

SISTEM SMART HOME BERBASIS ESP32 DENGAN RTOS DAN MQTT SERTA APLIKASI FLUTTER UNTUK MONITORING REAL-TIME

Cindy Amelia Prameswari¹, Irfan Sriyono Putro²

^{1,2}Universitas Singaperbangsa Karawang, Jl. HS. Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361, Telp. (0267) 641177.

Keywords:

Smart Home, IoT, ESP32.

Correspondent Email:

cindyameliaprameswari@gmail.com



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

Abstrak. Smart Home berbasis Internet of Things (IoT) menjadi salah satu solusi modern untuk meningkatkan kenyamanan, keamanan, dan efisiensi pemantauan lingkungan rumah. Penelitian ini dilakukan untuk merancang dan mengimplementasikan sistem Smart Home menggunakan ESP32 yang terintegrasi dengan enam sensor dan enam aktuator guna menghasilkan pemantauan real-time serta otomasi berbasis rule. Tujuan utama penelitian ini ialah membangun sistem yang mampu membaca kondisi lingkungan, mengirimkan data secara paralel, serta mengeksekusi tindakan otomatis tanpa intervensi pengguna. Metode yang digunakan meliputi perancangan perangkat keras, pemrograman berbasis FreeRTOS untuk multitasking, komunikasi data melalui protokol MQTT, penyimpanan historis menggunakan Firebase Firestore, serta pembuatan aplikasi Flutter sebagai antarmuka monitoring. Hasil pengujian menunjukkan bahwa seluruh sensor mampu memberikan data stabil, sedangkan aktuator merespons sesuai rule otomatis yang ditetapkan. Pengiriman data melalui MQTT dan Firebase berjalan lancar, sementara aplikasi Flutter menampilkan nilai sensor, status, serta riwayat secara real-time. Kesimpulan dari penelitian ini menunjukkan bahwa sistem Smart Home yang dikembangkan berfungsi responsif, akurat, dan dapat digunakan sebagai dasar pengembangan sistem otomasi rumah yang lebih canggih di masa depan.

Abstract. Smart Home systems based on the Internet of Things (IoT) have emerged as modern solutions for improving comfort, safety, and efficiency in residential environments. This study aims to design and implement a Smart Home system using an ESP32 microcontroller integrated with six sensors and six actuators to enable real-time monitoring and rule-based automation. The main objective is to develop a system capable of reading environmental conditions, transmitting data in parallel, and performing automated actions without user intervention. The methodology includes hardware design, FreeRTOS-based programming for multitasking, data communication using the MQTT protocol, historical data storage through Firebase Firestore, and the development of a Flutter application as the user interface. The test results show that all sensors provide stable readings, while the actuators respond accurately according to predefined automation rules. Data transmission through MQTT and Firebase performs smoothly, and the Flutter application successfully displays sensor values, statuses, and historical logs in real time. The findings indicate that the proposed Smart Home system operates

responsively and reliably, making it a suitable foundation for developing more advanced home automation applications in the future.

1. PENDAHULUAN

Perkembangan teknologi Internet of Things (IoT) dalam beberapa tahun terakhir telah membawa transformasi besar dalam otomatisasi dan monitoring perangkat elektronik. IoT memungkinkan perangkat fisik terhubung ke jaringan internet untuk mengumpulkan, memproses, dan mengirim data secara real-time. Menurut [1], IoT merupakan teknologi yang menghubungkan objek fisik agar dapat bertukar informasi secara otomatis dan efisien melalui jaringan terdistribusi. Teknologi ini menjadi fondasi dalam pengembangan berbagai sistem cerdas, termasuk Smart Home. Salah satu implementasi IoT yang paling berkembang adalah Smart Home. Sistem ini dirancang untuk meningkatkan keamanan, efisiensi energi, dan kenyamanan penghuni rumah melalui perangkat yang dapat bekerja secara otomatis. Menurut [2], *Smart Home* mampu mengintegrasikan sensor dan aktuator untuk memantau kondisi lingkungan dan mengambil tindakan otomatis seperti mengontrol cahaya, suhu ruangan, dan keamanan rumah. Penerapan sensor lingkungan menjadi aspek penting dalam *Smart Home*. Menurut [3], penggunaan beberapa sensor dalam satu sistem IoT meningkatkan akurasi deteksi serta memperluas fungsionalitas otomatisasi rumah, terutama dalam aspek keamanan dan keselamatan penghuni. Pemilihan mikrokontroler yang tepat juga menjadi faktor keberhasilan sistem Smart Home.

Menurut [4], mikrokontroler ESP32 sangat cocok digunakan pada sistem IoT karena memiliki konektivitas Wi-Fi, kemampuan pemrosesan yang cepat, serta dapat terintegrasi dengan berbagai sensor untuk mendeteksi kondisi lingkungan secara *real-time* kepada pengguna. ESP32 memungkinkan pengolahan data sensor secara paralel dan hemat daya. Teknologi pendukung seperti RTOS (*Real-Time Operating System*) dan MQTT semakin memperkuat keandalan sistem IoT. RTOS memungkinkan pembacaan sensor dilakukan secara multitasking dan *real-time*, sedangkan MQTT menyediakan komunikasi ringan antarperangkat. Aplikasi berbasis Flutter juga

mempermudah pengguna untuk memonitor kondisi rumah melalui smartphone. Pada arsitektur IoT modern, pengelolaan dan penyimpanan data menjadi komponen krusial. Data sensor yang dibaca oleh ESP32 menggunakan RTOS dan dipublikasikan melalui MQTT membutuhkan penyimpanan cloud yang stabil dan skalabel [5].

Menurut [6], Firebase mampu menyimpan data mikrokontroler secara real-time dan mudah diakses melalui aplikasi Android. Gap Analysis Penelitian sebelumnya atau proyek UTS yang telah dilakukan hanya menggunakan tiga sensor dan memiliki lingkup yang lebih sederhana. Sistem tersebut belum memanfaatkan keseluruhan potensi IoT modern seperti integrasi sensor yang lebih beragam, penggunaan RTOS untuk pemrosesan paralel, komunikasi MQTT yang stabil, serta penyimpanan cloud melalui Firebase untuk kebutuhan historis [7]. Pengembangan pada penelitian ini menghadirkan peningkatan signifikan melalui integrasi enam sensor dan enam aktuator, penggunaan ESP32 berbasis RTOS, pemanfaatan MQTT dan Firebase Firestore, serta visualisasi data pada aplikasi Flutter. Kombinasi komponen-komponen ini menciptakan sistem Smart Home yang lebih kompleks, responsif, serta lebih mendekati praktik implementasi pada sistem IoT nyata. Tujuan utamanya adalah menyelesaikan permasalahan terkait pemantauan kondisi lingkungan rumah secara real-time dan penyimpanan historis yang terstruktur untuk kebutuhan analisis maupun monitoring pengguna [8].

2. TINJAUAN PUSTAKA

2.1 *Internet of Things* (IoT)

Internet of Things (IoT) adalah konsep yang memungkinkan berbagai perangkat fisik saling terhubung melalui jaringan internet untuk mengumpulkan, mengirim, dan memproses data secara otomatis. Menurut [1], IoT merupakan teknologi yang menghubungkan objek fisik agar dapat melakukan pertukaran data secara real-time untuk meningkatkan efisiensi sistem. IoT terdiri dari sensor,

aktuator, mikrokontroler, protokol komunikasi, dan aplikasi pengguna.

2.2 Smart Home

Smart Home merupakan penerapan IoT dalam lingkungan rumah tangga yang bertujuan meningkatkan kenyamanan, keamanan, dan efisiensi energi. Sistem ini memungkinkan perangkat rumah seperti lampu, kipas, alarm, dan sensor lingkungan dikendalikan secara otomatis. Menurut [2], Smart Home mampu mengintegrasikan berbagai sensor untuk mendeteksi suhu, gas, cahaya, dan gerakan sehingga rumah dapat merespons kondisi lingkungan tanpa intervensi langsung pengguna.

2.3 MQTT (*Message Queue Telemetry Transport*)

MQTT adalah protokol komunikasi lightweight berbasis arsitektur publish-subscribe yang dirancang untuk perangkat IoT dengan keterbatasan bandwidth. Menurut [9], MQTT sangat efisien digunakan pada perangkat IoT karena hemat daya, cepat, dan mampu mengirim data secara real-time melalui broker sebagai penghubung [10].

2.4 RTOS (*Real-Time Operating System*)

RTOS merupakan sistem operasi yang dirancang khusus untuk menangani proses secara real-time dengan tingkat determinisme tinggi, dimana pada mikrokontroler ESP32, FreeRTOS memungkinkan pembuatan task paralel untuk setiap sensor dan aktuator sehingga setiap komponen dapat beroperasi secara independen tanpa menghambat proses lainnya, dan penggunaan FreeRTOS meningkatkan stabilitas serta responsivitas sistem IoT karena setiap proses dapat berjalan secara multitasking dengan prioritas yang terjadwal untuk memastikan task dengan prioritas tinggi seperti deteksi bahaya dapat dieksekusi tepat waktu [11].

2.5 Firebase

Firebase adalah *platform* pengembangan aplikasi berbasis *cloud* yang dikembangkan oleh Google, menyediakan berbagai layanan backend seperti *database real-time*, autentikasi, cloud storage, hosting, dan analytics yang terintegrasi untuk memudahkan pengembangan aplikasi mobile dan web. Platform ini sangat populer dalam ekosistem IoT karena menawarkan kemudahan integrasi, skalabilitas tinggi, dan kemampuan sinkronisasi data secara real-time tanpa memerlukan konfigurasi server

yang kompleks [5]. Firebase menyediakan SDK (*Software Development Kit*) untuk berbagai platform seperti Android, iOS, Web, dan Flutter, memungkinkan developer untuk mengimplementasikan fitur *backend* dengan cepat tanpa harus membangun infrastruktur server dari awal.

2.6 Flutter Untuk IOT

Flutter merupakan framework open-source yang dikembangkan oleh Google untuk membangun aplikasi mobile cross-platform dengan satu basis kode untuk Android dan iOS, yang sangat cocok digunakan dalam proyek IoT karena mendukung integrasi protokol komunikasi seperti MQTT dan dapat menampilkan data sensor secara real-time melalui antarmuka yang responsif dengan performa tinggi mendekati aplikasi [12].

2.8 Akuator

1. Relay 1 → Buzzer [13].
2. Relay 2 → Kipas
3. Kipas DC
4. LED
5. Microservo
6. LCD 16x2 I2C LCD 16x2 dengan modul I2C [7]

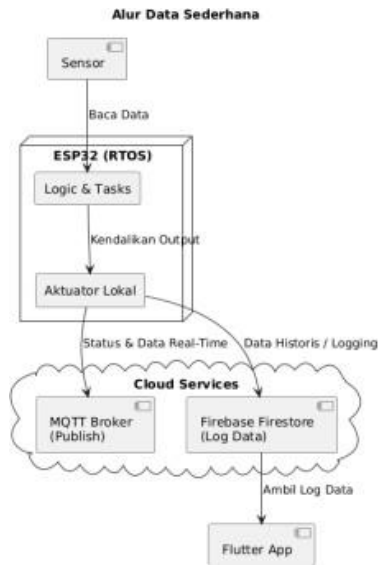
2.9 Power Adaptor 5v 2A

Power adaptor 5V 2A berfungsi sebagai sumber catu daya utama yang digunakan untuk memberikan tegangan stabil kepada rangkaian Smart Home berbasis ESP32. Adaptor ini menghasilkan keluaran sebesar 5 volt dengan arus maksimal 2 ampere, sehingga mampu memasok kebutuhan daya berbagai komponen seperti mikrokontroler, sensor, serta beberapa aktuator berdaya rendah seperti servo, modul relay, dan kipas DC kecil [14].

3. METODE PENELITIAN

3.1 Arsitektur Sistem

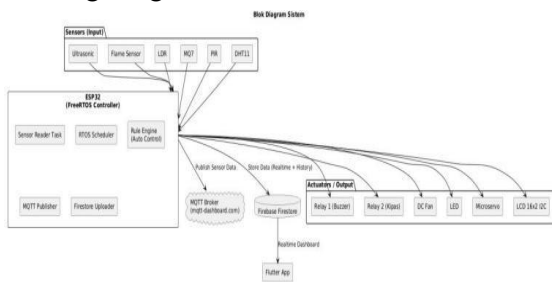
Perancangan arsitektur sistem Smart Home ini dibuat untuk mengintegrasikan enam sensor dan enam aktuator melalui mikrokontroler ESP32 yang berjalan menggunakan FreeRTOS, serta terhubung dengan aplikasi mobile berbasis Flutter melalui protokol komunikasi MQTT. Alur utama komunikasi dan pemrosesan sistem dapat digambarkan sebagai berikut:



Gambar 3.1 Arsitektur system

ESP32 membaca seluruh sensor secara paralel [15] menggunakan task-task FreeRTOS, kemudian mengirimkan data real-time ke MQTT Broker melalui topik tertentu. Secara paralel, ESP32 juga mengirimkan data sensor yang sama ke *Firebase Firestore*. Aplikasi Flutter akan berperan sebagai subscriber data dari Firestore dan menampilkannya kepada pengguna dalam bentuk dashboard, serta halaman detail sensor. Arsitektur ini memastikan sistem berjalan secara real-time, responsif, efisien, dan *scalable* untuk dikembangkan pada perangkat IoT lainnya.

3.2 Blog Diagram Sistem



Gambar 3.2 Blog Diagram Sistem Sistem Smart Home terdiri dari enam

sensor (Ultrasonik, Flame, LDR, MQ-7, PIR, dan DHT11) yang seluruhnya dibaca oleh ESP32. Mikrokontroler ini menangani pembacaan sensor, eksekusi task RTOS, pengiriman data ke MQTT Broker dan Firebase Firestore, serta kendali enam aktuator yang meliputi relay buzzer, relay kipas, kipas DC, buzzer, microservo, dan LCD I2C. Seluruh proses berjalan paralel melalui RTOS Task

Scheduler. Dua jalur komunikasi digunakan, yaitu MQTT untuk monitoring pihak ketiga dan Firebase Firestore untuk penyimpanan historis yang kemudian ditampilkan melalui aplikasi Flutter.

3.3 Perancangan RTOS

FreeRTOS digunakan agar setiap proses berjalan stabil. Task pembacaan sensor mencakup DHT11, MQ-7, LDR, PIR, ultrasonik, dan flame sensor. Task output digunakan untuk LCD dan buzzer, dan task komunikasi menangani MQTT. Prioritas task disesuaikan dengan tingkat urgensi, misalnya sensor gas dan api mendapat prioritas tertinggi. Dengan cara ini, sistem tetap real-time dan tidak saling mengganggu.

3.4 Perancangan Mode Otomatis

Mode otomatis memungkinkan sistem merespons sendiri kondisi lingkungan. MQ-7 mengaktifkan kipas dan alarm ketika CO tinggi, flame sensor memicu alarm darurat, LDR menyalakan atau mematikan lampu, PIR menggerakkan servo, DHT11 mengatur kipas berdasarkan suhu serta menampilkan “DHT ERR” bila gagal, dan sensor ultrasonik memicu buzzer ketika jarak objek terlalu dekat. Semua rule dijalankan paralel melalui FreeRTOS.



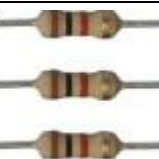
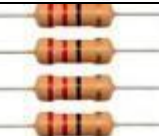
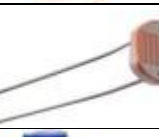





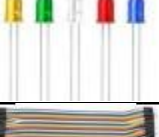

3.5 Perancangan Aplikasi Flutter



Aplikasi Flutter berfungsi sebagai pusat monitoring. Dashboard menampilkan enam card sensor lengkap dengan nilai, status, ikon, dan tombol detail. Pada halaman detail, pengguna dapat melihat tabel log berisi waktu dan nilai sensor serta status normal atau alert

4. HASIL DAN PEMBAHASAN



4.1 Implementasi Perangkat Keras


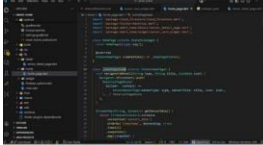
	Esp 32
	Relay
	ultrasonik

	Kipas Mini Dc 5 Volt
	Buzzer 5Volt
	Resist 1k
	Resistor 220ohm
	Sensor Ldr
	microservo
	Sensor pir
	Resistor 10k
	Sensor dht
	LCD 16x2 i2c
	LED
	Kabel Jumper

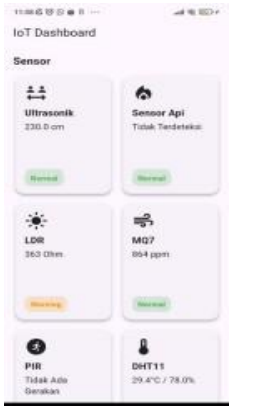
	Power Adaptor 5V 2A
	Rakitan semua alat


4.2 Implementasi Perangkat Lunak

	<p>proses monitoring data sensor secara real-time melalui HiveMQ WebSocket Client. ESP32 terlihat berhasil terhubung dengan status connected dan telah berlangganan pada topik “kelompok6/sensors”. Setiap beberapa detik, ESP32 mengirimkan data keenam sensor dalam format JSON, mencakup ultrasonic, flame, MQ-7, LDR, PIR, suhu, dan kelembapan. Seluruh data muncul pada bagian Messages, menunjukkan bahwa proses publish berjalan lancar dan broker HiveMQ mampu menerima data secara stabil</p>
	<p>Penyimpanan paralel ke Firebase Firestore menampilkan koleksi “sensors_data”</p>


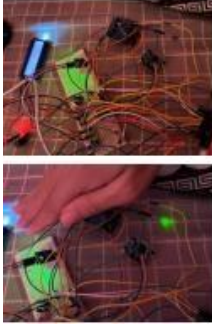
	<p>berisi paket lengkap nilai sensor dan timestamp, memungkinkan Flutter menampilkan riwayat, grafik, serta monitoring.</p>
	<p>bagian awal kode Arduino IDE yang memuat library untuk Firebase, FreeRTOS, WiFi, MQTT, sensor, servo, dan LCD. Struktur ini mendukung multitasking, pengiriman data, penulisan ke Firebase, serta kontrol sensor dan aktuator.</p>
	<p>pembuatan aplikasi Flutter yang menampilkan data sensor melalui Stream Firestore, navigasi detail, dan dashboard. Struktur folder lengkap membuktikan aplikasi terhubung dengan Firestore dan UI Smart Home berjalan responsif</p>



4.3 Implementasi flutter

	<p>Dashboard Flutter menampilkan enam sensor Smart Home dalam card berisi nilai real-time, status, dan ikon. Data diambil melalui Stream Firestore yang diperbarui ESP32. Pengguna dapat memantau kondisi rumah dengan cepat serta</p>
---	--

	<p>membuka detail untuk melihat informasi lanjutan dan riwayat sensor.</p>
	<p>Halaman detail Ultrasonic menampilkan jarak real-time, rule otomatis pemicu <10 cm yang mengaktifkan buzzer, serta riwayat pembacaan dari Firestore. Data tampil responsif, membantu pengguna memantau kondisi dan memahami status bahaya maupun normal dengan jelas.</p>

4.2 Pengujian sistem

	<p>Semua komponen dirangkai pada breadboard dan dihubungkan ke ESP32. Penyusunan wiring ini menjadi tahap penting untuk memastikan seluruh sensor dapat dibaca oleh mikrokontroler dan aktuator dikendalikan sesuai rule otomatis</p>
	<p>Gambar menunjukkan pengujian sensor LDR pada sistem Smart Home. LDR membaca intensitas cahaya melalui perubahan resistansi, diteruskan ke ESP32 melalui rangkaian pembagi tegangan, bersama LED, relay, dan LCD.</p>

	<p>flame sensor terpasang pada rangkaian utama bersama ESP32 dan LCD 16x2 I2C. Sensor ini dihubungkan dengan pin input pada mikrokontroler untuk mengirimkan sinyal deteksi ketika ada radiasi IR yang kuat</p>
	<p>Gambar menunjukkan pengujian flame sensor dengan sumber api. Sensor merespons cepat, menyalakan indikator, memicu buzzer, menampilkan peringatan pada LCD, dan menjalankan rule keselamatan, membuktikan deteksi kebakaran bekerja akurat.</p>

Smart Home berbasis ESP32 yang dikembangkan menunjukkan kinerja yang stabil dan selaras dengan tujuan utama perancangan, yaitu menghasilkan pemantauan dan otomatisasi yang berjalan real-time. Integrasi enam sensor dan enam aktuator terbukti memberikan cakupan pemantauan lingkungan rumah yang luas. Sensor ultrasonik, flame, LDR, MQ-7, PIR, dan DHT11 mampu memberikan data yang konsisten ketika diuji, sedangkan enam aktuator yang mencakup relay buzzer, relay kipas, kipas DC, buzzer, microservo, dan LCD merespons sesuai rule otomatis yang telah ditetapkan. Hal ini memperlihatkan bahwa alur pemrosesan berbasis FreeRTOS pada ESP32 berjalan selaras dan tidak saling menghambat antar-task. Pada tahap implementasi komunikasi data,

pengujian melalui HiveMQ WebSocket Client membuktikan bahwa pengiriman data menggunakan protokol MQTT berlangsung mulus. ESP32 berhasil mempertahankan koneksi ke broker publik dan mampu mempublikasikan nilai sensor secara periodik dalam format JSON.

Setiap perubahan pembacaan sensor langsung muncul di halaman pesan broker, menandakan bahwa mekanisme publish-subscribe bekerja efektif. Pengiriman paralel ke Firebase Firestore memberikan bukti tambahan bahwa data mampu disimpan secara terstruktur tanpa menyebabkan gangguan pada proses MQTT. Struktur koleksi dan dokumen di Firestore tersusun rapi, masing-masing menyimpan nilai sensor lengkap dengan timestamp yang memudahkan analisis historis. Respons sensor terhadap kondisi nyata juga menunjukkan konsistensi sistem. Pada sensor LDR, perubahan intensitas cahaya ketika sebagian permukaannya ditutup menghasilkan respon yang muncul langsung pada LCD maupun aplikasi. Hal ini menunjukkan bahwa pemrosesan nilai analog ke digital pada ESP32 dan rule otomatis terkait pencahayaan berjalan sesuai rancangan. Sensor flame memberikan reaksi cepat pada saat dikenai api, ditunjukkan dengan aktifnya indikator serta perubahan output yang kemudian memicu buzzer dan pesan peringatan pada LCD.

5. KESIMPULAN

- a. Sistem Smart Home berbasis ESP32 dengan FreeRTOS, MQTT, Firebase, dan Flutter berfungsi stabil, responsif, serta mampu memantau enam sensor dan enam aktuator secara real-time.
- b. RTOS meningkatkan multitasking, sementara pengiriman paralel ke MQTT dan Firebase memungkinkan monitoring dan penyimpanan historis berjalan lancar melalui aplikasi Flutter.
- c. Pengembangan dapat mencakup notifikasi darurat, sensor lebih akurat, modul keamanan tambahan, AI prediktif, serta antarmuka dan redundansi komunikasi yang lebih baik.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] A. A. Jalil, M. J. Abdullah, M. Simatupang, and H. T. Mokui, "Systematic Literature Review: Application of Internet of Things (IoT) for Smart Home Optimization," vol. 18, no. 2, pp. 79–90, 2025, [Online]. Available: <https://doi.org/10.47561/jtik.v18i2.338>
- [2] A. N. Fathoni and K. Khotimah, "Rancang Bangun Smart Home berbasis IoT Menggunakan Telegram Messenger Bot dan NodeMCU ESP 32 Smart Home Design Based on IoT Using Telegram Messenger Bot and NodeMCU," vol. 9, no. 1, pp. 34–43, 2023.
- [3] P. K. . Windesi, M. R. Sampebua, and R. M. Kmurawak, "IOT-BASED HOME AUTOMATION USING NODEMCU ESP8266," vol. 4, no. 4, pp. 391–396, 2022.
- [4] M. V. Gultom and I. S. Putro, "SISTEM DETEKSI KEBAKARAN BERBASIS DENGAN MIKROKONTROLER ESP32," vol. 13, no. 2, 2025, [Online]. Available: <https://doi.org/10.23960/jitet.v13i2.6236>
- [5] Siswanto, Firdiansyah, M. Anif, and B. H. Prasetyo, "Kendali dan Monitoring Ruang Server dengan Sensor Suhu DHT-11 , Gas MQ-2 serta Notifikasi SMS," pp. 122–130.
- [6] J. W. Jokanan, A. Widodo, N. Kholis, and L. Rakhmawati, "Rancang Bangun Alat Monitoring Daya Listrik Berbasis IoT Menggunakan Firebase Dan Aplikasi Android Arif Widodo , Nur Kholis , Lusia Rakhmawati," pp. 47–55, [Online]. Available: <https://doi.org/10.26740/jte.v11n1.p47-55>
- [7] A. Y. Rangan, A. Yusnita, and M. Awaludin, "Sistem Monitoring berbasis Internet of things pada Suhu dan KelembabanUdara di Laboratorium Kimia XYZ," vol. 4, no. 2, pp. 168–183, 2020, [Online]. Available: <https://doi.org/10.37339/e-komtek.v4i2.404>
- [8] B. Novriditiyo, Sugiri, and A. Nurcahyo, "PENGAPLIKASIAN SENSOR JARAK ULTRASONIK HC-SR04 PADA DRONE," vol. 7, no. 2, pp. 216–225, [Online]. Available: <https://doi.org/10.56521/teknika.v7i2.322>
- [9] P. Rahayu, Sularno, and I. U. Sari, "Perancangan Sistem Smart Home Berbasis IoT Menggunakan ESP32 dan Aplikasi Blynk untuk Otomatisasi Perangkat Rumah Tangga," vol. 3, no. 2, pp. 95–100, 2025.
- [10] J. P. Aljabbar, M. R. F. Rajaguguk, and N. Wijaya, "Studi Literatur Keamanan Data Dalam Sistem Smart Home Berbasis IoT," vol. 6, no. 1, pp. 245–255, 2025.
- [11] R. A. Setyawan, "Penerapan Firebase Realtime Database Pada Aplikasi Catatan Harian Diabetes Melitus," vol. 22, no. 1, pp. 1–9, 2024, [Online]. Available: <https://doi.org/10.61805/fahma.v22i1.102>
- [12] A. Herlina, A. D. Firdausi, S. Maulana, A. Habibi, and V. Dirgantara, "IMPROVEMENT OF SCIENCE AND TECHNOLOGY THROUGH INTERNET OF THINGS (IOT) FOR SMART HOME APPLICATION TRAINING FOR STUDENTS OF SMK NURUL JADID PROBOLINGGO," vol. 5, no. 2, pp. 274–286, 2021.
- [13] A. Kurnianto, J. D. Irawan, and F. T. Industri, "PENERAPAN IOT (INTERNET OF THINGS) UNTUK CONTROLLING LAMPU MENGGUNAKAN PROTOKOL MQTT BERBASIS WEB," vol. 6, no. 2, 2022, [Online]. Available: <https://doi.org/10.36040/jati.v6i2.5393>
- [14] V. K. Pratifi, A. T. Sasongko, and D. Afandi, "Integration of DHT11 and PIR Sensors in Indoor Temperature Automation and Motion Detection System Using Arduino Nano Microcontroller Integrasi Sensor DHT11 dan PIR dalam Sistem Otomatisasi Suhu dan Deteksi Gerakan dalam Ruangan Menggunakan

- Mikrokontroler ,” vol. 4, no. July, pp. 1148–1159, 2024, [Online]. Available: <https://doi.org/10.57152/malcom.v4i3.1490>
- [15] Zulhijayanto and A. Fadlil, “Desain Sistem Monitoring dan Penyiraman Tanaman Tomat Berbasis Internet of Things (IoT),” vol. 4, no. 2, pp. 94–104, 2022, doi: 10.12928/biste.v4i2.5884.