

Perancangan Sistem Deteksi Kebakaran Menggunakan ESP32 Dengan Notifikasi Blynk

Amelia Uswatun Khasanah¹, Nurul Azizah², Nurul Ilmah³, Taufik Bahtiar⁴,
Riswira⁵, Adis Aprilia Putri⁶

¹²³⁴⁵⁶Pendidikan Teknik Infotmatika dan Komputer, Universitas Negeri Makassar
Jl. Daeng Tata Raya Parang Tambung, Makassar

Email: ameliauswatunkhasanahhh@gmail.com , nununurul355@gmail.com,
nurulilmah06@gmail.com, 220209501124@student.unm.ac.id, riswira180603@gmail.com,
adisaprilialia06@gmail.com

Abstrak

Kebakaran merupakan salah satu jenis bencana yang berdampak besar terhadap keselamatan manusia, harta benda, dan lingkungan sekitar, yang sering kali disebabkan oleh korsleting listrik, kebocoran gas, atau kelalaian manusia. Ketika tidak terdeteksi sejak dini, kebakaran dapat berkembang pesat dan sulit dikendalikan, khususnya saat rumah atau bangunan dalam keadaan kosong. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem *Smart Fire Alarm* berbasis ESP32 yang terintegrasi dengan teknologi Internet of Things (IoT) sebagai solusi deteksi dini terhadap bahaya kebakaran. Sistem ini menggunakan sensor api (*Flame sensor*), serta aplikasi Blynk untuk mengirim notifikasi secara real-time ke perangkat pengguna saat potensi kebakaran terdeteksi. Pengujian dilakukan untuk mengukur keakuratan sistem dalam mendeteksi api, kecepatan aktivasi alarm, dan efektivitas notifikasi. Hasil menunjukkan bahwa sistem dapat mendeteksi keberadaan api dan gas dengan respons cepat dan mengirim peringatan secara real-time selama perangkat terhubung ke internet. Namun, deteksi mulai melemah pada jarak lebih dari 7 cm, menunjukkan keterbatasan jarak dan pengaruh kondisi lingkungan terhadap kinerja sistem. Untuk meningkatkan akurasi dan stabilitas, disarankan menambahkan sensor suhu serta melakukan pengujian dalam ruangan tertutup. Alternatif media notifikasi selain Blynk juga perlu dipertimbangkan agar sistem tetap bekerja meski koneksi tidak stabil.

Kata Kunci: Kebakaran, Internet of Things, ESP32, Sensor Api(Flame sensor), Sensor Gas MQ-2, Blynk, Sistem Deteksi

Abstract

Fire is one type of disaster that has a major impact on human safety, property, and the surrounding environment, which is often caused by electrical short circuits, gas leaks, or human negligence. If not detected early, fires can grow rapidly and be difficult to control, especially in empty houses or buildings. This study aims to design and implement an ESP32-based Smart Fire Alarm system integrated with Internet of Things (IoT) technology as an early detection solution for fire hazards. This system uses a flame sensor, and the Blynk application to send real-time notifications to the user's device when a potential fire is detected. Testing was carried out to measure the accuracy of the system in detecting fire, the speed of alarm activation, and the effectiveness of notifications. The results show that the system can detect the presence of fire and gas with a fast response and send real-time alerts as long as the device is connected to the internet. However, detection begins to weaken at a distance of more than 7 cm, indicating the limitations of distance and the influence of environmental conditions on system performance. To improve accuracy and stability, it is recommended to add a temperature sensor and conduct testing in a closed room. Alternative media notifications other than Blynk also need to be considered so that the system continues to work even though the connection is unstable.

Keywords: Fire, Internet of Things, ESP32, Flame sensor, MQ-2 Gas Sensor, Blynk, Detection System

1. Pendahuluan

Kebakaran merupakan salah satu jenis bencana yang sering terjadi dan berdampak besar terhadap keselamatan jiwa, harta benda, serta lingkungan sekitar [1]. Dalam banyak kasus, kebakaran disebabkan oleh hal-hal sepele seperti korsleting listrik, kebocoran gas, atau kelalaian manusia [2]. Ketika tidak terdeteksi sejak awal, kebakaran dapat berkembang dengan cepat dan

sulit dikendalikan, terutama saat bangunan dalam keadaan kosong atau tidak diawasi [3].

Kebakaran dapat dianggap sebagai satu diantara berbagai jenis musibah karena sifatnya yang membahayakan lingkungan dan masyarakat serta sulit dikendalikan[2]. Kebakaran merupakan suatu insiden yang dapat membahayakan dan mengganggu kelangsungan hidup manusia, dan dapat terjadi di berbagai

tempat seperti gedung, perumahan, dan hutan[4]. Kebakaran dapat disebabkan oleh banyak hal seperti membuang sisa rokok sembarangan, korsleting listrik, kebocoran gas, dan sebagainya, yang berpotensi menimbulkan kebakaran yang lebih besar lagi[5]. Pada saat ini, orang – orang keluar bekerja dari pagi hingga sore hari meninggalkan rumah atau tempat tinggal cukup lama. Sehingga musibah seperti kebakaran kadang tak bisa diatasi saat meninggalkan rumah[6].

Seiring dengan kemajuan teknologi, konsep *Internet of Things* (IoT) telah membuka peluang besar dalam menciptakan sistem deteksi dan peringatan dini terhadap bencana, termasuk kebakaran[7]. IoT memungkinkan perangkat fisik seperti sensor dan mikrokontroler untuk terhubung dengan jaringan internet, sehingga dapat mengirim dan menerima data secara otomatis dan real-time tanpa memerlukan interaksi langsung dari manusia [8], [18]. Konsep ini menjadikan IoT sebagai solusi yang sangat potensial dalam pengawasan lingkungan secara jarak jauh, termasuk dalam hal pengamanan dari risiko kebakaran..

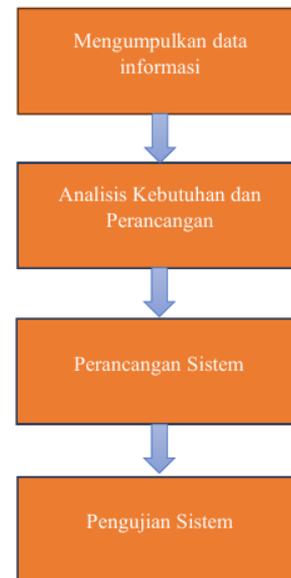
Dalam konteks pencegahan kebakaran, penerapan IoT dapat diwujudkan melalui sistem deteksi cerdas yang memanfaatkan perangkat seperti ESP32, sensor api, dan platform *Blynk*[9]. Sistem ini dirancang untuk mendeteksi adanya nyala api dan secara otomatis mengirimkan notifikasi ke smartphone pengguna melalui aplikasi *Blynk*, memungkinkan pengguna untuk segera mengambil tindakan meskipun sedang berada jauh dari lokasi [10].

Penelitian ini bertujuan untuk merancang dan membangun sistem *Smart Fire Alarm* berbasis ESP32 yang terintegrasi dengan teknologi IoT, sebagai solusi deteksi dini terhadap bahaya kebakaran. Sistem ini diuji untuk menilai epektifitas sensor api dalam mendeteksi keberadaan api, respons alarm, serta efektivitas pengiriman notifikasi ke perangkat pengguna secara real-time. Diharapkan dengan adanya sistem ini, pengguna dapat segera melakukan langkah pencegahan sebelum kebakaran meluas dan menimbulkan kerugian yang lebih besar.

2. Metode

2.1. Tahap Pengerjaan Penelitian

Untuk membuat sistem *Smart Fire Alarm* di perlukan beberapa tahapan, Adapun tahapannya dapat di lihat pada gambar di bawa ini.



Gambar 1. Tahap Pengerjaan Penelitian

2.2 Pengumpulan data

Dalam proses pengumpulan data, penelitian ini menggunakan dua metode utama, yaitu:

Observasi, dilakukan untuk mengidentifikasi permasalahan yang ada di lokasi penelitian. Setelah menemukan permasalahan, dilakukan analisis terhadap alat dan bahan yang diperlukan untuk merancang sistem deteksi kebakaran berbasis ESP32. Studi Literatur, dilakukan dengan mengkaji jurnal dan penelitian terdahulu yang relevan untuk memperoleh referensi terkait pengembangan sistem ini.

Saat sistem diaktifkan, sensor api akan mulai memantau kondisi lingkungan. Jika sensor mendeteksi keberadaan api yang melebihi ambang batas yang telah ditentukan, sistem akan memberikan respons berupa menyalakan LED merah, mengaktifkan *buzzer*, dan mengirim notifikasi ke aplikasi *Blynk*. Notifikasi ini memungkinkan pengguna menerima peringatan secara real-time melalui smartphone mereka. *Buzzer* akan tetap berbunyi hingga kondisi kembali normal atau sistem dihentikan secara manual.

2.3 Analisis Kebutuhan

Berdasarkan hasil analisis yang telah dilakukan *Smart Fire Alarm* memerlukan beberapa kebutuhan.

2.3.1 Kebutuhan Input dan Output

Smart Fire Alarm membutuhkan inputan dari data sensor api dan output atau keluarannya akan berupa bunyi *buzzer*, LED merah menyala dan LED hijau mati, serta notifikasai dari software blynk

2.3.2 Kebutuhan Hardware

Langkah awal dalam pengembangan sistem ini adalah mengidentifikasi perangkat yang ada. Proses ini mencakup pengumpulan alat-alat yang akan digunakan untuk membangun sistem deteksi kebakaran berbasis ESP32, yaitu:

a. ESP32

Mikrokontroler utama yang menjadi otak dari alat [11]. ESP32 akan mengontrol seluruh sistem, memproses data dari sensor api, serta mengirim notifikasi ke aplikasi *Blynk* melalui koneksi Wi-Fi [12], [19].



Gambar 2. ESP32

b. Sensor Api

Sensor yang berfungsi untuk mendeteksi adanya nyala api dalam jangkauan tertentu[13]. Jika mendeteksi keberadaan api, sensor akan mengirimkan sinyal ke ESP32 [14].



Gambar 3. Sensor Api

c. LED Merah dan Hijau

Indikator visual yang menunjukkan status lingkungan: LED Hijau menyala saat kondisi aman. LED Merah menyala jika terdeteksi potensi kebakaran.



Gambar 4. LED Merah dan Hijau

d. Buzzer

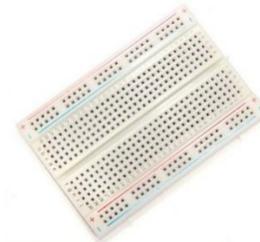
Berfungsi sebagai alarm peringatan yang berbunyi saat sistem mendeteksi kebakaran[14]



Gambar 5. Buzzer

e. Breadboard

Papan tempat merangkai komponen tanpa perlu menyolder, memudahkan proses perakitan dan pengujian alat [15].



Gambar 6. Breadboard

f. Kabel Jumper Male to Male

Kabel jumper adalah kabel listrik yang memiliki pin konektor di setiap ujungnya dan memungkinkan menghubungkan ESP32 dengan sensor, buzzer, dan LED pada breadboard[16].



Gambar 7. Kabel Jumper Male To Male

2.3.3 Kebutuhan Software

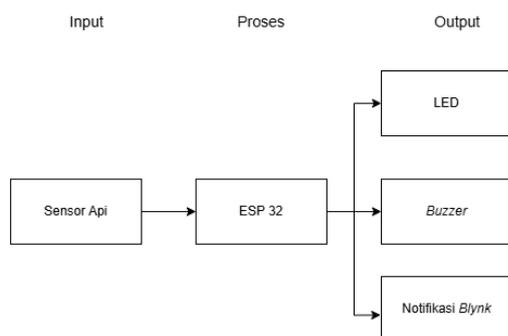
Dalam pembuatan *Smart Fire Alarm*, software yang dibutuhkan adalah *Arduino IDE* dan *Blynk*.

2.4 Perancangan Sistem

Perancangan sistem *Smart Fire Alarm* merupakan tahap krusial dalam membangun sistem pendeteksi kebakaran otomatis berbasis teknologi. Sistem ini dirancang agar mampu mendeteksi keberadaan api atau asap secara real-time dan memberikan notifikasi kepada pengguna secara cepat melalui aplikasi atau pesan[17].

2.4.1 Block Diagram

Untuk memahami bagaimana sistem bekerja, Gambar 8 menunjukkan blok diagram yang menggambarkan hubungan antara komponen perangkat keras dan perangkat lunak dalam sistem ini.



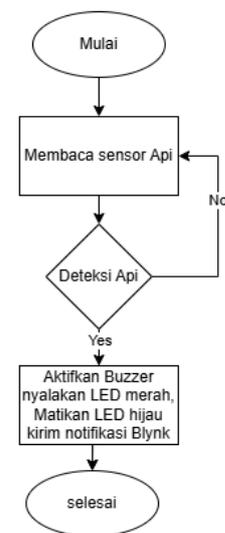
Gambar 8. Blok Diagram

Blok diagram tersebut menggambarkan alur kerja dari sistem *Smart Fire Alarm* berbasis IoT yang telah dirancang. Sistem ini menerima input dari sensor api (*Flame Sensor*), yang berfungsi untuk mendeteksi keberadaan nyala api di lingkungan sekitar. Sensor ini mendeteksi intensitas cahaya dari api dan mengirimkan data tersebut ke mikrokontroler.

Data dari sensor api kemudian diproses oleh ESP32, yang berperan sebagai pusat kendali sistem. ESP32 akan menganalisis data yang diterima dan mengevaluasi apakah api terdeteksi berdasarkan ambang nilai tertentu. Jika terdeteksi adanya api, maka sistem secara otomatis akan mengaktifkan beberapa output secara bersamaan, yaitu: buzzer sebagai alarm peringatan suara, LED merah sebagai indikator visual, serta notifikasi melalui aplikasi *Blynk* sebagai pemberitahuan jarak jauh kepada pengguna.

2.4.2 Flowchart

Gambar 8 menunjukkan flowchart dari sistem. Penjelasan dari alur flowchart yaitu proses dimulai dengan sistem membaca data dari sensor api. Jika tidak terdeteksi api, sistem akan terus membaca sensor. Namun, jika asap terdeteksi, sistem akan mengaktifkan buzzer, menyalakan LED merah (indikator adanya asap), mematikan LED hijau (indikator normal), dan mengirimkan notifikasi ke aplikasi *Blynk*. Setelah itu, sistem tetap berjalan untuk terus memantau kondisi lingkungan.



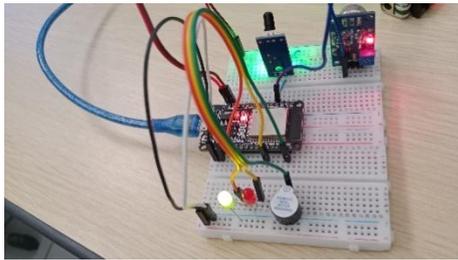
Gambar 9. Flowchart

3. Hasil dan Pembahasan

3.1 Hasil Implementasi

Gambar 10 menunjukkan implementasi *Smart Fire Alarm*. Pada pembuatan *Smart Fire Alarm*, sistem ini menggunakan mikrokontroler ESP32 yang terhubung dengan sensor api untuk mendeteksi keberadaan api. Sensor ini langsung terhubung ke ESP32 dan mengirimkan sinyal digital maupun analog yang kemudian diproses oleh sistem. ESP32 akan melakukan seleksi berdasarkan data yang diterima, dan jika kondisi yang telah ditentukan terpenuhi, *Smart Fire Alarm* akan mengirimkan notifikasi secara real-time melalui aplikasi *Blynk* menggunakan jaringan internet. Dengan adanya notifikasi ini, pengguna dapat segera mengetahui potensi kebakaran dan mengambil tindakan pencegahan dengan lebih cepat dan efektif.

Hasil implementasi sistem dapat dilihat pada Gambar 10.



Gambar 10. Implementasi Hardware

Setelah perancangan dan implementasi sistem dilakukan. Langkah selanjutnya adalah pengujian perangkat. Pengujian perangkat meliputi pengujian respon sensor api dan pengujian notifikasi.

3.1.2 Pengujian Perangkat

a. Pengujian Sensor Api

Sensor digunakan untuk mendeteksi api. Api diperoleh dari korek. Berikut ini pengujian sensor Api pada *Smart Fire Alarm*.

Tabel 1. Pengujian Sensor Api

Jarak	LED Hijau	LED Merah	Buzzer	Notifikasi	Dokumentasi
2 cm	Off	On	On	Ada	
3 cm	Off	On	On	Ada	
4 cm	Off	On	On	Ada	
5 cm	Off	On	On	Ada	
6 cm	Off	On	On	Ada	
7 cm	On	Off	Off	Tidak Ada	

b. Pengujian pengiriman notifikasi *Blynk*

Tujuan dari pengujian ini untuk mengetahui seberapa bisa ESP32 dalam

mengirimkan notifikasi ke aplikasi *Blynk*. Berikut ini tabel pengujian pengiriman notifikasi ke *Blynk*.

Tabel 2. Pengujian notifikasi Blynk

Respons	Jaringan WiFi	Notifikasi Blynk
2-3 detik	Normal	Bahaya! Api atau Asap/Gas terdeteksi
-----	Tidak normal	Tidak ada notifikasi

Tabel di atas menunjukkan hasil pengujian pengiriman notifikasi dari ESP32 ke aplikasi Blynk. Hasil pengujian menunjukkan bahwa saat koneksi WiFi dalam kondisi normal (ESP32 terhubung ke WiFi dari ponsel yang menggunakan aplikasi Blynk), notifikasi dapat dikirim dengan respon waktu sekitar 2–3 detik setelah sensor mendeteksi asap atau gas. Namun, ketika ESP32 tidak terhubung ke WiFi ponsel yang menggunakan Blynk, notifikasi tidak dapat dikirim, sehingga peringatan terhadap bahaya tidak diterima oleh pengguna. Gambar 10 adalah notifikasi Blynk jika berhasil terkoneksi.

**Gambar 10. Notifikasi Blynk**

3.2 Pembahasan

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 1, sistem *Smart Fire Alarm* menunjukkan performa yang baik dalam mendeteksi api dari jarak tertentu. Sensor yang digunakan mampu memicu respons sistem

berupa nyala LED, bunyi *buzzer*, dan pengiriman notifikasi ke aplikasi Blynk saat api terdeteksi.

Dari tabel, terlihat bahwa mulai dari jarak 2 cm hingga 6 cm, sistem merespons dengan menyalakan LED merah, mematikan LED hijau, membunyikan *buzzer*, serta mengirimkan notifikasi melalui aplikasi *Blynk*. Dokumentasi visual juga menunjukkan bahwa sensor mampu menangkap keberadaan api dari jarak dekat secara konsisten.

Namun, pada jarak 7 cm sistem tidak lagi memberikan respons: LED merah mati, *buzzer* tidak berbunyi, dan notifikasi tidak terkirim. Ini menandakan bahwa sensor api yang digunakan memiliki batas deteksi optimal maksimal pada jarak sekitar 6 cm.

Selanjutnya, berdasarkan Gambar 10, ditampilkan bukti visual bahwa aplikasi Blynk berhasil mengirimkan notifikasi secara real-time saat sensor mendeteksi adanya api atau asap. Pesan yang muncul pada notifikasi berisi peringatan “Bahaya! Api atau asap/gas terdeteksi!” yang sangat penting untuk memberikan respon cepat kepada pengguna.

Fitur notifikasi ini menjadi keunggulan sistem karena memungkinkan pemilik rumah atau pengguna untuk langsung mengetahui kejadian meskipun sedang berada jauh dari lokasi. Keberhasilan sistem ini tentunya bergantung pada kestabilan koneksi internet antara perangkat ESP32 dan server Blynk.

Penelitian ini memiliki kesamaan dengan penelitian yang dilakukan (Najib et al., 2023) yang berjudul “Sistem Deteksi Kebakaran Menggunakan ESP32 dan Arduino” namun perbedaan utama terletak pada metode notifikasi yang di terapkan. Selain itu, penelitian oleh (Wungow et al., 2024) dengan judul “Perencanaan Alat Deteksi Kebakaran Berbasis Arduino”, sistem deteksi kebakaran dirancang menggunakan mikrokontroler Arduino yang dilengkapi dengan sensor asap dan api. Sistem ini mampu mendeteksi potensi kebakaran dan memberikan notifikasi melalui bunyi alarm serta pengiriman pesan singkat (SMS) ke ponsel pengguna. Dengan demikian, baik penelitian ini maupun kedua penelitian lainnya sama-sama berfokus pada pembuatan alat deteksi kebakaran yang efektif, namun menggunakan pendekatan berbeda dalam hal media pemberitahuan kepada pengguna.

Berdasarkan hasil pengujian yang dilakukan dapat disimpulkan bahwa integrasi antara sensor api, mikrokontroler ESP32, dan platform IoT Blynk sangat efektif dalam membangun sistem deteksi dini kebakaran yang sederhana namun responsif. Meski demikian, keterbatasan dalam jangkauan deteksi sensor perlu menjadi perhatian dalam pengembangan sistem yang lebih luas dan akurat di masa mendatang.

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian, sistem Smart Fire Alarm yang dirancang menggunakan ESP32, sensor api menunjukkan kinerja yang baik dalam mendeteksi potensi kebakaran secara dini. Sensor api mampu merespons keberadaan nyala api dengan cepat. Sistem ini juga terintegrasi dengan platform Blynk, yang memungkinkan pengiriman notifikasi secara real-time ke perangkat pengguna, sehingga memberikan peringatan dini yang responsif selama ESP32 terhubung ke jaringan internet.

Namun, dalam pengujian ditemukan bahwa deteksi mulai melemah pada jarak lebih dari 7 cm, di mana sensor tidak lagi mengaktifkan buzzer maupun mengirim notifikasi. Hal ini menunjukkan adanya keterbatasan jarak deteksi serta pengaruh kondisi lingkungan terhadap performa sistem.

Oleh karena itu, pada penelitian selanjutnya disarankan untuk menambahkan sensor pendukung seperti sensor suhu guna meningkatkan akurasi sistem. Selain itu, uji coba sebaiknya difokuskan di dalam ruangan tertutup agar kondisi lebih stabil dan hasil pengujian lebih optimal. Perlu juga dipertimbangkan penggunaan alternatif notifikasi lain selain Blynk, untuk menjamin sistem tetap dapat mengirim peringatan walaupun koneksi tidak stabil.

5. Ucapan Terima Kasih

Peneliti mengucapkan terimakasih kepada seluruh pihak yang telah membimbing dan membantu dalam penelitian ini. Semoga dengan penelitian ini dapat menjadi referensi dan dapat dikembangkan lagi.

Referensi

- [1] U. Anggoro Saputro and A. Tuslam, "Sistem Deteksi Kebakaran Berbasis Internet Of Things Dengan Pesan Peringatan Menggunakan NodeMCU ESP8266 Dan Platform ThingSpeak," *Jurnal Infomedia: Teknik Informatika, Multimedia & Jaringan*, vol. 7, no. 1, 2022.
- [2] C. A. Haryani, A. R. Mitra, and A. E. Widjaja, "The Design of Microcontroller Based Early Warning Fire Detection System for Home Monitoring," *International Journal of New Media Technology*, vol. 9, no. 1, 2022, [Online]. Available: <https://firebase.google.com/docs/cloud-messaging/fcm->
- [3] I. N. B. Perwira and W. Broto, "PEMBUATAN ALAT PENDETEKSI API DAN ASAP BERBASIS MIKROKONTROLLER ARDUINO UNO DAN SENSOR MQ-2 KELUARAN SMS GATEWAY," *Universitas Negeri Jakarta Prosiding Seminar Nasional Fisika (E-Journal)*, vol. 6, 2017, doi: 10.21009/03.SNF2017.
- [4] S. Mahalingam, M. S. Deep, and K. S. Krishna, "Wireless Sensor Based Forest Fire Early Detection with Online Remote Monitoring," *Int J Eng Adv Technol*, vol. 10, no. 5, pp. 143–145, Jun. 2021, doi: 10.35940/ijeat.e2670.0610521.
- [5] X. Shi and L. Songlin, "Design and Implementation of a Smart Wireless Fire-Fighting System Based on NB-IoT Technology," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Aug. 2020. doi: 10.1088/1742-6596/1606/1/012015.
- [6] B. Kusumo and T. Ardiansyah, "RANCANG BANGUN SISTEM PENDETEKSI KEBAKARAN

- BERBASIS MIKROKONTROLER ESP32,” 2024.
- [7] Haryanto, L. Anifah, D. Rahmawati, A. K. Sahputra, and D. T. Laksono, “Smart Controlling system for Kitchen Fire Protection based Internet of Things,” *IOP Conf Ser Mater Sci Eng*, vol. 1125, no. 1, p. 012073, May 2021, doi: 10.1088/1757-899x/1125/1/012073.
- [8] Muhammad Ainun Najib, Adam Syuhada, Wahyu Dika Irfianton, and Sulartopo Sulartopo, “Sistem Deteksi Kebakaran Menggunakan Esp32 Dan Arduino,” *Seminar Nasional Teknologi dan Multidisiplin Ilmu (SEMNASTEKMU)*, vol. 3, no. 1, pp. 211–218, Oct. 2023, doi: 10.51903/semnastekmu.v3i1.216.
- [9] M. Pavitra, S. Khan, S. Jain, A. Mn, and P. Kalyan, “FOREST FIRE DETECTION SYSTEM USING IOT,” 2020. [Online]. Available: <http://www.ijeast.com>
- [10] G. G. Salindeho and T. Wellem, “PERANCANGAN DAN IMPLEMENTASI SISTEM PENDETEKSI DAN PERINGATANKEBAKARANBERBASIS IOT MENGGUNAKAN NODEMCU ESP8266DAN SENSOR API,” *Jurnal Penerapan Teknologi Informasi dan Komunikasi*, vol. 02, no. ISSN 2828-7940, e-ISSN 2829-1727, 2023.
- [11] S. Dhimas Ghoza *et al.*, “PERANCANGAN SMOKE DETECTOR BERBASIS SENSOR MQ-135 DAN MIKROKONTROLER ESP32 SEBAGAI DETEKSI DINI KEBAKARAN,” 2024.
- [12] Y. Subbarayudu, G. Vijendar Reddy, J. Bhargavi, and K. Latha, “An Efficient IoT-Based Novel Approach for Fire Detection Through Esp 32 Microcontroller in Forest Areas,” *MATEC Web of Conferences*, vol. 392, p. 01109, 2024, doi: 10.1051/matecconf/202439201109.
- [13] T. Juwariyah, S. Prayitno, L. Krisnawati, and S. Sulasminingsih, “Design of IoT-Based Home Fire Detection System Equipped with a Data Logger,” *IOP Conf Ser Mater Sci Eng*, vol. 1125, no. 1, p. 012079, May 2021, doi: 10.1088/1757-899x/1125/1/012079.
- [14] C. J. Wungow, A. R. A. Ferdinandus, A. A. Adam, and N. Komalig, “PERANCANGAN ALAT DETEKSI KEBAKARAN BERBASIS ARDUINO,” *Jurnal SAINTEK*, vol. 4(1), pp. 36–37, 2024.
- [15] A. Srakar, A. D. Sarkar, A. Argarwal, Dhar Palasri, and S. Roy, “Advanced Fire Controller- AGNI: Design & Implimentation,” Feb. 19, 2020. doi: 10.36375/prepare_u.a84.
- [16] M. S. Syamsudin *et al.*, “Rancang Bangun Alat Pendeteksi dan Monitoring Banjir Menggunakan ESP32,” *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis (SENATIB)*, 2023.
- [17] M. Ainun Najib, A. Syuhada, W. Dika Irfiantono, P. Studi Sistem Komputer, and U. Sains dan Teknologi Komputer, “SISTEM PENDETEKSI BENCANA KEBAKARAN MENGGUNAKAN ESP32 DAN ARDUINO BERBASIS WEB,” vol. 14, no. 2, pp. 384–394, 2023, [Online]. Available: <http://ejurnal.provisi.ac.id/index.php/JTI> KP□page384
- [18] Muhammad, Umar, and Mukhlisin Mukhlisin. "Desain Sistem Akuisisi Sensor Tegangan Berbasis IoT." *Joule (Journal of Electrical Engineering)*. 2.1 (2021): 72-75.
- [19] Mukhlisin, Gani, Purnamawati HA, and U. Muhammad. "Rancang Bangun Media Pembelajaran Mikrokontroler Berbasis Augmented Reality (AR)." *Journal Of Electrical Engineering (Joule)* 3.1 (2022): 125-129.