

**LAPORAN UJIAN TENGAH SEMESTER
MATA KULIAH INTERNET OF THINGS
SISTEM KEAMANAN/DETEKSI OBJEK**



Dosen Pengampu :

Ir. Subairi, ST., MT., IPM

Disusun Oleh :

Tiara Julyanti - 233140707111075

Desi Eka Mardiani - 233140707111084

Adam Ghonifirlandi - 233140707111102

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS VOKASI
SEMESTER 4 / TAHUN AJARAN 2025
UNIVERSITAS BRAWIJAYA**

ABSTRAK

Perkembangan mikrokontroler dan sensor memungkinkan sistem keamanan yang efektif dan terjangkau. Proyek ini merancang sistem deteksi objek berbasis ESP32 dengan sensor ultrasonik HC-SR04, LED, dan buzzer sebagai indikator. Sistem mendeteksi objek dalam jarak tertentu dan memberikan alarm peringatan. Pengembangan awal dilakukan melalui simulasi di platform Wokwi untuk pengujian virtual pemrograman ESP32, sensor, LED, dan buzzer. Setelah simulasi berhasil, implementasi dilanjutkan dengan pemrograman menggunakan Arduino IDE untuk pengujian langsung perangkat. Sensor ultrasonik mengukur jarak objek dengan mengirim dan menerima gelombang suara. Data jarak diproses ESP32 yang mengaktifkan LED dan buzzer saat objek terdeteksi dalam radius tertentu. Proyek ini membuktikan efektivitas ESP32 dalam deteksi jarak dan manfaat simulasi Wokwi sebagai alat pengembangan awal yang praktis.

Kata Kunci : ESP32, sensor ultrasonik HC-SR04, buzzer, LED, Arduino IDE, Wokwi, sistem deteksi objek, keamanan.

ABSTRACT

The development of microcontrollers and sensors enables an effective and affordable security system. This project designs an ESP32-based object detection system with an HC-SR04 ultrasonic sensor, LED, and buzzer as indicators. The system detects objects within a certain distance and gives a warning alarm. Initial development was carried out through simulation on the Wokwi platform for virtual testing of ESP32 programming, sensors, LEDs, and buzzers. After the simulation was successful, the implementation was continued with programming using the Arduino IDE for direct testing of the device. The ultrasonic sensor measures the distance of an object by sending and receiving sound waves. Distance data is processed by the ESP32 which activates the LED and buzzer when an object is detected within a certain radius. This project proves the effectiveness of the ESP32 in distance detection and the benefits of Wokwi simulation as a practical initial development tool.

Keywords : ESP32, HC-SR04 ultrasonic sensor, buzzer, LED, Arduino IDE, Wokwi, object detection system, security.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi mikrokontroler seperti ESP32 yang dilengkapi dengan fitur konektivitas Wi-Fi dan Bluetooth, semakin membuka peluang inovasi dalam pengembangan sistem deteksi objek yang cerdas dan responsif. ESP32 menjadi pilihan populer dalam berbagai aplikasi Internet of Things (IoT) karena kemampuannya yang tinggi, harga yang terjangkau, serta dukungan komunitas yang luas. Dengan kemampuan pemrosesan yang baik dan fitur komunikasi nirkabel, ESP32 memungkinkan pengembangan sistem yang tidak hanya mampu mengumpulkan data secara real-time, tetapi juga dapat berkomunikasi dengan perangkat lain atau server cloud untuk pemantauan jarak jauh.

Salah satu aplikasi yang banyak dikembangkan dengan ESP32 adalah sistem keamanan berbasis sensor ultrasonik yang mampu mendeteksi keberadaan objek atau hambatan dalam jarak tertentu. Sistem ini dirancang untuk memberikan peringatan dini jika ada objek yang mendekat atau memasuki area tertentu, sehingga dapat digunakan sebagai bagian dari sistem alarm atau pengawasan otomatis. Keunggulan ESP32 dalam hal konektivitas juga membuka kemungkinan integrasi dengan aplikasi mobile atau sistem monitoring berbasis web, sehingga pengguna dapat menerima notifikasi secara langsung di perangkat mereka.

Sensor ultrasonik HC-SR04 merupakan komponen utama yang sering digunakan dalam sistem deteksi jarak. Sensor ini bekerja dengan mengirimkan gelombang suara frekuensi tinggi dan mengukur waktu pantulan gelombang tersebut untuk menentukan jarak objek. Integrasi sensor ultrasonik dengan ESP32 menjadi sistem yang tidak hanya mampu mendeteksi objek, tetapi juga memberikan respons alarm secara visual dan audio melalui LED dan buzzer. LED berfungsi sebagai indikator visual yang mudah dikenali, sementara buzzer memberikan sinyal audio sebagai peringatan dini terhadap keberadaan objek atau potensi ancaman.

Penggunaan ESP32 dalam proyek ini juga didukung oleh kemudahan pemrograman menggunakan Arduino IDE, yang menyediakan lingkungan pengembangan yang familiar dan banyak digunakan oleh komunitas pembuat. Selain itu, dibantu oleh kemampuan simulasi di platform Wokwi sebagai pengujian rangkaian secara virtual sebelum implementasi fisik dilakukan. Simulasi ini sangat membantu dalam mengoptimalkan pengaturan sensor ultrasonik, LED, dan buzzer sehingga sistem dapat berfungsi secara efektif dan efisien tanpa risiko kerusakan perangkat saat pengujian awal. Dengan demikian, proses pengembangan menjadi lebih cepat dan hemat biaya.

1.2. Rumusan Masalah

Rumusan masalah dalam proyek ini adalah sebagai berikut :

1. Bagaimana cara merancang dan mengembangkan sistem deteksi objek berbasis mikrokontroler ESP32 yang mengintegrasikan sensor ultrasonik HC-SR04, LED sebagai indikator visual, dan buzzer sebagai indikator audio?
2. Bagaimana memanfaatkan platform simulasi Wokwi untuk merancang dan menguji rangkaian sensor ultrasonik, LED, dan buzzer secara virtual sebelum implementasi fisik?
3. Bagaimana mengimplementasikan logika pemrograman pada ESP32 menggunakan Arduino IDE agar dapat memproses data jarak dari sensor ultrasonik dan mengaktifkan LED serta buzzer sebagai respons alarm saat objek terdeteksi dalam jarak tertentu?
4. Bagaimana efektivitas penggunaan kombinasi sensor ultrasonik, LED, dan buzzer dalam memberikan peringatan dini terhadap keberadaan objek dalam aplikasi sistem keamanan sederhana?

1.3. Tujuan Proyek

Secara spesifik, berikut adalah beberapa tujuan dari proyek ini :

1. Merancang dan mengimplementasikan sistem deteksi objek berbasis mikrokontroler ESP32 yang mengintegrasikan sensor ultrasonik HC-SR04, indikator visual berupa LED, dan indikator audio berupa buzzer sebagai alarm peringatan.
2. Mengembangkan dan menguji rangkaian sistem secara virtual menggunakan platform simulasi Wokwi untuk memastikan fungsi sensor ultrasonik, LED, dan buzzer bekerja sesuai dengan desain sebelum implementasi perangkat keras.
3. Mengimplementasikan logika pemrograman pada mikrokontroler ESP32 menggunakan Arduino IDE untuk memproses data jarak dari sensor ultrasonik dan mengaktifkan LED serta buzzer sebagai respons terhadap keberadaan objek dalam jarak tertentu.
4. Melakukan implementasi dan pengujian langsung pada perangkat keras ESP32 dan sensor ultrasonik HC-SR04 untuk memastikan sistem dapat berfungsi secara efektif di lingkungan nyata.
5. Menghubungkan sistem deteksi objek dengan platform Node-RED untuk menampilkan data jarak sensor ultrasonik dalam bentuk chart secara *real-time*, sehingga memudahkan pemantauan dan analisis data secara visual.

1.4. Manfaat Proyek

Proyek implementasi sistem deteksi objek berbasis mikrokontroler ESP32 yang mengintegrasikan sensor ultrasonik HC-SR04, LED, buzzer, dan visualisasi data melalui Node-RED ini memberikan manfaat yang signifikan dari sisi akademis. Secara akademis, proyek ini memberikan pengalaman langsung dalam memahami konsep dasar dan penerapan Internet of Things (IoT) melalui pengembangan sistem yang terintegrasi antara perangkat keras dan perangkat lunak. Dengan memanfaatkan platform simulasi Wokwi, mahasiswa dapat melakukan perancangan dan pengujian rangkaian sensor ultrasonik, LED, dan buzzer secara virtual sebelum implementasi fisik, sehingga mempermudah pemahaman interaksi antar komponen tanpa memerlukan perangkat keras yang mahal atau kompleks. Proses pemrograman ESP32 menggunakan Arduino IDE juga memperkuat kemampuan dalam logika pemrograman mikrokontroler dan pengolahan data sensor secara real-time.

Selain itu, integrasi sistem dengan platform Node-RED untuk menampilkan data sensor secara visual dalam bentuk chart memberikan wawasan baru mengenai bagaimana data IoT dapat dimonitor dan dianalisis secara efektif. Hal ini memperluas pemahaman mahasiswa tentang konsep dashboard IoT dan pemanfaatan data untuk pengambilan keputusan. Proyek ini juga melatih keterampilan pemecahan masalah dan troubleshooting, mulai dari debugging kode hingga pengujian perangkat keras, yang merupakan kompetensi penting dalam pengembangan teknologi modern.

BAB II

PERENCANAAN PROYEK

2.1. Jadwal dan Timeline Proyek

TASK	ASSIGNED TO	PROGRESS	START	END
Planning and design				
Buat Skema Wokwi	Tiara Julyanti	100%	6/13/25	6/13/25
Implementasi Sensor di Wokwi	Tiara Julyanti	100%	6/13/25	6/13/25
Mengembangkan Logika Jarak	Adam Ghonifirlandi	100%	6/13/25	6/13/25
Integrasi LED dan Buzzer Simulasi	Adam Ghonifirlandi	100%	6/13/25	6/13/25
Penyempurnaan Logika Buzzer	Desi Eka Mardiani	100%	6/13/25	6/13/25
Hardware Assembly				
Persiapan Komponen Fisik	Desi Eka Mardiani	100%	6/13/25	6/13/25
Merakit Rangkaian Hardware	Adam Ghonifirlandi	100%	6/13/25	6/13/25
Penempatan Komponen	Tiara Julyanti	100%	6/13/25	6/13/25
Cek Koneksi & Troubleshooting	Adam Ghonifirlandi	100%	6/13/25	6/13/25
Programming				
Coding Logika di Arduino IDE	Desi Eka Mardiani	100%	6/15/25	6/15/25
Implementasi di Hardware	Semua Anggota	100%	6/15/25	6/15/25
Uji Fungsi Deteksi & Buzzer Fisik	Semua Anggota	100%	6/15/25	6/15/25
Setup ke Dashboard Node-RED	Desi Eka Mardiani	100%	6/16/25	6/16/25
Uji Visualisasi Chart Real-time	Semua Anggota	100%	6/16/25	6/16/25
Documentation & Evaluation				
Dokumentasi Simulasi Wokwi	Adam Ghonifirlandi	100%	6/16/25	6/16/25
Menyusun Draft Laporan	Tiara Julyanti	100%	6/16/25	6/16/25
Hasil Simulasi Hardware	Semua Anggota	100%	6/16/25	6/16/25
Meninjau Penulisan Laporan	Semua Anggota	100%	6/16/25	6/16/25
Finalisasi & Analisis Tujuan	Semua Anggota	100%	6/16/25	6/16/25

2.2. Sumber Daya yang Digunakan

Berikut adalah sumber daya yang digunakan dalam implementasi proyek ini :

1. Platform Simulasi Wokwi

- **Penjelasan** : Wokwi adalah sebuah platform simulasi online gratis yang dapat digunakan pengguna untuk membuat dan menjalankan berbagai proyek elektronika dan IoT. Platform ini menyediakan berbagai macam komponen virtual mulai dari

mikrokontroler ESP32, sensor ultrasonik HC-SR04, aktuator LED dan buzzer, serta modul konektivitas jaringan seperti Wi-Fi dan Bluetooth.

- **Peran dalam Proyek :** Wokwi digunakan sebagai lingkungan pengembangan utama untuk merancang rangkaian virtual, menulis kode program Arduino, dan mensimulasikan interaksi antar komponennya secara *real-time*. Simulasi ini membantu menguji fungsi sensor ultrasonik, LED, dan buzzer sebelum implementasi perangkat keras fisik.

2. Mikrokontroler ESP32

- **Penjelasan :** ESP32 adalah mikrokontroler canggih yang dilengkapi dengan fitur konektivitas Wi-Fi dan pengembangan aplikasi IoT karena fleksibilitas, performa, dan harga terjangkau. Pemrograman ESP32 dapat dilakukan menggunakan Arduino IDE, sehingga memudahkan perkembangan sistem yang kompleks.
- **Peran dalam Proyek :** Dalam simulasi ini, ESP32 berfungsi sebagai “otak” dari sistem deteksi objek. ESP32 menerima data jarak dari sensor ultrasonik HC-SR04, memproses data tersebut berdasarkan logika pemrograman yang dibuat, kemudian mengendalikan output ke LED dan buzzer. Selain itu, ESP32 juga mengirimkan data jarak ke platform Node-RED untuk visualisasi secara *real-time*.

3. Sensor Ultrasonik HC-SR04

- **Penjelasan :** HC-SR04 adalah sensor ultrasonik yang bekerja dengan mengirimkan gelombang suara frekuensi tinggi dan mengukur waktu pantulan gelombang tersebut untuk menentukan jarak objek tanpa kontak fisik.
- **Peran dalam Proyek :** Sensor HC-SR04 dalam simulasi ini bertugas untuk mendeteksi keberadaan objek dalam rentang jarak tertentu. Data jarak yang dihasilkan akan dibaca oleh ESP32 untuk menentukan apakah objek berada dalam zona deteksi, sehingga dapat mengaktifkan indikator LED dan buzzer.

4. LED atau *Light Emitting Diode* Virtual

- **Penjelasan :** LED adalah komponen semikonduktor yang memancarkan cahaya, ketika arus listrik melewatinya.
- **Peran dalam Proyek :** LED dalam simulasi ini digunakan sebagai indikator visual yang menyala ketika objek terdeteksi dalam jarak yang telah ditentukan. Hal ini memberikan sinyal visual yang mudah dikenali sebagai respons deteksi objek.

5. Buzzer

- **Penjelasan :** Buzzer piezoelektrik adalah sebuah komponen yang menghasilkan suara ketika diberikan tegangan listrik. Buzzer sering digunakan sebagai indikator audio atau alarm dalam aplikasi elektronika.
- **Peran dalam Proyek :** Buzzer dalam simulasi ini sama dengan LED, bertugas sebagai indikator audio yang berbunyi saat objek terdeteksi oleh sensor ultrasonik. Suara buzzer memberikan peringatan tambahan selain LED.

2.3. Pembagian Tugas

Dalam proyek sistem deteksi objek berbasis ESP32 dan sensor ultrasonik ini, setiap anggota memiliki tugas masing-masing untuk memastikan kelancaran pelaksanaan.

Fase 1 : Perancangan dan Implementasi di Simulasi Wokwi

1. Tiara Julyanti
 - Membuat rancangan awal rangkaian virtual di Wokwi, meliputi penempatan komponen-komponen utamanya, yaitu ESP32, sensor ultrasonik HC-SR04, LED, dan *buzzer*.
 - Mengimplementasi dasar interaksi sensor ultrasonik dengan Arduino Uno, meliputi inisialisasi pin, pengiriman trigger, pembacaan echo.
 - Menguji pembacaan jarak awal dari sensor ultrasonik di simulasi.
2. Adam Ghonifirlandi
 - Melanjutkan dan mengembangkan kode program ESP32 di Wokwi untuk membaca data jarak dari sensor ultrasonik.
 - Membuat logika deteksi jarak, dengan menentukan ambang batas jarak.
 - Mengintegrasikan output ke LED dan buzzer, ketika menyala saat terdeteksi.
3. Desi Eka Mardiani
 - Mengembangkan dan menyempurnakan logika alarm, durasi bunyi buzzer dan pola kedipan lampu LED.
 - Melakukan pengujian menyeluruh untuk skenario deteksinya dan respon alarm pada simulasi.

Fase 2 : Implementasi dan Pengujian pada Perangkat Keras

1. Desi Eka Mardiani

- Menyiapkan perangkat keras ESP32, sensor ultrasonik HC-SR04, LED, resistor, buzzer, breadboard, jumper.
- Melakukan pemrograman untuk perangkat keras melalui Arduino IDE.
- Menghubungkan sistem dengan platform Node-RED untuk menampilkan data jarak sensor dalam bentuk grafik/*chart* secara *real-time*.

2. Adam Ghonifirlandi

- Menyusun rangkaian hardware secara fisik.
- Melakukan pengecekan dan *troubleshooting* perangkat keras untuk memastikan sistem berjalan dengan baik.

3. Tiara Julyanti

- Berkontribusi melakukan perancangan rangkaian hardware.
- Membantu pengumpulan referensi dan dokumentasi pendukung terkait komponen hardware.

Fase 3 : Dokumentasi PowerPoint dan Laporan

1. Tiara Julyanti

- Membuat bagian pendahuluan dan latar belakang dalam laporan.
- Menjelaskan tujuan dan manfaat proyek.
- Menyusun daftar sumber daya yang digunakan.

2. Adam Ghonifirlandi

- Mendokumentasikan metodologi proyek dalam laporan dan.
- Menjelaskan langkah-langkah perancangan dan implementasi di Wokwi.
- Menyertakan *screenshot* rangkaian virtual di Wokwi dan potongan kode program.

3. Desi Eka Mardiani

- Membuat bagian hasil dan pembahasan, serta kesimpulan.
- Menganalisis hasil pengujian di simulasi Wokwi.
- Menyimpulkan keberhasilan proyek.

2.4. Peran dalam Proyek

Dalam proyek ini, peran saya memastikan sistem deteksi objek berbasis ESP32 dapat berfungsi dengan baik dan terdokumentasi. Pada fase 1, saya bertanggung jawab mengembangkan dan menyempurnakan logika alarm pada simulasi Wokwi. Hal ini mencakup

pengaturan durasi bunyi buzzer, pola kedipan LED sebagai indikator visual, serta memastikan respons alarm sesuai dengan skenario deteksi objek oleh sensor ultrasonik. Selain itu, saya juga melakukan pengujian menyeluruh untuk memastikan sistem dapat merespons dengan konsisten dan sesuai harapan.

Pada fase 2, saya melakukan persiapan dan pengujian pada perangkat keras. Tahap ini dimulai dengan menyiapkan perangkat ESP32, sensor ultrasonik HC-SR04, LED, buzzer, serta komponen pendukung seperti resistor, breadboard, dan kabel jumper. Saya juga melakukan pemrograman menggunakan Arduino IDE untuk mengimplementasikan logika yang telah dikembangkan ke perangkat fisik. Selanjutnya, saya menghubungkan sistem dengan platform Node-RED untuk menampilkan data jarak sensor secara *real-time* dalam bentuk grafik/*chart*, sehingga memudahkan pemantauan dan analisis data secara visual. Pengujian dilakukan untuk memastikan seluruh perangkat keras dan perangkat lunak bekerja secara sinergis.

Terakhir, pada fase 3, saya menyusun dokumentasi proyek berupa laporan tertulis. Bagian ini mencakup penulisan hasil pengujian, pembahasan terkait performa sistem, baik pada simulasi Wokwi maupun implementasi fisik, serta analisis keberhasilan proyek secara keseluruhan. Saya juga menyusun kesimpulan yang merangkum pencapaian dan manfaat dari pengembangan sistem deteksi objek berbasis ESP32 ini.

BAB III

IMPLEMENTASI DAN PROGRESS PROYEK

3.1. Deskripsi Teknis Proyek

Proyek ini bertujuan untuk mengembangkan dan mengimplementasikan sistem deteksi objek berbasis mikrokontroler ESP32 yang mengintegrasikan sensor ultrasonik HC-SR04, indikator visual berupa LED, dan indikator audio berupa buzzer. Sistem dirancang untuk mendeteksi keberadaan objek dalam jarak tertentu dan memberikan respons alarm sebagai bentuk peringatan. Selain itu, data yang diperoleh dari sensor ultrasonik juga dikirimkan secara *real-time* ke platform Node-RED untuk divisualisasikan dalam bentuk grafik/*chart*.

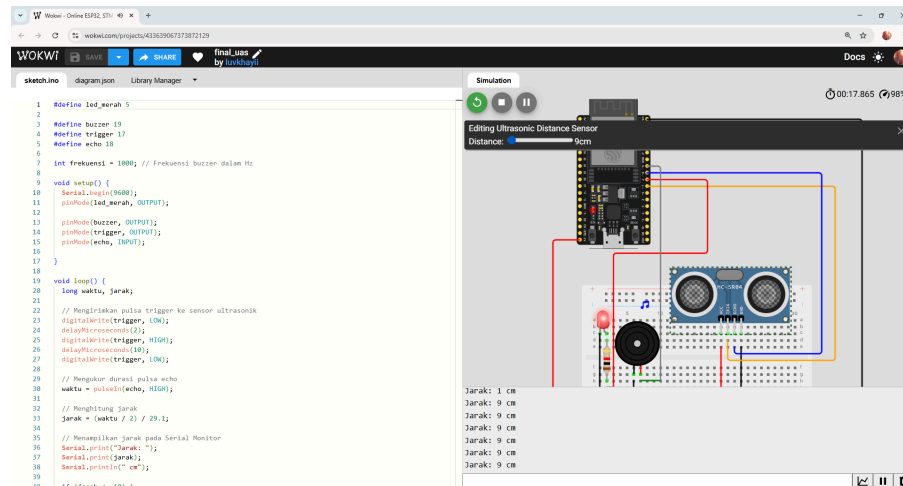
Komponen utama yang digunakan adalah mikrokontroler ESP32 sebagai pusat pengendali sistem dengan kemampuan konektivitas Wi-Fi. Sensor ultrasonik HC-SR04 mengukur jarak objek dengan prinsip pengiriman dan penerimaan gelombang suara ultrasonik, LED merah memberikan indikasi visual saat objek terdeteksi dalam jarak tertentu, serta buzzer yang memberikan indikasi audio sebagai alarm.

Prinsip Kerja Sistem

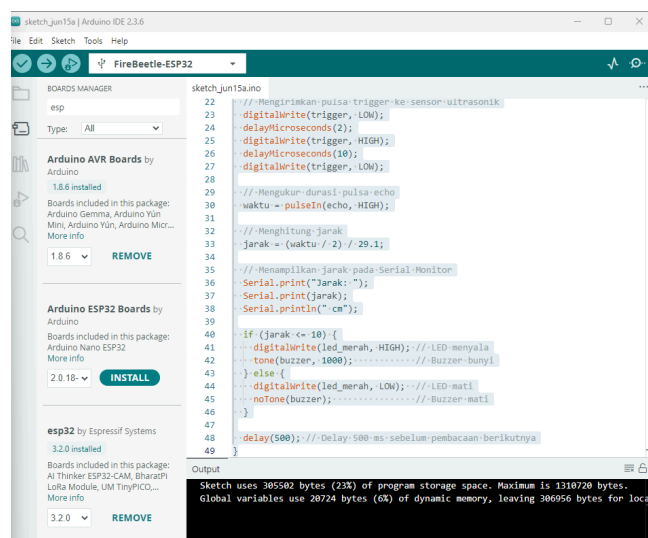
1. **Pengukuran Jarak :** ESP32 mengirimkan pulsa singkat ke pin trigger sensor HC-SR04 yang kemudian memancarkan gelombang ultrasonik.
2. **Penerimaan Pantulan :** Gelombang ultrasonik yang mengenai objek akan dipantulkan kembali dan diterima oleh pin echo sensor.
3. **Perhitungan Waktu Tempuh :** Sensor mengukur durasi waktu antara pengiriman pulsa echo.
4. **Konversi ke Jarak :** ESP32 menghitung jarak berdasarkan waktu tempuh gelombang suara dan kecepatan suara di udara (~ 343 m/s). Rumus yang digunakan :
$$\text{distance} = (\text{duration} / 2) \times 0.0343 \text{ cm}$$
5. **Indikasi Alarm :** Jika jarak objek berada di bawah ambang batas yang ditentukan (misal ≤ 10 cm), LED akan menyala dan buzzer bunyi sebagai tanda peringatan.
6. **Pengiriman Data ke Node-RED :** ESP32 menggunakan koneksi Wi-Fi untuk mengirimkan data jarak yang diperoleh dari sensor ultrasonik HC-SR04 ke server Node-RED secara berkala setiap 1 detik. Proses pengiriman data ini dilakukan melalui HTTP POST dalam format JSON. Data ini kemudian dapat divisualisasikan secara *real-time* dalam bentuk grafik/*chart* untuk memudahkan monitoring.

3.1. Dokumentasi Hasil Pengerjaan

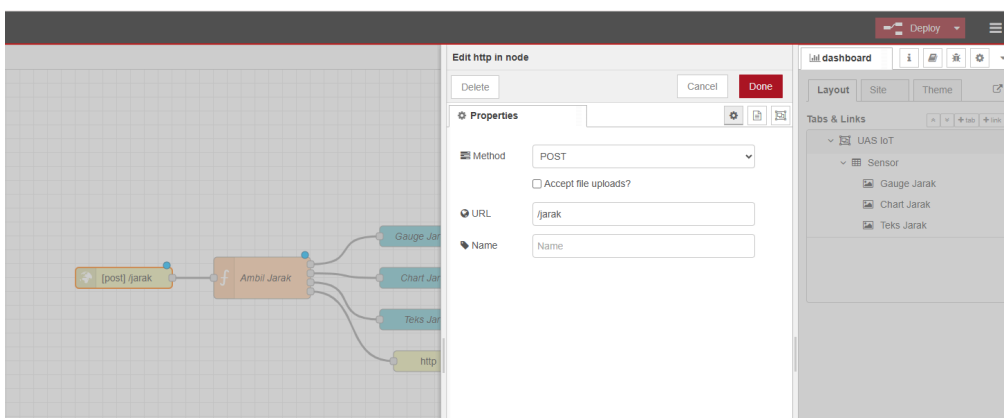
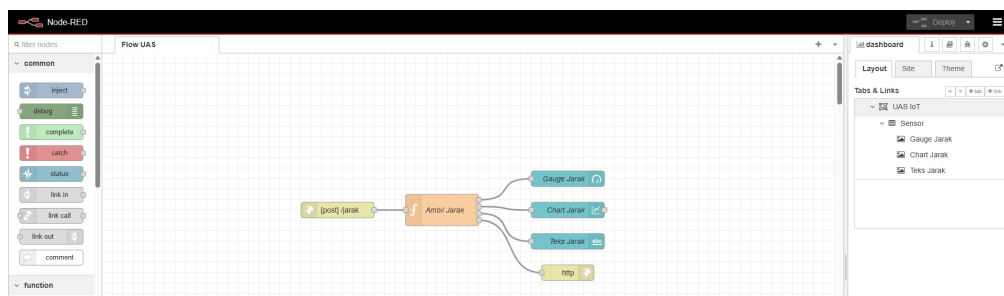
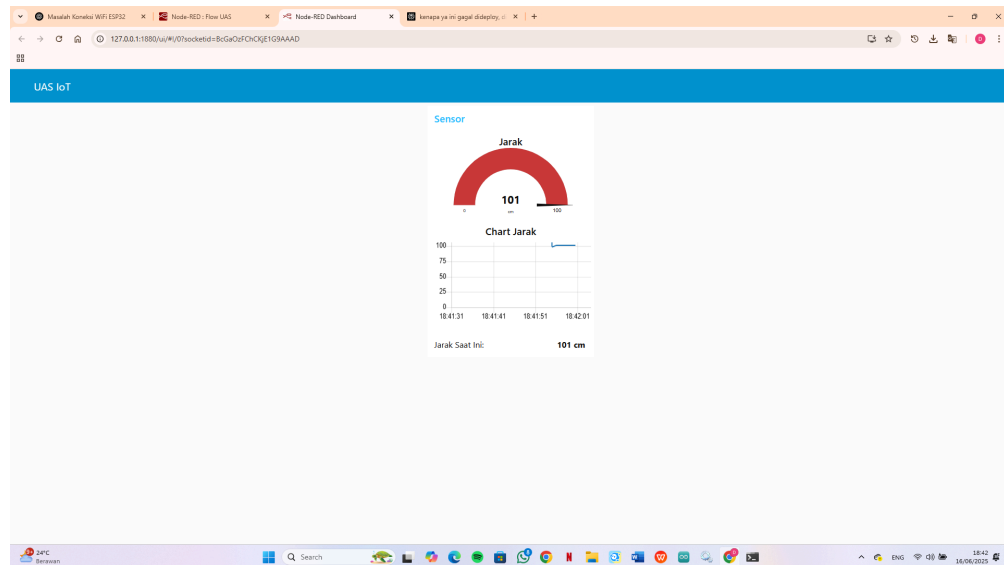
Proyek diawali dengan tahap simulasi menggunakan platform Wokwi. Pada tahap ini, rangkaian virtual yang terdiri dari mikrokontroler ESP32, sensor ultrasonik HC-SR04, LED, dan buzzer dirancang dan diprogram secara virtual. Simulasi ini bertujuan untuk menguji logika pengukuran jarak, pengendalian LED dan buzzer. Pengujian dengan berbagai skenario deteksi objek dilakukan agar memastikan respons sistem sesuai dengan yang diharapkan sebelum implementasi fisik.



Setelah simulasi berhasil dan logika program sudah berjalan dengan baik, tahap selanjutnya adalah mengimplementasikannya pada perangkat keras. ESP32 diprogram menggunakan Arduino IDE, dan dilakukan pengujian koneksi awal dengan membuka Serial Monitor untuk memastikan ESP32 dapat terhubung ke jaringan Wi-Fi yang telah disiapkan. Pada tahap ini, juga diuji fungsi dasar pembacaan sensor ultrasonik dan pengendalian output LED serta buzzer secara langsung pada perangkat fisik.

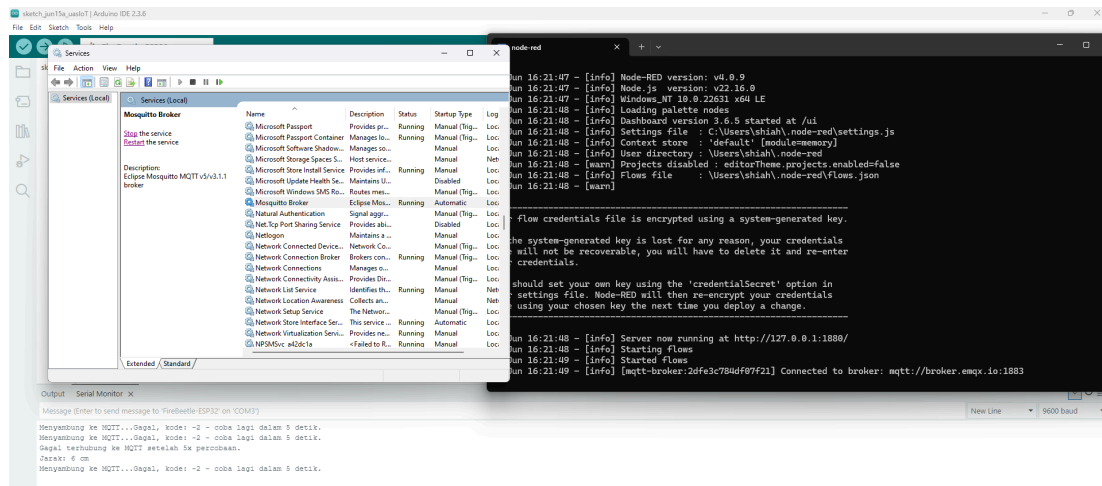


Proses konfigurasi Node-RED meliputi pembuatan flow yang menerima data HTTP POST, parsing data JSON, dan menampilkan nilai jarak sensor pada dashboard interaktif. Visualisasi ini memudahkan pemantauan kondisi sensor secara langsung dan memberikan gambaran performa sistem secara keseluruhan.



3.3. Kendala dan Solusi

Pada tahap awal implementasi, salah satu tantangan utama yang dihadapi adalah proses koneksi antara ESP32 dan platform Node-RED melalui jaringan Wi-Fi. Awalnya saya mencoba menggunakan protokol MQTT (*Message Queuing Telemetry Transport*) dengan bantuan Mosquitto Broker sebagai perantara komunikasi data antara ESP32 dan Node-RED. Namun, ternyata proses koneksi dengan MQTT-nya sering gagal.



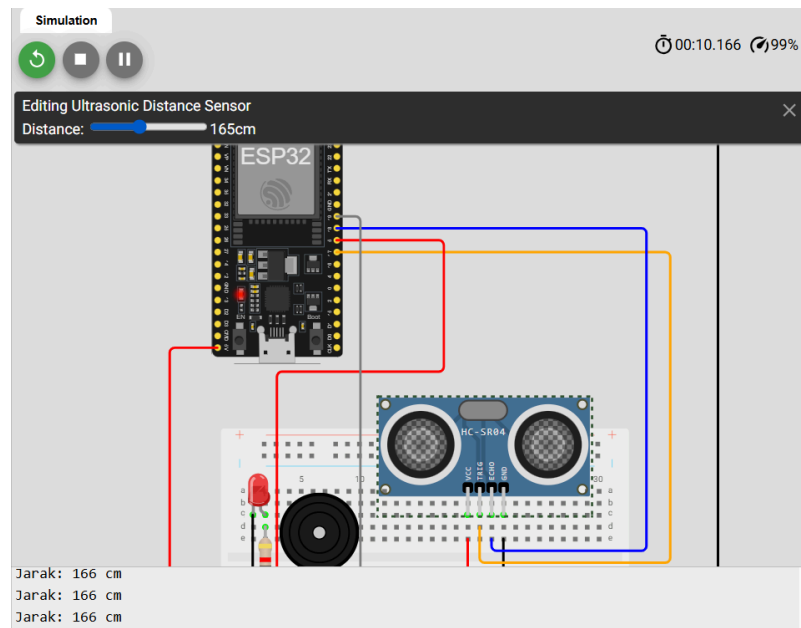
Kendala kemungkinan disebabkan oleh beberapa faktor seperti konfigurasi broker MQTT yang belum tepat, firewall pada sistem operasi atau ketidakcocokan antara library MQTT yang digunakan pada ESP32 dengan broker yang berjalan di lokal.

Karena sudah beberapa kali mencoba dan melakukan *troubleshooting* tetapi masih gagal, saya memutuskan untuk beralih dari protokol MQTT ke metode HTTP POST yang akan mengirimkan data dari ESP32 ke Node-RED. Dengan metode ini, ESP32 cukup mengirimkan data jarak dalam format JSON ke endpoint Node-RED (/jarak) menggunakan protokol HTTP, tanpa perlu konfigurasi broker atau autentikasi tambahan.

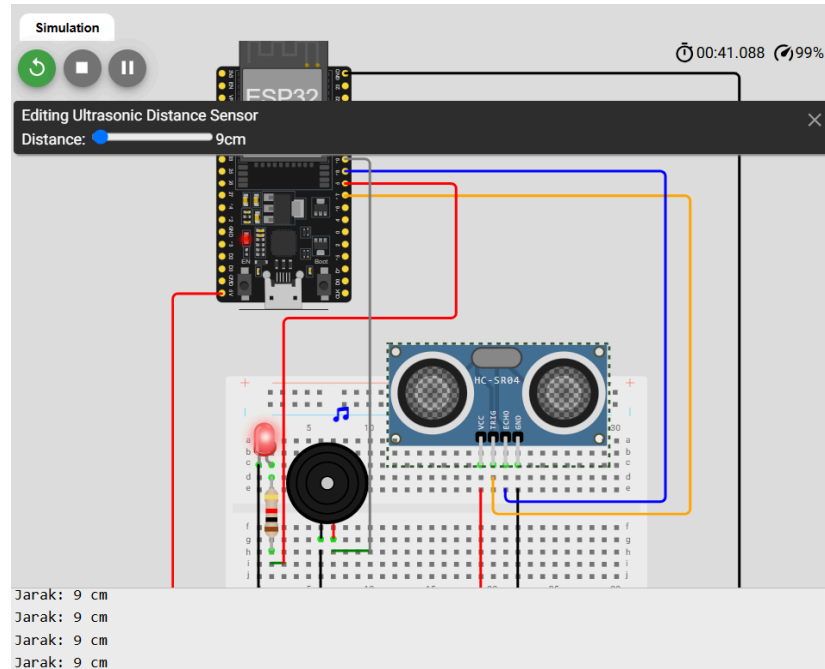
1. **Mengubah Kode ESP32 :** Menggunakan library WiFi.h dan HTTPClient.h untuk membuat koneksi Wi-Fi dan mengirim data menggunakan HTTP POST ke alamat endpoint Node-RED yang sudah dibuat (<http://192.168.114.8:1880/jarak>).
2. **Membuat Endpoint di Node-RED :** Pada Node-RED, dibuat flow dengan node HTTP In untuk menerima request pada path /jarak, kemudian data diproses dan ditampilkan pada dashboard menggunakan node gauge, chart, dan text.
3. **Pengiriman Data :** Pada Node-RED, dibuat flow dengan node HTTP In untuk menerima request pada path /jarak, kemudian data diproses dan ditampilkan pada dashboard menggunakan node gauge, chart, dan text.

3.4. Tampilan Mock Up

- Ketika objek tidak berada pada atau melewati jarak yang sudah ditentukan, maka lampu LED tidak menyala dan *buzzer* tidak akan berbunyi.



- Ketika objek berada pada atau melewati jarak yang ditentukan (200cm), maka lampu LED akan menyala dan *buzzer* akan berbunyi.



BAB IV

EVALUASI DAN HASIL

4.1. Perbandingan Hasil dengan Tujuan Awal

1. Merancang dan mengimplementasikan sistem deteksi objek berbasis mikrokontroler ESP32 yang mengintegrasikan sensor ultrasonik HC-SR04, indikator visual berupa LED, dan indikator audio berupa buzzer sebagai alarm peringatan.

Hasil : Sistem berhasil dirancang dan diimplementasikan dengan baik. Sensor ultrasonik HC-SR04 mampu mendeteksi objek dalam jarak yang telah ditentukan, dan ESP32 mengaktifkan LED serta buzzer sebagai alarm secara responsif. Indikator visual dan audio berfungsi sesuai harapan, memberikan peringatan yang jelas saat objek berada dalam radius deteksi.

2. Mengembangkan dan menguji rangkaian sistem secara virtual menggunakan platform simulasi Wokwi untuk memastikan fungsi sensor ultrasonik, LED, dan buzzer bekerja sesuai dengan desain sebelum implementasi perangkat keras.

Hasil : Simulasi di Wokwi berhasil dilakukan dengan lancar. Logika pemrograman dan interaksi antar komponen diuji secara menyeluruh, sehingga potensi kesalahan dapat diminimalisir sebelum implementasi fisik.

3. Mengimplementasikan logika pemrograman pada mikrokontroler ESP32 menggunakan Arduino IDE untuk memproses data jarak dari sensor ultrasonik dan mengaktifkan LED serta buzzer sebagai respons terhadap keberadaan objek dalam jarak tertentu.

Hasil : Logika pemrograman berhasil diimplementasikan dan diuji pada ESP32. Data jarak dari sensor ultrasonik diproses secara *real-time*, dan output LED serta buzzer dikendalikan sesuai kondisi jarak yang terdeteksi. Program berjalan stabil dan responsif.

4. Melakukan implementasi dan pengujian langsung pada perangkat keras ESP32 dan sensor ultrasonik HC-SR04 untuk memastikan sistem dapat berfungsi secara efektif di lingkungan nyata.

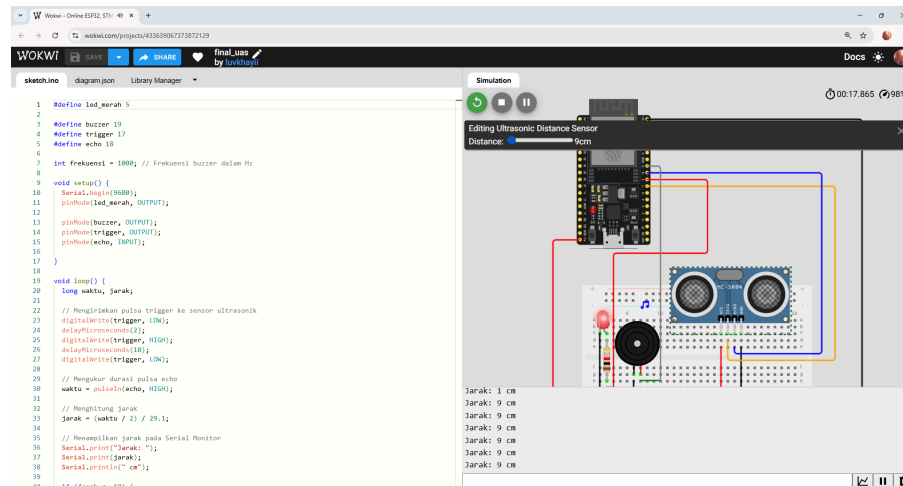
Hasil : Implementasi perangkat keras berhasil dilakukan dengan pengujian di lingkungan nyata. Sistem dapat mendeteksi objek secara akurat dan memberikan alarm sesuai dengan desain. Pengujian menunjukkan kestabilan sistem dan keandalan sensor dalam kondisi nyata.

5. Menghubungkan sistem deteksi objek dengan platform Node-RED untuk menampilkan data jarak sensor ultrasonik dalam bentuk chart secara *real-time*, sehingga memudahkan pemantauan dan analisis data secara visual.

Hasil : Integrasi dengan Node-RED berhasil dilakukan. Data jarak sensor dikirim secara berkala menggunakan HTTP POST dan divisualisasikan dalam bentuk gauge, chart, dan teks di dashboard Node-RED.

4.2. Uji Coba Sistem

<https://wokwi.com/projects/433639067373872129>



4.3. Feedback atau Saran Perbaikan

Proyek ini sepenuhnya telah berjalan dengan baik dan mencapai tujuan utama yang diharapkan, namun untuk saran perbaikan perkembangan selanjutnya adalah melakukan peningkatan akurasi sensor, mengembangkan dashboard Node-RED dengan fitur kontrol parameter jarak ambang dan log data historis yang lebih lengkap. Mungkin bisa juga melakukan pengembangan dengan menambahkan fitur notifikasi melalui aplikasi mobile untuk memperluas fungsi sistem keamanannya.

BAB V

KESIMPULAN DAN REKOMENDASI

5.1. Kesimpulan Akhir Proyek

Proyek sistem deteksi objek berbasis mikrokontroler ESP32 dengan sensor ultrasonik HC-SR04, indikator LED, buzzer, dan integrasi visualisasi data melalui Node-RED berhasil dirancang dan diimplementasikan sesuai dengan tujuan awal. Sistem mampu mendeteksi objek dalam jarak tertentu secara akurat dan memberikan respons alarm visual serta audio yang efektif. Simulasi di platform Wokwi membantu mempermudah pengembangan dan pengujian sebelum implementasi perangkat keras fisik. Integrasi dengan Node-RED memungkinkan monitoring data jarak secara real-time yang memudahkan analisis dan pengawasan.

5.2. Rekomendasi untuk Pengembangan lebih Lanjut

1. Peningkatan Akurasi dan Sensitivitas Sensor

Penggunaan sensor ultrasonik dengan resolusi lebih tinggi atau penggabungan sensor lain seperti sensor inframerah atau kamera ESP32-CAM dapat meningkatkan akurasi deteksi dan kemampuan sistem dalam mengenali objek dengan ukuran dan jarak yang lebih variatif.

2. Pengembangan Sistem Notifikasi dan Kontrol Jarak Jauh

Integrasi sistem dengan aplikasi mobile atau platform cloud dapat memungkinkan pengguna menerima notifikasi secara real-time dan melakukan kontrol sistem dari jarak jauh, sehingga meningkatkan fleksibilitas dan keamanan pengguna.

3. Optimasi Konsumsi Daya dan Efisiensi Sistem

Implementasi mode hemat daya (sleep mode) pada ESP32 dan optimasi algoritma pemrosesan data dapat memperpanjang masa pakai perangkat, terutama untuk aplikasi yang memerlukan operasi jangka panjang tanpa pengawasan langsung.

DAFTAR PUSTAKA

- Banzi, M., & Shiloh, A. (2014). *Getting started with Arduino: The open-source electronics prototyping platform*. Maker Media, Inc.
- Bogue, R. (2013). *Sensors for intelligent systems*. Artech House.
- LED (light-emitting diode). (n.d.). *Wikipedia*. Diakses dari https://en.wikipedia.org/wiki/Light-emitting_diode
- Piezoelectric speaker. (n.d.). *Wikipedia*. Diakses dari https://en.wikipedia.org/wiki/Piezoelectric_speaker
- Wokwi. (n.d.). *Wokwi online ESP32, Arduino, and Raspberry Pi simulator*. Diakses dari <https://wokwi.com/>
- Node-RED Documentation. (2023). *Node-RED User Guide*. Diakses dari <https://nodered.org/docs/>

LAMPIRAN

Kode Program

```
#include <WiFi.h>
#include <HTTPClient.h>

#define led_merah 2
#define buzzer 21
#define trigger 19
#define echo 18

// SSID & password WiFi (pakai hotspot HP)
const char* ssid = "Hayiii";
const char* password = "desiekaa";

// Alamat endpoint Node-RED
const char* serverName = "http://192.168.114.8:1880/jarak";

unsigned long lastSend = 0;

void setup() {
    Serial.begin(9600);
    pinMode(led_merah, OUTPUT);
    pinMode(buzzer, OUTPUT);
    pinMode(trigger, OUTPUT);
    pinMode(echo, INPUT);

    // Koneksi WiFi
    WiFi.begin(ssid, password);
    Serial.print("Menghubungkan ke SSID: ");
    Serial.println(ssid);

    int attempts = 0;
    while (WiFi.status() != WL_CONNECTED && attempts < 20) {
        delay(500);
        Serial.print(".");
        attempts++;
    }

    if (WiFi.status() == WL_CONNECTED) {
        Serial.println("\nWiFi terhubung!");
        Serial.print("IP ESP32: ");
```

```

        Serial.println(WiFi.localIP());
    } else {
        Serial.println("\nGagal terhubung ke WiFi!");
    }
}

void loop() {
    long waktu, jarak;

    // Trigger sensor ultrasonik
    digitalWrite(trigger, LOW);
    delayMicroseconds(2);
    digitalWrite(trigger, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger, LOW);

    // Baca durasi echo dengan timeout 30ms
    waktu = pulseIn(echo, HIGH, 30000);

    if (waktu == 0) {
        jarak = 0; // Tidak ada echo, anggaplah jaraknya 0
    } else {
        jarak = (waktu / 2) / 29.1; // Menghitung jarak dalam cm
    }

    // Tampilkan jarak di Serial Monitor
    Serial.print("Jarak: ");
    Serial.print(jarak);
    Serial.println(" cm");

    // Kontrol LED dan buzzer berdasarkan jarak
    if (jarak > 0 && jarak <= 10) {
        digitalWrite(led_merah, HIGH);
        tone(buzzer, 1000);
    } else {
        digitalWrite(led_merah, LOW);
        noTone(buzzer);
    }

    // Mengirim data ke Node-RED setiap 1 detik
    if (millis() - lastSend > 1000) {
        if (WiFi.status() == WL_CONNECTED) {
            HTTPClient http;

```

```

http.begin(serverName);
http.addHeader("Content-Type", "application/json");

String jsonData = "{\"jarak\":\"" + String(jarak) + "\"}";

int httpResponseCode = http.POST(jsonData);

if (httpResponseCode > 0) {
    Serial.print("Data terkirim! Kode: ");
    Serial.println(httpResponseCode);
} else {
    Serial.print("Gagal kirim. Error: ");
    Serial.println(httpResponseCode);
}
http.end();
} else {
    Serial.println("WiFi tidak terhubung saat kirim data.");
}
lastSend = millis();
}

delay(100); // Delay kecil sebagai pembacaan sensor lancar
}

```

Dokumentasi Pengujian

