

# Penerapan Tuya Application Programming Interface (API) pada Sistem IoT Monitoring Suhu Ruang Server

Very Kurnia Bakti<sup>\*1</sup>, Achmad Sutanto<sup>2</sup>, Mohammad Rizal Arfani<sup>3</sup>

<sup>1,2,3</sup> Program Studi Teknik Komputer, Politeknik Harapan Bersama, Tegal

<sup>1,2,3</sup> Jln. Mataram No.9 Pesurungan Lor, Kota Tegal, 52147, Indonesia,

email: <sup>\*1</sup>verykurniabakti@poltektegal.ac.id, <sup>2</sup>achmadsutanto@poltektegal.ac.id, <sup>3</sup>mohamadrijalarfani@poltektegal.ac.id

*Abstract – Server computers generally work 24 hours and continuously. The tasks of the server, starting from running the operating system, applications, and data storage, are heavy and oversized. All electronic devices inside the server generate heat when operating. If they are not conditioned in a room with low temperatures, then the temperature of the server computer will increase and cause a decrease in the performance of the server. This study tested the temperature sensor available from the Application Programming Interface (API) feature from Tuya. Utilizing the API will facilitate the design of a system that can focus more on monitoring application development because the built system is no longer focused on complicated and risky hardware designs. The results of monitoring the server room's temperature can be monitored with sensors integrated with Tuya on a web-based page.*

*Abstrak – Komputer server pada umumnya bekerja selama 24 jam dan terus menerus, tugas dari server mulai dari menjalankan sistem operasi, aplikasi dan penyimpanan data yang cukup berat dan besar, seluruh perangkat elektronik didalam server menghasilkan panas ketika beroperasi, jika tidak dikondisikan pada ruangan dengan suhu rendah maka suhu komputer server akan semakin meningkat dan menyebabkan menurunnya kinerja dari server. Dalam penelitian ini dilakukan pengujian terhadap sensor suhu yang tersedia fitur Application Programming Interface (API) dari Tuya. Dengan pemanfaatan API tersebut akan memudahkan perancangan sistem yang dapat lebih fokus pada pengembangan aplikasi monitoring, karena sistem dibangun tidak lagi fokus pada perancangan perangkat keras yang rumit dan beresiko. Hasil monitoring suhu ruang server dapat dimonitor dengan sensor yang terintegrasi dengan Tuya dalam sebuah halaman berbasis web.*

**Kata Kunci** – Sensor, Suhu, IoT, Tuya

## I. PENDAHULUAN

Dalam mengkondisikan kinerja komputer server agar memiliki kinerja yang optimal dan stabil diperlukan pengkondisian khusus agar suhu komputer server tetap terjaga, faktor agar komputer server memiliki suhu yang stabil salah satunya adalah suhu ruangan yang ideal dimana suhu tersebut berada pada kisaran 20 °C sampai dengan 25 °C [1] lebih spesifik temperatur ruang server di Indonesia baiknya pada kisaran suhu 21 – 23 °C[2]. apabila ruangan terlalu dingin atau rendah akan menimbulkan embun dilingkungan server tentunya akan membahayakan perangkat server[3], sedangkan jika suhu terlalu panas maka suhu server akan terus naik dan tentunya akan menurunkan kinerja server[4]. Menurut American Society of Heating Refrigerating and Air-Conditioning Engineers (ASHRAE)

rekomendasi suhu ruangan server yang baik yaitu antara 18 ° - 27 ° C [5] dari beberapa rekomendasi terkait adanya rentang ideal sebuah ruangan untuk server maka dapat dijadikan dasar bahwa suhu ruangan server sangat penting diperhatikan.

Membangun sistem monitoring suhu dengan komponen rakitan sangat mungkin dilakukan seperti pemanfaatan Message Queue Telemetry Transport (MQTT) communication protocol dan penggunaan sensor LM 35 yang dapat diintegrasikan dengan Raspberry Pi maupun Arduino[6] [7] [8] dengan menggunakan cara tersebut sudah banyak yang berhasil melakukan monitoring suhu yang dapat bekerja dengan baik dalam memperoleh nilai baca dari sensor suhu meskipun terdapat beberapa kendala utamanya dalam hal durability . Saat ini seiring perkembangan teknologi ada beberapa sensor yang terstandar pabrik yang memiliki fasilitas penggunaan API salah satu produk sensor dengan API adalah Tuya. Dimana sensor sudah tertanam fasilitas koneksi wifi dan telah terkoneksi API pada cloud yang disediakan oleh Tuya. Kelebihan penggunaan API Tuya adalah bisa memanfaatkan hasil bacaan sensor untuk diakuisisi datanya melalui cloud dan dikembangkan kembali kedalam bahasa pemrograman python atau php dan mewujudkannya kedalam sistem berbasis web. Dalam penelitian ini penggunaan API tuya adalah gap / pembeda dari penelitian – penelitian yang serupa sebelumnya dengan tujuan melakukan pengujian apakah sensor suhu melalui koneksi API dapat diakuisisi datanya yang akan diintegrasikan dalam pengkodean bahasa pemrograman php dan menghasilkan sistem monitoring suhu yang akurat.

## II. PENELITIAN TERKAIT

Memanfaatkan sensor suhu sebagai sistem monitoring ruangan sudah dilakukan oleh beberapa peneliti sebelumnya, seperti penelitian Periyaldi, dengan judul Implementasi Sistem Monitoring Suhu Ruang Server Satnetcom Berbasis Internet of Things(IOT) Menggunakan Protokol Komunikasi Message Queue Telemetry Transport(MQTT) hasil penelitian tersebut pada modul Mosquitto MQTT broker dapat berjalan dengan baik namun masih menggunakan komponen modular yang belum menerapkan API[7]. Demikian pada penelitian Faisal Arief Deswar, Monitoring suhu pada ruang server menggunakan wemos D1 R1 berbasis Internet of things (IoT) proses monitoring menggunakan beberapa perangkat seperti mikrokontroler Wemos D1 R1, sensor suhu DHT11, IR ,Transmitter dan LCD, dari seluruh perangkat tersebut merupakan komponen rakitan / berbentuk modul yang belum teruji dari durability[9]. Pada penelitian M. Safii , Novi

**\*) penulis korespondensi:** Very Kurnia Bakti

Email: verykurniabakti@poltektegal.ac.id

Indrayani dengan judul perancangan piranti lunak *responsive* untuk monitoring ruangan server menggunakan nodeMCU ESP 8266 berbasis internet of things, pada penelitian ini Masih berbentuk *prototype* monitoring suhu menggunakan nodeMCU ESP 2866 dan sensor DHT 21 dengan dibuat piranti lunak berbasis web, perancangan piranti lunak *responsive* untuk monitoring ruangan server namun masih menggunakan komponen modular dan belum memanfaatkan teknologi API[10]. Kemudian penelitian yang dilakukan Do-Hun Kang, yang berjudul *Room Temperature Control and Fire Alarm/Suppression IoT Service Using MQTT on AWS* penelitian ini menghasilkan prototipe dengan penggunaan komponen modular namun sudah berbasis *cloud* dan belum memanfaatkan API[11]. Banyaknya penelitian yang relevan dan gambaran dari beberapa hasil penelitian tersebut semakin memperjelas adanya jarak atau pembeda dari penelitian yang sudah dilakukan dengan hasil penelitian yang saat ini dilakukan, yaitu tentang pemanfaatan penggunaan *cloud* API dari Tuya.



Gbr 2. Sensor Suhu dengan Tuya

Selain perangkat sensor suhu, sebagai pendukung digunakan perangkat berupa *smart alarm* yang juga berbasis *cloud* API dari tuya. Perangkat ini dapat aktif jika terpicu oleh perubahan suhu yang telah ditentukan. Contoh *smart alarm* Tuya dapat dilihat pada Gbr 3.

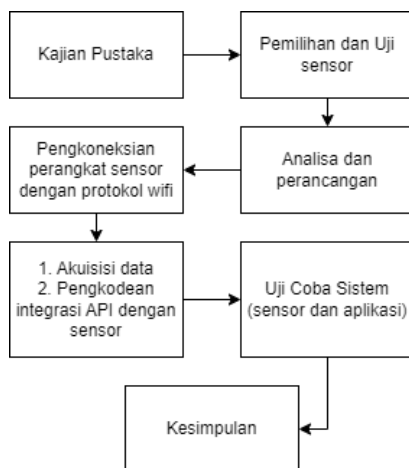


Gbr 3. Contoh Smart Alarm Tuya

### III. METODE PENELITIAN

#### A. Tahapan Penelitian

Pada bagian metode penelitian ini dibuat beberapa tahapan agar lebih mudah dalam pelaksanaan kegiatan penelitian dan juga agar menjaga konsistensi jalannya penelitian, adapun tahapan tersebut tersaji dalam bagan Gbr 1.

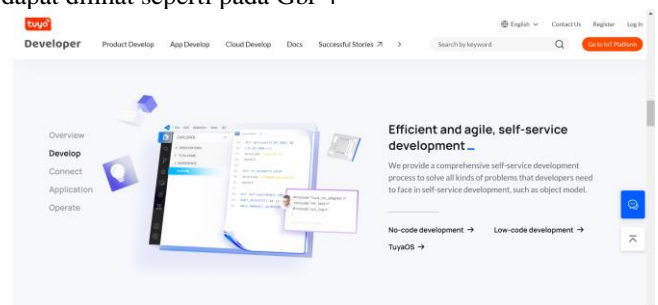


Gbr 1. Tahapan Penelitian

#### B. Alat dan Bahan

Pada tahapan pemilihan perangkat sensor yang digunakan adalah mengamati ketersediaan sensor dipasaran yang sudah terfasilitasi dengan Tuya API, dalam penelitian ini digunakan sensor suhu yang bentuknya dapat dilihat seperti pada Gbr 2.

dari Perangkat lain yang digunakan dalam penelitian adalah Memanfaatkan sebuah *cloud* API dari Tuya API Developer, dimana API tersebut dapat diakses dan dimanfaatkan sebagai alat bantu integrasi antara perangkat keras berupa sensor dengan perangkat lunak berbasis web yang dibangun sendiri, adapun halaman Tuya Developer dapat dilihat seperti pada Gbr 4



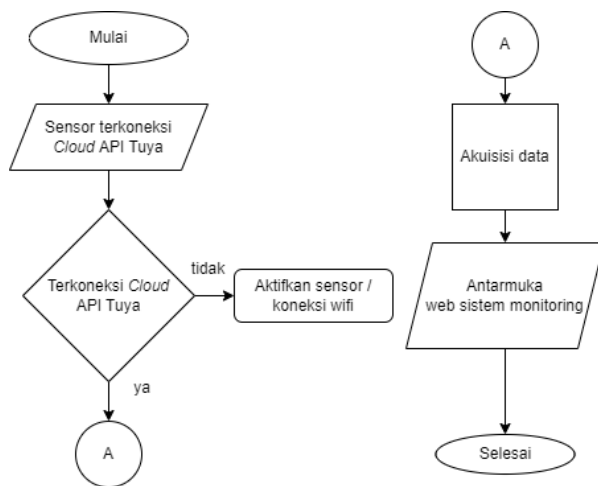
Gbr 4. Halaman Tuya Developer

Dari perangkat sensor suhu dan Tuya API dapat diintegrasikan sebagai kesatuan sistem IoT yang dapat dimanfaatkan sebagai sistem monitoring ruang server. Dimana IoT sendiri merupakan penggabungan antara perangkat keras dapat berupa sensor yang dapat diakses datanya dan dapat diintegrasikan kedalam perangkat lunak dengan dikendalikan dari jarak jauh[12][13][14]. Dan dalam tahapan penelitian ini dilakukan pengujian dari sisi sensor

terhadap akses *cloud* API dari Tuya yang selanjutnya adalah melakukan pengkodean dengan bahasa pemrograman php yang hasil dari bacaan sensor dapat ditampilkan dalam sebuah halaman web dan tentunya diakses melalui koneksi internet.

C. Perancangan

Tahapan ini dilakukan perancangan pembangunan sistem dengan cara yang pertama adalah dilakukan pengkoneksian antara sensor suhu dengan perangkat wifi melalui aplikasi yang tersedia pada Tuya, kemudian dilakukan akuisisi data. Akuisisi data sendiri adalah proses dimana perangkat keras berupa sensor memperoleh nilai atau besaran angka yang nilainya belum dapat dibaca dalam satuan dan atau sudah dapat terbaca dalam satuan[15], dalam penelitian ini dicontohkan satuan °C. Secara teknis penelitian ini dalam mengakuisisi data dengan cara mencoba Sensor suhu dikoneksikan pada API Tuya. Selanjutnya melakukan pengkodean dengan bahasa pemrograman php sebagai alat pemrograman yang dapat mengintegrasikan antara API Tuya dengan beberapa fitur yang akan ditambahkan. Bagan tahapan perancangan dapat dilihat pada Gbr 5.



Gbr 5. Tahapan Perancangan

D. Akuisisi Data

Tahapan akuisisi data pada sensor Suhu dilakukan dengan pemeberian catudaya dan terkoneksi internet melalui wifi, selanjutnya dilakukan integrasi antara sensor dengan *cloud* API kedalam pengkodean awal. Adapun pengkodean awal untuk akuisisi data dengan cara *Install Library* tuyaphpapi untuk koneksi ke Tuya API, kemudian Masukan *access\_key*, *secret\_key* dan *base\_url* kedalam konfigurasi tuya php api. Contoh pengkodean dapat dilihat pada Tabel 1.

TABEL I  
CONTOH PENGKODEAN AKSES API

No	Pengkodean
1	# add api_key, scret_key dan base_url
2	public function __construct()
3	{
4	\$this->CI = &get_instance();
5	\$config = [
6	'accessKey' => \$this->_getAccessKey(),

```

7         'secretKey' => $this->_getSecretKey(),
8         'baseUri' => $this->_getBaseUrl(),
9     ];
10    $this->tuya = new
        tuyapiphp(TuyaApi($config);
11    }
  
```

Langkah setelah melakukan akses API selanjutnya adalah melakukan pengkodean untuk access token, adapun pengkodean acces token dapat dilihat pada Tabel II.

TABEL II  
CONTOH PENGKODEAN AKSES TOKEN

No	Pengkodean
1	# Get Token
2	public function getToken()
3	{
4	\$data = \$this->tuya->token->get_new();
5	if (\$data->results->expire_time < 0) {
6	\$data = \$this->tuya->token->get_refresh(\$data->results->refresh_token);
7	}
8	return \$data->result->access_token;
9	}

Selanjutnya dilakukan pengkodean status *device* dengan memasukan *device\_id*, dan *access\_token*, untuk menghasilkan *output* parameter sesuai dengan perangkat yang dikoneksikan. Contoh pengkodean status *device* dapat dilihat pada Tabel III.

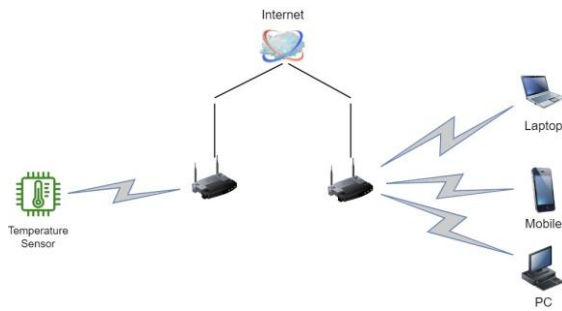
TABEL III  
CONTOH PENGKODEAN STATUS DEVICE

No	Pengkodean
1	# Get Status Device
2	public function getStatusDevice()
3	{
4	\$device_id = 'eb1e4c3fef315e9deexgbb';
5	\$access_token = \$this->tuyatoken->getToken();
6	\$status_device = \$this->tuya->devices(\$access_token->get_status(\$device_id);
7	echo json_encode(\$status_device);
8	}

IV. HASIL DAN PEMBAHASAN

A. Aktivasi Sensor

Sensor suhu sebelum digunakan perlu diberi diaktifkan dengan memberikan catudaya menggunakan baterai ukuran AA dan dikoneksikan ke wifi yang sudah ada sumber internet. Hal ini dilakukan agar sensor dapat terhubung secara *cloud* nantinya. Adapun model komunikasi jaringan dari implementasi ini adalah dibuatnya topologi yang diperlihatkan pada Gbr 5



Gbr 5. Topologi Komunikasi Jaringan

**B. Pengujian Sensor**

Pada pengujian yang dilakukan pada sensor suhu, dalam memperoleh data dilakukan dengan memicu sensor dan mengamati perubahan nilai baca pada sensor dan melihat beberapa *value* yang dihasilkan. Hasil pengujian pada sensor dapat dilihat seperti pada Tabel IV

TABEL IIIV  
HASIL PENGUJIAN PADA SENSOR SUHU

Sensor Uji	Pengujian	Parameter	Value / Nilai Baca
Temperature and Humidity Sensor	mengubah suhu pada AC diruangan,	Berhasil mendeteksi perubahan suhu. Dalam °C	temperature "values": [{"unit": "°C", "min": -200, "max": 600, "scale": 1, "step": 1}]" humidity "values": [{"unit": "%", "min": 0, "max": 1000, "scale": 1, "step": 1}]"

Pada bagian ini sensor suhu diberi pemicu dengan cara merubah setingan suhu pada *Air Conditioner (AC)* yang terdapat pada ruangan server, kemudian mengamati hasil nilai bacaan pada sensor dan melihat parameter apa saja yang dapat diakses dalam *cloud API Tuya*. Hasil dari Perubahan bacaan sensor terhadap perubahan suhu yang dipicu oleh perubahan seting suhu AC diperlihatkan pada Tabel V.

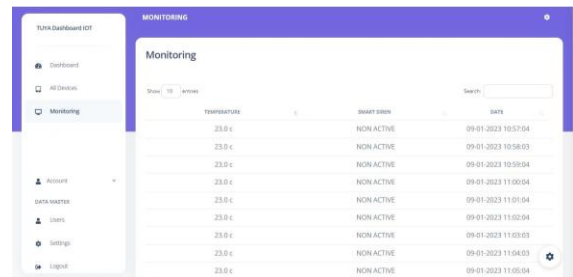
TABEL V  
DATA UJI PERUBAHAN BACAAN SENSOR SUHU

Temperatur AC°C	Temperature Sensor °C	Smart Sirine	Date
24	23.3	NON ACTIVE	03/01/2023 14.24
24	23.3	NON ACTIVE	03/01/2023 14.25
24	23.3	NON ACTIVE	03/01/2023 14.26
24	23.3	NON ACTIVE	03/01/2023 14.27
18	17.4	ACTIVE	03/01/2023 14.28
18	17.4	ACTIVE	03/01/2023 14.29
18	17.4	ACTIVE	03/01/2023 14.30
18	17.4	ACTIVE	03/01/2023 14.36
18	17.4	ACTIVE	03/01/2023 14.37
24	25.1	ACTIVE	03/01/2023 14.38
24	25.1	ACTIVE	03/01/2023 14.39
24	25.1	ACTIVE	03/01/2023 14.40
24	25.1	ACTIVE	03/01/2023 14.41
28	29	ACTIVE	03/01/2023 14.12

28	29	ACTIVE	03/01/2023 14.13
28	29	ACTIVE	03/01/2023 14.14
28	29	ACTIVE	03/01/2023 14.42
28	29	ACTIVE	03/01/2023 14.43

**C. Pengujian Keseluruhan**

Pengujian keseluruhan dilakukan setelah pengujian hasil akuisisi data pada sensor suhu terintegrasi dengan API Tuya, maka pengujian selanjutnya dengan mengintegrasikan hasil bacaan sensor suhu yang terdapat pada *cloud API Tuya* ditarik datanya dengan bahasa pemrograman php, dengan melakukan desain antarmuka dan pengkodean dalam bentuk web kemudian terhadap aplikasi web dilakukan hosting pada server, agar nantinya sistem monitoring dapat diakses secara daring. Web monitoring suhu server telah berhasil menarik data dari *cloud API Tuya* yang ditampilkan dalam menu. Selain halaman monitoring suhu juga diberikan fitur alarm yang akan berbunyi jika ada perubahan suhu diluar rentang batasan. Untuk dapat dilihat hasil pengujian antara sensor suhu, *Cloud API Tuya* dan web yang dibangun dapat diperlihatkan seperti pada Gbr 5.



Gbr. 5 Antarmuka Aplikasi Monitoring Suhu

Halaman pada aplikasi web menampilkan kondisi atau status bacaan sensor suhu dalam satuan °C, dimana sistem tersebut secara *realtime* akan menunjukkan data kondisi suhu didalam ruang server, dan dalma keadaan diluar rentang suhu yang ditentukan maka akan membunyikan sirine nengan status *ACTIVE* atau *NON ACTIVE* jika suhu normal.

## V. KESIMPULAN

Dalam penelitian ini dapat disimpulkan bahwa penggunaan API Tuya berhasil diintegrasikan dengan halaman web yang dirancang dan dibangun secara mandiri, hasil bacaan sensor suhu dapat terintegrasi dengan *cloud* API Tuya yang dapat diakuisisi datanya hingga terbaca pada halaman web dalam satuan °C, membangun sistem monitoring menggunakan sensor suhu Tuya dan API tuya dapat meminimalkan resiko kerusakan fisik dari pada penggunaan sensor komponen / modular. Dalam beberapa tahapan yang dilakukan untuk memonitor suhu ruang server telah berhasil dilakukan dengan menggunakan sensor suhu tuya yang terintegrasi dengan aplikasi berbasis web. Kekurangan dari sistem yang dibangun adalah belum tersedianya akses sensor secara *offline*, dan keberhasilan bacaan sensor sangat tergantung dengan ketersediaan internet dan akses *cloud* dari API Tuya Diharapkan pada penelitian selanjutnya dilakukan pengujian sensor dengan membangun sistem mandiri yang bisa menjalankan bacaan sensor dengan sistem lokal untuk mengurangi resiko ketidaksediaan koneksi internet dan akses *cloud*.

## UCAPAN TERIMA KASIH

Terimakasih disampaikan kepada Allah Yang Maha Kuasa, Politeknik Harapan Bersama, PT. Time Exelindo, PT. Supra Primatama Nusantara, PT. Primakom Interbuana, Nusantara Data Center, Jurnal Publikasi JPIT dan seluruh pihak yang mendukung dalam penelitian ini .

## DAFTAR PUSTAKA

### Journal Article

- [1] D. Fatra and A. Syazili, "Sistem Monitoring Suhu Jarak Jauh Pada Ruang Server Berbasis Internet of Things," *Bina Darma Conf. Comput. Sci.*, vol. 3, no. 2, pp. 401–408, 2021.
- [2] F. H. Purwanto, E. Utami, and E. Pramono, "Implementation and Optimization of Server Room Temperature and Humidity Control System using Fuzzy Logic Based on Microcontroller," *J. Phys. Conf. Ser.*, vol. 1140, no. 1, pp. 390–395, 2018, doi: 10.1088/1742-6596/1140/1/012050.
- [3] M. M. Islam, A. Rahaman, and M. R. Islam, "Development of Smart Healthcare Monitoring System in IoT Environment," *SN Comput. Sci.*, vol. 1, no. 3, pp. 1–11, 2020, doi: 10.1007/s42979-020-00195-y.
- [4] M. K. J. Ramphela, P. A. Owolawi, T. Mapayi, and G. Aiyetoro, "Internet of Things (IoT) integrated data center infrastructure monitoring system," *2020 Int. Conf. Artif. Intell. Big Data, Comput. Data Commun. Syst. icABCD 2020 - Proc.*, pp. 0–5, 2020, doi: 10.1109/icABCD49160.2020.9183873.
- [5] M. Ridwan, D. Djamaludin, and M. Roqib, "Prototype Monitoring Temperature and Humidity Sensor Room Server-Based Internet of Things (IOT)," 2020, doi: 10.4108/eai.23-11-2019.2301576.
- [6] D. E. Kurniawan, M. Iqbal, J. Friadi, R. I. Borman, and R. Rinaldi, "Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications," *J. Phys. Conf. Ser.*, vol. 1351, no. 1, 2019, doi: 10.1088/1742-6596/1351/1/012006.
- [7] P. Periyaldi, A. Bramanto, and A. Wajiansyah, "Implementasi Sistem Monitoring Suhu Ruang Server Satnetcom Berbasis Internet Of Things (Iot) Menggunakan Protokol Komunikasi Message Queue Telemetry Transport (Mqt)," *JIT (Jurnal Teknol. Terpadu)*, vol. 6, no. 1, p. 23, 2018, doi: 10.32487/jit.v6i1.435.
- [8] M. Adam and M. Zurairah, "UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA," 2021.
- [9] F. A. Deswar and R. Pradana, "Monitoring Suhu Pada Ruang Server Menggunakan Wemos D1 R1 Berbasis Internet of Things (Iot)," *Technol. J. Ilm.*, vol. 12, no. 1, p. 25, 2021, doi: 10.31602/tji.v12i1.4178.
- [10] M. Safii and N. Indrayani, "Perancangan Piranti Lunak Responsive

- Untuk Monitoring Ruang Server Menggunakan NodeMCU Esp8266 Berbasis Internet of Things," *J. Ilm. Matrik*, vol. 22, no. 3, pp. 270–277, 2020, doi: 10.33557/jurnalmatrik.v22i3.1121.
- [11] D. H. Kang *et al.*, "Room Temperature Control and Fire Alarm/Suppression IoT Service Using MQTT on AWS," *2017 Int. Conf. Platf. Technol. Serv. PlatCon 2017 - Proc.*, 2017, doi: 10.1109/PlatCon.2017.7883724.
- [12] M. O. Onibonoje, P. N. Bokoro, N. I. Nwulu, and S. L. Gbadamosi, "An IoT-Based Approach to Real-Time Conditioning and Control in a Server Room," *2019 Int. Conf. Artif. Intell. Data Process. Symp. IDAP 2019*, pp. 1–6, 2019, doi: 10.1109/IDAP.2019.8875880.
- [13] M. Natsir, D. B. Rendra, and A. D. Y. Anggara, "Implementasi IOT Untuk Sistem Kendali AC Otomatis Pada Ruang Kelas di Universitas Serang Raya," *J. PROSISKO (Pengembangan Ris. dan Obs. Rekayasa Sist. Komputer)*, vol. 6, no. 1, pp. 69–72, 2019.
- [14] R. K. Putra Asmara, "Rancang Bangun Alat Monitoring Dan Penanganan Kualitas Ait Pada Akuarium Ikan Hias Berbasis Internet Of Things (IOT)," *J. Tek. Elektro dan Komput. TRIAC*, vol. 7, no. 2, pp. 69–74, 2020, doi: 10.21107/triac.v7i2.8148.
- [15] Y. Wang, T. Sekine, Y. Takeda, K. Yokosawa, and H. Matsui, "Fully Printed PEDOT : PSS-based Temperature Sensor with High Humidity Stability for Wireless Healthcare Monitoring," *Sci. Rep.*, pp. 1–8, 2020, doi: 10.1038/s41598-020-59432-2.