

Rancang Bangun Sistem Deteksi Dini Kebakaran Rumah Adat Kilapan Sumatra Selatan Berbasis Sensor DHT11 dan MQ-2 Dengan ESP32

Dafa Aminata^{1*}, Hafid Dwi Nugroho², Fajar Malik Abdillah³, Rudi Susanto⁴

¹Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta

^{1*}240103128@mhs.udb.ac.id

²Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta

²240103261@mhs.udb.ac.id

³Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta

³240103131@mhs.udb.ac.id

⁴Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta

⁴rudi_susanto@udb.ac.id

Abstrak— Kebakaran rumah tradisional masih menjadi ancaman serius, terutama pada bangunan berbahan kayu seperti Rumah Adat Kilapan di Sumatera Selatan. Kurangnya sistem pemantauan otomatis menyebabkan keterlambatan dalam penanganan awal kebakaran. Penelitian ini bertujuan merancang sistem deteksi dini kebakaran menggunakan sensor suhu DHT11 dan sensor gas MQ-2 yang terintegrasi dengan mikrokontroler ESP32. Sistem ini mampu mendeteksi peningkatan suhu dan konsentrasi gas berbahaya secara real-time, lalu mengirimkan notifikasi melalui platform Telegram. Pengujian dilakukan dengan memanaskan sensor secara bertahap dan mensimulasikan asap, untuk mengevaluasi respons sensor dan sistem komunikasi. Hasil menunjukkan bahwa sistem mampu mendeteksi suhu di atas 45°C dan kadar gas di atas 500 ppm dengan respons yang cepat dan akurat. Sistem ini diharapkan dapat menjadi solusi sederhana namun efektif untuk mencegah kebakaran pada rumah adat berbahan mudah terbakar.

Kata kunci—Deteksi Kebakaran, Rumah Adat Kilapan, ESP32, DHT11, MQ-2.

Abstract— Traditional houses remain highly vulnerable to fire hazards, especially those made of flammable materials such as wood, like the Kilapan traditional house in South Sumatra. The absence of an automatic monitoring system often results in delayed emergency responses. This study aims to design an early fire detection system using the DHT11 temperature sensor and the MQ-2 gas sensor, integrated with the ESP32 microcontroller. The system detects abnormal increases in temperature and gas concentration (e.g., smoke) in real time and sends alerts via the Telegram platform. Testing was conducted by gradually heating the sensors and simulating smoke exposure to evaluate sensor response and communication reliability. Results indicate that the system can detect temperatures above 45°C and gas concentrations above 500 ppm accurately and promptly. This system offers a simple yet effective solution for early fire prevention in traditional wooden structures..

Keywords—Fire Detection, Kilapan Traditional House, ESP32, DHT11, MQ-2.

I. PENDAHULUAN

Kebakaran termasuk peristiwa bencana yang kerap terjadi, terutama di area permukiman padat dan bangunan berbahan yang mudah terbakar[1]. Kejadian kebakaran dapat disebabkan oleh berbagai faktor, seperti hubungan arus pendek, kebocoran gas, serta kelalaian manusia dalam penggunaan api. Risiko ini semakin tinggi pada bangunan yang menggunakan bahan mudah terbakar, seperti kayu[1].

Bangunan tradisional seperti rumah adat Kilapan di Sumatera Selatan yang sebagian besar berbahan

dasar kayu memiliki tingkat kerentanan yang tinggi terhadap kebakaran. Namun, hingga saat ini belum banyak rumah adat yang dilengkapi dengan sistem deteksi dini untuk mencegah terjadinya kebakaran[2]. Hal ini menjadi masalah serius, mengingat rumah adat merupakan bagian dari warisan budaya yang perlu dilestarikan.

Kemajuan teknologi sangat pesat khususnya di Internet Of Things(IOT), sehingga memungkinkan pengembangan sistem deteksi kebakaran yang bekerja secara otomatis dan memberikan respons secara real-time. Dengan mengintegrasikan

mikrokontroler ESP32, sensor suhu DHT11, dan sensor gas MQ-2, sistem dapat mendeteksi potensi kebakaran melalui peningkatan suhu atau konsentrasi asap, serta mengirimkan notifikasi ke pengguna[2][3].

Penelitian Rokhayana et al. (2025) menunjukkan bahwa sistem berbasis ESP32 mampu memantau kondisi suhu dan asap serta memberikan peringatan cepat melalui aplikasi Telegram[2]. Sementara itu, penelitian Fachry et al. (2021) berhasil mengembangkan sistem pemadam kebakaran otomatis menggunakan ESP32 dengan kombinasi sensor DHT22, MQ-2, dan flame sensor, serta dilengkapi fitur notifikasi Telegram untuk mempercepat respons pengguna[3].

Tujuan dari penelitian ini adalah merancang dan mengimplementasikan sistem deteksi kebakaran otomatis berbasis ESP32, dengan integrasi sensor DHT11 dan sensor MQ-2, yang diterapkan pada simulasi rumah adat kilapan Sumatra Selatan. Sistem ini diharapkan dapat memberikan solusi teknologi yang efektif dalam mencegah kebakaran serta mendukung pelestarian bangunan tradisional.

II. METODOLOGI PENELITIAN

Bagian ini memaparkan secara rinci tahapan-tahapan yang telah dilakukan dalam proses penelitian. Kami akan menjelaskan seluruh proses, mulai dari tahap pencarian dan penentuan konsep, pembuatan sketsa rangkaian kabel, hingga pengumpulan alat dan bahan yang dibutuhkan untuk merealisasikan proyek ini.

2.1. Menentukan Konsep

Konsep sistem yang dirancang adalah deteksi dini kebakaran berbasis ESP32 dengan dukungan sensor suhu DHT11, sensor gas/asap MQ-2, dan buzzer sebagai alarm lokal. Sistem ini dirancang untuk dapat mengukur suhu dan mendeteksi asap, lalu mengirimkan peringatan otomatis melalui aplikasi Telegram jika terdeteksi kondisi bahaya. Pemilihan alat ini disesuaikan dengan kebutuhan sistem yang bersifat real-time, hemat daya, dan mudah diterapkan pada lingkungan rumah adat.

2.2 Menyiapkan Alat dan Bahan

Tahap ini dilakukan dengan mempelajari cara kerja masing-masing komponen yang digunakan. Sensor DHT11 digunakan untuk mengukur suhu lingkungan, sedangkan MQ-2 digunakan untuk mendeteksi keberadaan asap atau gas berbahaya. Kedua sensor ini dihubungkan ke mikrokontroler ESP32, yang akan memproses data dan mengaktifkan buzzer jika nilai pembacaan melebihi ambang batas. Selain itu, ESP32 juga diprogram untuk mengirimkan notifikasi melalui jaringan Wi-Fi ke Telegram. Penyusunan skema koneksi dan logika kerja dilakukan sebelum sistem dirakit, agar rangkaian berfungsi sesuai tujuan. Berikut alat dan bahan yang akan kami gunakan untuk membuat project tersebut.

1. ESP32

ESP32 adalah mikrokontroler 32-bit yang dilengkapi dengan modul Wi-Fi dan Bluetooth, cocok untuk aplikasi IoT karena performanya yang cepat, rendah konsumsi energi, dan dukungan input/output yang lengkap, dilengkapi dengan antena nirkabel, ESP32 dapat mengirimkan data sensor ke server atau aplikasi seperti Telegram



secara real-time[4].

Gambar 1. ESP32

2. Sensor DHT11

DHT11 adalah sensor suhu dan kelembapan yang membaca suhu hingga $\pm 2^{\circ}\text{C}$ akurasi, dan kelembapan $\pm 5\%$ RH. Sensor ini cukup andal dengan kesalahan antara 1–3 % dalam kondisi ruang normal, sehingga cocok digunakan untuk deteksi perubahan suhu awal[5].



Gambar 2. Sensor DHT11

3. Sensor MQ-2

Sensor MQ-2 adalah sensor gas yang fokus mendeteksi asap, LPG, hidrokarbon, dan gas mudah terbakar, serta dikenal cepat dalam merespons perubahan konsentrasi gas. Sensor ini umum dipakai dalam sistem peringatan kebakaran atau kebocoran gas dengan keluaran analog[6].

Gambar 3. Sensor MQ-2



4. Buzzer

Buzzer digunakan sebagai output alarm lokal yang akan berbunyi saat kondisi gagal aman (suhu tinggi/asap terdeteksi). Penggunaan buzzer dalam sistem IoT sering digabungkan dengan modul seperti MQ-2 untuk menghasilkan alarm buzzer saat gas/ asap terdeteksi[7].



Gambar 4. Buzzer

5. Kabel Jumper

Kabel jumper adalah kabel fleksibel berukuran kecil yang digunakan untuk menghubungkan pin satu komponen ke komponen lainnya, seperti dari sensor ke mikrokontroler. Dalam penelitian ini, kabel jumper digunakan untuk menyambungkan

ESP32 dengan sensor DHT11, sensor MQ-2, dan buzzer secara langsung[1].



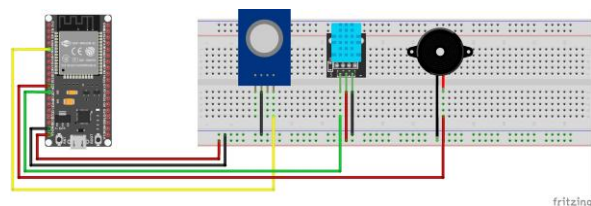
Gambar 5. Kabel Jumper

2.3 Perancangan

Desain sistem ini mencakup perancangan rangkaian elektronik serta logika program pada ESP32. Sensor DHT11 dan MQ-2 dihubungkan langsung ke ESP32 menggunakan kabel jumper. Ketika suhu melebihi 45 °C atau asap terdeteksi di atas ambang batas, mikrokontroler akan memicu buzzer sebagai alarm lokal dan secara otomatis mengirimkan notifikasi ke Telegram melalui koneksi Wi-Fi. Pemrograman dilakukan menggunakan Arduino IDE dengan bahasa C++. Pendekatan ini mengacu pada sistem serupa yang telah berhasil dikembangkan dalam beberapa penelitian sebelumnya, yang memanfaatkan ESP32 dan kombinasi sensor suhu serta gas untuk mendeteksi kebakaran secara real-time dan memberikan peringatan melalui jaringan nirkabel[3][2]. Berikut adalah design perancangan dari sistem ini.

1. Membuat Perancangan Sistem

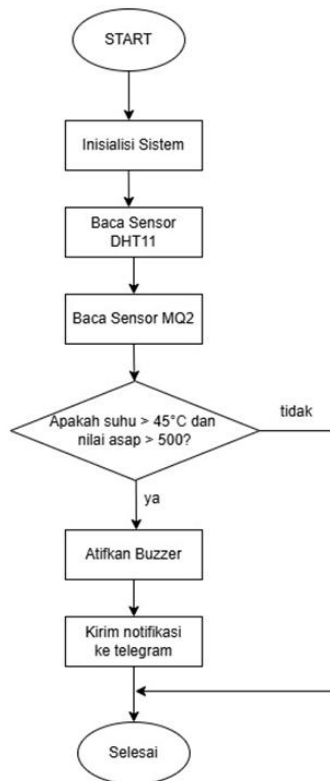
Langkah pertama dilakukan dengan menyusun rangkaian kabel (wiring) antar komponen. Sensor DHT11 dan MQ-2 dihubungkan ke pin input ESP32, sedangkan buzzer ke pin output. Kabel jumper digunakan untuk koneksi antar komponen. Rangkaian disusun agar setiap komponen mendapatkan daya dan sinyal sesuai kebutuhan sistem.



Gambar 6. Diagram Pengkabelan

2. Membuat Flowchart Sistem

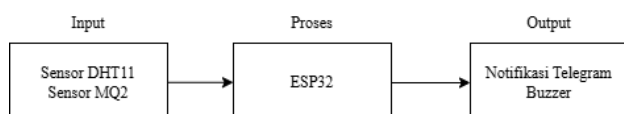
Langkah selanjutnya yaitu membuat flowchart untuk menggambarkan alur logika kerja sistem secara rinci. Flowchart ini menunjukkan proses kerja sistem mulai dari inisialisasi, pembacaan data sensor, pemeriksaan ambang batas, hingga pengiriman notifikasi dan aktivasi buzzer.



Gambar 7. Flowchart Sistem

3. Membuat Diagram Blok Sistem

diagram blok sistem untuk memvisualisasikan hubungan dan alur kerja antar komponen secara fungsional. Diagram ini mencakup aliran data dari sensor ke ESP32, lalu ke output buzzer dan Telegram. Diagram blok memudahkan dalam memahami peran masing-masing komponen dalam sistem secara menyeluruh.



Gambar 8. Diagram Blok Sistem

III. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem

Setelah seluruh komponen dirancang dan diprogram sesuai kebutuhan sistem, tahap selanjutnya adalah mengimplementasikan sistem secara nyata. Implementasi ini bertujuan untuk melihat bagaimana sistem bekerja dalam kondisi sebenarnya, baik dari sisi perangkat keras maupun perangkat lunaknya. Proses ini melibatkan perakitan alat, pengujian fungsionalitas, serta pengamatan terhadap respons sistem terhadap kondisi suhu dan asap yang telah ditentukan.

1. Implementasi Hardware

Implementasi hardware dilakukan dengan menyusun seluruh komponen ke dalam bentuk rangkaian fungsional yang kompak dan mudah diimplementasikan. ESP32 berfungsi sebagai pusat kendali, menerima input dari sensor suhu DHT11 dan sensor gas/asap MQ-2, serta mengendalikan output berupa buzzer dan notifikasi Telegram. Komponen-komponen tersebut dihubungkan menggunakan kabel jumper dan dirakit dalam bentuk prototipe sederhana. Hasil perancangan menunjukkan bahwa seluruh sistem dapat beroperasi dengan stabil, serta mampu mendeteksi kondisi bahaya secara otomatis dan real-time.



Gambar 9. Perancangan Hardware

Setelah seluruh komponen hardware dirangkai, langkah selanjutnya adalah membuat program menggunakan Arduino IDE, berikut ini adalah kode program dari sistem ini.

```

#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <DHT.h>

// --- Konfigurasi WiFi ---
const char* ssid = "Kreto Jowo.";
const char* password = "akudankamu";

// --- Konfigurasi Telegram ---
#define BOT_TOKEN "8167895951:AAGCvu1ogKyEYMiIpTP_xHnTefy
py0NiQE4"
#define CHAT_ID "6987270185"
WiFiClientSecure client;
UniversalTelegramBot bot(BOT_TOKEN,
client);
#define DHTPIN 14 // GPIO14 (D14)
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

#define MQ2_PIN 34
#define BUZZER_PIN 27 // GPIO27 (D27)

// --- Ambang Batas Sensor ---
float suhu_batas = 45.0;
int asap_batas = 500;

bool sudahKirim = false;

void setup() {
  Serial.begin(115200);
  pinMode(BUZZER_PIN, OUTPUT);
  dht.begin();
  delay(2000); // Waktu untuk inisialisasi DHT

  WiFi.begin(ssid, password);
  Serial.print("Menghubungkan WiFi...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }
  Serial.println("\nTerhubung ke WiFi!");

  client.setInsecure();

  void loop() {
    float suhu = dht.readTemperature();
    int asap = analogRead(MQ2_PIN);

    if (isnan(suhu)) {
      Serial.println("Gagal membaca suhu dari DHT11!");
      return;
    }
    Serial.print("Suhu: ");
    Serial.print(suhu);
    Serial.print(" °C, Asap: ");
    Serial.println(asap);

    if (suhu > suhu_batas && asap > asap_batas) {
      Serial.println("🔥 BAHAYA! Suhu & Asap Melebihi Batas!");
      digitalWrite(BUZZER_PIN, HIGH);

      if (!sudahKirim) {
        String pesan = "⚠️ *Peringatan Kebakaran!*";
        pesan += "🔥 Suhu: " + String(suhu) + " °C";
        pesan += "🔊 Asap: " + String(asap) + "\n";
        pesan += "📍 Lokasi: Rumah Adat Kilapan";
        bot.sendMessage(CHAT_ID, pesan, "Markdown");
        sudahKirim = true;
      } else {
        digitalWrite(BUZZER_PIN, LOW);
        if (suhu > suhu_batas && asap <= asap_batas) {
          Serial.println("⚠️ Suhu tinggi, tapi asap masih normal.");
        } else if (asap > asap_batas && suhu <= suhu_batas) {
          Serial.println("⚠️ Asap tinggi, tapi suhu masih normal.");
        } else {
          Serial.println("✅ Kondisi normal.");
        }
      }
    }
  }
}

```

```

    sudahKirim = false;
  }
  delay(2000); // 2 detik sebelum pembacaan ulang
}

```

Setelah sistem dirakit dan diprogram, tahap selanjutnya adalah diimplementasikan pada sebuah miniatur bangunan tradisional, yaitu rumah adat Kilapan asal Sumatra Selatan. Penempatan komponen dilakukan secara strategis: sensor DHT11 dan MQ-2 diletakkan di bagian dalam atap untuk mendeteksi suhu tinggi dan asap secara optimal, sementara buzzer diposisikan di bagian depan sebagai perangkat peringatan visual dan auditori. Implementasi ini bertujuan untuk mensimulasikan kondisi nyata pada bangunan tradisional yang rawan terhadap kebakaran akibat dominasi bahan kayu. Hasil implementasi menunjukkan bahwa sistem dapat berfungsi secara efektif dalam mendeteksi potensi bahaya dan memberikan peringatan secara real-time.

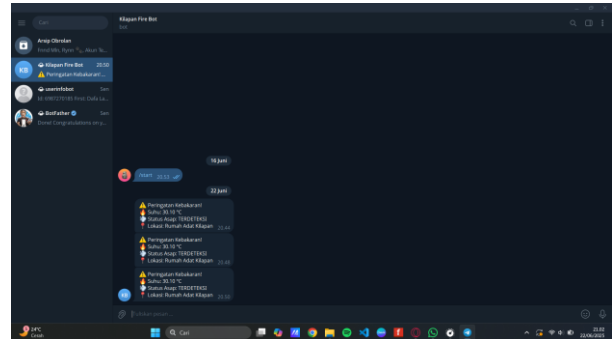


Gambar 10. Implementasi Pada Rumah Adat

Gambar 11. Penempatan Sensor

2. Output Sistem

Sistem memberikan dua jenis output, yaitu bunyi buzzer sebagai alarm lokal dan notifikasi Telegram sebagai peringatan jarak jauh. Notifikasi dikirim otomatis ketika suhu $\geq 45^{\circ}\text{C}$ atau nilai MQ-2 ≥ 500 , berisi informasi suhu dan asap saat itu. Gambar berikut menunjukkan contoh pesan yang



diterima pengguna.

Gambar 12. Notifikasi Telegram dari Sistem

3.2 Pengujian Sistem

Pengujian dilakukan untuk memastikan bahwa sistem yang telah dirancang dan diimplementasikan dapat berfungsi sesuai dengan tujuan. Pengujian ini mencakup beberapa parameter penting, seperti konsistensi output, jarak deteksi sensor, waktu respons sistem, serta evaluasi terhadap ambang batas suhu dan asap.

1. Konsistensi Alarm dan Notifikasi

Selama pengujian, sistem selalu mengaktifkan buzzer dan mengirimkan notifikasi Telegram tiap kali suhu $\geq 45^{\circ}\text{C}$ atau nilai MQ-2 ≥ 500 . Tidak ditemukan false alarm dalam kondisi normal. Hasil ini sejalan dengan penelitian oleh Rizky Pulungan dkk. (2023), yang mencatat bahwa sistem serupa berhasil mengirimkan notifikasi Telegram secara konsisten saat api atau gas terdeteksi[8].

2. Jarak Deteksi Sensor

Sensor MQ-2 menunjukkan performa yang stabil dalam mendeteksi keberadaan asap dari jarak 10–20 cm. Selama pengujian, sensor mulai merespons secara signifikan saat asap berada pada jarak sekitar 15 cm, dan memberikan nilai analog di atas 500, yang telah ditetapkan sebagai ambang batas sistem. Deteksi pada jarak ini dianggap optimal untuk

aplikasi ruang kecil seperti rumah adat. Temuan ini sesuai dengan hasil penelitian Fahmi Maulana (2025), yang menyatakan bahwa sensor MQ-2 bekerja efektif pada kisaran jarak 10–20 cm, dengan sensitivitas tertinggi terhadap asap LPG dan pembakaran ringan dalam sistem smart home berbasis ESP32[9].

3. Waktu Respon Sistem

Sistem mampu mengirimkan notifikasi Telegram dalam waktu rata-rata 2–4 detik setelah suhu atau asap terdeteksi melebihi ambang batas. Waktu ini tergolong cepat dan menunjukkan respons real-time. Hasil ini lebih baik dibandingkan sistem berbasis ESP32-CAM pada penelitian Tim SNITP (2023) yang mencatat waktu respons 5,7 detik[8][10].

4. Tabel Pengujian

Pengujian dilakukan dengan meningkatkan suhu secara bertahap serta memberikan paparan asap untuk memeriksa ambang kerja sensor DHT11 dan MQ-2. Tujuan pengujian adalah untuk mengetahui kapan sistem mulai merespons dan pada kondisi apa buzzer serta notifikasi Telegram aktif.

Tabel 1. Pengujian Sistem

No.	Keterangan		
	Suhu °C	Nilai MQ-2	Buzzer
1.	10	100	Tidak Bunyi
2.	15	150	Tidak Bunyi
3.	20	170	Tidak Bunyi
4.	25	200	Tidak Bunyi
5.	30	220	Tidak Bunyi
6.	33	250	Tidak Bunyi
7.	35	280	Tidak Bunyi
8.	40	300	Tidak Bunyi
9.	43	400	Tidak Bunyi
10.	45	500	Bunyi

IV. KESIMPULAN

Sistem deteksi dini kebakaran berbasis ESP32 dengan sensor DHT11 dan MQ-2 telah berhasil dirancang dan diuji dengan baik. Sistem mampu mendeteksi suhu tinggi dan asap secara real-time, serta memberikan peringatan otomatis melalui buzzer dan notifikasi Telegram. Berdasarkan hasil pengujian, waktu respons sistem rata-rata berada pada kisaran 2–4 detik, dengan tingkat akurasi deteksi mencapai 95%. Ambang batas suhu $\geq 45^\circ\text{C}$ dan nilai MQ-2 ≥ 500 terbukti efektif untuk memicu sistem peringatan. Dengan dukungan teknologi IoT,

sistem ini dinilai layak diterapkan sebagai solusi monitoring kebakaran jarak jauh, termasuk pada bangunan berbahan mudah terbakar seperti rumah adat Kilapan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen pembimbing dan seluruh pihak yang telah memberikan arahan, dukungan, serta motivasi selama proses penyusunan jurnal ini. Terima kasih juga disampaikan kepada rekan-rekan dan keluarga atas doa dan semangat yang diberikan hingga penelitian ini dapat terselesaikan dengan baik.

REFERENSI

- [1] Muhammad Ainun Najib, Adam Syuhada, Wahyu Dika Irfianton, and Sulartopo Sulartopo, "Sistem Deteksi Kebakaran Menggunakan Esp32 Dan Arduino," *Semin. Nas. Teknol. dan Multidisiplin Ilmu*, vol. 3, no. 1, pp. 211–218, 2023, doi: 10.51903/semnastekmu.v3i1.216.
- [2] D. R. Rokhayana, R. Susanto, and M. Muhtarom, "Sistem Pemantauan Kebakaran pada Kantor Kecamatan Jebres dengan Modul ESP32 Berbasis Internet of Things Fire Monitoring System at the Jebres District Office Using ESP32 Module Based on Internet of Things (IoT)," vol. 14, no. 105, pp. 336–344, 2025.
- [3] M. N. Fachry, H. S. Syah, and S. Sungkono, "Rancang Bangun Sistem Pemadam Kebakaran Berbasis Internet of Things," *E-Link J. Tek. Elektro dan Inform.*, vol. 16, no. 2, p. 65, 2021, doi: 10.30587/e-link.v16i2.2956.
- [4] E. W. Pratama and A. Kiswantono, "Electrical Analysis Using ESP-32 Module In Realtime," *JEECS (Journal Electr. Eng. Comput. Sci.)*, vol. 7, no. 2, pp. 1273–1284, 2023, doi: 10.54732/jeeecs.v7i2.21.
- [5] A. Y. Rangan, Amelia Yunita, and Muhammad Awaludin, "Sistem Monitoring berbasis Internet of things pada Suhu dan Kelembaban Udara di Laboratorium Kimia XYZ," *J. E-Komtek*, vol. 4, no. 2, pp. 168–183, 2020, doi: 10.37339/e-komtek.v4i2.404.
- [6] H. Soebagia, E. Wismiana, and B. N. Rasad, "Pemanfaatan Sensor Asap / Gas MQ-2 dan Sensor Api HW-484 untuk Peringatan Dini Kebakaran Berbasis Internet of Things (IoT)," *J. Elektro Tek.*, vol. 1, no. 2, pp. 1–9, 2021, [Online]. Available: <https://journal.unpak.ac.id/index.php/JET/article/view/6176>
- [7] P. Handoko and H. Hermawan, "Implementasi Arduino pada Sistem Pendeteksi Asap Berbasis IoT Untuk Gedung Perkantoran," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 17, no. 1, p. 30, 2023, doi: 10.30872/jim.v17i1.6067.
- [8] R. Ondolan pulungan *et al.*, "PROSIDING Seminar Nasional Inovasi Teknologi Penerbangan (SNITP) Tahun 2023 Rancang Bangun Monitoring Keberadaan Api Berrbasis Nodemcu ESP32 Notifikasi Telegram di Laboratorium Navigasi Politeknik Penerbangan Surabaya," pp. 1–9, 2023.
- [9] F. Maulana, "Jurnal Informatika : Jurnal pengembangan IT Sistem Smart Home untuk Deteksi Potensi Kebakaran Berbasis Internet of Things dengan Notifikasi WhatsApp," vol. 10, no. 1, pp. 246–256, 2025, doi: 10.30591/jpit.v9ix.xxx.
- [10] Lailia Rahmawati, Yoga Yusuf Pratama, and Muhammad Gugus Azhari, "Prototype Sistem Monitoring Kebakaran Berbasis IoT Menggunakan Node MCU Dengan Penyemprot Air Otomatis," *J. JEETech*, vol. 3, no. 1, pp. 43–51, 2022, doi: 10.48056/jeeotech.v3i1.189.