

APLIKASI BERBASIS WEB UNTUK MONITORING KETINGGIAN AIR DAN POMPA MENGGUNAKAN MIKROKONTROLER NODEMCU ESP8266

Hariz Pambudi¹, Subandi^{2*}, Utomo Budiyanto³, Indra⁴

^{1,2,3,4} Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur, DKI Jakarta, Indonesia

Email: ¹1811500659@student.budiluhur.ac.id, ^{2*}subandi@budiluhur.ac.id, ³utomo.budiyanto@budiluhur.ac.id, ⁴indra@budiluhur.ac.id
(* : corresponding author)

Abstrak- Banjir terjadi karena daya tampung saluran air melebihi kapasitas daya tampungnya, sehingga menyebabkan air menggenang dan menyebabkan banjir. Akibatnya, banjir dapat mempengaruhi kehidupan serta menyebabkan kerugian seperti dokumen penting hilang, korsleting listrik, kerusakan kendaraan, bahkan dapat menimbulkan korban jiwa. Pada penelitian ini menggunakan metode *prototype*, yang dimana metode *prototype* adalah metodologi pengembangan sistem untuk membuat desain secara cepat dan bertahap sehingga dapat dievaluasi dan diimplementasikan. Tujuan dari penelitian ini adalah untuk membuat prototipe sistem monitoring ketinggian air secara *online* dan *real-time* untuk memberikan informasi dini apabila sistem mendeteksi batas jarak dan ketinggian air dengan cara mengirimkan notifikasi lewat *web* dengan memanfaatkan *Internet of Things* (IoT), serta menggunakan LCD, LED, dan *buzzer* untuk peringatan secara langsung. Oleh sebab itu, dibutuhkan sistem untuk monitoring dan mendeteksi ketinggian air yang dibantu menggunakan mikrokontroler NodeMCU dengan kemampuan untuk terhubung ke jaringan Wi-Fi dan digunakan sebagai sistem kendali pada alat. Kemudian ditambahkan sensor untuk mendukung pembuatan prototipe ini menggunakan sensor ultrasonik HC-SR04 dan *water level sensor*. Sensor ultrasonik HC-SR04 digunakan untuk menghitung jarak air dan *water level sensor* sebagai alat pemberitahuan untuk menampilkan ketinggian air. Jika air sudah dalam jarak tertentu maka LED dan *buzzer* akan menyala dan bila air sudah menyentuh batas ketinggian air maka water pump akan menyala kemudian membuang air. Hasil pada pengujian yang dilakukan adalah aplikasi monitoring ketinggian air dapat menjalankan perintah sesuai dengan yang diinginkan. Status air rendah terjadi bila jarak \geq batas rendah, status air sedang terjadi bila jarak $<$ batas rendah dan jarak $>$ batas tinggi, status air tinggi terjadi bila jarak $<$ batas tinggi, dan pompa akan menyala bila nilai ketinggian \geq batas ketinggian.

Kata Kunci: Aplikasi Berbasis Web, Monitoring Ketinggian Air dan Pompa, Mikrokontroler NodeMCU, HC-SR04, *Water Level Sensor*.

WEB-BASED APPLICATION FOR MONITORING WATER LEVEL AND PUMP USING NODEMCU ESP8266 MICROCONTROLLER

Abstract- Flooding occurs because the capacity of waterways exceeds their capacity, causing water to stagnate and cause flooding. As a result, flooding can affect life and cause losses such as lost important documents, electrical short circuits, vehicle damage, and can even cause fatalities. In this research using the prototype method, where the prototype method is a system development methodology for making designs quickly and gradually so that they can be evaluated and implemented. The purpose of this research is to create a prototype of an online and real-time water level monitoring system to provide early information when the system detects distance and water level limits by sending notifications via the web by utilizing the Internet of Things (IoT), and using LCD, LED, and buzzer for immediate warning. Therefore, a system for monitoring and detecting water levels is needed which is assisted using a NodeMCU microcontroller with the ability to connect to a Wi-Fi network and is used as a control system for the device. Then added sensors to support the making of this prototype using HC-SR04 ultrasonic sensors and water level sensors. The HC-SR04 ultrasonic sensor is used to calculate the water distance and the water level sensor as a notification tool to display the water level. If the water is within a certain distance, the LED and buzzer will turn on and if the water has touched the water level limit, the water pump will turn on and then remove the water. The results of the tests carried out are that the water level monitoring application can carry out the commands as desired. Low water status occurs when the distance \geq low limit, medium water status occurs when the distance $<$ low limit and distance $>$ high limit, high water status occurs when the distance $<$ high limit, and the pump will turn on when the height value \geq height limit.

Keywords: Web-based application, water level monitoring and pumping, NodeMCU microcontroller, HC-SR04, water level sensor.

1. PENDAHULUAN

Banjir merupakan salah satu bencana alam paling sering terjadi di Indonesia, karena Indonesia merupakan kawasan beriklim tropis yang memiliki dua musim, yaitu musim kemarau dan hujan. Pada saat hujan, air terus menerus meningkat yang dapat menyebabkan banjir di beberapa tempat. Banjir terjadi karena daya tampung

saluran air melebihi kapasitas daya tampungnya, sehingga menyebabkan air menggenang di sekitar saluran dan menyebabkan banjir. Akibatnya, banjir dapat mempengaruhi kehidupan serta menyebabkan kerugian seperti dokumen penting hilang, konsleting listrik, kerusakan kendaraan, bahkan dapat menimbulkan korban jiwa. Pada tahun 2018, riset yang dilakukan oleh Achmad Muzakky, Akhmad Nurhadi, Ashuri Nurdiansyah, Galih Wicaksana, Istiadi membuat Perancangan Sistem Deteksi Banjir Berbasis IoT. Riset ini adalah untuk memantau ketinggian air secara online sebagai informasi awal jika terjadi banjir. Hasil dari penelitian ini adalah sebuah sistem deteksi ketinggian air yang dapat memberikan notifikasi, peringatan, peringatan, dan pemberitahuan tentang tingkat keamanan, maka dari itu sistem deteksi ini digunakan untuk informasi awal tentang terjadinya banjir [1].

Pada tahun 2020, riset yang dilakukan oleh Nicko Pratama, Ucu Darusalam, Novi Dian Nathasia membuat Perancangan Sistem Monitoring Ketinggian Air Sebagai Pendeteksi Banjir Berbasis IoT Menggunakan Sensor Ultrasonik. Pada riset ini, membuat sistem mendeteksi jarak antara sensor dengan permukaan air, ketika jarak permukaan air mendekati sensor maka akan terdengar sirine sebagai tanda air tinggi dan sistem menunjukkan status siaga dan jarak ke ketinggian air secara *real-time* [2]. Pada tahun 2019, riset yang dilakukan oleh Shania Putri Windiastik, Elsha Novia Ardhana, Joko Triono membuat Perancangan Sistem Pendeteksi Banjir Berbasis IOT (*Internet Of Thing*). Pada riset ini, mengacu pada jaringan yang menghubungkan beberapa perangkat fisik seperti *water level sensor* untuk mendeteksi ketinggian air, menggunakan NodeMCU ESP8266 untuk membaca data kemudian mengirimkannya melalui internet, dan *buzzer* mengeluarkan suara saat sensor mendeteksi air dan kemungkinan terjadi banjir [3].

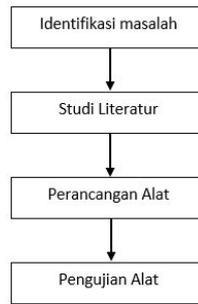
Untuk mengantisipasi kejadian diatas, diperlukan aplikasi monitoring air untuk memberikan peringatan menggunakan sistem komunikasi *wireless* untuk memberikan informasi kepada *user*. Seiring perkembangan zaman, konsep *Internet of Things* atau lebih dikenal dengan nama IoT yang merupakan sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. IoT dapat memberikan dampak positif bagi perkembangan beberapa teknologi saat ini, termasuk mempermudah sistem pengendalian. Oleh sebab itu, dibutuhkan sistem untuk monitoring dan mendeteksi ketinggian air dengan memanfaatkan IoT dan dibantu menggunakan mikrokontroler NodeMCU.

NodeMCU adalah *platform IoT open source*. NodeMCU merupakan mikrokontroler seperti Arduino dengan kemampuan untuk terhubung ke jaringan Wi-Fi. NodeMCU dapat mengirimkan data secara *wireless* yang akan dikoneksikan ke *website* untuk memonitoring bila terjadi luapan air. Data yang diterima kemudian dikirim ke *database* untuk menyimpan data yang didapat dari sensor ke *website* yang dapat memonitoring dan mendeteksi ketinggian air. Untuk mendukung aplikasi monitoring air, digunakan sensor ultrasonik untuk menentukan jarak suatu benda yang ada di depannya dengan tingkat akurasi yang tinggi serta deteksi yang stabil yaitu sensor ultrasonik HC-SR04. Sensor ultrasonik HC-SR04 merupakan sensor yang mempunyai fungsi mendeteksi ketinggian air dan menggunakan gelombang suara sebagai pengirim serta penerima data sehingga terbaca suatu jarak.

Serta menggunakan *water level sensor* yang berfungsi mendeteksi ketinggian air menggunakan *output analog* kemudian data yang diterima diolah dengan NodeMCU. *Water level sensor* bekerja dengan cara menggunakan konduktivitas air untuk mendeteksi ketinggian air dalam bentuk sinyal *analog* sebelum diproses di NodeMCU. *Water pump* digunakan untuk membuang air jika *water level sensor* sudah mencapai nilai yang telah ditentukan. Berdasarkan latar belakang di atas, pada penelitian ini dirancang “aplikasi monitoring ketinggian air dan pompa menggunakan Mikrokontroler Nodemcu ESP8266 berbasis *web*”.

2. METODE PENELITIAN

Pada penelitian ini menggunakan metode *prototype*, yang dimana metode *prototype* adalah metodologi pengembangan sistem untuk membuat desain secara cepat dan bertahap sehingga dapat dievaluasi dan diimplementasikan. Metode *prototype* ini adalah untuk mengembangkan rancangan desain produk sehingga menjadi produk akhir yang memenuhi kebutuhan dan permintaan pasar. Adapun tahap penelitian yang dapat dilihat pada Gambar 1 adalah sebagai berikut.



Gambar 1. Metode Penelitian

2.1 Identifikasi Masalah

Pada tahap ini dilakukan pengenalan suatu masalah dan tahap awal dalam proses penelitian. Permasalahan penelitian ini adalah pendeteksi dan pemberitahuan secara *online* dan *real-time* ketika sistem mendeteksi batas jarak dan ketinggian air menggunakan NodeMCU ESP8266 dibantu sensor ultrasonik HC-SR04 dan *water level sensor*.

2.2 Studi Literatur

Pada tahun 2018, riset yang dilakukan oleh Achmad Muzakky, Akhmad Nurhadi, Ashuri Nurdiansyah, Galih Wicaksana, Istiadi untuk memantau ketinggian air secara *online* sebagai informasi awal jika terjadi banjir [1]. Pada tahun 2020, riset yang dilakukan oleh Nicko Pratama, Ucu Darusalam, Novi Dian Nathasia membuat sistem mendeteksi jarak antara sensor dengan permukaan air, ketika jarak permukaan air mendekati sensor maka akan terdengar sirine sebagai tanda air tinggi dan sistem menunjukkan status siaga dan jarak ke ketinggian air secara *real-time* [2]. Pada tahun 2019, riset yang dilakukan oleh Shania Putri Windiastik, Elsha Novia Ardhana, Joko Triono mengacu pada jaringan yang menghubungkan beberapa perangkat fisik seperti *water level sensor* untuk mendeteksi ketinggian air, menggunakan NodeMCU ESP8266 untuk membaca data kemudian mengirimkannya melalui internet, dan *buzzer* mengeluarkan suara saat sensor mendeteksi air dan kemungkinan terjadi banjir [3].

Pada tahun 2020, riset yang dilakukan oleh Misnawati, Mardhiyah Nas, Fadlia, Megha Rahmawaty Marsing bertujuan untuk mewujudkan alat pemantau ketinggian air di bendungan dengan menggunakan sensor ultrasonik dan sistem IoT yang didukung beberapa komponen elektronik lainnya antara lain mikrokontroler NodeMCU, sensor ultrasonik, pompa air, *relay* dan *buzzer* [4]. Pada tahun 2022, riset yang dilakukan oleh Mahmudin, Sukisno Nugroho Adiprawiro, Mochammad Fadhil Octo Kurniawan membuat prototipe sistem pendeteksi banjir yang memantau ketinggian dan aliran air yang cepat yang terhubung satu sama lain perangkat fisik yang berbeda dengan protokol masing-masing ini menggunakan *waterflow sensor* untuk menghitung kecepatan air, *water level sensor* sebagai pendeteksi ketinggian air, Arduino Uno sebagai *hub* berbagai pembacaan sensor, Nodemcu ESP8266 sebagai pemancar pembacaan sensor ke *database Firebase* [5].

Pada tahun 2022, riset yang dilakukan oleh Muhammad Rega Alfiano Setiawan, Arif Rahman Sujatmika, Winarti mengembangkan prototipe sistem pendeteksi banjir berbasis IoT dengan notifikasi Blynk, menggunakan dua buah mikrokontroler, sensor ultrasonik dan sensor ketinggian air sebagai indikator ketinggian air [6]. Pada tahun 2020, riset yang dilakukan oleh Tomy Aditya Firmansah, Kunto Eko Susilo bertujuan untuk menyederhanakan sistem kerja dalam penanggulangan banjir di Indonesia menggunakan mikrokontroler Esp32, sensor ultrasonik dan motor servo, dengan informasi yang dihasilkan oleh sensor ditampilkan oleh lampu tanda bahaya [7]. Pada tahun 2021, riset yang dilakukan oleh Fuad Dwi Hanggara, Rama Dani Eka Putra bertujuan untuk memantau ketinggian air secara online untuk menginformasikan lebih awal kapan bencana ini terjadi yang didukung oleh sensor ultrasonik HC-SR 04 dan Arduino UNO R3 yang digunakan sebagai pengolah data kemudian mengirimkan data secara nirkabel ke aplikasi ThingsSpeak untuk dibaca di *Smartphone* atau situs web [8].

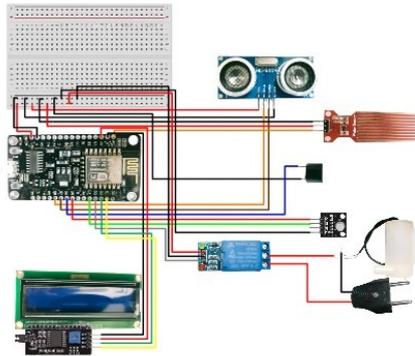
Pada tahun 2021, riset yang dilakukan oleh Ibnu Agung Deswiyani, Solikhun, Sumarno, Poningsih, Sundari Retno Andani bertujuan untuk mengetahui tinggi muka air dan bertindak sebagai peringatan sebelum banjir di masa mendatang untuk mengurangi jumlah kematian di suatu daerah atau tempat yang dapat diklasifikasikan sebagai rawan banjir [9]. Pada tahun 2019, riset yang dilakukan oleh Jonshon Tarigan, Agustinus Deka Betan bertujuan untuk merancang alat ukur untuk menentukan ketinggian banjir secara dini, dimana nilai ketinggiannya akan ditampilkan di LCD jika ketinggian banjir melebihi dari batas normal (> 60 cm) maka LED merah akan menyala dan alarm akan bunyi kemudian menampilkan data pada LCD [10].

2.3 Perancangan Alat

Pada tahap rangkaian alat ini merupakan rangkaian keseluruhan rancangan alat aplikasi monitoring air yang terdiri dari perancangan *hardware* dan *software*.

2.3.1 Perancangan *Hardware*

Pada tahap perancangan *hardware* dilengkapi dengan sensor, mikrokontroler, dan komponen output, serta komponen yang digunakan untuk mendukung alat tersebut yang dapat dilihat pada Gambar 2.

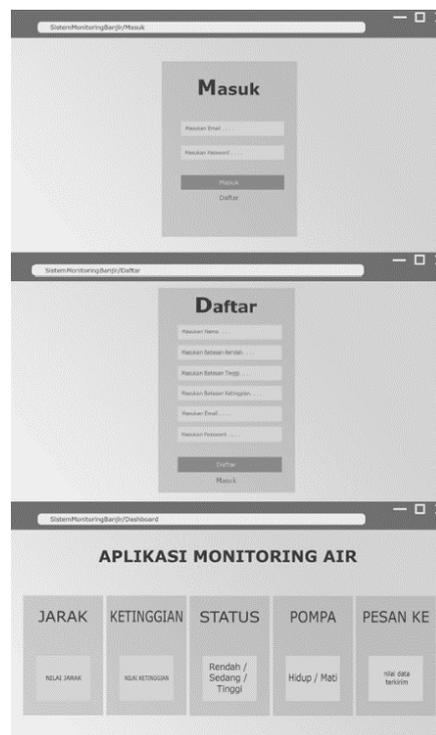


Gambar 2. Perancangan *Hardware*

Perancangan *hardware* terdiri dari NodeMCU ESP8266 sebagai sistem kendali alat, sensor ultrasonik HC-SR04 dan *water level sensor* sebagai masukan, serta LCD, LED, *buzzer*, dan *water pump* sebagai keluaran.

2.3.2 Perancangan *Software*

Pada tahap rancangan *software* merupakan tahap yang penting dalam pembuatan program karena berfungsi untuk menjembatani antara *user* dan sistem. Maka dari itu rancangan layar yang dibuat harus mudah dimengerti dan dipahami, agar *user* tidak dibuat kesulitan dan bingung saat menggunakan aplikasi monitoring air. Dibawah ini merupakan perancangan *software* yang dapat dilihat pada Gambar 3.



Gambar 3. Perancangan *Software*

Pada Gambar 3 terdapat rancangan *login*, rancangan registrasi, dan rancangan *dashboard*. Di bawah ini merupakan penjelasan dari perancangan *software*.

a. Rancangan *Login*

Pada rancangan *login* dibuat agar *user* dapat masuk ke halaman *dashboard*, *user* harus memasukan *email* dan *password* terlebih dahulu. Apabila *user* belum memiliki akun, maka *user* dapat meng-klik *button* daftar untuk membuat akun.

b. Rancangan Registrasi

Pada rancangan registrasi untuk membuat akun, *user* harus memasukan nama, batas rendah air, batas tinggi air, batas ketinggian, email, dan password kemudian klik *button* daftar agar akun dapat terbuat.

c. Rancangan *Dashboard*

Pada rancangan *dashboard*, *user* dapat memonitoring jarak dan ketinggian air secara *online* dan *real-time*. Dimana pada halaman *dashboard* terdapat nilai jarak air, nilai ketinggian air, status air, status pompa, dan nilai pesan.

2.4 Pengujian Alat

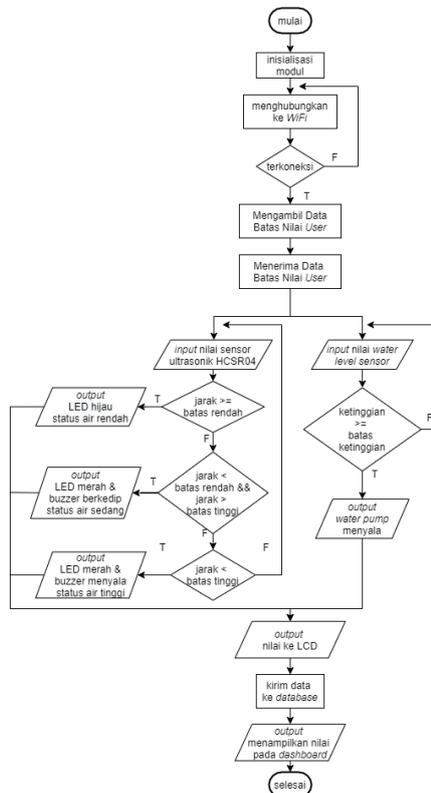
Setelah tahap perancangan alat, maka selanjutnya adalah tahap pengujian alat. Pada tahap pengujian alat bertujuan untuk mengetahui keberhasilan alat yang telah dibuat.

3. HASIL DAN PEMBAHASAN

Dibawah ini menjelaskan hasil dan dari " Aplikasi Monitoring Ketinggian Air dan Pompa Menggunakan Mikrokontroler NodeMCU ESP8266 Berbasis *Web*" yang terdiri dari *flowchart*, algoritme, tampilan perancangan alat, serta pengujian.

3.1 Flowchart

Flowchart adalah diagram yang menunjukkan langkah-langkah dan keputusan untuk menyelesaikan suatu proses pemrograman. *Flowchart* berfungsi untuk menggambarkan alur program dari satu proses ke proses lainnya agar alur program mudah dipahami. Berikut merupakan *flowchart* aplikasi monitoring air yang dirancang dalam penelitian ini pada Gambar 4.



Gambar 4. Flowchart Alat

Pada tahap pertama dilakukan inisialisasi modul terlebih dahulu kemudian mengkoneksikan ke Wi-Fi. Apabila gagal terkoneksi maka akan kembali menghubungkan ke Wi-Fi sampai terkoneksi, apabila berhasil terkoneksi maka selanjutnya adalah mengambil data batas nilai rendah, tinggi, dan ketinggian pada database. Selanjutnya sensor akan meng-*input* nilai yang dibaca. Saat sensor HC-SR04 mendeteksi jarak \geq batas rendah maka output akan menampilkan LED hijau kemudian akan ditampilkan pada LCD. Jika sensor HC-SR04 mendeteksi air berjarak $<$ dari batas rendah dan $>$ batas tinggi, maka sistem akan memberikan status air sedang kemudian LED akan berwarna merah berkedip dan *buzzer* juga berkedip kemudian akan ditampilkan pada LCD. Jika sensor HC-SR04 mendeteksi air mencapai \leq batas tinggi cm, maka sistem akan memberikan status air tinggi kemudian LED akan berwarna merah berkedip dan *buzzer* juga berkedip kemudian akan ditampilkan pada LCD. Jika *water level sensor* telah mendeteksi ketinggian air yang dimana nilai ketinggian \leq batas ketinggian, maka *water pump* menyala kemudian nilai yang dikirim *water level sensor* akan ditampilkan pada LCD. Selanjutnya data yang diterima akan dikirim ke *database* untuk ditampilkan pada halaman *dashboard* pada *web*.

3.2 Algoritme

Algoritme adalah suatu urutan proses logis dan sistematis pengambilan keputusan untuk pemecahan suatu masalah. Algoritme merupakan sebuah rancangan dari *flowchart*, dimana algoritme ini dapat menjabarkan cara kerja dari program. Berikut merupakan algoritme pada aplikasi monitoring air.

Algoritme 1. Algoritme Aplikasi Monitoring Air

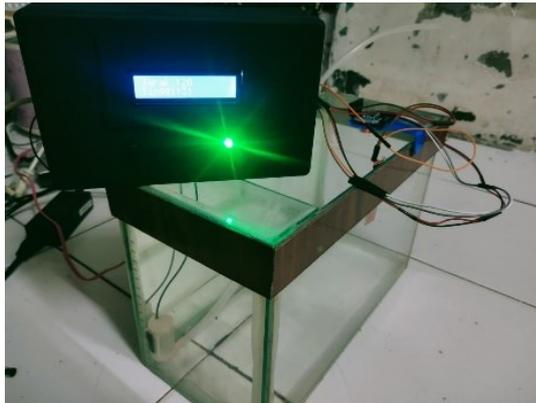
1. Mulai
2. Inisialisasi sensor ultrasonik HC-SR04
3. Inisialisasi *Water Level Sensor*
4. Inisialisasi *Buzzer*
5. Inisialisasi LED
6. Inisialisasi *Water Pump*
7. Inisialisasi LCD
8. Inisialisasi *Host*
9. *Output* LCD
10. Cek koneksi Wifi
11. Ambil data batas nilai pada *database user*
12. Inisialisasi batas nilai
13. Cetak batas rendah
14. Cetak batas tinggi
15. Cetak batas ketinggian
16. Sensor membaca jarak dan ketinggian air
17. *If* jarak \geq batas rendah *then*
18. LED hijau menyala
19. Cetak Status Air Rendah
20. *Else if* jarak $<$ batas rendah && jarak $>$ batas tinggi *then*
21. LED merah berkedip
22. *Buzzer* berkedip
23. Cetak Status Air Sedang
24. *Else if* jarak \leq batas tinggi *then*
25. LED merah menyala
26. *Buzzer* menyala
27. Cetak Status Air Tinggi
28. *Else*
29. Kembali ke baris ke 16
30. *If* ketinggian \geq batas ketinggian
31. *Water Pump* menyala
32. Cetak Pompa Menyala
33. *Else*
34. Cetak Pompa Mati
35. Kembali ke baris ke 16
36. Kirim data ke *database*
37. Data diterima *database*
38. Menampilkan nilai pada *dashboard*
39. Selesai

3.3 Tampilan Perancangan Alat

Pada tampilan aplikasi monitoring air menampilkan hasil dari rancangan pada alat dan *web*. Di bawah ini diperlihatkan hasil dari rancangan yang telah dibuat sebagai berikut.

3.3.1 Hasil Rancangan *Hardware*

Pada Gambar 5 merupakan tampilan dari hasil rancangan hardware yang terdiri dari NodeMCU, sensor Ultrasonik HC-SR04, water level sensor, LCD, buzzer, LED, relay, water pump, dan komponen lainnya sebagai pendukung yang telah disatukan, serta wadah dengan ukuran 20cm x 30cm x 15cm.



Gambar 5. Hasil Rancangan Hardware

3.3.2 Hasil Rancangan *Software*

Pada Gambar 6, 7, dan 8 merupakan tampilan dari hasil rancangan *software* yang terdiri dari halaman *login web* untuk masuk ke halaman *dashboard*, halaman registrasi *web* untuk membuat akun agar bisa *login*, dan halaman *dashboard web* untuk memonitoring ketinggian air secara *online* dan *real-time*.



Gambar 6. Halaman *Login Web*



Gambar 7. Halaman Registrasi *Web*



Gambar 8. Halaman *Dashboard Web*

3.4 Pengujian

Tahap pengujian dilakukan setelah perancangan alat dan *interface website* telah selesai dibuat. Selanjutnya dilakukan pengujian terhadap Aplikasi Monitoring Ketinggian Air Dan Pompa Menggunakan Mikrokontroler NodeMCU ESP8266 Berbasis *Web* dapat berjalan sebagaimana mestinya. Dibawah ini merupakan tahap pengujian dari Aplikasi Monitoring Ketinggian Air Dan Pompa Menggunakan Mikrokontroler NodeMCU ESP8266 Berbasis *Web*. Pada pengujian menggunakan data *user1* yang dimana data nilai rendah = 18 cm, nilai tinggi = 6 cm, dan batas ketinggian = 4 cm. berikut merupakan data dari *user1* pada gambar 4.20 .

id_user	nama	email	password	rendah	tinggi	bketinggian	status
1	user1	q@q	\$2y\$10\$OL3iM3ItGNgOQvqhleKd7OP1gWEDI9U1qK1cuYexAdP...	18	6	3	0

Gambar 9. Data *User*

3.4.1 Pengujian Sensor Ultrasonik HC-SR04

Pada tabel. 1 menampilkan pengujian pada sensor ultrasonik HC-SR04 yang dilakukan *user1* yang dimana data nilai rendah = 18 cm dan nilai tinggi = 6 cm.

Tabel 1. Pengujian Sensor Ultrasonik HC-SR04

No	Jarak Air	LED	Buzzer	LCD	Website
1	19 cm	Hijau	Mati	Jarak : 19cm	Status Air Rendah
2	18 cm	Hijau	Mati	Jarak : 18cm	Status Air Rendah
3	17 cm	Merah Berkedip	Berkedip	Jarak : 17cm	Status Air Sedang
4	10 cm	Merah Berkedip	Berkedip	Jarak : 10cm	Status Air Sedang
5	7 cm	Merah Berkedip	Berkedip	Jarak : 7cm	Status Air Sedang
6	6 cm	Merah	Menyala	Jarak : 6 cm	Status Air Tinggi
7	5 cm	Merah	Menyala	Jarak : 5 cm	Status Air Tinggi

Hasil pada pengujian yang dilakukan sensor ultrasonik HC-SR04 disimpulkan bahwa sensor dapat mendeteksi jarak pada air. Ketika sensor berada pada jarak \geq batas rendah user maka LED akan berwarna hijau, buzzer mati, LCD menampilkan nilai jarak, dan website menampilkan status air rendah. Pada jarak $<$ batas rendah dan jarak $>$ batas tinggi user maka LED akan berwarna merah berkedip, buzzer berkedip, LCD menampilkan nilai jarak, dan website menampilkan status air sedang. Pada jarak \leq batas tinggi user maka LED akan berwarna merah, buzzer menyala, LCD menampilkan nilai jarak, dan website menampilkan status air tinggi.

3.4.2 Pengujian *Water Level Sensor*

Pada Tabel 2 menampilkan pengujian pada sensor ultrasonik HC-SR04 yang dilakukan *user1* yang dimana data nilai batas ketinggian = 3 cm.

Tabel 2. Pengujian *Water Level Sensor*

No	Nilai Ketinggian	Water Pump	LCD	Website
1	0cm	Mati	Ketinggian : 0cm	Pompa Mati
2	1cm	Mati	Ketinggian : 1cm	Pompa Mati
3	2cm	Mati	Ketinggian : 2cm	Pompa Mati
4	3cm	Menyala	Ketinggian : 3cm	Pompa Hidup

Hasil pada pengujian yang dilakukan *water level sensor* disimpulkan bahwa sensor dapat mendeteksi ketinggian pada air. Ketika sensor berada pada ketinggian < batas ketinggian *user* maka *water pump* tidak menyala, LCD mencetak nilai ketinggian, dan *website* menampilkan status pompa mati. Dan ketika ketinggian \geq batas ketinggian *user* maka *water pump* menyala, LCD mencetak nilai ketinggian, dan *website* menampilkan status pompa hidup.

4. KESIMPULAN

Setelah membuat perancangan, pembuatan sistem, pengujian sistem, dan hasil yang didapatkan, maka dapat disimpulkan bahwa proses dan desain aplikasi monitoring air dapat berfungsi dengan baik, sistem dapat mendeteksi jarak dan ketinggian air, *output* seperti LCD, LED, *buzzer*, dan juga *water pump* sesuai dengan yang diharapkan, NodeMCU dapat terhubung dengan WiFi dengan baik sehingga data yang dikirim dapat diterima ke *database*, aplikasi monitoring air dapat diakses secara *online* dan *real-time*. Adapun saran yang dapat dilakukan untuk penelitian selanjutnya seperti menambahkan sensor tambahan agar dapat mendeteksi selain jarak dan ketinggian, Diharap menambahkan notifikasi agar *user* mendapat pemberitahuan apabila sedang tidak memonitoring *website*, dan diharap agar peneliti selanjutnya dapat membuat sistem berbasis android agar lebih mudah dalam memonitoring jarak dan ketinggian air.

DAFTAR PUSTAKA

- [1] A. Muzakky, A. Nurhadi, A. Nurdiansyah, and G. Wicaksana, "Perancangan Sistem Deteksi Banjir Berbasis IoT," *Conf. Innov. Appl. Sci. Technol. (CIASTECH 2018)*, vol. 7, no. 2, pp. 43–51, 2018.
- [2] N. Pratama, U. Darusalam, and N. D. Nathasia, "Perancangan Sistem Monitoring Ketinggian Air Sebagai Pendeteksi Banjir Berbasis IoT Menggunakan Sensor Ultrasonik," *J. Media Inform. Budidarma*, vol. 4, no. 1, p. 117, 2020, doi: 10.30865/mib.v4i1.1905.
- [3] S. P. Windiastik, N. Ardhana, and J. Triono, "Perancangan Sistem Pendeteksi Banjir Berbasis Iot (Internet of Thing)," *Semin. Nas. Sist. Inf.*, no. September, pp. 25–31, 2019.
- [4] Misnawati, M. Nas, Fadlia, and M. R. Marsing, "Prototipe Pemantauan Level Air Pada Bendungan Berbasis IOT," *J. Appl. SMART Electr. Netw. Syst.*, vol. Vol. 1 No., 2020.
- [5] Mahmudin, S. N. Adiprawiro, and M. F. O. Kurniawan, "Prototipe Sistem Pendeteksi Banjir Berbasis Iot (Internet Of Things)," vol. 9, no. 2, pp. 149–156, 2022.
- [6] M. R. A. Setiawan, "Prototipe deteksi banjir menggunakan sensor ultrasonik, dan water level sensor dengan notifikasi blynk," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 4, no. 2, pp. 462–468, 2022, [Online]. Available: <http://www.jurnal.unidha.ac.id/index.php/jteksis/article/view/573>
- [7] T. A. Tomy Aditya Firmansah, "Prototype Alat Monitoring dan Kontroling Banjir," *Techno Xplore J. Ilmu Komput. dan Teknol. Inf.*, vol. 5, no. 1, pp. 33–40, 2020, doi: 10.36805/technoxplore.v5i1.1081.
- [8] D. Hanggara, R. Dani, and E. Putra, "Purwarupa Perangkat Deteksi Dini Banjir Berbasis Internet of Things," *JIRE (Jurnal Inform. Rekayasa Elektron.*, vol. 4, no. 1, pp. 87–94, 2021.
- [9] I. A. Deswiyani, S. Solikhun, S. Sumarno, P. Poningsih, and S. R. Andani, "Rancang Bangun Alat Pendeteksi Ketinggian Air dan Alarm Pemberitahuan Antisipasi Datangnya Banjir Berbasis Arduino Uno," *J. Penelit. Inov.*, vol. 1, no. 2, pp. 155–164, 2021, doi: 10.54082/jupin.23.
- [10] J. Tarigan and A. D. Betan, "Sistem Perancangan Pendeteksi Banjir Secara Dini," *J. Tek. Mesin*, vol. 2, no. 2, pp. 63–67, 2019.