

PENGONTROLAN PERALATAN RUMAH TANGGA DENGAN PERINTAH SUARA MENGGUNAKAN ESP32 BERBASIS IOT

Nur Amril*¹, Andani Achmad², Abdul Latief Arda³, Dayanti⁴

^{1,2,3}Program Pascasarjana Universitas Handayani Makassar, ⁴Universitas Patria Artha
e-mail: *¹nuramril43@guru.smk.belajar.id, ²andani@unhas.ac.id, ³latiefarda@gmail.com, ⁴dayanti.fattah@gmail.com

Perkembangan Internet of Things (IoT) dan kecerdasan buatan mendorong terciptanya sistem otomasi rumah yang lebih cerdas dan efisien. Penelitian ini bertujuan merancang dan mengimplementasikan sistem pengontrolan peralatan rumah tangga berbasis IoT menggunakan mikrokontroler ESP32 yang terintegrasi dengan aplikasi perintah suara. ESP32 memanfaatkan konektivitas Wi-Fi dan terhubung ke platform cloud ESP RainMaker untuk memungkinkan pengendalian perangkat secara real-time melalui internet. Metode yang digunakan adalah prototyping, meliputi analisis kebutuhan, perancangan, pembangunan prototipe, serta evaluasi dan pengujian kinerja. Parameter pengujian mencakup waktu respons sistem dan tingkat akurasi pengenalan perintah suara. Hasil menunjukkan rata-rata waktu respons 2–2,7 detik tergantung kondisi jaringan. Akurasi pengenalan suara mencapai 100% pada lingkungan tenang dan 90% pada kondisi kebisingan normal. Sistem terbukti responsif, akurat, dan lebih praktis dibandingkan pengendalian manual, serta berpotensi mendukung pengembangan smart home yang terjangkau.

Kata Kunci — ESP32, ESP RainMaker, Perintah Suara, IoT.

I. PENDAHULUAN

mendorong implementasi konsep Internet of Things (IoT) dalam berbagai bidang, termasuk otomasi rumah. Istilah IoT mulai dikenal tahun 1999 yang saat itu disebutkan pertama kalinya dalam sebuah presentasi oleh Kevin Ashton [1]. IoT adalah jaringan luas dari objek-objek yang saling berhubungan yang dapat dialamati secara unik, berdasarkan pada protokol komunikasi standar [2]. IoT memungkinkan pengguna mengontrol seluruh perangkat yang ada disekelilingnya dari jarak jauh menggunakan internet [3].

Konsep smart home bertujuan meningkatkan kenyamanan, keamanan, serta efisiensi energi. Namun, sebagian besar peralatan rumah tangga masih dikendalikan secara manual menggunakan saklar fisik. Hal ini kurang efisien dan tidak ramah bagi lansia maupun penyandang disabilitas.

Pemanfaatan pengenalan suara (Automatic Speech Recognition/ASR) memungkinkan interaksi tanpa sentuhan fisik. Mikrokontroler ESP32 dipilih karena telah terintegrasi modul WiFi dan Bluetooth sehingga sangat sesuai untuk implementasi IoT [4].

Penelitian ini bertujuan merancang sistem pengontrolan lampu berbasis IoT menggunakan ESP32 dengan integrasi perintah suara melalui aplikasi Gemini dan platform ESP RainMaker.

Penelitian smart home sebelumnya banyak menggunakan Arduino dan ESP8266 [5],[6]. Perancangan smart home untuk pengendalian peralatan elektronik dan pemantauan keamanan rumah berbasis IoT juga banyak dilakukan dengan menggunakan Arduino dan sensor DHT11 [7].

Beberapa penelitian mengintegrasikan sistem berbasis web maupun SMS untuk monitoring dan kontrol jarak jauh. Selain Arduino dan ESP8266, beberapa juga menggunakan Rasbery Pi serbagai server yang akan menghubungkan antara hardware dan software yang dikontrol melalui web sebagai interface yang digunakan pengguna untuk memasukkan input dan menghasilkan output [8].

Berbeda dengan penelitian sebelumnya, sistem yang dikembangkan dalam penelitian ini:

1. Menggunakan ESP32 sebagai pusat kendali,
2. Mengintegrasikan cloud IoT,
3. Mengimplementasikan kontrol berbasis perintah suara real-time..

II. METODE PENELITIAN

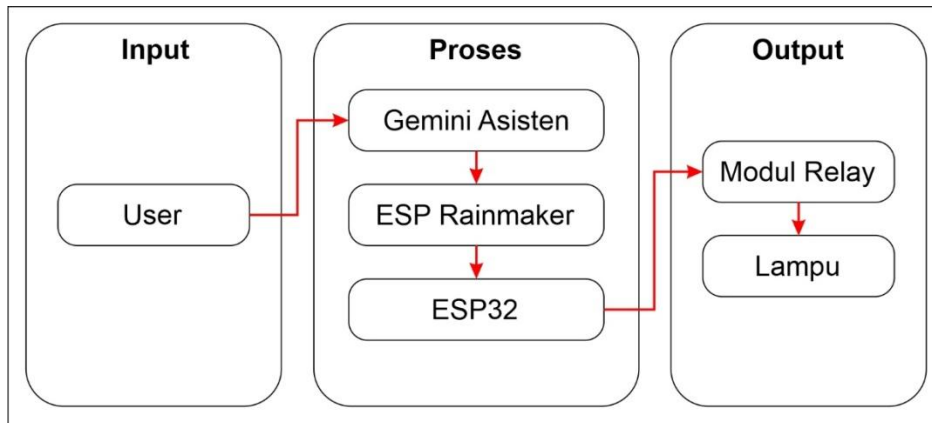
Penelitian ini menggunakan metode prototyping. Metode prototyping terdiri dari tahapan [9]:

1. Pengumpulan kebutuhan, meliputi identifikasi kebutuhan sistem pengontrolan lampu berbasis suara.
2. Perancangan sistem, meliputi desain arsitektur perangkat keras dan perangkat lunak.
3. Penelitian ini menggunakan metode prototyping. Metode prototyping terdiri dari tahapan [9]:
4. Pengumpulan kebutuhan, meliputi identifikasi

kebutuhan sistem pengontrolan lampu berbasis suara.

5. Perancangan sistem, meliputi desain arsitektur perangkat keras dan perangkat lunak.

A. Arsitektur Sistem



Gambar 1. Blok Diagram Sistem Kerja Pengontrolan Peralatan Rumah Tangga dengan Perintah Suara Menggunakan ESP32 Berbasis IoT

Blok diagram sistem pengontrolan lampu berbasis perintah suara terdiri dari beberapa komponen utama sebagai berikut:

1. User (Pengguna) memberikan perintah suara seperti “Nyalakan lampu” atau “Matikan lampu”.
2. Gemini Asisten. Perintah suara diproses menjadi perintah digital di aplikasi Gemini yang berfungsi sebagai antarmuka pengenalan suara kemudian dikirim melalui RainMaker Cloud API [10],[11].
3. Cloud Server (ESP RainMaker). Data perintah digital dari Gemini Asisten dikirim melalui RainMaker Cloud API ke platform cloud ESP RainMaker, yang berperan sebagai penghubung antara aplikasi dan perangkat IoT [12].
4. Mikrokontroler ESP32. Perintah dari cloud server (ESP RainMaker) diterima oleh ESP32 melalui koneksi Wi-Fi, kemudian diproses untuk mengontrol output [4],[13].
5. Modul Relay. Relay berfungsi sebagai sakelar elektronik untuk menghubungkan atau memutus arus listrik ke lampu [14].
6. Lampu. Lampu akan menyala atau mati sesuai dengan perintah suara yang diberikan.

B. Parameter Pengujian

Pengujian dilakukan berdasarkan:

1. Response time (waktu respons).
2. Tingkat keberhasilan pengenalan suara (akurasi).

Pengujian dilakukan menggunakan tiga kondisi jaringan:

1. Wi-Fi yang sama dengan ESP32.
2. Wi-Fi berbeda lokasi.
3. Jaringan seluler.

Pengujian akurasi dilakukan dalam kondisi lingkungan tenang dan kondisi kebisingan normal.

III. HASIL DAN PEMBAHASAN

A. Hasil Pengujian Responsivitas

Pengujian pertama dilakukan menggunakan jaringan Wifi IndieHome yang sama dengan yang terkoneksi dengan ESP32, dan dilakukan sebanyak 10 kali percobaan.

Tabel 1. Hasil Pengujian 1. Responsivitas Sistem menggunakan jaringan Wifi yang sama dengan yang terkoneksi dengan ESP32

No.	Perintah	Hasil Pengamatan Relay	Waktu Respon (Detik)
1	ON	Aktif	2 Detik
2	OFF	Tidak aktif	1 Detik
3	ON	Aktif	3 Detik
4	OFF	Tidak aktif	2 Detik
5	ON	Aktif	1 Detik
6	OFF	Tidak aktif	3 Detik
7	ON	Aktif	2 Detik
8	OFF	Tidak aktif	1 Detik
9	ON	Aktif	2 Detik
10	OFF	Tidak aktif	2 Detik

Rata-rata waktu respon sistem adalah 2 detik pada pengujian pertama yang dilakukan menggunakan jaringan Wifi IndieHome yang sama dengan yang terkoneksi dengan ESP32.

Pengujian kedua dilakukan menggunakan jaringan Wifi IndieHome yang berbeda dengan yang terkoneksi dengan ESP32, dalam hal ini menggunakan jaringan Wifi UPTD SMKN 1 Rangas yang berjarak 8 Km dari rumah tempat rangkaian terpasang.

Tabel 2. Hasil Pengujian 2. Responsivitas Sistem menggunakan jaringan Wifi berbeda dengan yang terkoneksi dengan ESP32

No.	Perintah	Hasil Pengamatan Relay	Waktu Respon (Detik)
1	ON	Aktif	3 Detik
2	OFF	Tidak aktif	2 Detik
3	ON	Aktif	3 Detik
4	OFF	Tidak aktif	3 Detik
5	ON	Aktif	2 Detik
6	OFF	Tidak aktif	3 Detik
7	ON	Aktif	3 Detik
8	OFF	Tidak aktif	2 Detik
9	ON	Aktif	2 Detik
10	OFF	Tidak aktif	2 Detik

Rata-rata waktu respon sistem adalah 2,5 detik pada pengujian kedua yang dilakukan menggunakan jaringan Wifi IndieHome yang berbeda dengan yang terkoneksi dengan ESP32.

Pengujian ketiga dilakukan menggunakan jaringan seluler Telkomsel yang terkoneksi dengan smart phone

yang digunakan untuk perintah suara dan ESP32 tetap terkoneksi dengan jaringan Wifi IndiHome yang terpasang pada rumah tempat rangkaian terpasang.

Tabel 3. Hasil Pengujian 3. Responsivitas Sistem menggunakan jaringan Selular Telkomsel.

No.	Perintah	Hasil Pengamatan Relay	Waktu Respon (Detik)
1	ON	Aktif	3 Detik
2	OFF	Tidak aktif	3 Detik
3	ON	Aktif	3 Detik
4	OFF	Tidak aktif	3 Detik
5	ON	Aktif	2 Detik
6	OFF	Tidak aktif	3 Detik
7	ON	Aktif	3 Detik
8	OFF	Tidak aktif	3 Detik
9	ON	Aktif	2 Detik
10	OFF	Tidak aktif	2 Detik

Rata-rata waktu respon sistem adalah 2,7 detik pada pengujian ketiga yang dilakukan menggunakan jaringan selular Telkomsel yang terkoneksi dengan smart phone yang digunakan untuk perintah suara dan ESP32 tetap terkoneksi dengan jaringan Wifi IndiHome yang terpasang pada rumah tempat rangkaian terpasang.

B. Tingkat keberhasilan perintah suara

Pengujian selanjutnya dilakukan untuk menguji keberhasilan sistem mengenali perintah suara. Pengujian keempat dilakukan sebanyak 20 kali di dalam ruangan dengan tingkat kebisingan yang sunyi.

Tabel 4. Hasil Pengujian 4. Tingkat Keberhasilan Perintah Suara di dalam Ruangan.

No.	Perintah	Hasil Pengamatan Relay
1	Nyalakan Lampu Ruang Tamu	Berhasil dikenali
2	Matikan Lampu Ruang Tamu	Berhasil dikenali
3	Nyalakan Lampu Ruang Tengah	Berhasil dikenali
4	Matikan Lampu Ruang Tengah	Berhasil dikenali
5	Nyalakan Lampu Teras 1	Berhasil dikenali
6	Matikan Lampu Teras 1	Berhasil dikenali
7	Nyalakan Lampu Teras 2	Berhasil dikenali
8	Matikan Lampu Teras 2	Berhasil dikenali
9	Nyalakan Lampu Kamar 1	Berhasil dikenali
10	Matikan Lampu Kamar 1	Berhasil dikenali
11	Nyalakan Lampu Kamar 2	Berhasil dikenali
12	Matikan Lampu Kamar 2	Berhasil dikenali
13	Nyalakan Lampu Ruang Tamu	Berhasil dikenali
14	Nyalakan Lampu Ruang Tengah	Berhasil dikenali
15	Nyalakan Lampu Teras 1	Berhasil dikenali
16	Nyalakan Lampu Teras 2	Berhasil dikenali
17	Nyalakan Lampu Kamar 1	Berhasil dikenali
18	Nyalakan Lampu Kamar 2	Berhasil dikenali
19	Matikan semua Lampu	Berhasil dikenali
20	Nyalakan semua lampu	Berhasil dikenali

Hasil pengujian keempat dengan jumlah percobaan sebanyak 20 kali di dalam ruangan dengan tingkat kebisingan yang sunyi, mendapatkan hasil semua perintah suara berhasil dikenali dan tidak ditemukan perintah suara gagal dikenali. Dengan demikian, Persentase keberhasilan yaitu:

$$Akurasi = \frac{20}{20} \times 100\% = 100\%$$

Tingkat akurasi sebesar 100% menunjukkan aplikasi Gemini Asisten yang diintegrasikan dengan platform cloud ESP RainMaker bekerja dengan sangat baik pada kondisi pengujian di dalam ruangan dengan tingkat kebisingan yang sunyi.

Pengujian kelima dilakukan sebanyak 20 kali di luar ruangan dengan tingkat kebisingan yang normal dimana terdengar suara beberapa orang sedang berbicara dan suara kendaraan lalu-lalang.

Tabel 5. Hasil Pengujian 5. Tingkat Keberhasilan Perintah Suara di luar Ruangan.

No.	Perintah	Hasil Pengamatan Relay
1	Nyalakan Lampu Ruang Tamu	Berhasil dikenali
2	Matikan Lampu Ruang Tamu	Berhasil dikenali
3	Nyalakan Lampu Ruang Tengah	Berhasil dikenali
4	Matikan Lampu Ruang Tengah	Gagal dikenali
5	Nyalakan Lampu Teras 1	Berhasil dikenali
6	Matikan Lampu Teras 1	Berhasil dikenali
7	Nyalakan Lampu Teras 2	Berhasil dikenali
8	Matikan Lampu Teras 2	Berhasil dikenali
9	Nyalakan Lampu Kamar 1	Berhasil dikenali
10	Matikan Lampu Kamar 1	Berhasil dikenali
11	Nyalakan Lampu Kamar 2	Berhasil dikenali
12	Matikan Lampu Kamar 2	Gagal dikenali
13	Nyalakan Lampu Ruang Tamu	Berhasil dikenali
14	Nyalakan Lampu Ruang Tengah	Berhasil dikenali
15	Nyalakan Lampu Teras 1	Berhasil dikenali
16	Nyalakan Lampu Teras 2	Berhasil dikenali
17	Nyalakan Lampu Kamar 1	Berhasil dikenali
18	Nyalakan Lampu Kamar 2	Berhasil dikenali
19	Matikan semua Lampu	Berhasil dikenali
20	Nyalakan semua lampu	Berhasil dikenali

Hasil pengujian kelima dengan jumlah percobaan sebanyak 20 kali di luar ruangan dengan tingkat kebisingan yang normal, mendapatkan hasil 18 perintah suara berhasil dikenali dan 2 perintah suara gagal dikenali. Dengan demikian, Persentase keberhasilan yaitu: Akurasi= $18/20 \times 100\% = 90\%$

Tingkat akurasi sebesar 90% menunjukkan aplikasi Gemini Asisten yang diintegrasikan dengan platform cloud ESP RainMaker bekerja dengan baik pada kondisi pengujian di luar ruangan dengan tingkat kebisingan yang normal.

C. Perbandingan dengan Sistem Manual

Tabel 6. Menunjukkan perbandingan antara sistem manual atau menggunakan saklar manual dengan sistem berbasis IoT yang dinilai dari aspek cara operasi, kemudahan, akseibilitas, dan efisiensi.

Tabel 6. Perbandingan dengan Sistem Manual.

Aspek	Sistem Manual	Sistem Berbasis IoT
Cara Operasi	Menggunakan saklar fisik	Perintah suara
kemudahan	Harus mendekati saklar	Dapat dilakukan dari mana saja
Akseibilitas	Terbatas	Lebih ramah lansia/disabilitas
Efisiensi	Bergantung pengguna	Lebih praktis

Dari perbandingan pada tabel 6 menunjukkan sistem berbasis IoT memberikan kemudahan dan fleksibilitas dibandingkan dengan metode konvensional atau sistem manual.

IV. KESIMPULAN DAN SARAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem pengontrolan lampu berbasis IoT menggunakan ESP32 yang terintegrasi dengan Gemini Asisten dan ESP RainMaker. Sistem mampu bekerja secara *real-time* dengan rata-rata waktu respons 2–2,7 detik. Tingkat akurasi pengenalan suara mencapai 100% pada kondisi tenang dan 90% pada kondisi kebisingan normal.

Sistem terbukti responsif, akurat, dan lebih praktis dibandingkan metode manual konvensional, serta berpotensi mendukung pengembangan *smart home* yang terjangkau dan mudah diimplementasikan.

Secara sederhana, Gemini asisten merupakan AI pintar yang bisa diajak berpikir dan berdiskusi, sedangkan Google Assistant dan Amazon Alexa lebih ke alat perintah cepat. Keunggulan Gemini dibanding sistem lama antara lain, memiliki kemampuan berpikir dan analisis jauh lebih tinggi, bahasa alami dan percakapan lebih fleksibel, kontrol data dan privasi lebih fleksibel, serta arah masa depan (AI utama Google).

UCAPAN TERIMA KASIH

Pada penyusunan penelitian ini ditemukan banyak kendala, namun berkat bantuan dari berbagai pihak, baik berupa saran, kritikan, bimbingan dan nasehat maka penulis dapat menyelesaikan penelitian ini dengan baik.

Tak ada gading yang tak retak, penulis menyadari dalam penulisan penelitian ini masih terdapat kekurangan dan kekeliruan. Oleh karena itu, dengan penuh kerendahan hati penulis mohon maaf. Akhir kata semoga bantuan yang telah diberikan dari semua pihak baik moril maupun materil mendapat balasan dari Allah SWT, Amin yaa rabbal alamin.

DAFTAR PUSTAKA

- [1] A. D. Limantara, Y. C. S. Purnomo, and S. W. Mudjanarko, "Pemodelan Sistem Pelacakan Lot Parkir Kosong Berbasis Sensor Ultrasonic dan Internet Of Things (IoT) Pada Lahan Parkir Diluar Jalan" Prosiding - Seminar Nasional Sains dan Teknologi 2017.
- [2] P.P. Ray, "A survey on Internet of Things architectures", Journal of King Saud University - Computer and Information Sciences, 2018.
- [3] A. Junaidi, "Internet of Things, Sejarah, Teknologi dan Penerapannya," JITTER, vol. 1, no. 3, 2015.
- [4] A. Imran and M. Rasul, "Pengembangan Tempat Sampah Pintar Menggunakan ESP32," Jurnal Media Elektrik, vol. 17, no. 2, 2020.
- [5] H. Andrianto and G. I. Saputra, "Smart Home System Berbasis IoT dan SMS," TELKA, vol. 6, no. 1, pp. 40–48, 2020.
- [6] R. I. Nugraha and A. R. Nugraha, "Simulasi Smart Home Berbasis Arduino," 2018.
- [7] E. S. Rahayu and R. A. M. Nurdin, "Perancangan Smart Home Untuk Pengendalian Peralatan Elektronik Dan Pemantauan Keamanan Rumah Berbasis Internet Of Things', *Jurnal Teknologi*, 2019.
- [8] M. Muslihudin. "Implementasi Aplikasi Rumah Pintar Berbasis Android dengan Arduino Mikrokontroler" 2018.
- [9] Ogedebe, Peter M., and Babatunde Peter Jacob. "Software Prototyping: A Strategy to Use When User Lacks Data Processing Experience". *ARPJ Journal of Systems and Software*, vol. 2, no. 6, 2012.
- [10] A. V. Oppenheim and R. W. Schaffer. 'Discrete-Time Signal Processing, 3rd ed. Pearson, 2010.
- [11] D. Jurafsky and J. H. Martin. 'Speech and Language Processing', 3rd ed., Pearson, 2023.
- [12] Martins, Oluwaseun O., et al. "Development of a Sequential Neural Network Model for Bottle-Fill Level Detection and Classification." *FUOYE Journal of Engineering and Technology*, vol. 8, no. 3, 2023.
- [13] H. Kusumah and R. A. Pradana, "Penerapan Trainer Interfacing Mikrokontroler dan Internet Of Things Berbasis ESP32 pada Mata Kuliah Interfacing" *Jurnal Cerita*, 2019.
- [14] K. Fayakun, A. Afandi, F. Afifah, and H. Ramza, "Universitas 17 Agustus 1945 Jakarta," 2018.