

RANCANG BANGUN SISTEM KONTROL DAN MONITORING STOP KONTAK CERDAS BERBASIS INTERNET OF THINGS

Yasril Harviandra

Politeknik Negeri Pontianak; Jalan Jendral Ahmad Yani, Telp (0561) 736180

Jurusan Teknik Elektro, Politeknik Negeri Pontianak, Pontianak

e-mail: yasrilharviandra@gmail.com

Abstract

The presence of IoT technology helps humans carry out their daily activities, one of which is its use for control and monitoring. It is hoped that the design of this tool can help reduce the use of electrical energy that is wasted due to excessive use. This tool can help users control and monitor electrical power consumption remotely using the Arduino IoT Cloud platform which can be accessed via smartphone or desktop devices. The interface displays the on/off button function, timer, scheduling, control with voice commands, display of power usage, and also power usage graphs which will make it easier for users to analyze the electrical power consumption used. The designed system can function as planned, only there is a slight delay between the time the instruction is given and the response from the device.

Abstrak

Hadirnya teknologi IoT membantu manusia dalam melaksanakan aktivitasnya sehari-hari, salah satunya adalah pemanfaatannya untuk melakukan kontrol dan monitoring. Perancangan alat ini diharapkan dapat membantu untuk mengurangi penggunaan energi listrik yang terbuang sia-sia karena digunakan secara berlebihan. Alat ini dapat membantu penggunaannya dalam mengontrol dan juga memonitoring penggunaan konsumsi daya listrik dari jarak jauh dengan menggunakan platform Arduino IoT Cloud yang dapat diakses melalui perangkat smartphone maupun desktop. Pada tampilan antarmuka dirancang fungsi on/off button, timer, penjadwalan, kontrol dengan perintah suara, tampilan dari penggunaan daya, dan juga grafik penggunaan daya yang akan memudahkan penggunaannya dalam menganalisis konsumsi daya listrik yang digunakan. Sistem yang dirancang dapat berfungsi sebagaimana yang telah direncanakan, hanya saja terdapat sedikit delay antara waktu pemberian instruksi dan respon dari perangkat.

Article History

Submitted: 2 September 2024

Accepted: 7 September 2024

Published: 14 September 2024

Key Words

Control and Monitoring System, Internet of Things, ESP32, PZEM-004T, Arduino IoT Cloud, Smart Socket, Smart Plug.

Sejarah Artikel

Submitted: 2 September 2024

Accepted: 7 September 2024

Published: 14 September 2024

Kata Kunci

Sistem Kontrol dan Monitoring, Internet of Things, ESP32, PZEM-004T, Arduino IoT Cloud, Stop Kontak

1. PENDAHULUAN

Arduino IoT Cloud merupakan platform pada perangkat smartphone dan juga desktop yang dapat mengimplementasikan suatu sistem kontrol dan monitoring berbasis IoT yang pada penelitian ini diterapkan pada sebuah sistem stop kontak. Karena sistem tersebut berbasis IoT maka dari itu pengguna dapat melakukan kontrol dan juga monitoring dari jarak yang jauh selagi terkoneksi dengan internet. Tujuan utama dari penelitian ini lebih berfokus pada kemudahan penggunaannya dalam mengoperasikan peralatan elektronik rumah tangga dari jarak jauh serta dapat dilakukannya monitoring atas konsumsi daya yang digunakan. Dengan demikian, maka durasi penggunaan peralatan yang tidak dibutuhkan dapat ditekan sehingga diharapkan akan memberikan dampak berupa penggunaan energi listrik yang lebih hemat.

Pada penelitian ini, permasalahan yang diangkat berupa bagaimana cara agar stop kontak dapat dikendalikan dari jarak jauh, bagaimana cara untuk memonitoring penggunaan daya listrik dari jarak jauh, bagaimana agar kontrol dari stop kontak tersebut dapat diberikan timer serta dilakukannya penjadwalan, bagaimana cara agar dapat dilakukannya kontrol dengan menggunakan perintah suara (voice command), bagaimana cara untuk memaksimalkan layanan hosting gratis dari Arduino IoT Cloud, bagaimana cara untuk membuat sistem proteksi arus berlebih pada alat tersebut.

2. TINJAUAN PUSTAKA

Kasus pertama dibahas oleh Suryadi Eka Saputra dalam tugas akhir yang berjudul “Monitoring dan kontrol listrik rumah tinggal berbasis IoT/ESP32” pada tahun 2022. Sistem yang dirancang pada penelitian ini memiliki konsep berupa kontrol serta monitoring dari pemakaian energi listrik rumah tinggal. Menggunakan sensor Pzem untuk mengukur daya yang kemudian diteruskan ke ESP32 sehingga dapat dimonitoring melalui perangkat smartphone. Kekurangan dari sistem ini ialah kontrol yang yang dapat dilakukan sangat terbatas yaitu hanya untuk on/off lampu rumah saja dan. (Satria, 2022).

Kasus kedua dibahas oleh Abdul Hanif dalam skripsi yang berjudul “Rancangan Sistem Kontrol dan Monitoring Instalasi Listrik Berbasis Internet of Things” pada tahun 2022. Sistem yang dirancang juga berupa kontrol dan monitoring pemakaian energi listrik yang menggunakan Pzem sebagai sensor pengukur dayanya. Terdapat 3 buah lampu dan sebuah stop kontak yang memungkinkan penggunaanya untuk lebih leluasa dalam memilih peralatan yang akan di kontrol. Kekurangan dari sistem ini adalah pengontrolan yang dapat dilakukan hanya sebatas on/off saja. (Hanif, 2022)

Kasus ketiga dibahas oleh Iqbal Pramudya, Syaifurrahman dan Seno D. Panjaitan dalam Jurnal yang berjudul “Penerapan Aplikasi Blynk untuk sistem kontrol dan monitoring stop kontak cerdas”. Sistem yang dirancang pada penelitian ini memiliki fitur kontrol on/off serta monitoring pemakaian daya listrik yang dapat dipantau baik melalui monitor pada alat itu sendiri maupun melalui perangkat smartphone atau website dari Blynk itu sendiri. Berbentuk sebuah stop kontak sehingga penggunaanya dapat lebih fleksibel dalam memilih peralatan apa saja yang ingin dikontrol. Terdapat sensor DHT 22 yang berfungsi sebagai sensor suhu sebagai fitur tambahan dari alat ini. Kekurangan pada sistem ini terdapat pada kontrolnya yang hanya bisa digunakan untuk menyambungkan ataupun memutus kontak saja. Pramudya dkk. (2022)

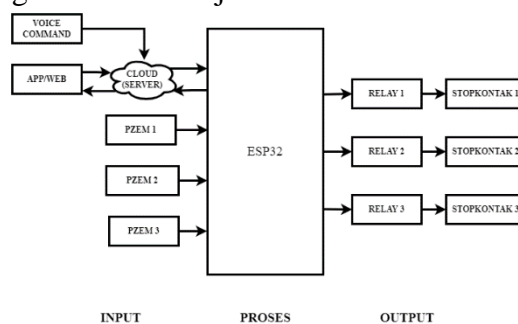
Berbeda dengan referensi acuan yang digunakan, pada penelitian ini ditambahkan beberapa fitur tambahan berupa pewaktu (timer dan penjadwalan), kontrol dengan menggunakan perintah suara (voice command), penambahan jumlah stop kontak menjadi 3 buah, penggunaan layanan IoT yang berbeda, serta adanya proteksi dari arus berlebih. Diharapkan dengan penambahan fitur ini akan dapat meningkatkan kemudahan serta kenyamanan penggunaanya dalam mengoperasikan alat ini.

3. PERANCANGAN SISTEM

Pada perancangan sistem, akan dirancang berbagai macam aspek yang dibutuhkan didalam pengerjaan alat ini. Aspek tersebut berupa diagram blok, flowchart, perancangan prototype, perancangan rangkaian kelistrikan, perancangan pengawatan sistem, penentuan variabel hingga perancangan dashboard Arduino IoT Cloud.

3.1 Diagram Blok

Diagram ini menggambarkan struktur dan interkoneksi komponen utama yang membentuk sistem ini. Diagram blok ditunjukkan oleh Gambar 1 :

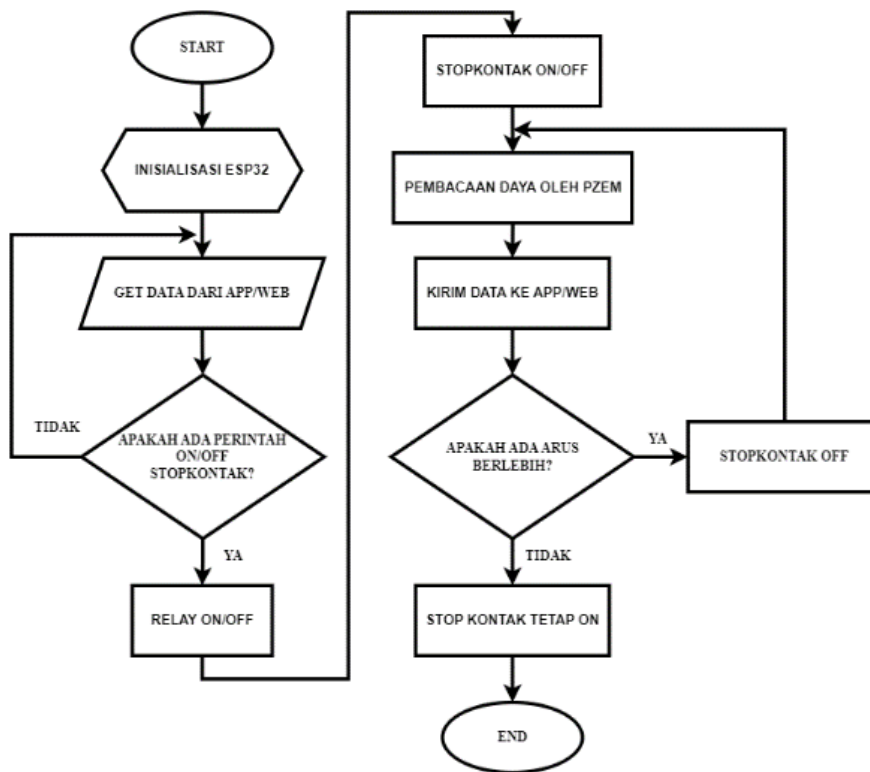


Gambar 1 Diagram Blok

Diagram blok pada gambar 1 merupakan gambaran sederhana dari sistem yang akan dibuat, terdiri atas beberapa bagian berupa blok input, blok proses dan juga blok output.

3.2 Prinsip Kerja

Alat ini merupakan sebuah stop kontak yang dapat dikontrol dan dipantau dari jarak jauh melalui perangkat desktop maupun smartphone. Instruksi yang dapat diberikan berupa kontrol on/off, timer, penjadwalan serta kontrol dengan perintah suara dan untuk monitoring sendiri data yang akan ditampilkan berupa pembacaan daya beban oleh sensor PZEM 004T di tiap-tiap kontak. Data historis pemakaian daya tersebut dapat didownload pada bagian dashboard dari Arduino Iot Cloud sehingga akan memudahkan pengguna dalam melakukan analisis dari penggunaan daya peralatan elektronik yang digunakan. Perancangan Flowchart akan ditunjukkan oleh Gambar 2:



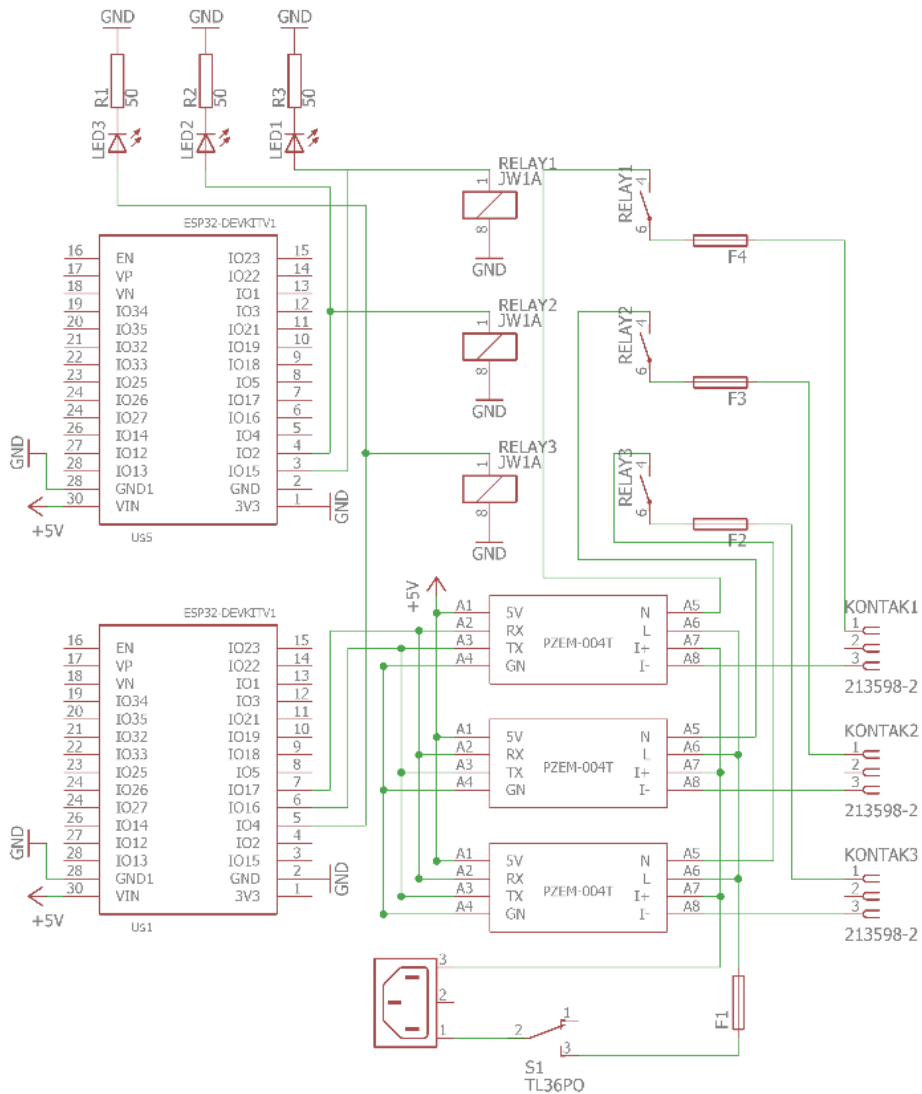
Gambar 2 Flowchart Sistem

Pada awal mula stop kontak dinyalakan maka sistem akan melakukan inialisasi yang dimana ESP32 akan menghubungkan koneksi ke WiFi yang kemudian akan terhubung ke database dari layanan hosting Arduino IoT Cloud. Instruksi dapat diberikan untuk menyalakan ataupun mematikan stop kontak dengan menggunakan perangkat smartphone ataupun melalui website dari Arduino IoT Cloud itu sendiri yang diakses melalui perangkat desktop. Terdapat fitur on/off dengan menggunakan kontrol tombol manual pada app/web, fitur pewaktu yang mencakup timer dan penjadwalan, serta kontrol dengan menggunakan perintah suara (voice command) dengan alexa. Ketika ada instruksi yang diberikan oleh app/web selanjutnya akan diteruskan menuju ke ESP32 kemudian akan dikirimkan sinyal menuju ke modul relay untuk mengaktifkan ataupun mematikan stop kontak. Ketika terdapat beban yang terhubung ke stop kontak maka akan dideteksi oleh sensor PZEM lalu dilakukannya pembacaan daya beban dan akan ditampilkan pada tampilan interface (dashboard) dari app/web yang akan disajikan dalam bentuk pembacaan real-time dan grafik historis. Jika daya yang dibaca terdeteksi lebih dari 660 Watt maka sistem akan mematikan kontak secara otomatis agar rangkaian komponen terhindar

dari kerusakan dan juga untuk proteksi jika terjadi hubung singkat arus listrik. Walaupun diberikan perintah untuk menyalakan kontak, aliran dari kontak tersebut akan terus terputus hingga ditekan tombol reset pada dashboard.

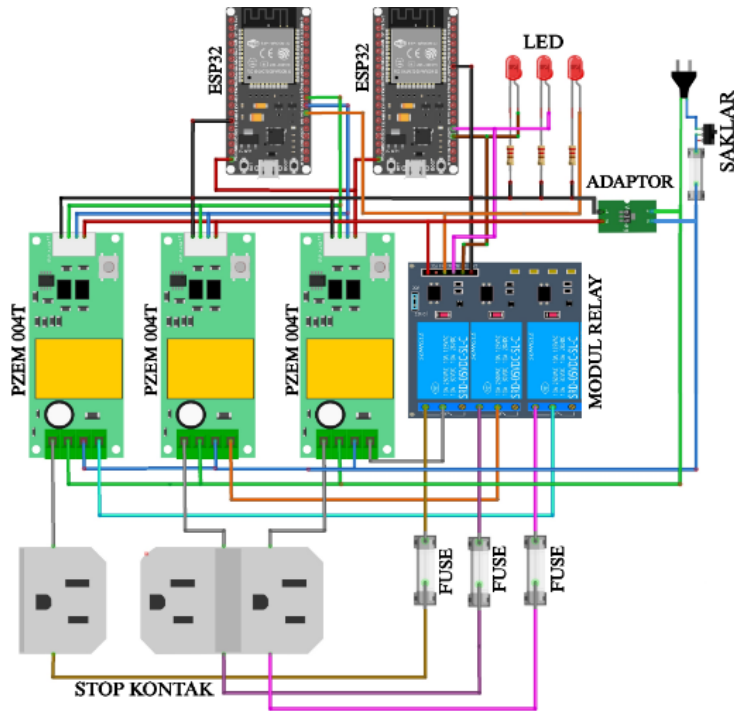
3.3 Perancangan Rangkaian Kelistrikan dan Pengawatan Sistem

Merupakan rancangan dari konektivitas antar komponen yang digunakan. Terdiri atas dua buah ESP 32, tiga buah sensor PZEM 004T, satu buah modul relay 3 channel, tiga buah resistor dan led sebagai indikator, adaptor DC 5V/2A serta tiga buah kontak. Perancangan rangkaian kelistrikan akan ditunjukkan oleh Gambar 3:



Gambar 3 Rangkaian Kelistrikan Sistem

Setelah dirancangnya rangkaian kelistrikan sistem maka selanjutnya adalah perancangan pengawatan sistem dari setiap komponen yang ada secara keseluruhan berdasarkan dari rancangan kelistrikan yang telah dibuat sebelumnya. Berikut adalah rancangan pengawatan sistem secara keseluruhan yang ditunjukkan oleh Gambar 4.



Gambar 4 Rangkaian Pengawatan Keseluruhan Sistem

3.4 Perancangan Perangkat Lunak

a. Penentuan Variabel

Dikarenakan keterbatasan pada layanan versi gratis dari Arduino IoT Cloud maka digunakanlah dua buah ESP32 dalam mengatasi permasalahan tersebut. Jumlah variabel yang dapat digunakan pada versi gratis hanyalah 5 buah variabel dan 2 buah things. Dapat disimpulkan bahwa pada layanan gratis tiap tiap mikrokontroler hanya diberikan batas 5 buah variable dan maksimal mikrokontroler yang dapat digunakan adalah 2 buah. Dengan demikian maka total variabel yang dapat digunakan adalah sebanyak 10 variabel. Berikut merupakan variabel-variabel yang digunakan pada Proyek Tugas Akhir ini, baik itu variabel untuk konektivitas ESP32 dan Arduino IoT Cloud maupun variabel spesifikasi pada alat. Untuk variabel konektivitas ESP32 dan Arduino IoT Cloud, variabel tersebut dibuat sebagai elemen yang digunakan untuk menyimpan dan memperbaharui data sehingga memungkinkan perangkat ESP32 berkomunikasi dengan cloud, baik untuk mengirim data ke cloud atau menerima instruksi dari cloud. Untuk variabel spesifikasi pada alat merujuk kepada penentuan spesifikasi komponen yang akan digunakan pada perangkat sesuai dengan kebutuhan dan juga kemampuan komponen.

Tabel 1 Penentuan Variabel Konektivitas

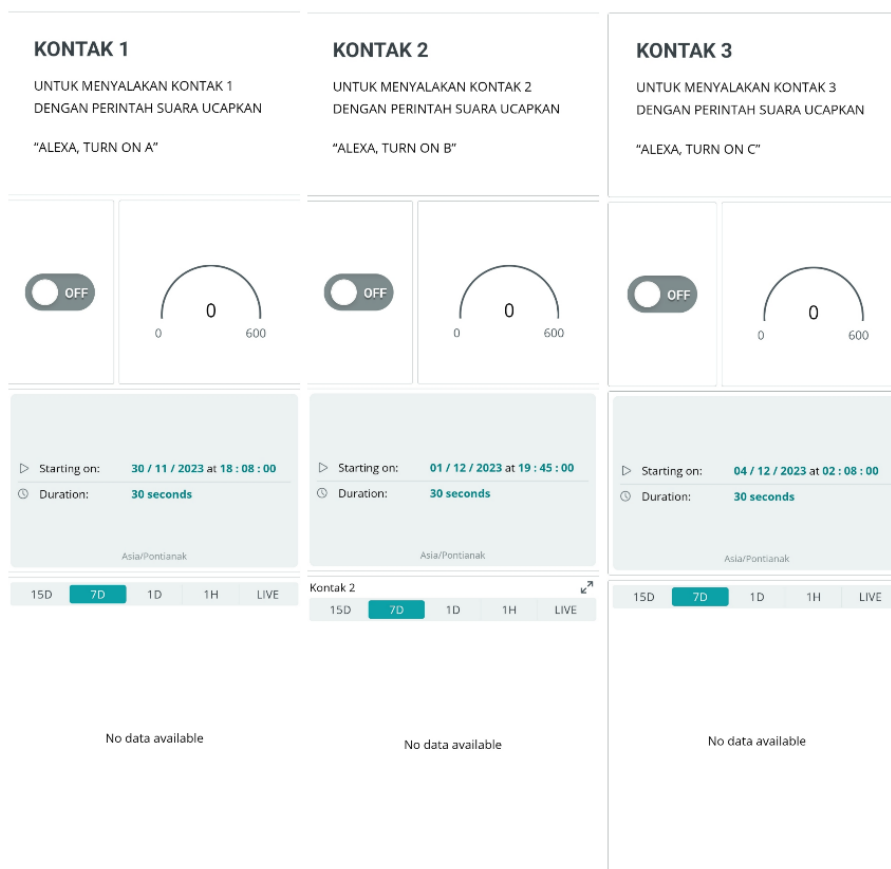
| No. | Nama Variabel | Keterangan |
|-----|---------------|---|
| 1. | A | On/Off/Voice Command Kontak 1 - ESP32 (1) |
| 2. | B | On/Off/Voice Command Kontak 1 - ESP32 (1) |
| 3. | C | On/Off/Voice Command Kontak 1 - ESP32 (2) |
| 4. | D | Kontrol Pewaktu (Timer, Penjadwalan) kontak 1 ESP32 (1) |
| 5. | E | Kontrol Pewaktu (Timer, Penjadwalan) kontak 2 ESP32 (1) |
| 6. | F | Kontrol Pewaktu (Timer, Penjadwalan) kontak 3 ESP32 (2) |

| | | |
|-----|--------|--------------------------------------|
| 7. | G | Reset Pemutus Arus Berlebih (1) |
| 8. | power1 | Variabel pembacaan daya kontak 1 (2) |
| 9. | power2 | Variabel pembacaan daya kontak 2 (2) |
| 10. | power3 | Variabel pembacaan daya kontak 3 (2) |

Dapat dilihat pada Tabel 3, untuk variabel a, b, d, e, g (5 buah variabel) merupakan variabel yang dikhususkan untk ESP32 pertama (1) dan untuk variabel c, f, Power1, Power2, Power 3 (5 buah variabel) merupakan variabel yang dikhususkan untuk ESP32 kedua (2). Selain itu, dirancang pula variabel lainnya berupa nilai-nilai yang berpengaruh terhadap kinerja dari alat tersebut, misalnya saja spesifikasi dari alat itu sendiri dan lain sebagainya.

b. Perancangan Dashboard Arduino IoT Cloud

Merupakan tahapan dimana dirancangnya tampilan dashboard ataupun *user interface* dari Arduino IoT sebagai pusat kontrol dan monitoring. Untuk bagian dashboard ini sengaja dirancang terlebih dahulu guna memudahkan dalam proses pengujian tiap komponen dikarenakan didalam proses pembuatan alat ini membutuhkan pengujian satu per satu komponen yang baru kemudian digabungkan menjadi sebuah satu kesatuan sistem utuh. Berikut merupakan hasil rancangan bagian dashboard pada Arduino IoT Cloud untuk tampilan smartphone yang akan ditunjukkan oleh Gambar 5.



Gambar 5 Tampilan Dashboard Versi Smartphone Kontak 1, 2 dan 3

4. HASIL DAN PEMBAHASAN

Setelah perangkat selesai dibuat maka dilanjutkanlah dengan pengujian guna mengetahui kinerjanya secara keseluruhan. Proses pengujian memerlukan beberapa peralatan tambahan berupa alat ukur seperti kWh Meter dengan spesifikasi 1000VA 230V 50Hz,

stopwatch serta kalkulator yang diakses pada perangkat smartphone. Hasil keseluruhan dari perangkat Stop Kontak Cerdas yang dibuat ditunjukkan oleh Gambar 6:



Gambar 6 Rangkaian Sistem Stop Kontak

4.1 Pengujian Respon Instruksi Widget Button untuk On/Off

Pengujian yang dilakukan berupa menghitung waktu respons alat terhadap instruksi yang diberikan melalui dashboard Arduino IoT Cloud yang berupa kontrol On/Off. Perhitungan waktu menggunakan stopwatch yang dimulai ketika menekan tombol widget pada dashboard baik itu pada smartphone maupun desktop. Berikut merupakan data hasil perhitungan pada percobaan yang telah dilakukan, ditunjukkan oleh Tabel 2:

Tabel 2. Data Hasil Pengukuran Responsibilitas Instruksi Widget Button untuk On/Off

| No. | Smartphone | Desktop | No. | Smartphone | Desktop |
|-----|------------|---------|-----|------------|---------|
| 1. | 3.00s | 1.80s | 11. | 2.25s | 1.70s |
| 2. | 3.25s | 2.31s | 12. | 1.37s | 1.74s |
| 3. | 1.61s | 3.18s | 13. | 2.69s | 2.37s |
| 4. | 1.73s | 3.47s | 14. | 1.60s | 2.43s |
| 5. | 2.86s | 2.23s | 15. | 2.23s | 1.82s |
| 6. | 2.20s | 1.87s | 16. | 1.71s | 2.03s |
| 7. | 1.20s | 1.94s | 17. | 1.07s | 3.22s |
| 8. | 3.10s | 3.57s | 18. | 2.53s | 3.45s |
| 9. | 1.00s | 3.49s | 19. | 2.57s | 2.81s |
| 10. | 2.56s | 2.29s | 20. | 2.78s | 2.75s |

| | Smartphone | Desktop |
|------------------|------------|---------|
| Total | 46.01s | 50.47s |
| Rata-rata | 2.30s | 2.52s |

Berdasarkan data pada Tabel 2 diketahui bahwa responsibilitas dari perangkat smartphone dan juga desktop memiliki kecepatan waktu yang relatif sama dengan perbedaan yang tidaklah terpaut jauh. Perbedaan hasil dapat disebabkan oleh fluktuasi kecepatan jaringan dan juga human error saat melakukan perhitungan.

4.2 Instruksi Perintah Suara

Pengujian yang dilakukan berupa percobaan menghidupkan dan mematikan kontak dengan menggunakan perintah suara dari Amazon Alexa. Langkah pengujian yang dilakukan berupa mengucapkan perintah “Alexa, Turn On A” Untuk menyalakan kontak 1, “Alexa, Turn On B” untuk menyalakan kontak 2 dan “Alexa, Turn On C” untuk menyalakan kontak 3 disaat pengguna membuka software Amazon Alexa pada perangkat smartphone. Akan tetapi ketika *command* layanan asisten pintar alexa sudah aktif, pengguna tidak perlu menyebutkan “Alexa”

lagi dan hanya perlu menyebutkan perintah “Turn On (A/B/C)” sesuai dengan keinginan. Untuk pengujiannya, dilakukan dengan ketentuan ketika instruksi sudah terkonfirmasi oleh asisten pintar Alexa barulah perhitungan dengan stopwatch dimulai. Berikut merupakan hasil dari perhitungan pertama yang akan ditunjukkan oleh Tabel 3:

Tabel 3 Data Perhitungan Responsibilitas Instruksi Perintah Suara Pertama

| No. | Waktu Respons | No. | Waktu Respons |
|------------------|---------------|---------------|---------------|
| 1. | 1.12s | 11. | 1.11s |
| 2. | 1.02s | 12. | 1.28s |
| 3. | 1.19s | 13. | 0.71s |
| 4. | 1.21s | 14. | 0.96s |
| 5. | 1.43s | 15. | 0.46s |
| 6. | 0.75s | 16. | 1.62s |
| 7. | 1.48s | 17. | 0.83s |
| 8. | 0.65s | 18. | 1.15s |
| 9. | 1.66s | 19. | 0.51s |
| 10. | 0.91 | 20. | 0.97s |
| Total | | 21.02s | |
| Rata-rata | | 1.05s | |

Berdasarkan data pada Tabel 3 dapat diketahui bahwa respons perangkat terhadap instruksi perintah suara ternyata lebih cepat jika dibandingkan dengan menggunakan widget button. Hal tersebut disimpulkan dengan catatan bahwa perhitungan baru dimulai disaat instruksi terkonfirmasi oleh asisten pintar ketika terdengar suara *feedback* berupa “Okay”.

4.3 Lama Transfer Data oleh Stop Kontak Menuju Dashboard

Pengujian yang dilakukan berupa percobaan untuk mengetahui kecepatan transfer data pembacaan pemakaian daya oleh perangkat. Berikut merupakan data hasil perhitungan pada percobaan yang telah dilakukan dan akan ditunjukkan oleh Tabel 4:

Tabel 4 Lama Transfer Data oleh Stop Kontak Menuju Dashboard

| No. | Waktu Respons | No. | Waktu Respons |
|------------------|---------------|----------------|---------------|
| 1. | 5.93s | 11. | 4.80s |
| 2. | 4.64s | 12. | 4.67s |
| 3. | 5.81s | 13. | 6.00s |
| 4. | 5.71s | 14. | 3.97s |
| 5. | 5.22s | 15. | 6.19s |
| 6. | 5.71s | 16. | 6.56s |
| 7. | 6.06s | 17. | 5.44s |
| 8. | 4.48s | 18. | 6.46s |
| 9. | 4.07s | 19. | 3.94s |
| 10. | 6.86s | 20. | 5.26s |
| Total | | 107.78s | |
| Rata-rata | | 5.39s | |

Percobaan dilakukan dengan cara melihat seberapa lama waktu yang dibutuhkan oleh perangkat untuk mengirimkan data pembacaan daya oleh sensor PZEM 004T hingga ditampilkan ke dashboard Arduino IoT Cloud. Skenarionya adalah ketika pengguna menekan tombol widget button dan relay sudah merespon barulah dihitung dengan menggunakan stopwatch. Berdasarkan data pada Tabel 3 dapat dilihat bahwa perangkat membutuhkan sedikit waktu untuk mengirimkan data pembacaan daya menuju ke dashboard. Hal tersebut disebabkan oleh keterbatasan kecepatan koneksi dan juga dikarenakan data yang dikirim harus melalui cloud terlebih dahulu baru bisa ditampilkan menuju ke dashboard, maka dari itu terdapat sedikit keterlambatan didalam pengiriman data.

4.4 Perbandingan Pembacaan Daya Stop Kontak Cerdas dan kWh Meter

Pada mulanya dilakukan pengukuran pembacaan daya oleh perangkat Stop Kontak Cerdas kemudian dilanjutkan dengan dilakukannya pengukuran pembacaan daya oleh kWh Meter sebagai acuan dan diasumsikan sebagai nilai yang sebenarnya. Perbandingan ini dimaksudkan untuk mengetahui kehandalan dan keakuratan dari sistem yang dibangun dalam melakukan pengukuran daya listrik. Data perbandingan diambil dengan cara mematikan terlebih dahulu seluruh peralatan elektronik di rumah yang terhubung dengan jalur kWh meter, baru kemudian dilakukannya pengukuran dari beban yang ingin diukur satu persatu. Berikut merupakan perbandingan data hasil pengukuran dari Stop Kontak Cerdas dan juga kWh Meter berdasarkan pengujian yang telah dilakukan yang akan ditunjukkan oleh Tabel 5:

Tabel 5. Data Perbandingan Pengukuran Perangkat dengan kWh Meter

| No. | Nama Alat | Pengukuran Stop Kontak Cerdas | Pengukuran kWh Meter | Persentase Error |
|-----|------------------|-------------------------------|----------------------|------------------|
| 1. | Setrika | 427.7W | 423W | 1.11% |
| 2. | Solder | 34.3W | 33W | 3.94% |
| 3. | Rice Cooker | 72.2W | 69W | 4.63% |
| 4. | Lampu | 8.4W | 8W | 5% |
| 5. | Charger Laptop | 35.7W | 31W | 15.16% |
| 6. | Mini Grinder | 35.1W | 31W | 13.23% |
| 7. | Bor | 171,1W | 164W | 4.33% |
| 8. | Grinder | 228W | 224W | 1.79% |
| 9. | Rice Cooker Mini | 51.3W | 50W | 2.6% |
| 10. | Charger Hp | 5.9W | 6W | 1.67% |

Pengukuran yang dilakukan dengan menggunakan Stop Kontak Cerdas dan kWh meter ternyata memiliki sedikit selisih, hal tersebut dapat disebabkan oleh toleransi dan tingkat presisi dari alat ukur yang berbeda-beda serta kemungkinan terjadinya human error disaat melakukan pengukuran.

4.5 Data Pada Datalogger

Datalogger yang digunakan merupakan datalogger *default* dari Arduino IoT Cloud itu sendiri. Data historis hasil pembacaan disimpan pada cloud dan dapat diunduh pada bagian dashboard yang kemudian dapat dipilih data historis variabel mana saja yang ingin diunduh. Data hasil pembacaan disimpan pada variabel power1 untuk kontak1, power2 untuk kontak2 dan power3 untuk kontak3.

| | A | B | C | D | E | F | | A | B | C | D | E | F |
|------|--|---|---|---|---|---|--|---|---|---|---|---|---|
| 3461 | 2024-01-11T15:45:01.474350017Z,74.9000015258789 | | | | | | | 2024-01-11T15:45:01.474350017Z,53.099998474121094 | | | | | |
| 3462 | 2024-01-11T15:45:05.369897109Z,74.9000015258789 | | | | | | | 2024-01-11T15:45:05.369897109Z,53 | | | | | |
| 3463 | 2024-01-11T15:45:09.253916518Z,75 | | | | | | | 2024-01-11T15:45:09.253916518Z,53.099998474121094 | | | | | |
| 3464 | 2024-01-11T15:45:13.150752367Z,75.5999984741211 | | | | | | | 2024-01-11T15:45:13.150752367Z,53.5 | | | | | |
| 3465 | 2024-01-11T15:45:17.034698389Z,75.5999984741211 | | | | | | | 2024-01-11T15:45:17.034698389Z,53.5 | | | | | |
| 3466 | 2024-01-11T15:45:20.934708693Z,75.5 | | | | | | | 2024-01-11T15:45:20.934708693Z,53.5 | | | | | |
| 3467 | 2024-01-11T15:45:24.822188422Z,75.5 | | | | | | | 2024-01-11T15:45:24.822188422Z,53.599998474121094 | | | | | |
| 3468 | 2024-01-11T15:45:28.714747175Z,75.5999984741211 | | | | | | | 2024-01-11T15:45:28.714747175Z,53.599998474121094 | | | | | |
| 3469 | 2024-01-11T15:45:32.602949774Z,75.5999984741211 | | | | | | | 2024-01-11T15:45:32.602949774Z,53.599998474121094 | | | | | |
| 3470 | 2024-01-11T15:45:36.51025231Z,75.69999694824219 | | | | | | | 2024-01-11T15:45:36.51025231Z,53.599998474121094 | | | | | |
| 3471 | 2024-01-11T15:45:40.384251693Z,75.69999694824219 | | | | | | | 2024-01-11T15:45:40.384251693Z,53.599998474121094 | | | | | |
| 3472 | 2024-01-11T15:45:44.274101029Z,75.69999694824219 | | | | | | | 2024-01-11T15:45:44.274101029Z,53.599998474121094 | | | | | |
| 3473 | 2024-01-11T15:45:48.174429715Z,76 | | | | | | | 2024-01-11T15:45:48.174429715Z,53.900001525878906 | | | | | |
| 3474 | 2024-01-11T15:45:52.062776392Z,76.0999984741211 | | | | | | | 2024-01-11T15:45:52.062776392Z,53.900001525878906 | | | | | |
| 3475 | 2024-01-11T15:45:55.954627829Z,76 | | | | | | | 2024-01-11T15:45:55.954627829Z,53.79999923706055 | | | | | |
| 3476 | 2024-01-11T15:45:59.834954857Z,75.5 | | | | | | | 2024-01-11T15:45:59.834954857Z,53.5 | | | | | |
| 3477 | 2024-01-11T15:46:03.744980703Z,75.5 | | | | | | | 2024-01-11T15:46:03.744980703Z,53.5 | | | | | |
| 3478 | 2024-01-11T15:46:07.633016728Z,75.5 | | | | | | | 2024-01-11T15:46:07.633016728Z,53.5 | | | | | |
| 3479 | 2024-01-11T15:46:11.521814912Z,75.5999984741211 | | | | | | | 2024-01-11T15:46:11.521814912Z,53.5 | | | | | |
| 3480 | 2024-01-11T15:46:15.412470706Z,75.5999984741211 | | | | | | | 2024-01-11T15:46:15.412470706Z,53.5 | | | | | |
| 3481 | 2024-01-11T15:46:19.329436172Z,75.5 | | | | | | | 2024-01-11T15:46:19.329436172Z,53.400001525878906 | | | | | |

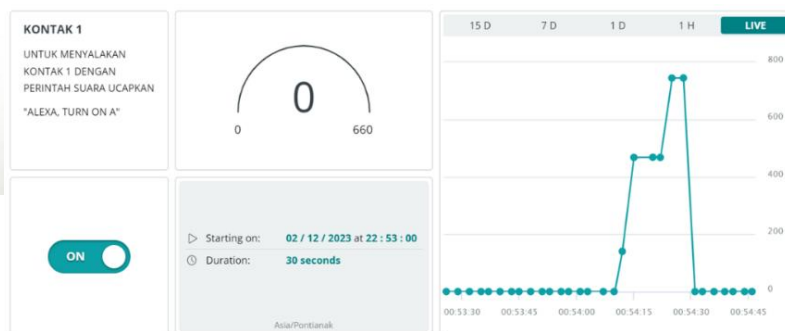
| A | B | C | D | E | F |
|---|---|---|---|---|---|
| 2024-01-11T15:45:01.474350017Z,12.800000190734863 | | | | | |
| 2024-01-11T15:45:05.369897109Z,16.100000381469727 | | | | | |
| 2024-01-11T15:45:09.253916518Z,11.5 | | | | | |
| 2024-01-11T15:45:13.150752367Z,17.5 | | | | | |
| 2024-01-11T15:45:17.034698389Z,17 | | | | | |
| 2024-01-11T15:45:20.934708693Z,18.100000381469727 | | | | | |
| 2024-01-11T15:45:24.822188422Z,15.5 | | | | | |
| 2024-01-11T15:45:28.714747175Z,13.100000381469727 | | | | | |
| 2024-01-11T15:45:32.602949774Z,17.200000762939453 | | | | | |
| 2024-01-11T15:45:36.51025231Z,11.5 | | | | | |
| 2024-01-11T15:45:40.384251693Z,15.800000190734863 | | | | | |
| 2024-01-11T15:45:44.274101029Z,19.299999237060547 | | | | | |
| 2024-01-11T15:45:48.174429715Z,19.399999618530273 | | | | | |
| 2024-01-11T15:45:52.062776392Z,16.100000381469727 | | | | | |
| 2024-01-11T15:45:55.954627829Z,15.300000190734863 | | | | | |
| 2024-01-11T15:45:59.834954857Z,13.899999618530273 | | | | | |
| 2024-01-11T15:46:03.744980703Z,16.100000381469727 | | | | | |
| 2024-01-11T15:46:07.633016728Z,12.100000381469727 | | | | | |
| 2024-01-11T15:46:11.521814912Z,13.699999809265137 | | | | | |
| 2024-01-11T15:46:15.412470706Z,13.5 | | | | | |
| 2024-01-11T15:46:19.329436172Z,15.399999618530273 | | | | | |

Gambar 7 Datalogger Kontak 1, Kontak 2 dan Kontak 3

Berdasarkan data pada Gambar 7, dapat diketahui bahwa interval pengiriman tiap data menuju ke datalogger berkisar antara setiap 3 detik sampai dengan 4 detik. Mencakup waktu dari data tersebut dikirim berupa tahun, bulan, tanggal, jam, menit hingga detik. Terdapat pula Id dari tiap-tiap data tersebut dan tentunya terdapat data hasil pengukuran daya oleh perangkat. Pada bagian waktu ternyata terdapat perbedaan zona waktu yang digunakan, waktu pengujian yang sebenarnya adalah pukul 22.45 WIB dengan zona waktu GMT +7 sedangkan pada sampel, waktu yang ditunjukkan adalah pukul 15.45. Dapat diketahui bahwa zona waktu yang digunakan adalah GMT 0. Data historis dari tiap variabel berada pada file CSV yang terpisah didalam file Rar yang akan dikirimkan ke alamat email pengguna.

4.6 Pengujian Fitur Arus Berlebih

Pada percobaan ini didapati hasil berupa Stop Kontak Cerdas berhasil memutus aliran listrik pada tiap-tiap kontak ketika terdeteksi daya beban melebihi dari 660 Watt. Disaat terdeteksi adanya daya berlebih, sistem tidak serta merta langsung memutus aliran arus listrik pada kontak, akan tetapi terdapat sedikit delay sehingga daya yang akan tampil pada bagian dashboard akan melewati dari batas ambang maksimal yaitu 660 Watt baru kemudian kontak dimatikan.



Gambar 8 Dashboard Kontak 1 Ketika Mendeteksi Adanya Daya Berlebih

Dapat dilihat pada Gambar 8, penggunaan daya mulai naik seiring dengan kontak diaktifkan sehingga peralatan yang terhubung dapat menyala. Ketika terbaca daya beban melebihi dari 660 Watt maka sistem akan merespon dengan mematikan kontak dan ketika aliran listrik pada kontak terputus, maka pembacaan daya akan kembali lagi menuju ke 0. Disaat kontak mati karena sistem mendeteksi adanya daya beban berlebih maka kontak tidak akan dapat diaktifkan walaupun diberikan instruksi untuk menghidupkan kontak. Didalam kondisi tersebut, tombol reset pada dashboard haruslah ditekan agar kontak tersebut dapat dinyalakan kembali.

5.1 Kesimpulan

Setelah melalui serangkaian uji coba dan analisis terhadap Rancang Bangun Sistem Kontrol dan Monitoring Stop Kontak Cerdas Berbasis IoT ini, maka dapat ditarik kesimpulan sebagai berikut:

- Proyek Tugas Akhir ini berhasil menciptakan sistem yang dapat mengontrol peralatan listrik yang terhubung sesuai instruksi yang diberikan oleh pengguna melalui tombol widget, timer, penjadwalan dan perintah suara serta dapat memantau penggunaan daya dari beban.
- Pada saat diberikannya instruksi pada perangkat, terdapat jeda waktu respons (delay) sebelum perangkat memberikan reaksi yang berdasarkan hasil pengujian tercatat waktu tercepat 0.46 detik dan paling lambat 3.57 detik.
- Agar dapat menggunakan lebih dari satu sensor PZEM 004T didalam satu pin RX/TX yang sama pada ESP32, Id dari tiap-tiap sensor haruslah diubah terlebih dahulu agar data yang dikirim oleh sensor tidak konflik.
- Versi gratis dari layanan Arduino IoT Cloud hanya memperbolehkan penggunaan 5 buah variabel dan 2 buah things.
- Data yang ditampilkan pada pengukuran daya oleh sensor PZEM 004T merupakan daya aktif yang mendekati nilai daya sebenarnya berdasarkan dari hasil alat ukur yang digunakan sebagai acuan.
- Perangkat berhasil dan dapat digunakan untuk mencatat data historis penggunaan daya pada setiap kontak (datalogger) serta juga dapat menampilkan penggunaan daya yang dikonsumsi secara real-time.
- Data historis dari pemakaian daya pada Stop Kontak Cerdas dapat didownload pada dashboard dan data tersebut akan dikirimkan ke alamat email yang terdaftar.
- Voice Command* pada perangkat hanya mendukung penggunaan bahasa Inggris

5.2 Saran

Adapun saran untuk pengembangan dari alat ini pada penelitian di waktu yang akan datang agar dapat menjadi lebih baik lagi kedepannya berupa:

- a. Ditambahkan pengukuran untuk tegangan (V), arus (I), energi (kWh) dan juga faktor daya pada tiap kontak.
- b. Menambahkan jumlah dari kontak yang hanya berjumlah 3.
- c. Mengupayakan agar dapat digunakan satu buah mikrokontroler saja.
- d. Membuat agar ukuran dimensi dari alat menjadi lebih kompak dan didesain agar lebih menarik.
- e. Mengoptimalkan datalogger agar data yang ditampilkan lebih mudah untuk dibaca.
- f. Menggunakan komponen dengan spesifikasi yang lebih tinggi agar dapat menangani beban listrik yang lebih besar.
- g. Mengupayakan agar perangkat menjadi lebih responsif dalam menerima instruksi.
- h. Membuat langkah antisipasi agar perangkat tetap dapat beroperasi disaat situasi darurat seperti mati lampu.
- i. Mengupayakan agar dapat digunakannya bahasa Indonesia pada fitur *Voice Command*.

UCAPAN TERIMA KASIH

Penyusunan dapat terlaksana dengan baik berkat dukungan dari banyak pihak baik secara teknis, maupun dukungan moral dan semangat. Untuk itu, diucapkan terimakasih sebesar-besarnya kepada kedua orangtua yang telah memberikan dukungan dan motivasi selama mengerjakan tugas akhir ini, selanjutnya Bapak Dr. H. Widodo P.S., S.T., M.T. selaku Direktur Politeknik Negeri Pontianak, Bapak Hasan, ST., MT selaku Kepala Jurusan Teknik Elektro, Bapak Agus Riyanto, ST., MT selaku Koordinator Prodi TRSE, Bapak Satriyo, ST., M.Kom dan Bapak Eko Mardiyanto, ST., S.ST selaku dosen pembimbing tugas akhir, dan seluruh rekan mahasiswa jurusan Teknik Elektro Politeknik Negeri Pontianak.

DAFTAR PUSTAKA

- [1] Agustianingsih, W. N., Kurniawan, F., & Setiawan, P. (2020). Analisis Ketepatan Pengukur Daya dan Faktor Daya Listrik Berbasis Arduino Uno R3 328P. *Aviation Electronics, Information Technology, Telecommunications, Electricals, Controls*, 3(1), 15-28.
- [2] Anwar, S., Artono, T., Nasrul, N., Dasrul, D., & Fadli, A. (2019). Pengukuran Energi Listrik Berbasis PZEM-004T. *Prosiding Seminar Nasional Politeknik Negeri Lhokseumawe*, 3(1), 272–276.
- [3] Furqon, A., Prasetyo, A. B., & Widiyanto, E. D. (2019). Rancang Bangun Sistem Monitoring dan Kendali Daya Listrik pada Rumah Kos Menggunakan NodeMCU dan Firebase Berbasis Android. *Techné: Jurnal Ilmiah Elektroteknika*, 18(2), 93-104.
- [4] Hanif, A. (2022). Rancangan Sistem Kontrol Dan Monitoring Instalasi Listrik Berbasis Internet of Things (Doctoral dissertation, UIN Ar-Raniry).
- [5] Irfan, M., Suryadi, D., & Saleh, M. Rancang Bangun Sistem Monitoring Baterai serta Besaran Fisis Sepeda Listrik Berbasis IoT. *Jurnal Teknik Elektro Universitas Tanjungpura*, 1(1).
- [6] Kurniawan, I. H., & Hayat, L. (2014). Perancangan Dan Implementasi Alat Ukur Tegangan, Arus Dan Frekuensi Listrik Arus Bolak-Balik Satu Fasa Berbasis Personal Computer. *Techno (Jurnal Fakultas Teknik, Universitas Muhammadiyah Purwokerto)*, 15(1), 21-31.
- [7] Melipurbowo, B. G. (2016). Pengukuran Daya Listrik Real Time Dengan Menggunakan Sensor Arus Acs. 712. *Orbith: Majalah Ilmiah Pengembangan Rekayasa dan Sosial*, 12(1).
- [8] Mordecai, A. R. (2022). Smart Home System Menggunakan Voice Command Alexa Berbasis Nodemcu Dan Smart Security Menggunakan Esp32-Cam Dan Pir Motion Sensor (Doctoral dissertation).

- [9] Muliadi, M., Imran, A., & Rasul, M. (2020). Pengembangan tempat sampah pintar menggunakan ESP32. *Jurnal Media Elektrik*, 17(2), 73-79.
- [10] Munthe, G. H. (2020). RANCANG BANGUN ALAT BANTU PENGELOLAAN Pengerjaan KEGIATAN TERJADWAL DENGAN INDIKATOR LAMPU LED (Doctoral dissertation, Universitas Muhammadiyah Yogyakarta).
- [11] Noor, F. A., Ananta, H., & Sunardiyo, S. (2017). Pengaruh Penambahan Kapasitor Terhadap tegangan, arus, faktor daya, dan daya aktif pada beban listrik di minimarket. *Jurnal Teknik Elektro*, 9(2), 66-73.
- [12] Pramudya, I., Syaifurrahman, & Panjaitan, S. D. (2022). Penerapan Aplikasi Blynk untuk pada Sistem Kontrol dan Monitoring Stop Kontak Cerdas.
- [13] Rifaldi, M. (2022). Penerapan Internet Of Things Pada Prototype Smart Home Menggunakan Pola Suara Dengan Mikrokontroler Nodemcu (Doctoral dissertation, Universitas Islam Riau).
- [14] Samsugi, S. (2017). Internet of Things (iot): Sistem Kendali jarak jauh berbasis Arduino dan Modul wifi Esp8266. *ReTII*.
- [15] Siregar, R., Akbar, S. R., & Maulana, R. (2019). Implementasi Alexa Voice Command Untuk Pembacaan Informasi Sensor Pada Rumah Pintar. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(2), 1804-1813.
- [16] Susanto, F., Prasiani, N. K., & Darmawan, P. (2022). Implementasi Internet of Things Dalam Kehidupan Sehari-Hari. *Jurnal Imagine*, 2(1), 35-40.
- [17] Syafriyudin, & Ledhe, N. T. (2015). Analisis Pertumbuhan Tanaman Krisan Pada Variabel Warna Cahaya Lampu Led. *Jurnal Teknologi*, 8(1), 83–87.