

Sistem IoT Pemantauan Ketinggian Air Sungai Berbasis Arduino dan Bot Telegram

Muhammad Asla Labiib Badaly¹, Julianto Aji Milanesta², M Afiv Hidayatulloh³

^{1,2,3}Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: ¹aslabiib26@gmail.com, ²juliantoajimilanestaofficial@gmail.com,
³mafivhidayatulloh@gmail.com

Abstrak – Banjir adalah bencana yang sering terjadi di Indonesia yang menyebabkan korban jiwa dan kerugian harta benda. Salah satu alasan mengapa banjir sering terjadi adalah fakta bahwa informasi tentang kenaikan ketinggian air sungai belum dikumpulkan dengan cepat karena proses pengawasan masih dilakukan secara manual. Tujuan dari penelitian ini adalah untuk membuat prototipe sistem pemantauan ketinggian air sungai berbasis *Internet of Things* (IoT) yang menggunakan mikrokontroler Arduino dan sensor ultrasonik. Sensor dipasang di atas air untuk mengukur jarak ke permukaan secara berkala dengan tingkat galat pembacaan yang kecil, yaitu sekitar 0,3 hingga 0,4 meter dari hasil pengujian. Arduino mengolah data jarak berdasarkan ambang batas tertentu dan menunjukkan status aman atau bahaya sesuai perubahan ketinggian air. Pada jarak manual 7 meter, 6 meter, dan 4 meter, sistem menunjukkan status aman karena jarak masih berada di atas ambang batas yang ditentukan. termasuk ketika jarak turun hingga 1 meter dan 7 meter yang terbaca sebagai kondisi bahaya. Selain itu, data dikirim ke komputer melalui komunikasi serial untuk disimpan dalam bentuk dataset; kemudian, ketika ketinggian air mendekati sensor, pesan peringatan dikirim ke aplikasi Telegram melalui bot. Hasil implementasi menunjukkan bahwa sistem dapat memantau ketinggian air secara *real-time* dengan akurasi sensor yang stabil dan secara otomatis mengirimkan notifikasi peringatan. Ini dapat membantu masyarakat dan pemangku kepentingan mengantisipasi banjir lebih awal.

Kata Kunci — *Arduino, banjir, Internet of Things, ketinggian air, Telegram*

1. PENDAHULUAN

Indonesia adalah negara yang berada di garis khatulistiwa sehingga Indonesia termasuk wilayah yang beriklim tropis, yang mengakibatkan sepanjang tahun, memiliki suhu hangat/panas stabil, kelembaban tinggi, dan curah hujan tinggi, dengan dua musim utama (hujan dan kemarau), bukan musim dingin, yang menciptakan hutan hujan lebat[1]. Akibat curah hujan yang tinggi, kondisi geografis, dan pengelolaan tata ruang yang buruk, Indonesia menjadi salah satu negara dengan risiko banjir yang tinggi[2]. Banjir yang sering terjadi berulang mengganggu aktivitas sosial dan perekonomian masyarakat selain menyebabkan kerugian material[3]. Tidak adanya data ketinggian debit air sungai yang cepat dan berkelanjutan adalah salah satu faktor yang membuat banjir lebih sulit di prediksi sehingga menyebabkan masyarakat kesulitan untuk mengantisipasi dan evakuasi.

Banyak petugas masih melakukan pemantauan kondisi sungai secara manual, seperti melihat papan duga air secara visual[4]. Metode ini memiliki beberapa kelemahan, seperti bahwa itu tidak selalu berlangsung, bergantung pada jumlah petugas yang tersedia di lapangan, dan rentan terhadap kesalahan pembacaan.

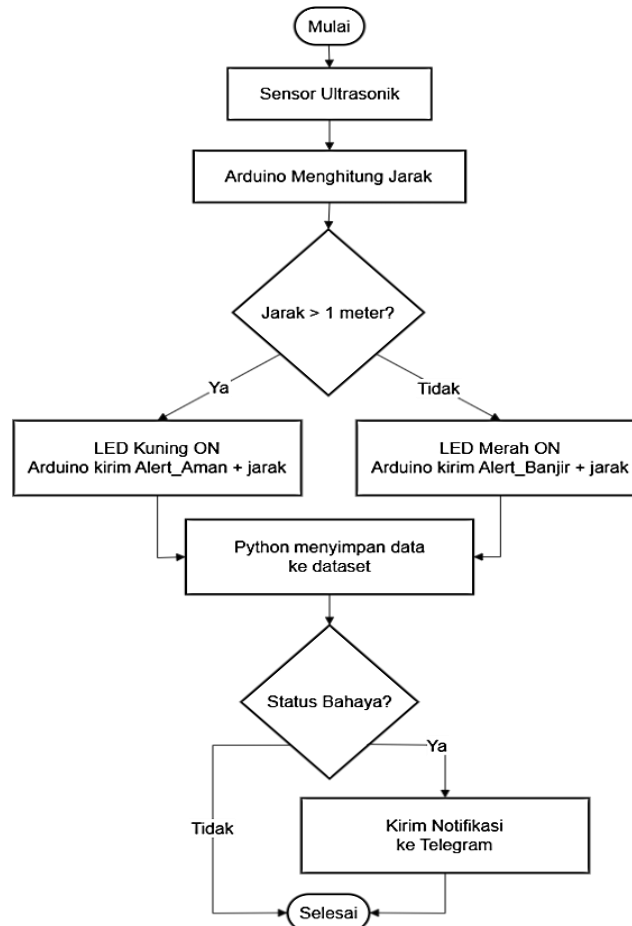
Perkembangan *Internet of Things* (IoT) memungkinkan berbagai perangkat fisik, seperti sensor dan mikrokontroler, terhubung ke internet secara otomatis untuk mengirim dan menerima data[5]. Akibatnya, beberapa penelitian telah mengembangkan sistem pemantauan ketinggian air berbasis IoT yang menggunakan sensor ultrasonik, mikrokontroler NodeMCU[6]. Beberapa menggunakan bot Telegram untuk menyimpan data pengamatan dan mengirimkan notifikasi ketika ketinggian air melewati ambang tertentu[7].

Meskipun demikian, prototipe yang sederhana dan mudah direplikasi yang menggunakan komponen yang biasa digunakan dalam pendidikan, seperti sensor ultrasonik dan Arduino Uno, masih diperlukan[8]. Namun, mereka masih dapat diintegrasikan dengan aplikasi Telegram dan menyimpan data dalam bentuk dataset. Diharapkan sistem seperti ini berfungsi sebagai alat pembelajaran dan solusi awal yang dapat dikembangkan lebih lanjut untuk diterapkan di lapangan[9].

Berdasarkan latar belakang ini, penelitian ini bertujuan untuk: (1) membuat sistem pemantauan ketinggian air sungai dengan sensor ultrasonik berbasis Arduino yang memiliki indikator LED lokal, (2) mengintegrasikan sistem dengan bot Telegram untuk memberikan peringatan ketika ketinggian air mendekati sensor, dan (3) membuat dataset hasil pengamatan ketinggian air yang dapat digunakan untuk analisis dan pengembangan sistem di masa mendatang. Artikel ini distrukturkan dengan pendahuluan, metode penelitian, hasil, pembahasan, dan saran.

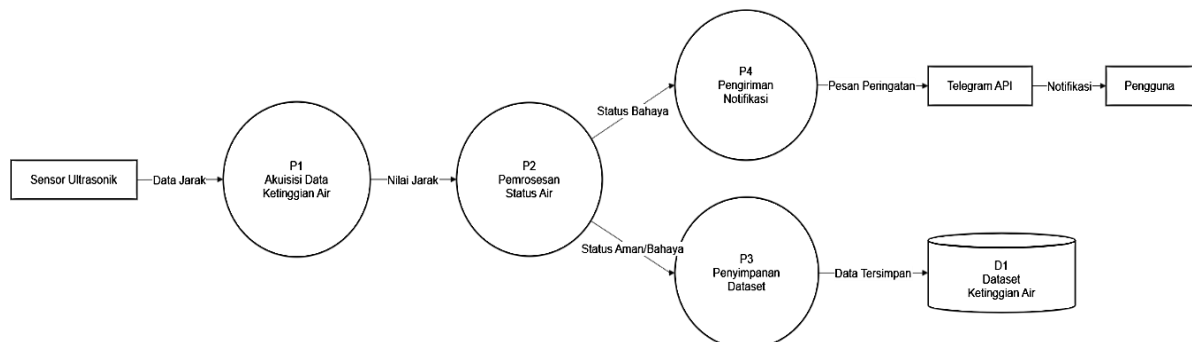
2. METODE PENELITIAN

Proses penelitian yang digunakan untuk membuat prototipe sistem pemantauan ketinggian air sungai berbasis *Internet of Things* (IoT) ini meliputi analisis kebutuhan, perancangan arsitektur sistem, pemilihan dan perancangan perangkat keras, perancangan perangkat lunak, implementasi, pengujian dan pengumpulan data. Setiap langkah dibahas dalam subbab berikut.



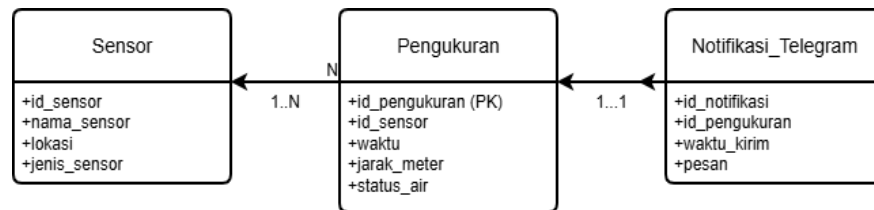
Gambar 1. Metode Penelitian

Gambar 1 merupakan gambar roses kerja sistem pemantauan kualitas udara berbasis Internet of Things (IoT), yang dimulai dengan pengumpulan data menggunakan sensor ultrasonik dan pemrosesan data menggunakan Arduino. Hasil pengukuran dibandingkan dengan batas ambang 1 meter untuk menentukan status aman atau berbahaya. Sedangkan LED merah menyala dalam kondisi berbahaya, LED kuning menyala dalam kondisi aman. Python digunakan untuk mengimpor data dan status sistem ke dalam dataset. Ketika terjadi masalah, sistem secara otomatis mengirimkan peringatan melalui Telegram.



Gambar 2. DFD Level 1

Gambar 2 merupakan gambar arsitektur sistem pemantauan banjir, di mana Sensor Tingkat Air beroperasi dengan mengukur tekanan air dan mengirimkan data ke sistem Banjir sebagai pusat pengolahan. Sistem Banjir menganalisis data sensor untuk menentukan kualitas air dan memasukkannya ke dalam basis data. Selain itu, sistem secara otomatis menghasilkan hasil berdasarkan kondisi yang telah ditentukan sebelumnya. Informasi ini kemudian disajikan kepada pengguna sebagai peringatan dini mengenai potensi banjir.



Gambar 3. Entity Relationship Diagram (ERD) Sistem Pemantauan Ketinggian Air Sungai Berbasis IoT

Gambar 3 merupakan gambar sistem ERD untuk pengendalian kualitas udara, yang terdiri dari tiga entitas utama: Sensor, Pengukuran, dan Notifikasi Telegram. Karena satu sensor dapat melakukan beberapa pengukuran, Entitas Sensor menyimpan informasi perangkat dan memiliki hubungan satu-ke-banyak dengan pengukuran. Hasil data pengukuran meliputi kualitas udara, status udara, dan waktu udara, yang menjadi dasar untuk analisis kondisi sistem. Selanjutnya, setiap set data dapat menghasilkan satu notifikasi Telegram yang dikirimkan kepada pengguna sebagai peringatan dini ketika situasi menjadi tidak stabil.

2.1 Analisis Kebutuhan

Analisis Kebutuhan: Pada tahap ini, analisis kebutuhan dilakukan untuk mengetahui fungsi dan batasan sistem yang akan dibangun[10]. Tujuan utama sistem adalah untuk memberikan prototipe pemantauan ketinggian air sungai yang dapat diandalkan:

1. Secara berkala mengukur jarak antara sensor dan permukaan air,
2. Membagi kondisi air menjadi status aman dan bahaya berdasarkan nilai ambang batas tertentu, menampilkan status lokal melalui indikator LED,
3. Mengirimkan data pengukuran ke komputer untuk disimpan dalam bentuk dataset,
4. Secara otomatis mengirimkan pesan peringatan ke pengguna melalui aplikasi Telegram ketika kondisi bahaya terdeteksi.

Kebutuhan nonfungsional meliputi biaya yang relatif rendah, pemeliharaan sistem yang mudah, kompatibilitas dengan komponen yang umum di lingkungan pendidikan, dan kemudahan replikasi oleh peneliti lain[11]. Analisis persyaratan ini menentukan teknologi dan desain arsitektur[12].

2.2 Perancangan Arsitektur Sistem

Aliran data dan fungsi setiap komponen sistem dijelaskan dengan pendekatan berlapis saat membangun arsitektur sistem. Secara umum, sistem terdiri dari tiga lapisan utama.

1. Lapisan pertama adalah akuisisi data, yang melibatkan mikrokontroler Arduino Uno dan sensor ultrasonik HC-SR04. Lapisan ini memproses dan menyimpan data, mengkonversi ketinggian air menjadi jarak dengan menggunakan waktu tempuh gelombang ultrasonik.
2. Lapisan kedua adalah pemrosesan dan penyimpanan. Di sini, komputer atau Raspberry Pi memproses dan mengelompokkan data dari Arduino melalui komunikasi serial.
3. Lapisan layanan informasi yang menggunakan bot Telegram untuk menyampaikan peringatan dini kepada pengguna. Pada tahap ini, program Python menggunakan API Telegram untuk mengirimkan pesan peringatan ketika nilai jarak menunjukkan kondisi bahaya.

2.3 Metode Pengembangan Sistem

Metode pengembangan sistem model prototyping merupakan pendekatan pengembangan sistem, di mana pengembang membuat model awal yang sederhana dari sistem untuk menentukan bentuk dan fungsinya. Prototipe ini kemudian diuji dan dievaluasi oleh pengguna, memungkinkan pengembangan untuk melakukan perbaikan dan peningkatan secara bertahap sesuai dengan kebutuhan spesifik mereka[13]. Metode pengembangan yang digunakan dalam penelitian ini.

1. Pengumpulan kebutuhan: mengidentifikasi fungsi dan spesifikasi sistem sebagaimana dijelaskan pada Bab 2.1.
2. Perancangan cepat: membuat rancangan awal arsitektur sistem, rangkaian perangkat keras, dan alur kerja perangkat lunak,
3. Pembuatan prototipe: membuat rancangan dalam bentuk rangkaian Arduino dengan sensor ultrasonik dan LED, serta menulis program Python dan Arduino .
4. Evaluasi prototipe

2.4 Implementasi Perangkat Keras

Implementasi Perangkat Keras Sistem Perangkat keras terdiri dari:

1. Unit pemroses Arduino Uno R3,
2. Sensor ultrasonik HC-SR04 untuk mengukur jarak antara sensor dan permukaan air,
3. Dua LED (merah dan kuning) sebagai indikator visual status bahaya dan aman,
4. Dua resistor 1 kΩ untuk membatasi arus LED,
5. Sumber daya 5 V dari adaptor atau port USB. Untuk memastikan bahwa koneksi pin dan logika dasar sudah benar, rangkaian awal disimulasikan menggunakan Tinkercad. Rangkaian ini kemudian diuji secara fisik pada breadboard.

2.5 Implementasi Perangkat Lunak

Implementasi perangkat lunak melibatkan dua komponen utama, yaitu program Arduino dan program Python pada komputer.

1. Program Arduino

Program Arduino dapat ditulis menggunakan IDE Arduino menggunakan C/C++. Dalam fungsi *setup*, *pin trigger*, *echo*, dan LED diaktifkan, dan komunikasi serial diaktifkan pada baud rate 9600 bps[14]. Pada fungsi *loop*, Arduino mengirimkan pulsa trigger selama 10 mikrodetik, menggunakan *pulseIn* untuk mengukur durasi pantulan sinyal pada *pin echo*, dan kemudian menggunakan persamaan untuk menghitung jarak:

$$\text{jarak} = \frac{\text{durasi}/2}{29,1} \dots\dots\dots (1)$$

Selanjutnya, jarak diukur dengan ambang batas, misalnya 1 meter. Jika jarak kurang dari ambang batas dan masih dalam rentang pengukuran yang sah, sistem menganggap ada bahaya dan menyalakan LED merah dan mematikan LED kuning. Sistem kemudian mengirimkan pesan *ALERT_BANJIR;jarak;m* melalui serial. Selain itu, untuk memudahkan pencatatan dataset, Arduino juga mengirimkan data berkala dalam format data, *jarak,m*.

2. Program python dan integrasi bot telegram

Program Python dijalankan dengan integrasi bot Telegram komputer atau Raspberry Pi yang terhubung ke *port serial* Arduino [15]. Setelah mengidentifikasi apakah data bertipe DATA atau *ALERT_BANJIR*, program memproses setiap baris data dalam *serial*. Nilai jarak dan status aman atau bahaya diubah menjadi data, dan kemudian disimpan ke dalam berkas *log_banjir.csv* dan *timestamp* waktu sistem. Data *ALERT_BANJIR*: Memicu pengiriman pesan peringatan ke pengguna melalui API Telegram dengan *token* bot dan *chat_id* yang telah dikonfigurasi. Pesan peringatan menunjukkan waktu peristiwa dan jarak sensor ke air.

2.6 Pengujian dan Pengumpulan Data

Pengujian sistem dilakukan untuk memastikan bahwa setiap komponen bekerja sesuai dengan fungsinya. Setiap bagian sistem diuji untuk memastikan bahwa mereka melakukan fungsinya dengan benar[16]. Metode pengujian adalah sebagai berikut: Pertama, uji sensor dengan menjauhkannya dari air atau objek pantul, kemudian bandingkan hasilnya dengan penggaris manual. Selanjutnya, uji indikator LED untuk mengetahui apakah LED kuning menyala dalam kondisi aman dan LED merah menyala dalam kondisi yang tidak aman. Terakhir, uji pencatatan data.

Tabel 1. Skenario Uji Coba.

No	Kondisi Ketinggian Air	Jarak Sensor-Air (M)	Status yang Diharapkan	LED yang Menyala	Notifikasi Telegram
1	Air rendah	≥ 7	Aman	LED kuning	Tidak
2	Air mendekati ambang	3 – 4	Aman (siaga)	LED kuning	Tidak
3	Air melampaui ambang	≤ 1	Bahaya	LED merah	Ya
4	Di luar jangkauan sensor	< 0 atau > 7	<i>Out of range</i>	Tidak pasti	Tidak

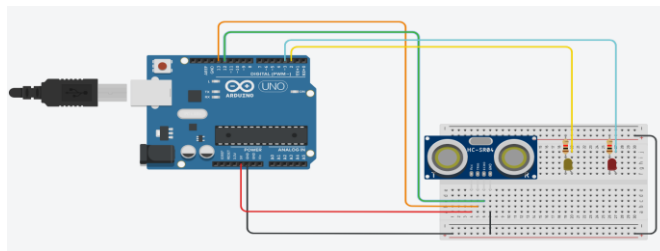
3. HASIL DAN PEMBAHASAN

Pada bagian Ini menunjukkan hasil dari implementasi dan pengujian sistem pemantauan ketinggian air sungai yang telah dibangun berbasis *Internet of Things* (IoT). Hasil yang ditampilkan meliputi kinerja sensor ultrasonik yang dibaca, perubahan status indikator LED terhadap variasi ketinggian air, pencatatan data ke dalam

dataset, dan waktu tanggap pesan melalui aplikasi Telegram. Selanjutnya, temuan kuantitatif dan kualitatif dibahas secara teoritis. Hasil ini dibandingkan dengan spesifikasi rancangan dan penelitian yang relevan untuk mengevaluasi keakuratan dan keefektifan sistem sebagai model peringatan dini banjir.

3.1 Hasil Implementasi Sistem

Hasilnya Adalah membuat rencana untuk mengimplementasikan protokol pengawasan debit air. *Breadboard* ini termasuk rangkaian pendukung, mikrokontroler Arduino Uno, dan sensor ultrasonik HC-SR04 yang diletakkan tegak di atas permukaan debit air. Arduino sering menggunakan lampu kilat untuk menunjukkan status deteksi air. Lampu kilat kuning menunjukkan semua aman, dan lampu kilat merah menunjukkan semua berbahaya. Setelah data diterima dari Arduino melalui komunikasi *serial*, gunakan program Python di komputer untuk menyimpan data ke dalam berkas *log_banjir.csv*. Waktu pengukuran, nilai jarak dalam meter, dan status aman atau bahaya disertakan dalam setiap baris data. Saat Arduino mengirimkan pesan *ALERT_BANJIR*, bot Telegram akan mengirimkan notifikasi ke akun atau grup tertentu. Ini langsung menunjukkan kepada pengguna tentang kondisi penting dengan waktu kejadian dan jarak air dari sensor.



Gambar 4. Arsitektur Sistem IoT

Gambar 4 merupakan gambar jangkauan sistem Keras untuk penilaian kualitas udara, yang terdiri dari Arduino Uno dan sensor ultrasonik HC-SR04. Sensor ultrasonik terhubung ke Arduino melalui pin Trig dan Echo untuk mengirim dan menerima sinyal ultrasonik. Fungsi utama Arduino adalah memproses sinyal pantulan untuk mengurangi tekanan udara. Hasil pengukuran jarak ini kemudian digunakan sebagai dasar untuk menentukan status kualitas udara dalam sistem pemantauan banjir.



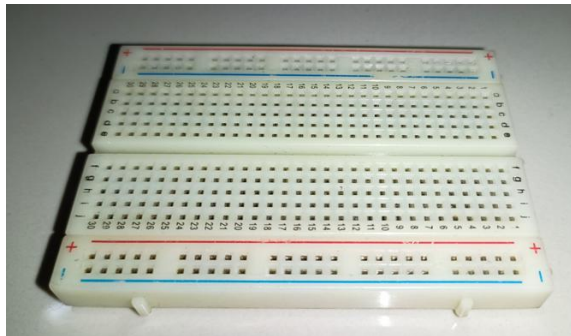
Gambar 5. Arduino Uno dan ESP

Gambar 5 merupakan gambar Arduino Uno yang terhubung ke modul ESP8266 sebagai perangkat komunikasi IoT. Tujuan utama Arduino Uno adalah untuk membaca dan memproses data dari sensor kualitas udara. Modul ESP8266 digunakan untuk mengirimkan hasil pengukuran data ke sistem atau aplikasi melalui koneksi internet. Kabel USB berfungsi sebagai sumber daya sekaligus media untuk pemrograman Arduino. Kombinasi perangkat ini memungkinkan pemantauan banjir secara real-time dan integrasi dengan pemberitahuan daring.



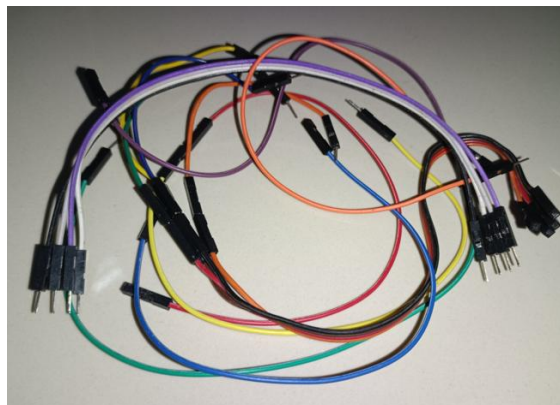
Gambar 6. Sensor Ultrasonik

Gambar 6 merupakan gambar modul sensor ultrasonik HC-SR04. Ini adalah sensor non-kontak yang umumnya digunakan dalam proyek elektronik seperti robotika. Sensor ini bekerja dengan mendeteksi gelombang ultrasonik (melalui pin Trig) dan mengukur waktu yang dibutuhkan untuk pantulan gelombang tersebut kembali (melalui pin Echo). Waktu tempuh (Time of Flight) kemudian digunakan untuk mengukur jarak ke objek yang dimaksud, dengan hasil yang akurat dan tepat dalam situasi saat ini.



Gambar 7. *BreadBoard*

Gambar 7 merupakan gambar Breadboard (Papan Roti) yang berfungsi sebagai prototipe. Perangkat ini memungkinkan pengoperasian komponen elektronik (seperti resistor, IC, dan kabel) tanpa perlu melepas solder. Lubang-lubang di tengah (area kerja/area komponen) terhubung secara vertikal, sedangkan jalur di tepi (jalur daya/power rails) terhubung secara horizontal, ditandai dengan garis merah (+) dan biru (-). Sebelum sirkuit akhir dibuat, papan roti sangat penting untuk mengevaluasi dan memperbaiki desain.



Gambar 8. Kabel *Jumper*

Gambar 8, yaitu kabel *jumper*, merupakan konektor fleksibel untuk perangkat elektronik. Fungsi utamanya adalah menghubungkan komponen, sensor, dan modul (seperti yang ada di breadboard) ke mikrokontroler (seperti Arduino) atau perangkat serupa lainnya. Kabel ini sering tersedia dalam konfigurasi Male-to-Male, Male-to-Female, atau Female-to-Female dan sangat penting untuk prototipe cepat karena kemudahan modifikasi koneksinya.



Gambar 9. LED

Gambar 9 ini menggambarkan LED (*Light Emitting Diode*), jenis semikonduktor yang memancarkan cahaya saat terkena arus listrik. LED adalah komponen polaritas, artinya memiliki kutub positif (Anoda, kaki lebih panjang) dan negatif (Katoda, kaki lebih pendek), sehingga perlu digunakan dengan hati-hati. Komponen ini cukup populer karena efisiensi energinya, umur pakainya yang panjang, dan ukurannya yang kecil, sehingga menjadi standar dalam proyek elektronik, pencahayaan, dan indikator.

3.2 Hasil Pengujian Pembacaan Jarak

Pengujian pembacaan jarak dilakukan dengan mengubah ketinggian air sesuai skenario pada Tabel 1 dan membandingkan hasil pembacaan sensor dengan pengukuran manual menggunakan penggaris.

Tabel 2. Tabel Hasil Pengujian Pembacaan Jarak

No	Jarak Manual (M)	Jarak Sensor (M)	Galat (M)	Status Sistem
1	7	6,7	0,3	Aman
2	6	5,4	0,4	Aman
3	4	4,8	0,4	Aman
4	1	0,9	0,3	Bahaya
5	<1	<1	0,4	Bahaya

Nilai galat pembacaan sensor terhadap jarak manual relatif kecil, kurang dari 1 meter untuk seluruh sampel, seperti yang ditunjukkan dalam Tabel 2. Hal ini menunjukkan bahwa sensor *ultrasonik* HC-SR04 cukup akurat untuk mengukur jarak antara sensor dan permukaan air dalam jangkauan pengukuran yang digunakan dalam penelitian. Grafik hubungan waktu dan jarak secara kualitatif menunjukkan bahwa nilai jarak menurun seiring naiknya permukaan air dan kemudian meningkat kembali seiring penurunan permukaan air. Sistem mengklasifikasikan kondisi sebagai bahaya di daerah di mana titik data berada di bawah garis ambang batas 10 meter.

3.3 Hasil Pengujian Notifikasi Telegram

Menaikkan permukaan air di atas ambang batas memungkinkan Arduino mengirimkan pesan *ALERT_BANJIR* untuk menguji notifikasi Telegram. Waktu terjadinya kondisi bahaya dicatat dari tampilan *serial monitor*, sedangkan waktu penerimaan pesan dicatat dari timestamp aplikasi Telegram. Waktu tanggap pengiriman pesan selama percobaan berkisar antara 1 hingga 4 detik, dengan rata-rata sekitar 2 hingga 3 detik. Perbedaan waktu respon terutama dipengaruhi oleh kualitas koneksi internet pada komputer yang menjalankan program Python dan perangkat pengguna. Tetapi rentang waktu ini masih cukup cepat untuk digunakan sebagai metode peringatan dini di masyarakat. Selain itu, pengujian menunjukkan bahwa pesan peringatan hanya dikirim satu kali saat terjadi perubahan status dari aman menjadi bahaya. Ini sesuai dengan logika program yang menggunakan variabel status, sehingga sistem tidak mengirimkan pesan lagi selama kondisi bahaya berlangsung. sehingga, pengguna tidak terganggu oleh banyak notifikasi yang sama dan dapat berkonsentrasi pada satu peringatan yang paling penting sehingga dapat mengevakuasi diri.



Gambar 10. Notifikasi Telegram

Gambar 10 ini adalah notifikasi Telegram yang dikirim secara otomatis dengan kode “*ALERT_BANJIR*” ketika sensor mendeteksi bahwa debit air telah mencapai tingkat yang ditentukan, sistem akan mengirimkan sebuah notifikasi atau pesan tersebut. Sebagai hasilnya, sistem ini mendukung konsep Internet of Things (IoT) dengan menggunakan sensor untuk mengumpulkan data, memprosesnya, dan menyediakan data *real-time* kepada pengguna.

3.4 Pembahasan

Hasil pengujian dan implementasi menunjukkan bahwa sistem yang dirancang dapat menjalankan dua tugas utama: memantau ketinggian air sungai secara *real-time* dan mengirimkan peringatan melalui aplikasi Telegram. Sensor ultrasonik HC-SR04 dapat mengamati perubahan ketinggian air dengan akurasi yang cukup tinggi pada skala prototipe dengan jarak pembacaan kurang dari 1 meter. Pencatatan kumpulan data sangat memungkinkan

proses analisis tambahan terhadap pola perubahan ketinggian air, seperti mengamati frekuensi dan durasi kondisi bahaya. Fungsi ini membantu kebutuhan penelitian dan evaluasi kebijakan penanggulangan banjir di masa depan. Dibandingkan dengan sistem yang hanya menggunakan indikator lokal, integrasi bot Telegram meningkatkan nilai. Selama pengguna telah terhubung ke internet, maka orang yang jauh dari sensor dapat mengakses informasi penting tentang kondisi. Sehingga Sistem ini dapat digunakan oleh masyarakat untuk mengantisipasi banjir dengan sangat cepat. Selama implementasi di lapangan, perlu dipertimbangkan penggunaan sumber daya mandiri dan jalur komunikasi alternatif karena sistem masih bergantung pada jaringan listrik dan internet.

4. SIMPULAN

Ini adalah hasil dari perancangan, implementasi, dan pengujian.

1. Sensor ultrasonik HC-SR04 dan Arduino Uno telah digunakan untuk membuat prototipe sistem pemantauan ketinggian air sungai berbasis *Internet of Things* (IoT). Sensor ini dapat mengukur jarak permukaan air secara berkala dan menampilkan status aman dan bahaya melalui indikator LED.
2. Sistem juga dapat mencatat hasil pengukuran ke dalam dataset CSV yang mengandung waktu pengukuran, nilai jarak, dan status kondisi air, yang kemudian dikirim ke gambaran pol
3. Dengan menggunakan bot Telegram dan dapat secara otomatis mengirimkan peringatan dini ketika ketinggian air melampaui ambang batas.
4. Prototipe yang baik adalah sistem peringatan dini karena mengandung informasi tentang waktu kejadian dan jarak air ke sensor, sehingga notifikasi dikirim lebih cepat dibandingkan dengan pemantauan manual di lapangan. Namun, protokol memiliki beberapa keterbatasan. Ini mencakup jumlah titik pemantauan yang terpisah yang bergantung pada koneksi internet dan sumber daya listrik. Selain itu, antarmuka visualnya tidak terintegrasi secara langsung dalam aplikasi yang memanfaatkan peta atau grafik. Keterbatasan ini membuka peluang untuk penelitian yang lebih luas.

5. SARAN

Berikut adalah beberapa rekomendasi untuk pengembangan penelitian selanjutnya.

1. Agar data ketinggian air yang diperoleh lebih rinci dan representatif, tambahkan jumlah sensor dan tempatkan titik pemantauan di sepanjang aliran sungai.
2. Menggabungkan sistem dengan sumber daya independen, seperti panel surya yang dipadukan dengan baterai, memungkinkan perangkat IoT tetap beroperasi di lapangan tanpa bergantung pada pasokan listrik PLN.
3. merancang antarmuka visual berbasis web dan aplikasi mobile yang memungkinkan pengguna melihat perubahan ketinggian air secara *real-time* dengan grafik.

DAFTAR PUSTAKA

- [1] A. Kurniadi, "Future projections of extreme rainfall events in Indonesia," no. June 2023, pp. 160–182, 2024, doi: 10.1002/joc.8321.
- [2] B. A. Adnan, "Tropical forest conservation efforts as climate change mitigation in Indonesia : A Review," vol. 1, no. 2, pp. 80–89, 2023.
- [3] N. V. Tamara, J. T. Informatika, and F. Teknik, "Sistem Ketinggian Air dan Pendeteksi Dini Banjir menggunakan Wemos D1 Mini Berbasis Esp8266 dengan Notifikasi Telegram dan Whatsapp," vol. 15, no. 1, pp. 39–46, 2024.
- [4] M. Charir, A. Ghifari, M. R. Z. Abidin, and U. Y. Pasuruan, "SISTEM MONITORING TINGGI AIR SUNGAI DAN MENGANTISIPASI BANJIR," vol. 6, no. 2, 2025.
- [5] A. W. Tiyas, D. Erwanto, I. Yanuartanti, T. Elektro, F. Teknik, and U. I. Kadiri, "Peningkatan Akurasi Sensor Suhu dan Kelembaban DHT11 dengan Kalibrasi Suhu Berbasis IoT pada Platform Thingspeak Improved Accuracy of DHT11 Temperature and Humidity Sensor with IoT-Based Temperature Calibration on the Thingspeak Platform," vol. 5, no. 3, pp. 625–633, 2025.
- [6] J. Informatika, Z. M. Arrasyidy, A. Hallim, and A. Rahim, "Perancangan Sistem Deteksi Banjir Berbasis IoT Untuk Peringatan Dini dengan Sistem Integrasi Web".
- [7] H. S. Jurnal, Y. Ratmini, V. Atina, and E. Purwanto, "Jurnal Ilmiah Teknologi Informasi Asia Sistem monitoring dan peringatan dini banjir berbasis Internet of Things (IoT)," vol. 19, pp. 1–8, 2025.
- [8] D. Fernanda and A. A. Syukron, "Implementasi Sensor Ultrasonic Untuk Pemantauan Debit Sungai Serayu Sebagai Antisipasi Banjir Berbasis Arduino JURNAL MEDIA INFORMATIKA [JUMIN]," vol. 6, no. 3, pp. 1920–1929, 2025.
- [9] E. Flood, W. System, I. N. Loireng, and S. District, "No Title," vol. 8, no. 2022, pp. 1303–1313, 2025.
- [10] S. Kasus *et al.*, "Analisis Kebutuhan dan Perancangan Sistem Informasi Seleksi Perangkat Desa pemerintahan desa yang baik . Sayangnya , pendekatan manual yang masih banyak digunakan Analisis

- Kebutuhan Sistem lunak yang bertujuan untuk mengenali dan memahami kebutuhan pengguna akhir . Seperti Model Waterfall dalam Pengembangan Sistem Model Waterfall adalah metode pengembangan sistem yang berurutan dan sistematis , di mana setiap tahapan harus diselesaikan sebelum melangkah ke tahap,” 2025.
- [11] M. E. Ortiz *et al.*, “Development of Low-Cost IoT System for Monitoring Piezometric Level and Temperature of Groundwater,” pp. 1–16, 2023.
- [12] S. D. Brito, G. J. Azinheira, J. F. Semião, N. M. Sousa, and S. P. Litrán, “Non-Intrusive Low-Cost IoT-Based Hardware System for Sustainable Predictive Maintenance of Industrial Pump Systems,” pp. 1–34, 2025.
- [13] V. Issue, S. Aprilisa, and R. Aulia, “JUTIN : Jurnal Teknik Industri Terintegrasi Penerapan Metode Prototype dalam Pengembangan Sistem Informasi Inventory Barang Berbasis Web,” vol. 7, no. 1, pp. 333–340, 2024.
- [14] E. S. Rahman, N. Almunawati, Y. Tjandi, A. M. Mappalotteng, and A. Imran, “DESIGN , IMPLEMENTATION , AND EVALUATION OF AN ELECTRICAL ENERGY CONSUMPTION,” vol. 2, no. 2, pp. 92–98, 2025.
- [15] G. W. Marshal and R. Soelistijadi, “ChatBot SPBU 44 . 501 . 01 Menggunakan Telegram Bot API Dengan Metode Long-Polling,” vol. 17, no. 2, pp. 125–132, 2023.
- [16] M. K. Rosidin, S. Ilman, A. Farhan, C. A. Pradhana, C. Irawan, and B. Dharmawan, “Rancang Bangun Alat Pendeteksi Ketinggian Air Banjir Berbasis Arduino Uno,” vol. 4, no. 1, pp. 51–57, 2023.