

# RANCANG BANGUN APLIKASI ANDROID UNTUK MENGONTROL LAMPU MENGUNAKAN MIKROKONTROLER NODEMCU

---

Evan Fauzi Pranendra, Dega Surono Wibowo, Rosid Mustofa

D IV Teknik Informatika Politeknik Harapan Bersama Tegal

Jln. Mataram No. 09 Tegal

Telp/Fax 90283) 35200

E-mail: evanfauzi0@gmail.com

---

## ABSTRAK

Lampu merupakan sumber penerangan pengganti sinar matahari. Pada umumnya penerangan lampu masih menggunakan sistem *on/off* pada saklar. Produk saklar yang sering digunakan masih berupa tombol yang harus kita tekan untuk mengoperasikannya. Pada malam hari perlu penerangan untuk SDN Kraton 3 Tegal agar tidak terjadi hal yang tidak diinginkan misalnya pencurian. Selain itu, untuk memudahkan karyawan dalam menyalakan lampu lewat aplikasi android. Sistem kendali lampu dengan menggunakan *Mikrokontroler NodeMCU* yang terhubung dengan internet dapat dikendalikan melalui aplikasi android di smartphone yang juga terhubung dengan internet dan menggunakan *database Firebase* yang dapat diimplementasikan dengan *mikrokontroler*, android, dan *website*. Menggunakan sensor cahaya/*LDR* yang dapat mendeteksi lampu sudah menyala atau belum, sensor arus *ACS712* dan sensor tegangan *ZMPT101B* yang dimanfaatkan untuk menghitung nilai arus, tegangan dan daya yang dikeluarkan. Dan juga menghitung biaya listrik yang dikeluarkan dari daya yang telah didapat dari sensor arus dan tegangan.

*Kata Kunci : Lampu, Internet of Things, Mikrokontroler, NodeMCU, Firebase, Sensor Cahaya/LDR, Sensor ACS712, Sensor ZMPT101B*

## PENDAHULUAN

Lampu merupakan sumber cahaya yang sangat penting sebagai pengganti matahari. Lampu sebagai sumber penerangan pada malam hari yang sangat berguna bagi masyarakat. Penggunaan

lampu di masyarakat sekarang ini dinilai kurang efektif dan masih sering mengabaikan penggunaannya, sering kali lampu masih tetap menyala walaupun tidak dipakai. Hal semacam ini merupakan suatu pemborosan. Disisi lain proses

mematikan dan menhidupkan lampu secara manual masih dirasa banyak membuang banyak waktu. Penggunaan lampu umumnya masih menggunakan sistem *on/off* pada saklar, misalnya pada gedung-gedung perkantoran, industri maupun sekolahan. Pada gedung seperti ini biasanya memiliki beberapa lantai, jika ingin menyalakan lampu harus menekan saklar yang dinyalakan satu persatu antar ruangan. Jika gedung bertingkat maka petugas atau karyawan harus naik turun lantai untuk menyalakan lampu satu persatu.

Seperti pada gedung sekolah, di malam hari juga perlu penerangan untuk sekolah agar tidak terjadi hal yang tidak diinginkan misalnya pencurian. Ini merupakan tugas penjaga sekolah untuk selalu memastikan lampu sekolah tetap menyala pada malam hari sampai pagi hari. Penjaga sekolah harus buka tutup pintu kelas untuk menyalakan lampu pada saklar di dalam kelas untuk lampu yang ada di depan kelas setiap hari, apalagi pada

SDN Kraton 3 Tegal sudah lantai tingkat 2 maka harus naik tangga lagi untuk menyalakan lampu. Dan pada pagi hari harus sudah mematikan lampu dengan cara seperti tadi untuk mematikan lampu. Hal ini sering dikeluhkan oleh penjaga sekolah SDN Kraton 3 Tegal.

Penelitian terdahulu telah dibuat tentang “Implementasi Teknologi *Internet Of Things (IoT)* Pada Rumah Pintar Berbasis *Mikrokontroler Esp8266*” yang telah berhasil membuat sistem kontrol lampu dengan perintah suara *google assistant* dengan media koneksi internet [1]. Penelitian yang lain dengan judul “Sistem Kendali Berbasis *Mikrokontroler* Menggunakan Protokol *MQTT* pada *Smarthome*” telah berhasil membuat sistem kendali untuk melakukan *controlling* dan *monitoring* pada *smarthome* menggunakan protokol *Message Queue Telemetry Transport (MQTT)* sebagai komunikasi di dalam sistem *smarthome* [2]. Selanjutnya penelitian dengan judul “Membangun

Sistem Kendali Jarak Jauh Pada Ruang Kelas Menggunakan *Thingspeak* Berbasis *Web*” telah dibuat sistem kendali ruang jarak jauh menggunakan *website* sebagai halaman kendali dengan memanfaatkan *thingspeak* sebagai *cloud server* [3]. Berikutnya penelitian dengan judul “Penerapan *Internet Of Things (IoT)* Untuk Kontrol Lampu Menggunakan Arduino Berbasis *Web*” telah berhasil membuat sistem kontrol dan monitor lampu dengan memanfaatkan *cayenne* sebagai *server* [4].

Berdasarkan permasalahan yang sedang terjadi, maka dilakukan penelitian yang berjudul “Rancang Bangun Aplikasi Android untuk Mengontrol Lampu Menggunakan *Mikrokontroler NodeMCU*”. Aplikasi ini dapat menjadi solusi dari masalah karyawan dan penjaga sekolah dalam menyalakan lampu di lorong kelas tanpa harus menekan saklar di tiap kelas. Aplikasi ini dapat menyalakan lampu dan mengecek kondisi lampu dengan jarak jauh dengan syarat harus terhubung ke koneksi internet.

## METODOLOGI PENELITIAN

Penelitian ini menggunakan metode penelitian dan pengembangan atau *Research and Development (R&D)*. Pengumpulan data melalui Studi Literatur dan Wawancara. Pada perancangan sistem melalui tahap yaitu : Perancangan *UML*, Perancangan *User Interface*. Pada Pengujian sistem menggunakan Pengujian *Blackbox*.

### Internet of Thinks

*Internet of Thinks (IoT)* saat ini merupakan teknologi yang sedang berkembang tersebar secara global. *IoT* telah melangkah di berbagai bidang terdiri dari industri, pemerintahan, akademis, dan masih berbagai penelitian dilakukan. Sekarang ini *IoT* memainkan peran penting dalam sektor bisnis, banyak bisnis yang suram dan berkembang bergantung pada *IoT* dan Industri *IoT*. *IoT* berkembang mulai dari sipil hingga sektor keamanan. Bidang seperti pertanian, *hortikultura*, perawatan kesehatan, *space*, manufaktur,

konstruksi, air, dan pertambangan, sedang transit dari pengaturan infrastruktur lama ke pengaturan *IoT* modern [5].

### **NodeMCU**

*NodeMCU* merupakan sebuah *open source platform IoT* dan pengembangan *kit* yang menggunakan bahasa pemrograman *Lua* untuk membantu dalam membuat *prototype* produk *IoT* atau bisa dengan memakai *sketch* dengan *arduino IDE*. Pengembangan *kit* ini didasarkan pada modul *ESP8266*, yang mengintegrasikan *GPIO*, *PWM (Pulse Width Modulation)*, *IIC*, *1-Wire* dan *ADC (Analog to Digital Converter)* semua dalam satu *board*. *NodeMCU* berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur *WiFi* dan *Firmware*-nya yang bersifat *opensource* [6].

### **Relay**

*Relay* merupakan komponen elektronika berupa saklar atau *switch* elektrik yang dioperasikan secara listrik

dan terdiri dari 2 bagian utama yaitu *Elektromagnet (coil)* dan *mekanikal* (seperangkat kontak Saklar/*Switch*). Komponen elektronika ini menggunakan prinsip *elektromagnetik* untuk menggerakkan saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi [6].

### **Firestore**

*Firestore Realtime Database* adalah *database* yang *host* nya berada di *cloud*. Data disimpan sebagai *JSON* dan disinkronkan secara *realtime* ke setiap klien yang terhubung. *Firestore Realtime Database* menggunakan sinkronisasi data setiap kali data berubah, semua perangkat yang terhubung akan menerima *update* dalam waktu milidetik [7].

### **XAMPP**

*XAMPP* merupakan program *Apache Web Server*. Program *XAMPP* selain memiliki kemampuan sebagai penyedia layanan *web*, juga memiliki basis data

terintegrasi yaitu *MySQL*. Basis data diperlukan untuk menyimpan data-data dari proses aktifitas pengguna pada sistem. Proses instalasi *web server XAMPP* begitu mudah tinggal mengeksekusi file setup *XAMPP*, dan proses instalasi pun berjalan. Setelah selesai proses instalasi, melakukan pengujian terhadap layanan *web server*. Masukkan alamat *web* dengan mengetikkan teks “*localhost*”, maka muncul halaman *web XAMPP* yang menandakan bahwa layanan *web server* telah bekerja [8].

### **Arduino IDE**

*Integrated Development Environment (IDE)* yang diperuntukan untuk membuat perintah atau *source code*, melakukan pengecekan kesalahan, kompilasi, upload program, dan menguji hasil kerja *arduino* melalui *serial monitor*. *Arduino IDE* adalah perangkat lunak yang sangat canggih ditulis dengan menggunakan *Java*. *Arduino IDE* terdiri dari: *Editor* program, sebuah *window* yang memungkinkan pengguna menulis dan

pengedit program dalam bahasa *Processing* [6].

### **Bahasa Lua**

*Lua* merupakan bahasa pemrograman dinamis yang ditujukan untuk digunakan sebagai bahasa skrip, dan cukup ringkas untuk disisipkan dalam berbagai jenis platform utama. *Lua* hanya mendukung beberapa jenis struktur data atomik seperti; boolean, floating point, serta string. Jenis-jenis tipe/struktur data lainnya seperti; larik, set, ataupun list direpresentasikan dalam *Lua* melalui satu bentuk tipe data, table.

### **Unified Modeling Language**

*Unified Modeling Language (UML)* adalah metode pemodelan secara visual sebagai sarana untuk merancang atau membuat *software* berorientasi objek. Karena *UML* ini merupakan bahasa visual untuk pemodelan bahasa berorientasi objek, maka semua elemen dan diagram berbasiskan pada paradigma *object oriented*.

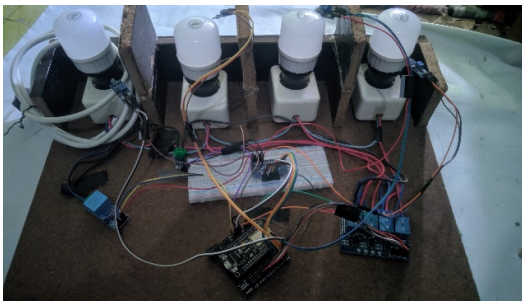
## HASIL DAN PEMBAHASAN

### Hasil Penelitian

Dari tahapan perancangan hingga pembuatan Rancang Bangun Aplikasi Android untuk Mengontrol Lampu Menggunakan Mikrokontroler NodeMCU (Studi Kasus: SDN Kraton 3 Tegal) diperoleh hasil sebagai berikut.

### Rangkaian *Prototype*

Rangkaian Rancang Bangun Aplikasi Android untuk Mengontrol Lampu Menggunakan Mikrokontroler NodeMCU yang telah dibuat menjadi *prototype* yang ada pada gambar dibawah.

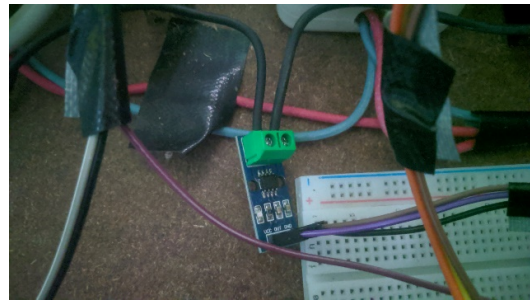


Gambar 1. Rangkaian *Prototype*

### Rangkaian Pemasangan Sensor *ACS712*

Dibawah ini adalah gambar peletakan sensor arus *ACS712* yang di sambungkan

dengan salah satu kabel lampu agar dapat mendeteksi arus listrik.



Gambar 2. Rangkaian Pemasangan Sensor  
*ACS712*

### Rangkaian Pemasangan Sensor

#### *ZMPT101B*

Dibawah ini adalah gambar peletakan sensor tegangan *ZMPT101B*, dari kabel lampu di *ACS712* disambungkan ke sensor ini dan kabel lampu plus (+) dan min (-) untuk di colokkan ke sumber listrik.



Gambar 3. Rangkaian Pemasangan Sensor  
*ZMPT101B*

## Rangkaian Pemasangan Sensor Cahaya/LDR

Dibawah ini adalah gambar peletakan sensor cahaya, menggunakan 4 sensor cahaya yang masing-masing didekatkan dengan setiap lampu untuk mendeteksi cahaya dari lampu.

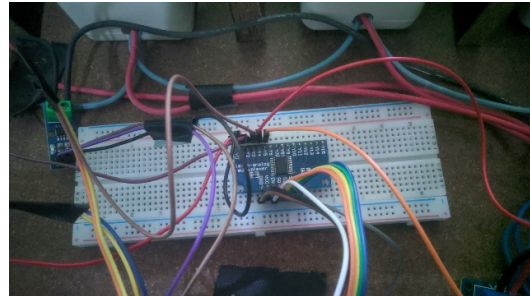


Gambar 4. Rangkaian Pemasangan Sensor Cahaya/LDR

## Rangkaian Kabel Jumper

Penggunaan kabel *jumper* sebagai penyambung antar komponen yang digunakan, dengan adanya kabel *jumper* memudahkan konektifitas antar komponen. Sehingga dapat memaksimalkan fungsi sistem, selain itu juga penggunaan kabel *jumper* tergolong mudah. Dan juga penggunaan 16-Channel Analog

*Multiplexer* dengan ini sensor dapat membaca nilai *analog* dari sensor agar bisa dibaca *mikrokontroler*. Berikut ini gambar tentang rangkaian penggunaan kabel *jumper*.



Gambar 5. Rangkaian Kabel *Jumper*

## Pengujian Sensor

Pada tahap ini, menguji ketiga sensor apakah sudah bekerja sesuai dengan yang diinginkan, dimana data yang dihasilkan oleh sensor sama dengan data yang akan dikirim ke *database Firebase*.

Tabel 1. *Test Case* Pengujian Sensor LDR/Cahaya

Nilai Analog	Lampu 1	Lampu 2	Lampu 3	Lampu 4	Keterangan
Dibawah 300	ON	ON	ON	ON	Berhasil dinyalakan
Diatas 300	Menyala/Error karena seharusnya lampu mati	Menyala/Error karena seharusnya lampu mati	Menyala/Error karena seharusnya lampu mati	Menyala/Error karena seharusnya lampu mati	Lampu dalam keadaan menyala tetapi sensor membaca nilai analog diatas 300 maka lampu rusak / ada kabel yang putus.
Nilai Analog	Lampu 1	Lampu 2	Lampu 3	Lampu 4	Keterangan
Dibawah 300	OFF	OFF	OFF	OFF	Lampu dalam keadaan mati tetapi sensor membaca nilai analog dibawah 300 maka kabel rangkaian lampu ada yang salah / mati lampunya.
Diatas 300	Mati/Error karena seharusnya lampu menyala	Mati/Error karena seharusnya lampu menyala	Mati/Error karena seharusnya lampu menyala	Mati/Error karena seharusnya lampu menyala	Berhasil dimatikan

Dari tabel tersebut dapat disimpulkan bahwa jika sensor cahaya membaca nilai sensornya dibawah 300 dalam keadaan lampu ON maka lampu berhasil dinyalakan, sebaliknya jika sensor membaca diatas 300 dalam keadaan lampu OFF maka lampu berhasil dimatikan. Dan jika sensor membaca diatas 300 tetapi lampu sedang ON maka ada lampu yang rusak / ada kabel yang putus. Kemudian jika sensor membaca dibawah 300 tetapi lampu sedang OFF maka ada kabel rangkaian lampu yang salah / mati lampunya.

Tabel 2. Test Case Pengujian Sensor ACS712 dan ZMPT101B

Lampu	Sensor arus ACS712 (A)	Sensor tegangan ZMPT101B (V)	Daya (W)	Error (%)
1	4	56	224	15
2	5	60	240	20
3	8	100	800	40
4	5	55	220	10
ON ALL	16	180	2880	50

Dari tabel diatas dapat disimpulkan bahwa keakuratan sensor arus dan tegangan ini masih belum cukup, karena arus dan tegangan yang dibaca tidak hanya lampu saja melainkan besarnya arus dari *power supply* yang diperoleh dari *mikrokontroler*. Hasil dari daya (W) diperoleh dari hasil arus dikalikan dengan tegangan.

## Pembahasan

Pada penelitian ini telah dihasilkan sebuah alat yang bisa digunakan untuk mengontrol lampu dan bisa *monitoring* besarnya arus dan tegangan yang digunakan pada lampu tersebut. Dengan menggunakan aplikasi android diharapkan pengguna bisa dengan mudah menyalakan lampu tanpa harus menekan saklar. Adapun *backend website* untuk mudah melihat data laporan lampu yang telah



tersimpan pada *firebase*. Melalui *website* ini pengguna bisa melihat informasi dari lampu secara *realtime* jika sudah menyala atau mati dan melihat *history* data tanggal saat menyalakan lampu atau mematikan lampu. Selain itu, *website* ini untuk menyimpan data laporan stok lampu jika ada kerusakan pada lampu sebagai cadangan lampu yang ada. Jika kemungkinan setiap bulannya ada pergantian lampu maka dapat simpan data laporannya di *website*. Dengan sensor cahaya/*LDR*, sensor arus *ACS712*, dan sensor tegangan *ZMPT101B* sebagai deteksi lampu jika ada kesalahan ketika dinyalakan bisa cukup baik di gunakan karena pengguna bisa mendapatkan informasi jika ada lampu yang rusak. Dengan tingkat akurasi dari sensor arus dan tegangan yang lumayan digunakan untuk mengetahui perangkat *mikrokontroler* sudah berjalan dengan baik atau belum. Dengan menggunakan sensor arus *ACS712* dan tegangan *ZMPT101B* ini, bisa digunakan untuk menghitung kira-kira

biaya listrik yang akan dikeluarkan setiap bulannya. Untuk menghitung biaya listrik, perlu mengetahui golongan tarif listrik yang digunakan. Diketahui pada SDN Kraton 3 kira-kira menggunakan golongan 2.200 *VA (Volt Ampere)*. Jika tarif dasar listrik untuk golongan 1.301 - 2.200 *VA* adalah Rp. 1.444,70 per *kWh (Kilowatt per Hour)*. Langkah selanjutnya menghitung total daya listrik yang dikeluarkan, berdasarkan hasil pengujian sensor arus dan tegangan diperoleh total daya yang digunakan untuk menhidupkan 4 buah lampu dan *mikrokontroler* yaitu 2.880 *Watt*. Jika pemakaian per hari setidaknya 12 jam, maka estimasi penggunaan dayanya menjadi  $2880 \times 12 = 34.560 \text{ Watt}$ . Selanjutnya diubah dulu ke *kWh* dengan membagi jumlah penggunaan daya dengan 1.000, atau  $34.560 : 1.000 = 34,56 \text{ kWh}$ . Kemudian dikalikan dengan tarif dasar listrik,  $34,56 \times 1.444,70$  hasilnya Rp. 49.929,00. Untuk menghitung biaya listrik dalam sebulan maka dikalikan dengan 30 yaitu  $49.929 \times 30 = 1.497.870$  Rupiah.

Maka dalam sebulan kira-kira biaya listrik yang dikeluarkan Rp. 1.497.870,00. Penggunaan sensor arus dan tegangan ini cukup mudah dan berguna untuk menghitung kira-kira biaya listrik yang harus dikeluarkan per bulannya.

Berdasarkan hasil “*Test Case*” pengujian sensor cahaya dapat disimpulkan bahwa sensor cahaya akan menampilkan notifikasi berhasil dinyalakan pada aplikasi android jika nilai sensornya dibawah 300, sebaliknya sensor cahaya akan menampilkan notifikasi berhasil dimatikan jika nilai sensor diatas 300. Dan sensor cahaya akan menampilkan notifikasi lampu yang mati / rusak / ada kabel yang putus jika nilai sensor diatas 300 tetapi kondisi lampu sedang *ON*. Kemudian sensor cahaya akan menampilkan notifikasi *error* kabel rangkaian lampu yang salah / mati lampunya jika nilai sensor dibawah 300 tetapi kondisi lampu sedang *OFF*. Setelah sensor cahaya, pengujian keakuratan sensor arus dan tegangan ini masih belum

cukup, karena arus dan tegangan yang dibaca tidak hanya lampu saja melainkan besarnya arus dari *power supply* yang diperoleh dari *mikrokontroler*. Dan tingkat *error*-nya terbilang cukup tinggi, maka dari itu dibutuhkan sensor arus dan tegangan yang lebih bagus lagi tingkat keakuratannya.

## **KESIMPULAN DAN SARAN**

### **Kesimpulan**

Berdasarkan hasil penelitian dan pengembangan yang dilakukan, maka disimpulkan bahwa :

1. Telah dibuat aplikasi kontrol lampu untuk mengontrol lampu dengan mudah menggunakan aplikasi android untuk *turn ON/OFF*. Menggunakan *server database* dari *Firebase* yang dengan mudah di integrasikan ke 3 *service* antara lain, menyimpan *input*-an dari *relay* ataupun sensor-sensor yang dikendalikan oleh *mikrokontroler NodeMCU* ke *firebase*, aplikasi android

yang terhubung *firebase* untuk mengontrol lampunya.

2. Berdasarkan pengujian "*Test Case*", penelitian ini menggunakan 3 sensor yang digunakan yaitu, sensor cahaya/*LDR*, sensor arus *ACS712*, dan sensor tegangan *ZMPT101B*. Hasil yang diperoleh dari sensor cahaya sudah baik karena bisa digunakan untuk mendeteksi nyala/mati dari lampu dengan baik, sensor arus yang dihasilkan juga cukup baik digunakan untuk mendeteksi arus yang masuk dari sumber listrik dan sensor tegangan yang memperoleh hasil yang cukup baik juga untuk mendeteksi tegangan yang masuk dari sumber listrik walaupun kedua sensor arus dan tegangan ini tingkat keakuratannya kurang tetapi bisa digunakan dengan baik sesuai penggunaannya.
3. *Website* digunakan untuk mengecek kondisi lampu dalam urutan tanggal, menyimpan data laporan penggunaan lampu, dan mengatur jumlah lampu

yang akan digunakan sesuai jumlah *relay* yang terdapat pada *mikrokontroler*.

### **Saran**

Berdasarkan pembahasan hasil penelitian, maka beberapa dapat di ajukan sebagai berikut :

1. Pada penelitian ini masih banyak kekurangan karena tingkat ke akuratan sensor masih ada yang belum cukup dan karena itu diperlukan pengembangan lagi agar bisa lebih akurat lagi dengan mengganti sensor yang lebih akurat atau dengan mengganti *mikrokontroler* yang mampu menjalankan beberapa sensor dengan cepat tanpa harus menunggu *delay* yang seperti ada di *NodeMCU*.
2. Pada aplikasi android masih banyak kekurangan pada fitur oleh karena itu untuk tahapan produksi massal di butuhkan pengembangan lagi agar lebih banyak memiliki fitur yang dimiliki.
3. Hasil penelitian ini masih berupa *prototype*, diharapkan dapat di

integrasikan ke lapangan agar bisa mendapatkan manfaatnya secara langsung bagi pengguna.

## DAFTAR PUSTAKA

- [1] R. Rizky, Z. Hakim, A. M. Yunita, and N. N. Wardah, "Implementasi Teknologi Iot ( Internet Of Think ) Pada Rumah Pintar Berbasis Mikrokontroler ESP 8266," vol. 4, no. 2, pp. 278–281, 2020.
- [2] H. A. Rochman, R. Primananda, and H. Nurwasito, "Sistem Kendali Berbasis Mikrokontroler Menggunakan Protokol MQTT pada Smarthome," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 1, no. 6, pp. 445–455, 2017, [Online]. Available: <http://j-ptiik.ub.ac.id>.
- [3] P. H. Weng, A. B. M. W, and G. C. Setyawan, "Membangun Sistem Kendali Jarak Jauh Pada Ruang Kelas Menggunakan Thingspeak Berbasis Web," vol. 1, no. 1, pp. 1–12, 2020.
- [4] B. Artono and R. G. Putra, "Penerapan Internet Of Things (IoT) Untuk Kontrol Lampu Menggunakan Arduino Berbasis Web," *J. Teknol. Inf. dan Terap.*, vol. 5, no. 1, pp. 9–16, 2019, doi: 10.25047/jtit.v5i1.73.1, no. 1, pp. 1–12, 2020.
- [5] S. Uma, R. Eswari, R. Bhuvanya, and G. S. Kumar, "IoT based Voice/Text Controlled Home Appliances," *Procedia Comput. Sci.*, vol. 165, no. 2019, pp. 232–238, 2019, doi: 10.1016/j.procs.2020.01.085.
- [6] A. Purwanto and S. Lutfi, "Pengendalian Lampu Rumah Berbasis Google Asisstant Melalui Smartphone Menggunakan NodeMCU-12E ESP8266 di Nuke Komputer Service," *J. Himsya Tech*, vol. 20, no. 2, pp. 1–6, 2019.
- [7] Pratiwi, "Rancang Bangun Sistem Telecontrolling Pada Ruangan Oven Batang Rokok Berbasis Android Secara Realtime ( Studi Kasus di Pabrik Indokretek ) Program Studi Jaringan Telekomunikasi Digital , Teknik Elektro ,," *Pratiwi*, pp. 8–13, 2018.
- [8] T. F. Siallagan, "Rancang Bangun Sistem Peringatan Dini Terhadap Kebakaran Berbasis Bot Telegram Menggunakan Pendahuluan Kajian Pustaka," vol. VI, no. 1, pp. 61–70, 2019.