak sepenuhnya sama dengan nilai acuan (meteran taraf), sehingga menghasilkan nilai selisih dan error dalam batas yang masih dapat ditoleransi. Nilai error terbesar terjadi pada pengujian jarak 400 cm dengan nilai 2.25%, sedangkan error terkecil tercatat pada pengujian 350 cm sebesar 0.86%. Adapun pengujian pada jarak 50 cm hingga 131 300 cm menunjukkan nilai error berkisar antara 1% hingga 2%, sementara pada 450 cm, error tercatat sebesar 1.11%. Untuk summary pengujian dari jarak 50 cm - 450 cm tercatat error sebesar 1.62%.

**C. Delay Bot Telegram**

Bagian ini akan memaparkan tentang hasil uji coba delay Bot Telegram yang digunakan sebagai media notifikasi. Pengujian ini dilakukan untuk mengukur kecepatan respon Bot dalam menjalankan perintah sederhana dan merespon perubahan ketinggian air pada Supabase. Data hasil pengujian dapat dilihat pada tabel dibawah ini :

TABEL 2  
(B) Hasil Pengujian Waktu Respon Bot Telegram

| Fitur                            | Sub-Fitur              | Tujuan Tes   | Waktu Respon |
|----------------------------------|------------------------|--|--------------|
| Simple Commands                  | /status                | Memastikan perintah berfungsi dan memberikan respons yang benar  | 1.18 detik   |
|                                  | /report                |  | 847 ms       |
|                                  | /history               |  | 826 ms       |
| Automatic Notification           | 60 Minute Interval     | Memastikan bot dapat mengirimkan pembaruan status otomatis setiap jam                                  | 3.73 detik   |
|                                  | 10 Minute Interval     | Memastikan bot dapat mengirimkan notifikasi darurat setiap 10 menit jika ketinggian melebihi threshold | 140 ms       |
| Alert Status Change Notification | Penurunan Status Siaga | Memastikan bot bisa mendeteksi dan memberikan notifikasi perubahan status siaga                        | 10 ms        |
|                                  | Kenaikan Status Siaga  |  | 2.26 detik   |

Dari tabel hasil pengujian waktu delay Bot Telegram diatas, dapat disimpulkan bahwa semua fitur Bot Telegram sudah berfungsi secara baik berdasarkan waktu respon yang diberikan. Dengan rentang waktu dari 10 ms hingga 3.73 detik.

**D. Analisis Hasil Implementasi**

Melalui sistem ini, Bot Telegram dapat mengirimkan notifikasi otomatis dalam bentuk :

**1. Kinerja Sistem IoT**

Implementasi sistem IoT berbasis ESP32, sensor ultrasonik, database Supabase, dan Bot Telegram berjalan stabil sesuai perancangan. Data dari sensor disimpan secara periodik setiap 1 menit pada array sementara, kemudian dirata-ratakan setiap 10 menit sebelum dikirim ke Supabase. Mekanisme ini berhasil meminimalkan fluktuasi pembacaan akibat gangguan sesaat. Selama pengujian lapangan, pengiriman data ke Supabase berjalan tanpa gangguan, dan sistem mampu memberikan notifikasi otomatis, laporan rutin, serta respon terhadap perintah pengguna dengan konsistensi yang baik. Hal ini membuktikan bahwa integrasi perangkat keras dan perangkat lunak dalam sistem IoT ini dapat bekerja secara handal dan berkesinambungan.

**2. Akurasi Sensor Ultrasonik**

Pengujian sensor ultrasonik A02YYUW menunjukkan tingkat akurasi yang tinggi dengan rata-rata error **1,62%** pada pengukuran jarak 50–450 cm. Nilai error terkecil tercatat pada jarak 350 cm sebesar **0,86%**, sedangkan nilai error terbesar sebesar **2,25%** pada jarak 400 cm. Seluruh nilai error masih berada dalam batas toleransi yang dapat diterima untuk aplikasi pemantauan tinggi muka air. Hasil ini menegaskan bahwa sensor mampu memberikan pembacaan yang konsisten dan dapat diandalkan untuk mendeteksi variasi permukaan air secara real-time.

**3. Delay Test Bot Telegram**

Hasil pengujian delay menunjukkan bahwa Bot Telegram mampu merespons berbagai jenis perintah dan notifikasi dengan sangat cepat. Perintah sederhana seperti /status, /report, dan /history memiliki waktu tunda di bawah 1,2 detik, yang tergolong responsif untuk komunikasi real-time. Notifikasi kritis saat tinggi air melebihi 300 cm memiliki delay hanya **140 ms**, sehingga informasi dapat sampai ke pengguna hampir seketika. Perubahan status siaga juga terdeteksi sangat cepat, dengan penurunan status memerlukan hanya **10 ms** dan kenaikan status **2,26 detik**. Waktu tunda ini membuktikan bahwa sistem peringatan berbasis Telegram dapat diandalkan untuk memberikan informasi tepat waktu dalam kondisi darurat.

**V. KESIMPULAN**

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, sistem pemantauan ketinggian air berbasis Internet of Things (IoT) dengan integrasi ESP32, sensor ultrasonik A02YYUW, layanan database Supabase, dan Telegram Bot berhasil berfungsi sesuai dengan tujuan perancangan. Sistem mampu membaca perubahan ketinggian air secara real-time, menyimpan data secara periodik, serta mengirimkan notifikasi peringatan otomatis dengan waktu respons yang cepat dan akurat. Pengiriman data ke Supabase berjalan stabil tanpa gangguan selama pengujian, dan sistem notifikasi melalui Telegram terbukti efektif dalam menyampaikan informasi baik secara otomatis maupun melalui perintah sederhana dari pengguna. Hasil ini menunjukkan bahwa sistem memiliki potensi besar untuk



digunakan sebagai solusi peringatan dini banjir yang terjangkau, efisien, dan mudah diakses masyarakat. Implementasi ini juga membuka peluang pengembangan lebih lanjut, seperti integrasi multi-sensor, prediksi banjir berbasis data historis, serta peningkatan sistem energi mandiri untuk keandalan jangka panjang di lapangan.

#### REFERENSI

- [1] BNPB, "Bencana Hidrometeorologi Dominasi Kejadian Bencana di Indonesia," *Badan Nasional Penanggulangan Bencana*, 2021.
- [2] M. A. Nugroho et al., "Design of IoT-based Flood Monitoring and Early Warning System," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 794, no. 1, 2021.
- [3] A. Putra and R. H. Nugroho, "Implementasi Sensor Ultrasonik untuk Pemantauan Ketinggian Air Berbasis Mikrokontroler," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 2, pp. 75–81, 2021.
- [4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [5] Espressif Systems, "ESP32 Technical Reference Manual," [Online]. Available: <https://www.espressif.com/en/products/socs/esp32>
- [6] A. Putra dan R. Nugroho, "Implementasi Sensor Ultrasonik untuk Pemantauan Ketinggian Air," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 2, pp. 75–81, 2021.
- [7] Supabase Docs, "Introduction to Supabase," [Online]. Available: <https://supabase.com/docs>
- [8] Telegram, "Telegram Bot API," [Online]. Available: <https://core.telegram.org/bots/api>
- [9] D. Setiawan, "Rancang Bangun Sistem Peringatan Dini Banjir Berbasis Mikrokontroler," *Jurnal Rekayasa Elektrika*, vol. 15, no. 1, pp. 34–41, 2021.