

**SISTEM KONTROL DAN *MONITORING* PERANGKAT  
*SMART HOME* MENGGUNAKAN IoT DENGAN  
MENERAPKAN TOPOLOGI *STAR***

**PROYEK AKHIR**

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Ari Alamsyah            NIM : 0031803

Bulandari                NIM : 0031804

**POLITEKNIK MANUFAKTUR NEGERI  
BANGKA BELITUNG  
TAHUN 2021**

## LEMBAR PENGESAHAN

### JUDUL PROYEK AKHIR

Sistem Kontrol dan Monitoring Perangkat  
*Smart Home* Menggunakan IoT dengan  
Menerapkan Topologi *Star*

Oleh :

Ari Alamsyah / 0031803

Bulandari / 0031804

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



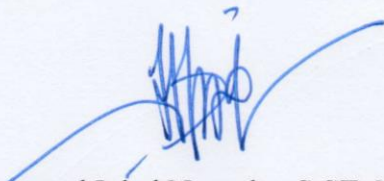
Indra Dwisaputra, M.T.

Pembimbing 2



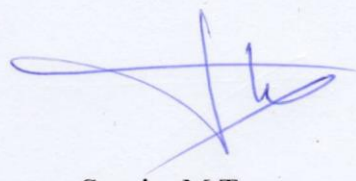
Nofriyani, M.Tr.T.

Penguji 1



Muhammad Iqbal Nugraha, S.ST, M.Eng.

Penguji 2



Surojo, M.T.



## ABSTRAK

*Pesatnya perkembangan teknologi sekarang ini membawa dampak positif salah satunya terciptanya smart home. Sebelumnya telah dibuat alat untuk mengontrol dan memonitoring perangkat smart home yaitu fitting lampu dan stop kontak, tetapi pada perangkat tersebut masih memiliki kekurangan yaitu belum berbasis aplikasi android dan juga belum IoT. Oleh karena itu dibuatlah suatu alat yang sudah kompleks dengan IoT menggunakan topologi star. Penggunaan sistem topologi star untuk memudahkan pengguna jika ingin menggunakan semua perangkat sekaligus tanpa harus mengkoneksikan perangkat satu ke perangkat lainnya. Pengguna hanya perlu memasukkan id dan password untuk mengontrol dan memonitoring perangkat dengan mengakses aplikasi yang telah dibuat melalui MIT App Inventor. Dalam implementasinya menggunakan komunikasi wifi untuk pengiriman data ke server dan modul NodeMCU ESP8266 sebagai web server dan juga sebagai upload data ke server Firebase. Hasil pengujian, bisa dilakukan di dalam rumah atau jarak jauh, namun dibutuhkan delay di setiap pengiriman data ke server firebase. Perangkat dapat terhubung ke aplikasi android secara bersamaan tetapi masih memiliki waktu antrian data 1-2 detik, tergantung koneksi hotspot. Fungsi monitoring lampu memiliki error maksimal sebesar 0,21% untuk pengukuran tegangan, error maksimal sebesar 1,42% untuk pengukuran arus dan error maksimal sebesar 8,18% untuk pengukuran daya dan fungsi monitoring outlet memiliki error maksimal sebesar 6,99% untuk pengukuran tegangan, kesalahan maksimum 5,5% untuk pengukuran arus dan kesalahan maksimum 10,8% untuk pengukuran daya.*

*Kata kunci: Smart Home, IoT, Topologi Star, Firebase, MIT App Inventor*

## **ABSTRACT**

*The rapid development of technology today has a positive impact, one of which is the creation of a smart home. Previously, tools to control and monitor smart home devices have been made, namely light fittings and sockets, but these devices still have drawbacks, namely they are not based on android applications and also not IoT. Therefore, a complex tool with IoT was made using a star topology. The use of a star topology system to make it easier for users if they want to use all devices at once without having to connect one device to another. Users only need to enter an id and password to control and monitor the device by accessing the application that has been created through MIT App Inventor. In its implementation it uses wifi communication for sending data to the server and the NodeMCU ESP8266 module as a web server and also as data uploads to the Firebase server. The test results, can be done at home or remotely, but it takes a delay in every data transmission to the firebase server. Devices can connect to android apps at the same time but still have 1-2 seconds of data queuing time, depending on the hotspot connection. The lamp monitoring function has a maximum error of 0.21% for voltage measurements, a maximum error of 1.42% for current measurements and a maximum error of 8.18% for power measurements and the outlet monitoring function has a maximum error of 6.99% for measurements voltage, maximum error of 5.5% for current measurement and maximum error of 10.8% for power measurement.*

*Keywords: Smart Home, IoT, Star Topology, Firebase, MIT App Inventor*

## KATA PENGANTAR

Puji dan syukur kehadirat Allah SWT yang telah memberi rahmat dan karunia-Nya, sehingga dapat menyelesaikan laporan serta proyek akhir ini dengan baik dan tepat waktu.

Adapun judul proyek akhir ini adalah “Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*”. Tujuan dari pembuatan laporan proyek akhir ini yaitu sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung.

Dalam pelaksanaan pembuatan proyek akhir ini, ingin menyampaikan rasa terimakasih yang sebesar-besarnya kepada:

1. Bapak I Made Andik Setiawan, M.Eng, Ph.D, selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
2. Bapak Muhammad Iqbal Nugraha, S.ST, M.Eng, selaku Ketua Jurusan Teknik Elektro dan Informatika Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Ocsirendi, M.T. selaku Kepala Program Studi Diploma III Teknik Elektronika Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak Indra Dwisaputra, M.T. selaku Dosen Pembimbing I yang telah membantu memberikan arahan dan bimbingan dalam penulisan serta pembuatan alat pada penyelesaian proyek akhir ini.
5. Ibu Nofriyani, M.Tr.T. selaku Dosen Pembimbing II yang telah membantu memberikan arahan dan bimbingan dalam penulisan serta pembuatan alat pada penyelesaian proyek akhir ini.
6. Seluruh Dosen, Instruktur dan Staf pengajar Jurusan Teknik Elektro dan Informatika Politeknik Manufaktur Negeri Bangka Belitung yang telah mengajarkan banyak ilmu pengetahuan.
7. Kepada keluarga yang telah memberikan dukungan kepada penulis berupa dukungan moril maupun materi.

8. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan proyek akhir.

Dalam penulisan laporan proyek akhir ini masih memiliki kekurangan dikarenakan keterbatasan ilmu yang dimiliki. Untuk itu dengan tidak mengurangi rasa hormat, dengan segala kerendahan hati, penulis sangat mengharapkan saran atau kritik yang sifatnya membangun dan bermanfaat untuk kesempurnaan proyek akhir ini.

Selanjutnya penulis berharap semoga proyek akhir ini dapat bermanfaat bagi para pembaca umumnya dan dapat dikembangkan untuk memperlancar dalam melaksanakan tugas yang berkaitan dengan proyek akhir ini.

Sungailiat, 26 Mei 2021

Penulis

## DAFTAR ISI

HALAMAN JUDUL .....	i
LEMBAR PENGESAHAN.....	<b>Error! Bookmark not defined.</b>
PERNYATAAN BUKAN PLAGIAT .....	ii
ABSTRAK .....	iv
<i>ABSTRACT</i> .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI .....	viii
DAFTAR TABEL .....	xi
DAFTAR GAMBAR .....	xii
DAFTAR LAMPIRAN .....	xv
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan Proyek Akhir .....	2
BAB II DASAR TEORI.....	4
2.1 <i>Internet of Thing (IoT)</i> .....	4
2.2 <i>Smart Home</i> .....	5
2.3 Topologi .....	5
2.4 Teori Daya, Arus, dan Tegangan.....	6
2.5 NodeMCU ESP8266.....	8
2.6 Sensor PZEM-0004T .....	9
2.7 Modul <i>Solid State Relay (SSR)</i> .....	10
2.8 <i>Power Supply</i> .....	11
2.9 Android .....	12
2.10 <i>Firebase Realtime Database</i> .....	13
2.11 MIT App Inventor.....	14



2.12 <i>Hotspot</i> .....	14
BAB III METODE PELAKSANAAN .....	16
3.1 Tahapan Perancangan dan Pembuatan Alat .....	16
3.1.1 Pengumpulan data, pengolahan data, dan analisa data.....	17
3.1.2 Perancangan <i>hardware</i> kontrol dan <i>monitoring</i> perangkat <i>smart home</i> .	18
3.1.3 Perakitan <i>hardware</i> elektrik kontrol dan <i>monitoring</i> perangkat <i>smart home</i>	18
3.1.4 Pembuatan program <i>hardware</i> elektrik kontrol dan <i>monitoring</i> perangkat <i>smart home</i> .....	18
3.1.5 Pengujian <i>hardware</i> elektrik kontrol dan <i>monitoring</i> perangkat <i>smart home</i>	18
3.1.6 Pembuatan <i>software</i> kontrol dan <i>monitoring</i> perangkat <i>smart home</i> .....	19
3.1.7 Pengujian fungsi aplikasi kontrol dan <i>monitoring</i> perangkat <i>smart home</i>	19
3.1.8 Pembuatan laporan proyek akhir.....	19
3.2 Rancangan <i>Hardware</i> .....	20
3.2.1 Rangkaian diagram kontrol.....	20
3.2.2 Rangkaian diagram utama .....	20
3.2.3 Sistematis elektrik kontrol dan <i>monitoring</i> perangkat <i>smart home</i> .....	21
3.3 Rancangan <i>Software</i> .....	22
3.3.1 Menu <i>Login</i> Aplikasi.....	23
3.3.2 Menu Perangkat Fiting Lampu Aplikasi .....	23
3.3.3 Menu Perangkat Stop Kontak Aplikasi .....	24
3.3.4 Menu <i>Add</i> .....	25
3.3.5 Menu <i>Edit</i> .....	26
BAB IV PEMBAHASAN .....	28
4.1 Deskripsi alat .....	28
4.2 Diagram blok.....	28
4.3 Prinsip kerja.....	29
4.4 Pengujian kontrol.....	29
4.4.1 Pengujian kontrol fitting lampu .....	30

4.4.2 Pengujian kontrol stop kontak .....	33
4.5 Pengujian Monitoring .....	36
4.5.1 Pengujian <i>monitoring</i> pada fitting lampu .....	37
4.5.2 Pengujian <i>monitoring</i> pada stop kontak .....	38
4.6 Pengujian Sensor Banding dengan Alat Ukur .....	40
4.7 Data <i>Monitoring</i> Perangkat Kombinasi .....	55
4.8 Pengujian Aplikasi .....	57
BAB V KESIMPULAN DAN SARAN .....	61
5.1 Kesimpulan .....	61
5.2 Saran .....	61
DAFTAR PUSTAKA .....	63
LAMPIRAN	

## DAFTAR TABEL

Tabel 2. 1 Spesifikasi NodeMCU ESP8266 [14] .....	9
Tabel 2. 2 Spesifikasi dari Sensor PZEM-004T [15].....	10
Tabel 2.3 Spesifikasi <i>Solid State Relay</i> [11] .....	11
Tabel 4.1 Hasil pengujian fitting lampu 1 .....	31
Tabel 4.2 Hasil pengujian kontrol lampu 2 .....	32
Tabel 4. 3 Hasil pengujian stop kontak 1 .....	34
Tabel 4.4 Hasil pengujian stop kontak 2.....	35
Tabel 4.5 Tabel pengujian stop kontak 3 .....	36
Tabel 4.6 Data pengujian lampu 1 .....	40
Tabel 4. 7 Hasil data daya lampu 1 .....	41
Tabel 4.8 Data pengujian lampu 2.....	43
Tabel 4.9 Hasil data daya lampu 2.....	44
Tabel 4.10 Tabel pengujian stop kontak 1 .....	46
Tabel 4. 11 Hasil data daya stop kontak 1 .....	47
Tabel 4. 12 Data pengujian stop kontak 2.....	49
Tabel 4.13 Hasil data daya stop kontak 2 .....	50
Tabel 4. 14 Data pengujian stop kontak 3.....	52
Tabel 4. 15 Hasil data daya stop kontak 3 .....	53
Tabel 4.16 Data tegangan dengan beban kombinasi.....	56
Tabel 4. 17 Data arus dengan beban kombinasi .....	56
Tabel 4. 18 Data daya dengan beban kombinasi .....	57

## DAFTAR GAMBAR

Gambar 2.1 Sistem <i>Smart Home</i> [8].....	5
Gambar 2.2 Ilustrasi Topologi <i>Star</i> [3].....	6
Gambar 2.3 NodeMCU ESP8266 [14] .....	9
Gambar 2. 4 Sensor PZEM-004T [15].....	10
Gambar 2.5 Modul <i>Solid State Relay</i> (SSR) [11].....	11
Gambar 2.6 <i>Power Supply</i> [17] .....	12
Gambar 2.7 Versi Android [18].....	13
Gambar 2.8 Logo <i>Firebase</i> [20].....	14
Gambar 2.9 Logo MIT App Inventor [21] .....	14
Gambar 2.10 Logo <i>Hotspot</i> [23].....	15
Gambar 3.1 <i>Flowchart</i> Sistem Kontrol dan <i>Monitoring</i> Perangkat <i>Smart Home</i> Menggunakan IoT dengan Menerapkan Topologi <i>Star</i> .....	17
Gambar 3.2 Rangkaian diagram kontrol .....	20
Gambar 3.3 Rangkaian diagram utama lampu .....	20
Gambar 3.4 Rangkaian diagram utama stop kontak .....	21
Gambar 3.5 Sistematik elektrik keseluruhan kontrol dan <i>monitoring</i> perangkat <i>smart home</i> (lampu) .....	21
Gambar 3.6 Sistematik elektrik keseluruhan kontrol dan <i>monitoring</i> perangkat <i>smart home</i> (stop kontak).....	22
Gambar 3.7 Tampilan menu utama aplikasi.....	22
Gambar 3.8 Menu login pada aplikasi .....	23
Gambar 3.9 Tampilan menu fitting lampu .....	24
Gambar 3.10 Menu Stop Kontak Aplikasi .....	25
Gambar 3.11 Menu <i>add</i> aplikasi.....	26
Gambar 3.12 Menu Edit Aplikasi.....	26
Gambar 4.1 Diagram blok sistem kontrol dan <i>monitoring</i> perangkat <i>smart home</i> .....	29
Gambar 4.2 Sistematik pemasangan pin SSR pada fitting lampu .....	30

Gambar 4.3 (a) Kondisi Lampu 1 Menyala (b) Kondisi Lampu 1 Padam.....	30
Gambar 4.4 (a) Kondisi Lampu 2 Menyala (b) Kondisi Lampu 2 Padam.....	31
Gambar 4.5 Sistematis pemasangan pin SSR dan <i>NODEMCU</i> pada stop kontak	33
Gambar 4. 6 (a) Kondisi Stop Kontak 1 Menyala (b) Kondisi Stop Kontak 1 Padam.....	33
Gambar 4.7 (a) Kondisi Stop Kontak 2 Menyala (b) Kondisi Stop Kontak 2 Padam.....	34
Gambar 4. 8 Kondisi Stop Kontak 3 Menyala (b) Kondisi Stop Kontak 3 Padam	35
Gambar 4.9 Sistematis pemasangan pin PZEM dan NodeMCU pada fitting lampu .....	37
Gambar 4.10 Sistematis pemasangan pin PZEM dan NodeMCU pada stop kontak .....	38
Gambar 4.11 Perbandingan nilai tegangan lampu 1 .....	42
Gambar 4.12 Perbandingan nilai arus lampu 1 .....	42
Gambar 4.13 Perbandingan nilai daya lampu 1.....	43
Gambar 4.14 Perbandingan nilai tegangan lampu 2 .....	45
Gambar 4.15 Perbandingan nilai arus lampu 2 .....	45
Gambar 4.16 Perbandingan nilai daya lampu 2.....	46
Gambar 4.17 Perbandingan nilai tegangan stop kontak 1.....	48
Gambar 4.18 Perbandingan nilai arus stop kontak 1 .....	48
Gambar 4. 19 Perbandingan nilai daya stop kontak 1 .....	49
Gambar 4.20 Perbandingan nilai tegangan stop kontak 2.....	51
Gambar 4. 21 Perbandingan nilai arus stop kontak 2 .....	51
Gambar 4. 22 Perbandingan nilai daya stop kontak 2 .....	52
Gambar 4.23 Perbandingan nilai tegangan stop kontak 3.....	54
Gambar 4.24 Perbandingan nilai arus stop kontak 3 .....	54
Gambar 4.25 Perbandingan nilai daya stop kontak 3 .....	55
Gambar 4.26 Pilihan <i>logout</i> .....	58
Gambar 4.27 Tampilan notifikasi aplikasi .....	59
Gambar 4.28 (a) Data responden lampu <i>off</i> (b) Data responden lampu <i>on</i> .....	59

Gambar 4. 29 (a) Data responden stop kontak *off* (b) Data responden stop kontak  
*on* ..... 60

## **DAFTAR LAMPIRAN**

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Program NodeMCU

Lampiran 3 : Program Aplikasi

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi di bidang telekomunikasi dan jaringan modern saat ini tidak terlepas bahwa internet sangat dibutuhkan oleh kalangan masyarakat. Pesatnya perkembangan ini juga menghasilkan dampak positif bagi manusia. Salah satu dampak positif tersebut adalah terciptanya *smart home*. *Smart home* adalah aplikasi khusus untuk lingkungan rumah dengan fungsi meningkatkan kenyamanan dan keamanan penghuninya. Sistem *smart home* biasanya terdiri dari perangkat kontrol, pemantauan, dan pengoptimalan atau peralatan rumah tangga yang dapat diakses melalui perangkat elektronik seperti ponsel dan komputer [1].

Sri Astuti dan Muhammad Taufik [2], menggunakan sistem *website* pada IoT yang dibangun. Hal ini kurang efektif dikarenakan penggunaannya harus mengakses melalui *browser* terlebih dahulu dan tidak dapat diunduh. Yogie dan Habi [3], pada alat yang dibangun menggunakan sistem berbasis android untuk *monitoring* perangkatnya. Dengan ini memudahkan pemilik rumah untuk mengontrol dan *monitoring* daya, biaya pemakaian dan mampu beroperasi dengan kontrol *timer* melalui jaringan *wireless (wifi)*.

Ayu dan Deswanto [4], menggunakan *hotspot* pada alat yang dibangun untuk *monitoring* perangkat. Hal ini membuat alat yang dibangun masih terkendala oleh jarak. Oleh karena itu, munculah sebuah inovasi menggabungkan proyek tersebut yang dapat dikendalikan jarak jauh melalui internet dan kapanpun agar lebih efisien. Saat ini kalangan masyarakat tidak lepas dari *smartphone*. Tipe *smartphone* yang banyak digunakan masyarakat adalah *smartphone* dengan sistem operasi android. Pada sistem android berbasis aplikasi ini memiliki keuntungan yaitu lebih cepat dan praktis dibandingkan *website*. Sistem ini sangat cocok untuk para penghuni rumah apabila sedang berpergian jauh tanpa mengawatirkan perangkat-perangkat rumah dalam keadaan menyala atau mati. Inovasi tersebut



adalah “Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*”.

## **1.2 Rumusan Masalah**

Dari latar belakang tersebut, adapun rumusan masalah yang akan dikaji dalam laporan proyek akhir ini adalah :

1. Bagaimana cara mengkoneksikan *smartphone* dengan perangkat-perangkat *smart home*.
2. Bagaimana cara mengirim data-data pada perangkat *smart home* ke dalam *smartphone*.
3. Bagaimana cara membuat aplikasi pada *smartphone* yang bisa terhubung ke perangkat *smart home*.
4. Bagaimana cara mengintegrasikan fitting lampu dan stop kontak agar bisa bekerja saat diberi perintah.

## **1.3 Batasan Masalah**

Untuk menghindari ruang lingkup permasalahan yang bersifat cukup kompleks, maka dalam pembuatan proyek akhir ini dibuat beberapa batasan masalah, yaitu :

1. Aplikasi hanya dapat digunakan pada *smartphone*.
2. Setiap perangkat yang dibuat terdiri dari 2 fitting lampu dan 3 stop kontak.
3. Sistem *smart home* ini hanya bisa digunakan ditempat yang memiliki jaringan internet stabil.

## **1.4 Tujuan Proyek Akhir**

Adapun tujuan dalam penulisan laporan proyek akhir yang berjudul “Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*” adalah sebagai berikut :

1. Merancang dan membuat sistem kontrol *hardware* pada fitting lampu dan stop kontak.
2. Merancang dan membuat sistem kontrol *software* pada fitting lampu dan stop kontak yang dapat di monitor melalui *smartphone* dengan menerapkan sistem IoT.

3. Membuat aplikasi pada *smartphone* untuk mengontrol dan *monitoring* perangkat *smart home*.

## **BAB II**

### **DASAR TEORI**

#### **2.1 *Internet of Thing (IoT)***

*Internet of Thing (IoT)*, suatu sebutan yang belum lama ini mulai ramai diperbincangkan, tetapi belum terdapat definisi yang baku dari IoT itu sendiri. Burange serta Misalkar [5] melaporkan *Internet of Thing* merupakan struktur dimana objek, orang disediakan dengan bukti diri eksklusif serta keahlian buat pindah informasi lewat jaringan tanpa membutuhkan 2 arah antara manusia ke manusia ialah sumber ke tujuan ataupun interaksi manusia ke PC.

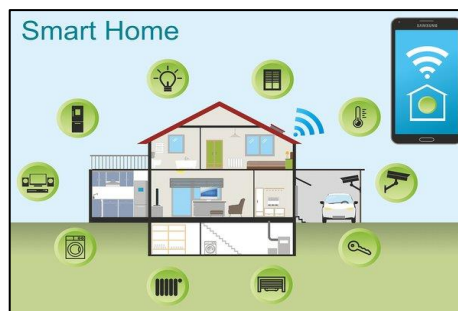
Pada tahun 1999 Kevin Ashton pertama kali mencetuskan ide *Internet of Thing*. Dalam artikel RFID Journal, “*That ‘Internet of thing’ Things*”, Kevin Asthon [6] menyatakan: “Hampir semua dari sekitar 50 petabyte (1.024 terabyte) dari data yang tersedia di internet diciptakan oleh manusia untuk memahami, rekaman, memotret, atau membaca *barcode*. Namun disebabkan oleh keterbatasan waktu dan informasi dari didunia nyata tidak bisa diperoleh secara langsung. Mengenai ini diakibatkan teknologi informasi disaat ini sangat bergantung dari data yang berasal dari manusia. Sehingga kita memerlukan PC yang diketahui dapat mencerna informasi tanpa dorongan manusia, supaya dapat melacak serta menghitung suatu perihal yang butuh ditukar, diperbaiki, diingatkan, serta apakah mereka masih baru ataupun telah usang.

*Internet of Thing* mempunyai kemampuan untuk mengganti dunia. Lebih dari itu, *Internet of Thing* memakai tata cara nirkabel ataupun pengendalian secara otomatis tanpa memahami jarak [7]. Pengimplementasian *Internet of Thing* sendiri umumnya disesuaikan dengan kemauan pengembang dalam pengembangan aplikasi yan diciptakannya. Oleh sebab itu *Internet of Thing* bisa membagikan inovasi kemudahan dalam pengontrolan serta monitoring dari jarak jauh dengan konektifitas internet yang baik.

## 2.2 *Smart Home*

*Smart home* merupakan sebuah aplikasi yang terprogram melalui komputer yang bisa memberikan kenyamanan, keamanan, serta penghematan energi secara otomatis sesuai dengan kendali pengguna [1]. Pada referensi yang berjudul “Implementasi Aplikasi Rumah Pintar Berbasis Android Dengan Arduino *Microcontroller*” sistem ini dibuat dengan tujuan mempermudah penggunaanya dalam mnendalikan perangkat elektronik yang terdapat didalam rumah tersebut [5].

*Smart home* sebagai rumah pintar menjadi “cerdas” karena memiliki kemampuan untuk jarak memantau berbagai peralatan yang membantu manusia mengontrol berbagai aspek kehidupan sehari-hari [7]. Untuk gambar *sart home* ditunjukkan pada gambar 2.1 berikut.



Gambar 2.1 Sistem *smart home* [8].

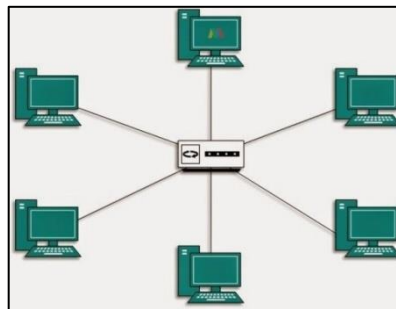
## 2.3 Topologi

Topologi merupakan sebuah sistem jaringan yang membentuk struktur jaringan yang bisa menghubungkan perangkat antar jaringan dengan menggunakan kabel ataupun tanpa kabel (*nirkabel*) [9]. Adapun macam–macam topologi jaringan adalah [10].

1. Topologi *Ring*
2. Topologi *Mesh*
3. Topologi *Star*
4. Topologi *Tree*
5. Topologi *Hybrid*

6. Topologi *Linear*
7. Topologi *Peer to Peer*

Untuk topologi yang digunakan dalam proyek ini adalah topologi *star*. Topologi *star* merupakan suatu bentuk hubungan antar perangkat dimana setiap perangkat memiliki jalur komunikasi tersendiri yang terhubung ke perangkat pusat [3]. Berikut ilustrasi topologi *star* ditunjukkan pada gambar 2.2.



Gambar 2.2 Ilustrasi topologi *star* [3].

## 2.4 Teori Daya, Arus, dan Tegangan

Daya (*watt*), arus (*ampere*), dan tegangan (*volt*) adalah 3 hal yang mempunyai hubungan yang saling berkaitan. Contohnya ketika kita menggunakan perangkat elektronik di rumah, maka daya, arus dan tegangan harus terpenuhi.

### 2.4.1 Daya Listrik

Daya listrik didefinisikan sebagai tingkat di mana energi listrik disampaikan dalam sebuah rangkaian listrik. Daya listrik dibagi menjadi tiga, yaitu daya aktif, daya reaktif dan daya semu [11].

### 2.4.2 Daya Aktif

Daya aktif adalah kekuatan dikonsumsi oleh beban dengan simbol  $P$ . Daya aktif dapat dihitung pada persamaan 2.1 [11].

$$P = V \times I \times \cos \varphi \quad \dots\dots\dots (2.1)$$

Keterangan :  $V$  = Tegangan, *volt*

$I$  = Arus, *ampere*

$\varphi$  = Sudut phi, derajat

### 2.4.3 Daya Reaktif

Daya reaktif adalah jumlah daya yang diperlukan untuk pembentukan medan magnet yang tidak hilang oleh beban atau kekuatan yang diserap namun kembali ke sumber dengan simbol Q. Daya reaktif dapat dihitung dengan persamaan 2.2 [11].

$$Q = V \times I \times \sin \varphi \quad \dots\dots\dots (2.2)$$

Keterangan : V = Tegangan, *volt*

I = Arus, *ampere*

$\varphi$  = Sudut phi, derajat

### 2.4.4 Daya Semu

Hasil penjumlahan trigonometri dari daya aktif dan daya reaktif disebut daya semu yang disimbolkan S, dengan VA (*Voltampere*). Daya tampak dapat dihitung menggunakan persamaan 2.3 [11].

$$S = V \times I \quad \dots\dots\dots (2.3)$$

Keterangan : V = Tegangan, *volt*

I = Arus, *ampere*

Perhitungan persentase *error* daya yang didapat merupakan perbandingan pada alat ukur dan alat yang dibuat, sehingga rumusnya adalah:

- Persentase *error* daya =  $\left| \frac{W_{uk} - W_a}{W_{uk}} \right| \times 100\% \quad \dots\dots\dots (2.4)$

Keterangan :  $W_a$  = Daya pada alat yang dibuat (W)

$W_{uk}$  = Daya pada alat ukur (W)

### 2.4.5 Arus Listrik

Arus listrik aliran elektron yang bergerak atau mengalir dalam satuan waktu. Arus listrik (I) yang mengalir melalui konduktor didefinisikan sebagai jumlah muatan listrik (Q) yang mengalir per satuan waktu (t). Secara matematis dapat dituliskan [12].

$$I = Q/t \quad \dots\dots\dots (2.5)$$

I = Arus Listrik (A)

Keterangan : C = Muatan Listrik (Q)

t = Selang waktu

- Persentase *error* arus  $= \left| \frac{I_{uk} - I_a}{I_{uk}} \right| \times 100\%$  ..... (2.6)

Keterangan : I<sub>a</sub> = Arus pada alat yang dibuat (A)

I<sub>uk</sub> = Arus pada alat ukur (A)

### 2.4.6 Tegangan Listrik

Sumber tegangan listrik adalah beda potensial antara dua titik terus menerus. Tegangan listrik dibangkitkan dari pembangkit listrik. Perbedaan potensial listrik diukur dalam *volt* (V). Alat yang digunakan adalah *voltmeter* [12].

$$V = W / Q \quad \text{..... (2.7)}$$

V = beda potensial listrik dalam *volt* (V)

Keterangan : W = energi listrik dalam *joule* (J)

Q = muatan listrik dalam *coulomb* (C).

Perhitungan persentase *error* tegangan yang didapat merupakan perbandingan pada alat ukur dan alat yang dibuat, sehingga rumusnya adalah:

- Persentase *error* tegangan  $= \left| \frac{V_{uk} - V_a}{V_{uk}} \right| \times 100\%$  ..... (2.8)

Keterangan : V<sub>a</sub> = Tegangan pada alat yang dibuat (V)

V<sub>uk</sub> = Tegangan pada alat ukur (V)

### 2.5 NodeMCU ESP8266

NodeMCU adalah platform *Internet of Thing* (IoT) yang bersifat *open source*. NodeMCU terdiri dari perangkat keras dalam bentuk *Sistem On Chip Wi-Fi* ESP8266 dari ESP8266 dibuat oleh *ESPRESSIF SYSTEM*. NodeMCU ESP8266 menggunakan ESP8266 dengan seri *ESP-12E (ESP8266MOD)* dengan dilengkapi *drive CH340G*. Pengembangan penggunaan NodeMCU ESP8266 dapat

menggunakan bahasa pemrograman Lua atau dengan menggunakan *sketch* atau program dengan *software* Arduino IDE [13]. Berikut NodeMCU ESP8266 ditunjukkan pada gambar 2.3.



Gambar 2.3 NodeMCU ESP8266 [14]

Spesifikasi NodeMCU ESP8266 dapat dilihat pada tabel 2.1 berikut.

Tabel 2. 1 Spesifikasi NodeMCU ESP8266 [14]

No.	Spesifikasi	Keterangan
1.	<i>Mikrokontroller</i>	ESP8266
2.	<i>Ukuran Board</i>	57 mm x 30 mm
3.	<i>Tegangan Input</i>	3,3V – 5V
4.	<i>GPIO</i>	13 PIN
5.	<i>Kanal PWM</i>	10 Kanal
6.	<i>10 bit ADC Pin</i>	1 Pin
7.	<i>Flash Memory</i>	4 MB
8.	<i>Clock Speed</i>	40/26/24 MHz
9.	<i>WiFi</i>	IEEE 802.11 b/g/n
10.	<i>Frekuensi</i>	2.4 GHz – 22.5 Ghz
11.	<i>USB Port</i>	<i>Micro USB</i>
12.	<i>Card Reader</i>	Tidak Ada
13.	<i>USB to Serial Converter</i>	CH340G

## 2.6 Sensor PZEM-004T

Sensor PZEM-004T berguna untuk mengukur arus, tegangan dan daya pada listrik AC (*Alternatif Current*). Sensor ini mengeluarkan output dengan komunikasi serial. Pada dasarnya sensor PZEM-004T tidak mampu membaca arus dengan ketelitian *mili ampere* [15]. Berikut sensor PZEM-004T ditunjukkan pada gambar 2.4.





Gambar 2. 4 Sensor PZEM-004T [15]

Spesifikasi dari sensor PZEM-004T dapat dilihat pada Tabel 2.2 berikut.

Tabel 2. 2 Spesifikasi dari Sensor PZEM-004T [15]

No.	Spesifikasi	Keterangan
1.	Tegangan Kerja	80~260VAC
2.	Pengukuran Arus	0-100A
3.	Nilai Daya	22Kw
4.	Frekuensi Operasi	45-65Hz
5.	Akurasi Pengukuran	1.0 <i>grade</i>

## 2.7 Modul *Solid State Relay* (SSR)

Sebuah modul *solid state relay* (SSR) merupakan sakelar elektronik yang menyala atau mati ketika tegangan eksternal diterapkan pada terminal kontrol. *Solid state relay* yang dikemas menggunakan perangkat semikonduktor untuk beralih arus hingga sekitar seratus *ampere* dan memiliki kecepatan *switching* cepat dibandingkan dengan *relay* elektromekanik, serta tidak memiliki kontak fisik aus. Penerapan *relay solid state* harus memperhitungkan kemampuan lebih rendah mereka untuk menahan kelebihan beban sesaat dibandingkan dengan kontak elektromekanis [11]. Modul *solid state relay* berikut ditunjukkan pada gambar 2.5.



Gambar 2.5 Modul *solid state relay* [11]

Berikut spesifikasi dari *solid state relay* dapat dilihat pada tabel 2.3.

Tabel 2.3 Spesifikasi *Solid State Relay* [11]

No.	Spesifikasi	Keterangan
1.	Tegangan <i>Input</i>	5V
2.	Beban Keluaran	100~240V dengan arus 2A
3.	Tegangan Pemicu	0 – 1.5V

## 2.8 *Power Supply*

*Power supply* atau catu daya merupakan rangkaian elektronika yang digunakan sebagai pengubah listrik tegangan AC menjadi listrik tegangan DC sehingga listrik tegangan DC dapat digunakan sebagai sumber energi listrik *hardware* elektronika [16].

*Power supply* terdiri dari beberapa komponen utama yaitu *transformator* atau trafo, *diode*, kapasitor jenis *elco* dan transistor. *Transformator* atau trafo berfungsi sebagai penurun tegangan misalnya tegangan AC yang awalnya sebesar 220V menjadi keluaran tegangan AC sebesar 12V, kemudian keluaran dari trafo masuk kerangkaian *diode bridge* yang berguna sebagai penyearah tegangan AC menjadi tegangan DC. Kapasitor jenis *elco* bertugas sebagai penyaring tegangan *ripple* yang masih bocor, sedangkan transistor berfungsi sebagai penstabil tegangan. *Output* dari rangkaian *power supply* akan menjadi sumber *hardware* elektronika [17]. Berikut *power supply* ditunjukkan pada gambar 2.6.



Gambar 2.6 *Power supply* [17]

## 2.9 Android

Android, Inc. didirikan di Palo Alto, California, pada bulan Oktober 2003 oleh Andy Rubin, Rich Miner, Nick Sears, dan Chris White untuk mengembangkan perangkat ponsel pintar yang lebih sadar lokasi dan preferensi pengguna mereka. Kemudian dilakukan pengembangan Android dengan tujuan mengembangkan sistem operasi canggih yang diperuntukan untuk kamera digital, namun disadari pasar perangkat tersebut tidak cukup besar dan pengembangan Android digeser ke pasar *smartphone* untuk bersaing dengan *Symbian* dan *Windows Phone* [18].

Pada 17 Agustus 2005 Google mengakuisisi Android Inc. sehingga menjadikannya anak perusahaan yang dimiliki sepenuhnya oleh Google. Pendiri Android Inc. seperti Andy Rubin, Rich Miner dan Chris White tetap bersama perusahaan setelah diakuisisi oleh Google. Google mulai mengembangkan *platform* perangkat *mobile* menggunakan kernel Linux ke pasar untuk produsen perangkat *mobile* dan operator nirkabel dengan janji bahwa mereka akan menyediakan sistem yang fleksibel dan terbarukan [18].

Sejak 2008 sampai sekarang Android telah dikembangkan secara bertahap oleh Google dan *Open Handset Alliance* (OHA) yang sebelumnya telah membuat perbaikan untuk meningkatkan kinerja sistem, menambahkan fitur baru dan bug fix yang terkandung dalam versi. Setiap versi utama yang dirilis dinamai menurut abjad dengan nama makanan penutup, dengan pengecualian Android 1.0 dan Android 1.1 [18]. Berikut gambar versi android ditunjukkan pada gambar 2.7.



Gambar 2.7 Versi android [18]

## 2.10 *Firestore Realtime Database*

*Database* yang disimpan pada *cloud* dan mendukung *multi-platform* misalnya *Android*, *iOS* dan *Web* disebut *Firestore realtime database*. Data *Firestore* akan disimpan pada *JSON (Javascript Object Notation)*. Data tersebut akan disinkronkan secara otomatis dengan aplikasi klien yang terhubung. Aplikasi yang menggunakan *SDK Android*, *iOS* dan *JavaScript* akan menerima *update* data terbaru secara otomatis pada saat aplikasi terhubung ke *server firestore* [19]. Ketika terjadi perubahan data, maka aplikasi yang terhubung dengan *Firestore* akan disinkronisaikan secara otomatis melalui perangkat baik *website* ataupun *mobile*. *Framework* lain seperti *node*, *java*, *javascript*, dan lain-lain dapat digabungkan dengan *Firestore* [19].

Fitur yang disediakan oleh *Firestore* yaitu Adapun fitur yang disediakan oleh *firebase* ini antara lain *Firestore Analytics*, *Firestore Authentication*, *Firestore Cloud Messaging and Notification*, *Firestore Crash Reporting*, *Firestore Remote Config* dan *Firestore Real Time Database*. Fitur yang memiliki peran cukup penting adalah *Firestore Real Time Database* dan *Firestore Remote Config*.

Tujuan utama dari *database realtime firestore* akses data time kinerja. *Database realtime* mengoptimalkan waktu akses, sehingga akses data di kisaran mikrodetik dan bahkan nanodetik, sehingga biaya akses data dapat diminimalkan. Beberapa kemampuan *Firestore* yang ditawarkan ke pengguna dan pengembang sistem adalah *Realtime database*, *Responsif*, dapat diakses dari perangkat klien[20]:



Gambar 2.8 Logo *Firebase* [20]

### 2.11 MIT App Inventor

MIT App Inventor adalah alat pengembangan yang fungsinya untuk membuat aplikasi yang dapat di-*install* pada Android. MIT App Inventor ini diciptakan oleh MIT (*Massachusetts Institute of Technology*) dengan tujuan untuk memudahkan pembuatan aplikasi Android. MIT menciptakan alat pengembangan yang mudah digunakan oleh siapa saja dengan menggunakan metode *click* dan *drag* untuk desain tampilannya dan metode penyusunan blok untuk algoritma programnya MIT App Inventor dapat diakses melalui *browser* internet [21]. Berikut logo MIT App Inventor dapat dilihat pada gambar 2.9.



Gambar 2.9 Logo MIT App Inventor [21]

### 2.12 Hotspot

*Hotspot* adalah istilah untuk area di mana orang atau pengguna dapat mengakses jaringan internet, selama mereka menggunakan PC, laptop atau perangkat lainnya dengan fitur yang memiliki Wi-Fi (*Wireless Fidelity*) sehingga mereka dapat mengakses internet tanpa media yang kabel [22]. Berikut kelebihan dari *hotspot* adalah sebagai berikut [23].

1. Bisa dibawa kemana-mana (*portable*).
2. Mudah dilakukan.
3. Hemat biaya pembelian modem.

4. Dapat digunakan kapanpun dan dimanapun selama jaringan penyedia internet dan telepon (*provider*) tersedia.
5. Bisa mengontrol *user* (pengguna) jaringan *hotspot* wifi.



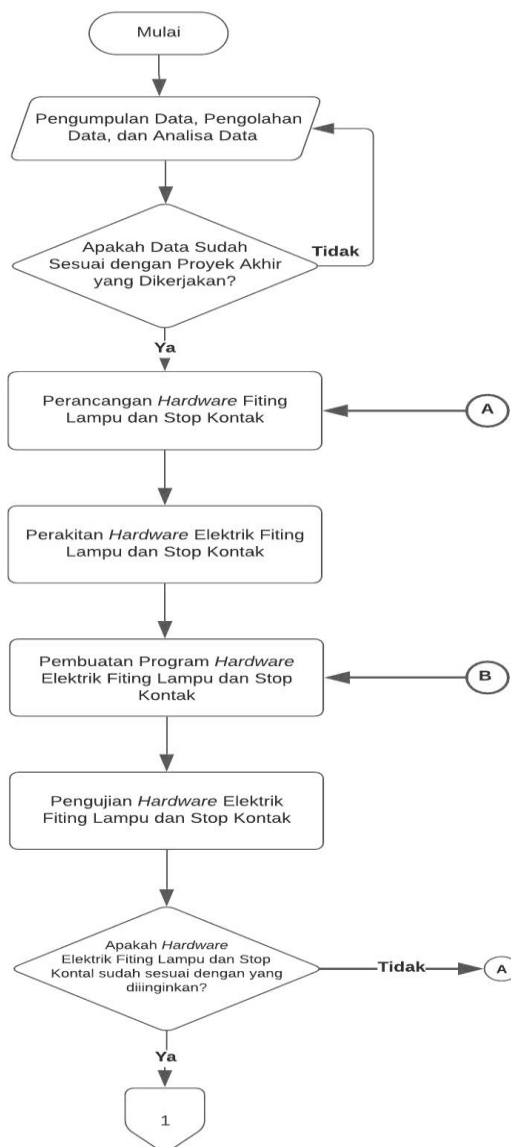
Gambar 2.10 Logo *hotspot* [23]

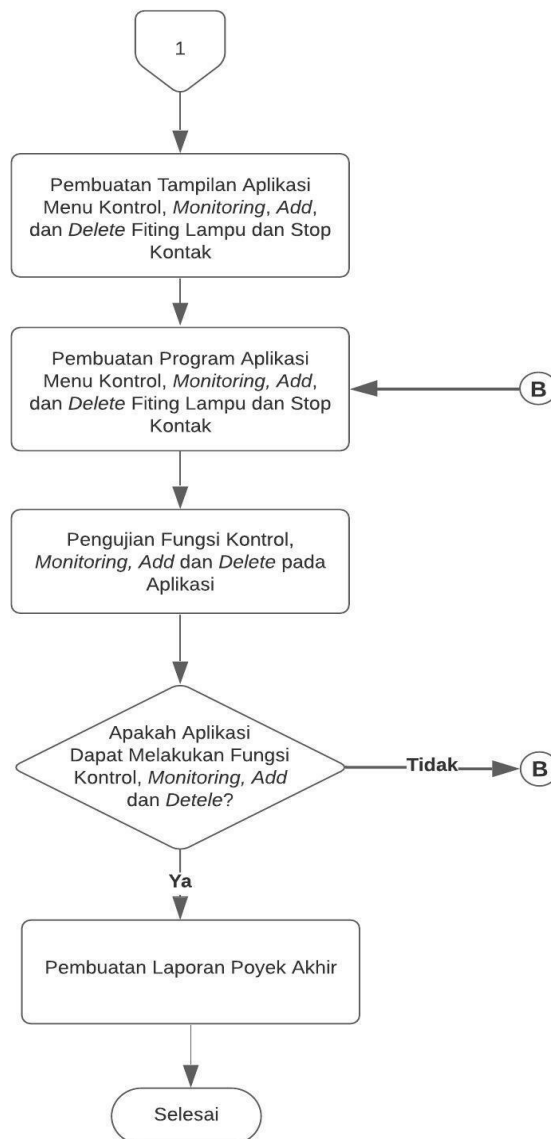
# BAB III

## METODE PELAKSANAAN

### 3.1 Tahapan Perancangan dan Pembuatan Alat

Pada proses pembuatan proyek akhir ini terdiri dari beberapa tahapan yang bertujuan untuk mempermudah dalam penyelesaian proyek akhir. Tahapan tersebut dijelaskan pada *flowchart* yang ditunjukkan pada gambar 3.1 berikut.





Gambar 3.1 *Flowchart* sistem kontrol dan *monitoring* alat

### 3.1.1 Pengumpulan data, pengolahan data, dan analisa data

Pengumpulan data yang diperlukan dalam menyelesaikan proyek akhir. Proses pengumpulan data dapat dilakukan secara langsung dan secara tidak langsung. Pengumpulan data secara langsung merupakan pengumpulan data yang diperoleh dari konsultasi dengan dosen pembimbing sedangkan untuk pengumpulan data secara tidak langsung merupakan data yang diperoleh dari



referensi makalah proyek akhir tahun sebelumnya dan dari referensi–referensi dari penelitian yang masih berhubungan dengan proyek akhir yang ingin dibuat. Data-data yang didapatkan dari proses pengumpulan data selanjutnya diolah dan dianalisa. Data yang telah diolah dan dianalisa dikumpulkan lalu dipilih untuk menjadi referensi dan acuan dalam pembuatan proyek akhir.

### **3.1.2 Perancangan *hardware* kontrol dan *monitoring* perangkat *smart home***

Proses Perancangan *hardware* fitting lampu dan stop kontak dibuat agar *hardware* dan prinsip kerja sesuai dengan spesifikasi proyek akhir. Proses perancangan *hardware* fitting lampu dan stop kontak juga dilakukan untuk menentukan komponen elektrik yang akan digunakan seperti NodeMCU ESP8266, sensor PZEM-004T, dan *Solid State Relay*.

### **3.1.3 Perakitan *hardware* elektrik kontrol dan *monitoring* perangkat *smart home***

Proses perakitan *hardware* fitting lampu dan stop kontak harus sesuai dengan spesifikasi koneksi pin dari *hardware* fitting lampu dan stop kontak yang digunakan. Perakitan dilakukan dengan cara merakit dari setiap bagian dengan menggunakan *box sealpack* 1000ml. Selain itu pemasangan spiral kabel berguna untuk melindungi kabel.

### **3.1.4 Pembuatan program *hardware* elektrik kontrol dan *monitoring* perangkat *smart home***

Pembuatan program *hardware* fitting lampu dan stop kontak dibuat dengan menggunakan *software* Arduino IDE dengan database menggunakan *firebase*. Pembuatan pemrograman pada NodeMCU ESP8266 sebagai pengontrolan sistem.

### **3.1.5 Pengujian *hardware* elektrik kontrol dan *monitoring* perangkat *smart home***

Pengujian *hardware* elektrik fitting lampu dan stop kontak ini bertujuan untuk mengetahui apakah komponen yang digunakan bekerja sesuai dengan fungsi yang diinginkan dan dibutuhkan. Berikut uji coba *hardware* pada komponen elektrik :

- 1) Uji coba NodeMCU ESP8266
- 2) Uji coba Sensor PZEM-004T
- 3) Uji coba *Solid State Relay*

### **3.1.6 Pembuatan *software* kontrol dan monitoring perangkat *smart home***

Pembuatan *software* menggunakan MIT App Inventor yang diakses menggunakan *browser* internet. Pembuatan tampilan menu dilakukan pada bagian *designer* MIT App Inventor. Pembuatan tampilan menu kontrol, *monitoring*, *add* dan *delete* harus mudah dipahami serta digunakan dan sesuai fungsi dari perangkat yang akan dibuat. Pembuatan program menu kontrol, *monitoring*, *add* dan *delete* perangkat *smart home* dilakukan pada bagian *blocks* MIT App Inventor.

### **3.1.7 Pengujian fungsi aplikasi kontrol dan *monitoring* perangkat *smart home***

Pengujian fungsi dilakukan secara keseluruhan, baik dari segi kontrol, *monitoring*, *add* dan *delete* harus dapat bekerja sesuai dengan fungsinya. Berikut pengujian yang dilakukan:

- 1) Pengujian *SSR* apakah dapat berfungsi untuk mengontrol perangkat *smart home* yaitu fitting lampu dan stop kontak.
- 2) Pengujian sensor PZEM-004T dalam membaca tegangan, arus, daya, dan biaya penggunaan fitting lampu dan stop kontak yang keluar pada tampilan *monitoring smartphone*.

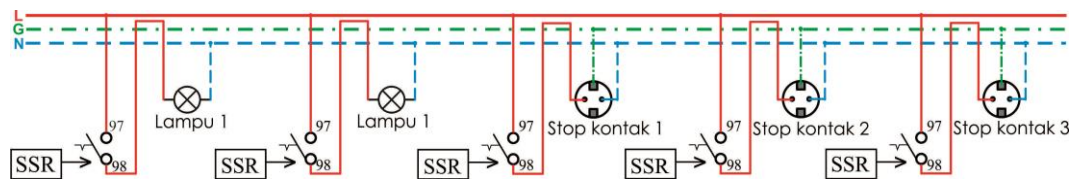
### **3.1.8 Pembuatan laporan proyek akhir**

Proses pembuatan laporan merupakan proses terakhir dalam pembuatan proyek akhir. Pembuatan laporan bertujuan merangkum keseluruhan yang berhubungan dengan proyek akhir, seperti latar belakang, tujuan, rumusan masalah, batasan masalah, teori dasar, metode pelaksanaan, pembahasan serta kesimpulan dan saran.

### 3.2 Rancangan *Hardware*

Dalam proses pembuatan kontrol dan *monitoring* perangkat *smart home* terdapat rangkaian diagram kontrol, rangkaian diagram utama, dan sistematik elektrik untuk memenuhi spesifikasi pembuatan alat yang diinginkan.

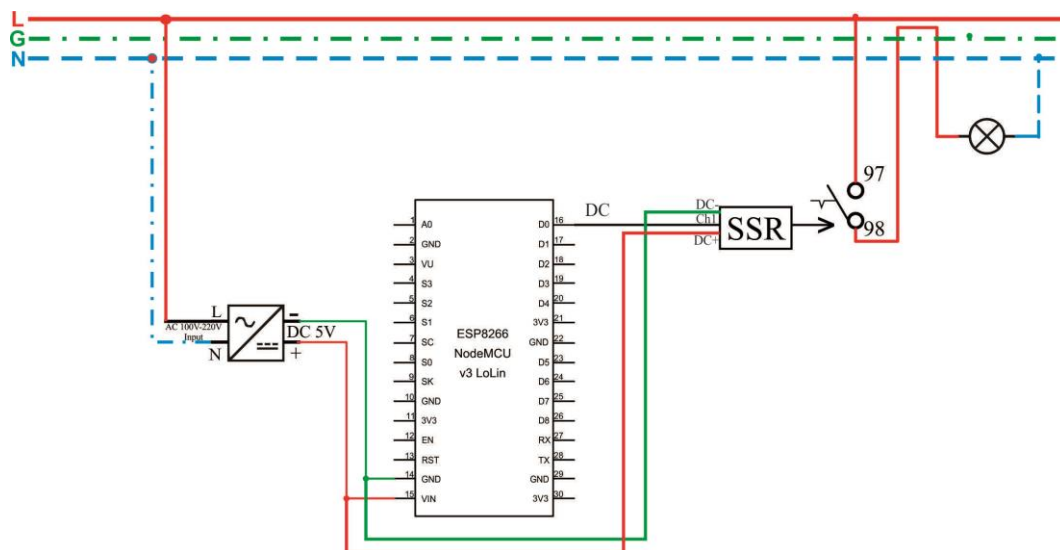
#### 3.2.1 Rangkaian diagram kontrol



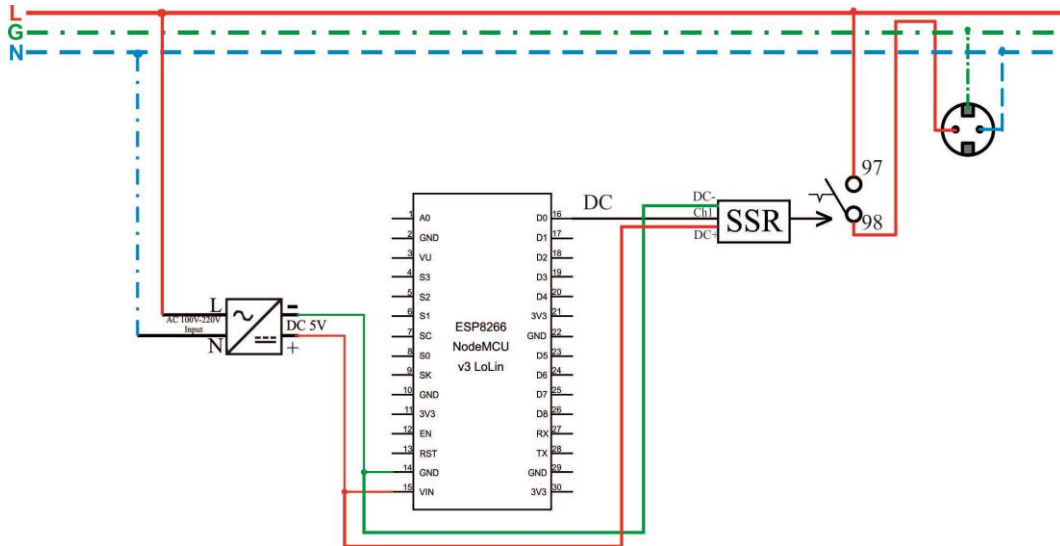
Gambar 3.2 Rangkaian diagram kontrol

Pada rangkaian diagram kontrol fitting lampu dan stop kontak SSR berguna sebagai saklar untuk mematikan dan menghidupkan perangkat.

#### 3.2.2 Rangkaian diagram utama



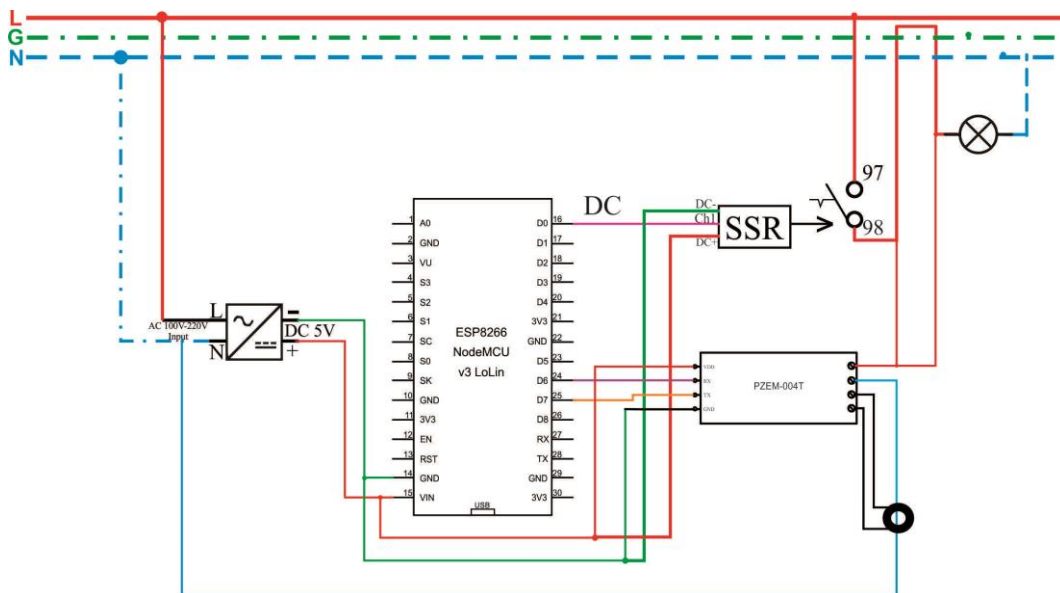
Gambar 3.3 Rangkaian diagram utama lampu



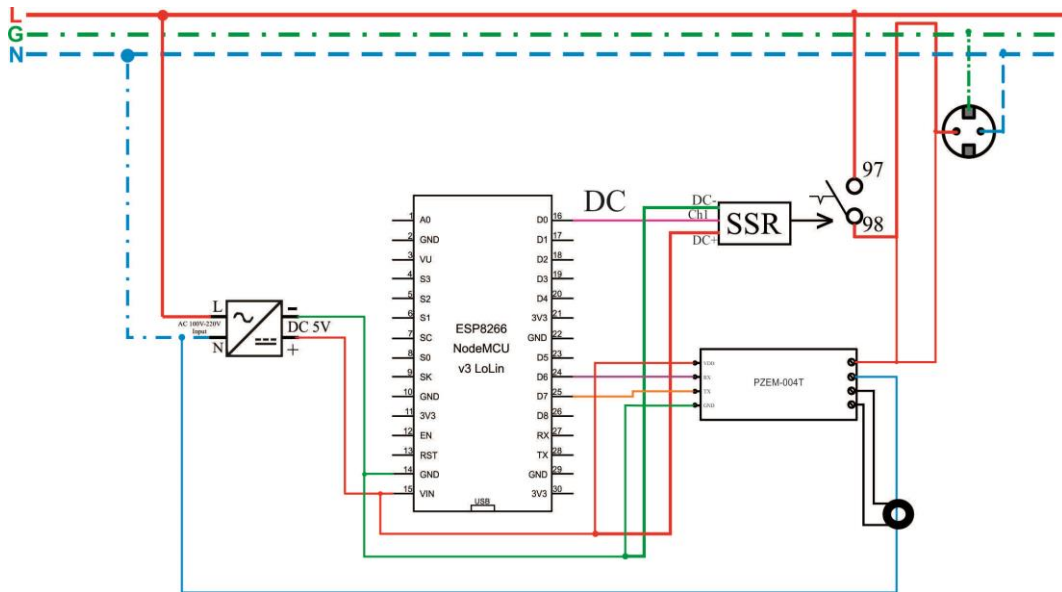
Gambar 3.4 Rangkaian diagram utama stop kontak

Pada rangkaian diagram utama terdapat *power supply* sebagai pengubah tegangan AC ke DC. Keluaran *power supply* akan masuk ke pin NodeMCU, sehingga NodeMCU aktif untuk men-trigger *solid state relay* untuk mengontrol perangkat.

### 3.2.3 Sistematis elektrik kontrol dan *monitoring* perangkat *smart home*



Gambar 3.5 Sistematis elektrik keseluruhan kontrol dan *monitoring* perangkat *smart home* (lampu)

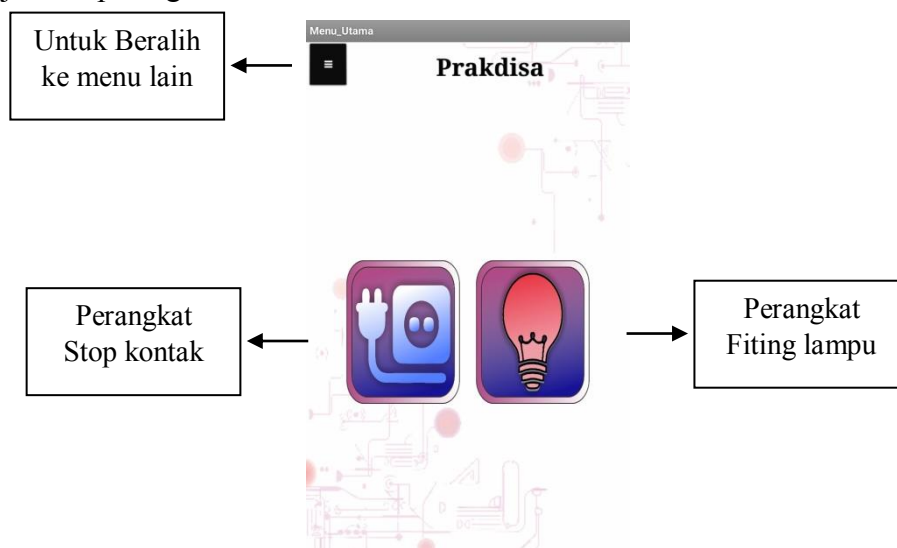


Gambar 3.6 Sistematis elektrik keseluruhan kontrol dan *monitoring* perangkat *smart home* (stop kontak)

Pada rangkaian keseluruhan ini, terdapat sensor PZEM-004T yang berfungsi sebagai pembaca tegangan, arus, dan daya pada rangkaian.

### 3.3 Rancangan *Software*

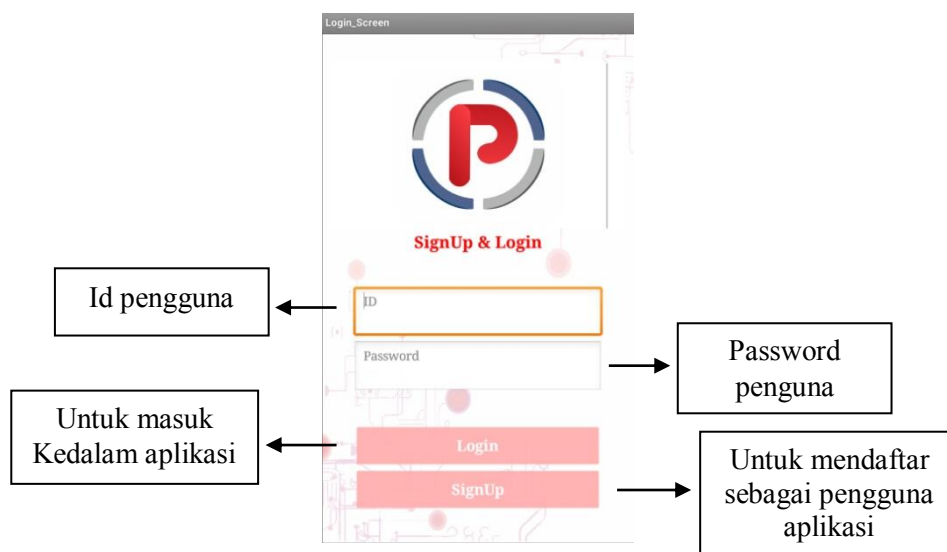
Aplikasi ini dirancang menggunakan aplikasi MIT App Inventor. Aplikasi yang telah dibuat dapat di *install* melalui android. Berikut gambar pada aplikasi ditunjukkan pada gambar 3.7.



Gambar 3.7 Tampilan menu utama aplikasi

### 3.3.1 Menu *Login* Aplikasi

Menu *login* ini digunakan untuk masuk kedalam sebuah aplikasi. Menu *login* ini dapat dikatakan sebagai sistem keamanan aplikasi bagi pengguna untuk dapat mengakses aplikasi tersebut. *Login* yang dimaksudkan untuk mengatur proses identifikasi. Proses *login* membutuhkan hal spesifik tertentu, yaitu *username* dan *password*. Berikut tampilan menu *login* pada aplikasi ditunjukkan pada gambar 3.8.

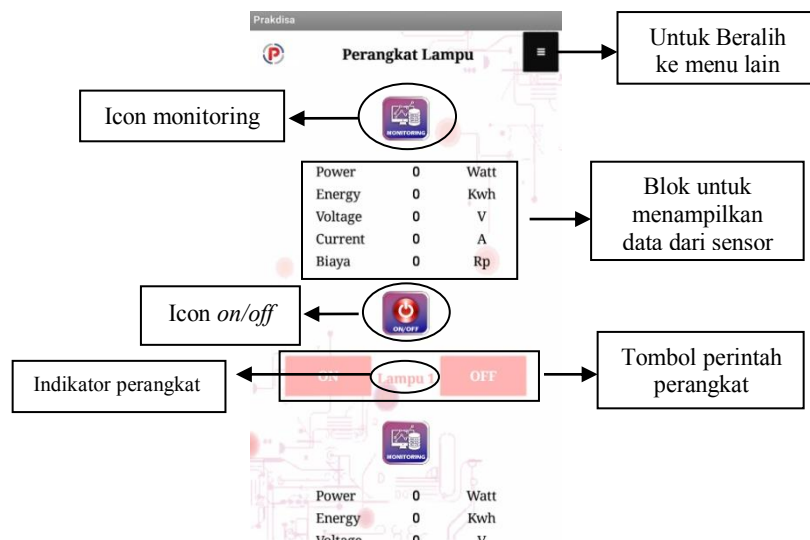


Gambar 3.8 Menu *login* pada aplikasi

Pada menu login ini, pengguna harus memasukkan id dan *password* untuk masuk kedalam aplikasi. Jika pengguna belum memiliki akun, pengguna dapat mengisi id dan *password* yang mudah diingat lalu pilih *SignUp*, maka pengguna baru sudah terdaftar sebagai pemilik akun.

### 3.3.2 Menu Perangkat Fiting Lampu Aplikasi

Pada menu perangkat fitting lampu ini, terdapat sub menu *on/off* dan sub menu *monitoring*. Sub menu *on/off* berfungsi untuk mengaktifkan atau tidak mengaktifkan perangkat yaitu lampu dengan kendali *smartphone*. Sedangkan sub menu *monitoring* berfungsi untuk melihat data-data pada perangkat yaitu lampu dengan kendali *smartphone*. Berikut gambar dari menu perangkat fitting lampu yang ditunjukkan pada gambar 3.9.

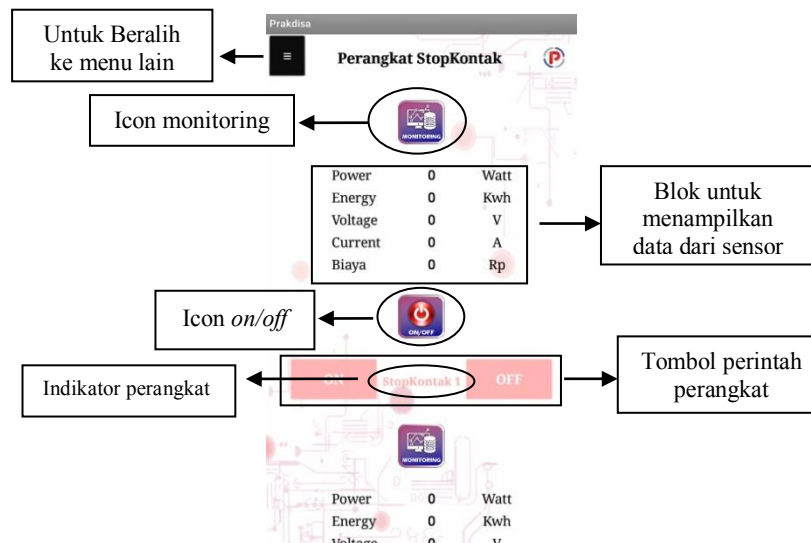


Gambar 3.9 Tampilan menu fitting lampu

Jika menekan tombol *on* maka perangkat lampu akan aktif dengan ditandai indikator tulisan *Lampu 1* berwarna merah pada *smartphone*. Pada blok *monitoring* menampilkan data sensor, nilai dari *power*, *energy*, *voltage*, *current*, dan biaya yang semulanya 0 akan menampilkan angka karena perangkat terhubung ke aplikasi. Jika menekan tombol *off* maka perangkat mati dengan ditandai lampu indikator pada tulisan *lampu 1* berwarna pink dan blok *monitoring* menampilkan data sensor, nilai dari *power*, *energy*, *voltage*, *current*, dan biaya akan kembali ke 0.

### 3.3.3 Menu Perangkat Stop Kontak Aplikasi

Pada menu perangkat stop kontak, terdapat sub menu *on/off* dan sub menu *monitoring*. Sub menu *on/off* berfungsi untuk mengaktifkan atau tidak mengaktifkan perangkat yaitu stop kontak dengan kendali *smartphone*. Sedangkan sub menu *monitoring* berfungsi untuk melihat data-data pada perangkat yaitu stop kontak dengan kendali *smartphone*. Berikut gambar dari menu perangkat stop kontak yang ditunjukkan pada gambar 3.10.



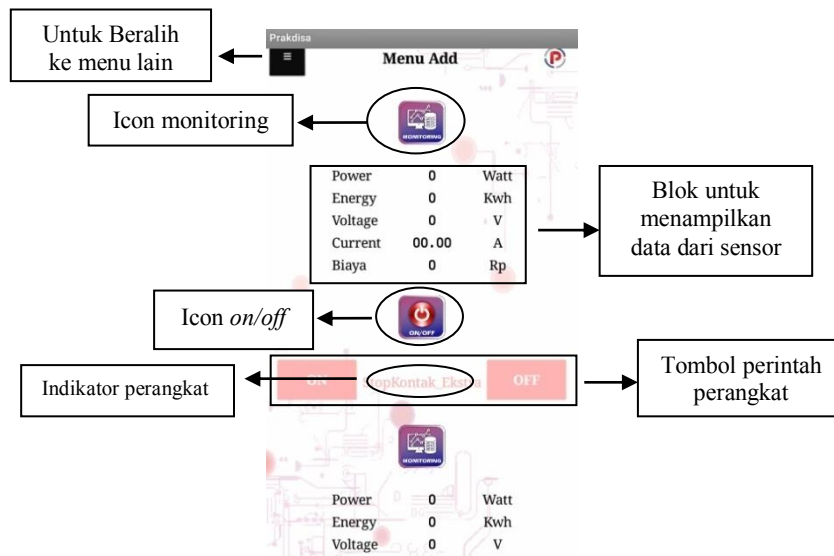
Gambar 3.10 Menu stop kontak aplikasi

Jika menekan tombol *on* maka perangkat stop kontak akan aktif dengan ditandai indikator tulisan stop kontak 1 berwarna merah pada *smartphone*. Pada blok *monitoring* menampilkan data sensor, nilai dari *power*, *energy*, *voltage*, *current*, dan biaya yang semulanya 0 akan menampilkan angka karena perangkat terhubung ke aplikasi. Jika menekan tombol *off* maka perangkat mati dengan ditandai lampu indikator pada tulisan stop kontak 1 berwarna pink dan blok *monitoring* menampilkan data sensor, nilai dari *power*, *energy*, *voltage*, *current*, dan biaya akan kembali ke 0.

### 3.3.4 Menu *Add*

Saat menu *add* ditekan, maka akan keluar tampilan seperti tampilan menu pada stop kontak dan menu pada fitting lampu. Menu *add* tersebut digunakan jika pengguna ingin menambahkan perangkat seperti stop kontak ekstra atau lampu ekstra. Perangkat tambahan dapat digunakan jika perangkat sudah terprogram terlebih dahulu. Berikut menu *add* ditunjukkan pada gambar 3.11.

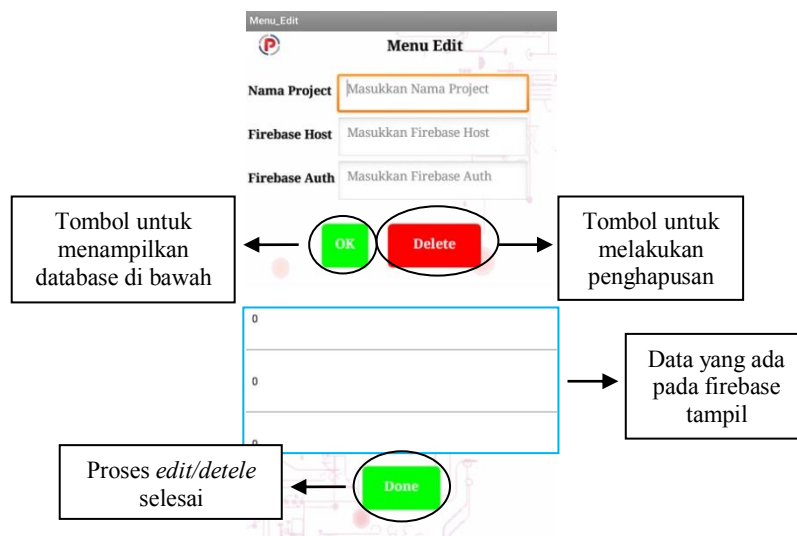




Gambar 3.11 Menu *add* aplikasi

### 3.3.5 Menu *Edit*

Menu *edit* ini digunakan untuk para pengguna yang ingin menghapus perangkat. Berikut tampilan pada menu *edit* ditunjukkan pada gambar 3.12.



Gambar 3.12 Menu *edit* aplikasi

Jika pengguna tidak ingin lagi menggunakan salah satu perangkat, pengguna dapat menghapus perangkat tersebut. Pengguna harus memasukkan nama *project*, *firebase host*, dan *firebase authnya*. Lalu pengguna dapat mengklik tombol *delete*

dan setelah itu pengguna dapat mengklik tombol *done*. Artinya penghapusan data perangkat sudah selesai dilakukan dan perangkat tidak dapat lagi digunakan.

## BAB IV

### PEMBAHASAN

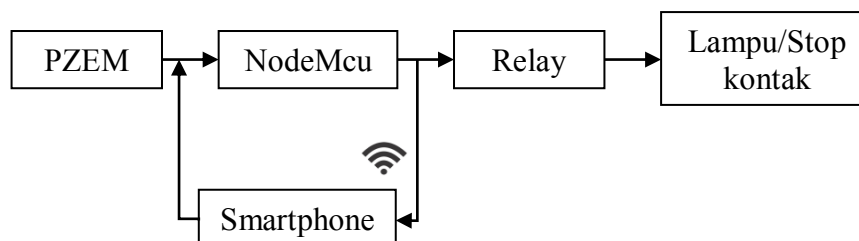
Pada bab ini akan dibahas mengenai proses pengujian alat proyek akhir yang berjudul “Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*”.

#### 4.1 Deskripsi alat

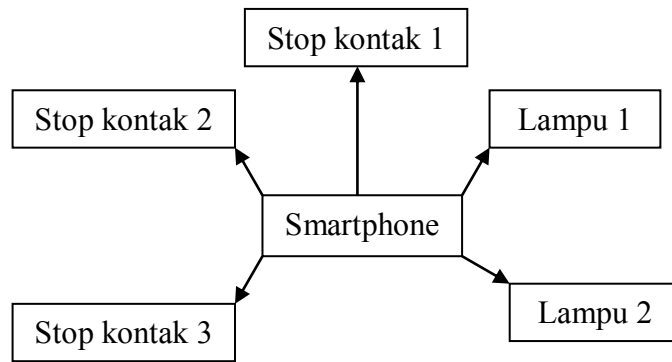
Sistem kontrol dan *monitoring* perangkat *smart home* menggunakan IoT dengan menerapkan topologi *Star* ini merupakan perangkat yang digunakan untuk mempermudah penggunaanya dalam mengontrol dan *monitoring* perangkat dari jauh dengan menggunakan *smartphone*. Sistem pengontrolan dan *monitoring* dari alat ini menggunakan sensor PZEM-004T dalam mengambil data berupa daya, tegangan, dan arus serta NodeMCU ESP8266 sebagai pengirim data yang akan ditampilkan pada aplikasi *smartphone*. Sistem ini menggunakan *database* berupa *firebase* sehingga perubahan data pada sensor PZEM-004T akan tetap tersinkronisasi oleh NodeMCU ESP8266.

#### 4.2 Diagram blok

Diagram blok dari Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star* ditunjukkan pada gambar 4.1 berikut.



Gambar 4. 1 Diagram blok sistem kontrol dan *monitoring*



Gambar 4.2 Diagram blok topologi *star* pada sistem

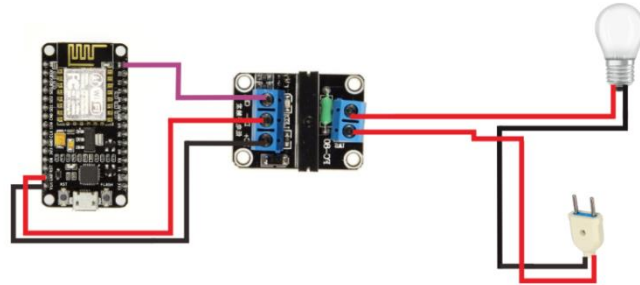
### 4.3 Prinsip kerja

Prinsip kerja dari blok kontrol dan *monitoring* diatas adalah semua komponen elektrik pada sistem ini terhubung dengan *power supply* sehingga sistem ini hanya dapat bekerja ketika mendapatkan tegangan listrik. NodeMCU ESP8266 akan membaca tegangan, arus, dan daya dengan bantuan sensor PZEM-004T kemudian mengirimkan data tersebut ke *database firebase* untuk ditampilkan ke aplikasi *smartphone*. NodeMCU ESP8266 juga akan membaca perintah yang dikirimkan oleh aplikasi *smartphone* yang akan di ubah dengan memberikan logika “HIGH” atau “LOW” pada pin tertentu pada *solid state relay* (SSR) guna mengatur *on/off* lampu atau stop kontak. *Database firebase* dengan memanfaatkan *wifi* menjadi pusat koneksi antara sistem dan aplikasi MIT App Inventor, dengan ini sistem dapat berjalan sesuai dengan yang diharapkan.

### 4.4 Pengujian kontrol

Pada tahap pengujian kontrol ini bertujuan untuk mengetahui apakah sistem kontrol pada perangkat *smart home* berbasis IoT dapat bekerja sesuai dengan target yang diinginkan. Pengujian kontrol ini dilakukan dengan cara mengontrol perangkat dari aplikasi yang telah dibuat pada *smartphone*.

#### 4.4.1 Pengujian kontrol fitting lampu



Gambar 4.3 Sistematis pemasangan pin SSR pada fitting lampu

Pada pengujian ini dilakukan dengan menekan saklar *on* dan *off* untuk lampu di aplikasi yang dibuat pada *smartphone*. Pengguna dapat masuk pada aplikasi yang telah dibuat dan pilih perangkat lampu untuk mengontrol lampu 1 atau lampu 2. Tujuan dilakukan pengujian kontrol pada fitting lampu ini adalah untuk mengetahui apakah lampu akan merespon atau tidak saat diberi perintah.

##### a. Pengujian fitting lampu 1 (11 watt)

Setelah dilakukan pengujian maka kondisi lampu yang didapatkan dapat dilihat pada gambar 4.3 berikut.



(a)



(b)

Gambar 4.4 (a) Kondisi lampu 1 menyala (b) kondisi lampu 1 padam

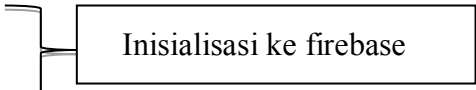
Adapun kondisi fitting lampu 1 yang telah diuji akan ditampilkan dalam bentuk tabel 4.1 berikut ini.

Tabel 4.1 Hasil pengujian fitting lampu 1

NO	Perintah	Kondisi Lampu
1	<i>ON</i>	Menyala
2	<i>OFF</i>	Tidak Menyala

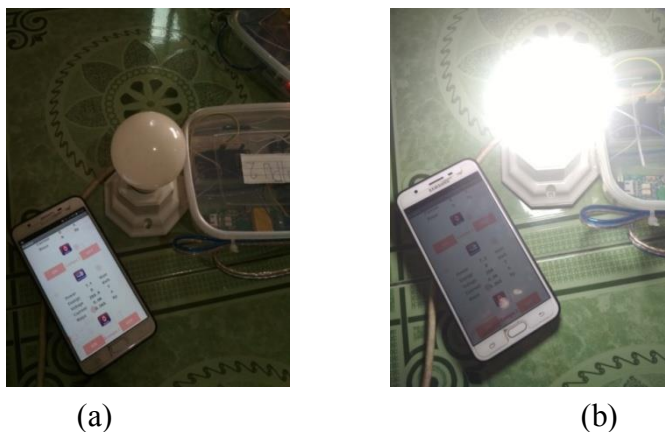
Ketika perintah *on* dikirimkan dari *smartphone* maka kondisi lampu 1 menyala. Ketika perintah *off* dikirimkan dari *smartphone* maka kondisi lampu 1 akan padam.

```
void loop() {
b=Firebase.getString("S4");
Serial.println(b);
if (b=="1"){ // perintah yang diberikan
Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
digitalWrite(S4,HIGH); //Mengeluarkan perintah
delay(5000);
}
else{
Serial.println("Lampu MATI");
digitalWrite(S4,LOW); //Device1 if OFF
}
}
```



b. Pengujian kontrol fitting lampu 2 (7 watt)

Setelah dilakukan pengujian maka kondisi lampu yang didapatkan dapat dilihat pada gambar 4.4 berikut.



Gambar 4.5 (a) Kondisi lampu 2 menyala (b) Kondisi lampu 2 padam

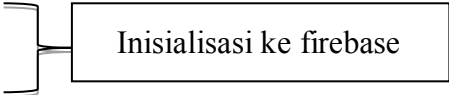
Adapun kondisi fitting lampu 2 yang telah diuji akan ditampilkan dalam bentuk tabel 4.2 berikut ini.

Tabel 4.2 Hasil pengujian kontrol lampu 2

NO	Perintah	Kondisi Lampu
1	<i>ON</i>	Menyala
2	<i>OFF</i>	Tidak Menyala

Ketika perintah *on* dikirimkan dari *smartphone* maka kondisi lampu 2 menyala. Ketika perintah *off* dikirimkan dari *smartphone* maka kondisi lampu 2 akan padam.

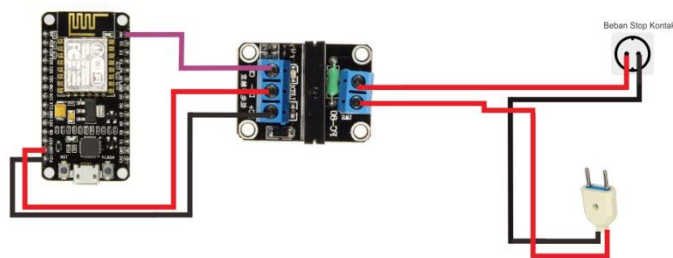
```
void loop() {
  f=Firebase.getString("S5");
  Serial.println (f);
  if (f=="1"){ // perintah yang diberikan
    Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
    digitalWrite(S4,HIGH); //Mengeluarkan perintah
    delay(5000);
  }
  else{
    Serial.println("Lampu MATI");
    digitalWrite(S4,LOW); //Device1 if OFF
  }
}
```



c. Analisa dan kesimpulan

Pada hasil pengujian kontrol lampu 1 dan lampu 2, diketahui bahwa perangkat dan aplikasi yang telah dibuat dapat bekerja sesuai dengan perintah yang diberikan. Perangkat juga berfungsi baik pada sistem IoT.

#### 4.4.2 Pengujian kontrol stop kontak



Gambar 4.6 Sistematis pemasangan pin SSR dan NodeMCU pada stop kontak

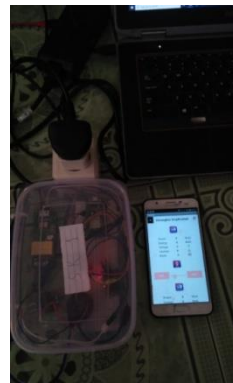
Pada pengujian ini dilakukan dengan menekan saklar *on* dan *off* untuk stop kontak di aplikasi yang dibuat pada *smartphone*. Pengguna dapat masuk pada aplikasi yang telah dibuat dan pilih perangkat stop kontak untuk mengontrol stop kontak 1, stop kontak 2, dan stop kontak 3.

a. Pengujian stop kontak 1

Setelah dilakukan pengujian maka kondisi stop kontak 1 yang didapatkan dapat dilihat pada gambar 4.6 berikut.



(a)



(b)

Gambar 4. 7 (a) Kondisi stop kontak 1 menyala (b) Kondisi stop kontak 1 padam

Adapun kondisi stop kontak 1 yang telah diuji akan ditampilkan dalam bentuk tabel 4.3 berikut ini.

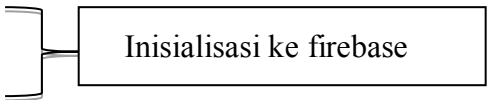


Tabel 4. 3 Hasil pengujian stop kontak 1

NO	Perintah	Kondisi Stop kontak
1	<i>ON</i>	Menyala
2	<i>OFF</i>	Tidak Menyala

Ketika perintah *on* dikirimkan dari *smartphone* maka kondisi stop kontak 1 menyala. Ketika perintah *off* dikirimkan dari *smartphone* maka kondisi stop kontak 1 akan padam.

```
void loop() {
c=Firebase.getString("S1");
Serial.println(c);
if (c=="1"){ // perintah yang diberikan
Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
digitalWrite(S1,HIGH); //Mengeluarkan perintah
delay(5000);
}
else{
Serial.println("Lampu MATI");
digitalWrite(S1,LOW); //Device1 if OFF
}
}
```



b. Pengujian stop kontak 2

Setelah dilakukan pengujian maka kondisi stop kontak 2 yang didapatkan dapat dilihat pada gambar 4.7 berikut.



(a)



(b)

Gambar 4.8 (a) Kondisi stop kontak 2 menyala (b) Kondisi stop kontak 2 padam

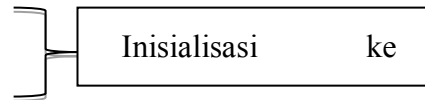
Adapun kondisi stop kontak 2 yang telah diuji akan ditampilkan dalam bentuk tabel 4.3 berikut ini.

Tabel 4.4 Hasil pengujian stop kontak 2

NO	Perintah	Kondisi Stop kontak
1	<i>ON</i>	Menyala
2	<i>OFF</i>	Tidak Menyala

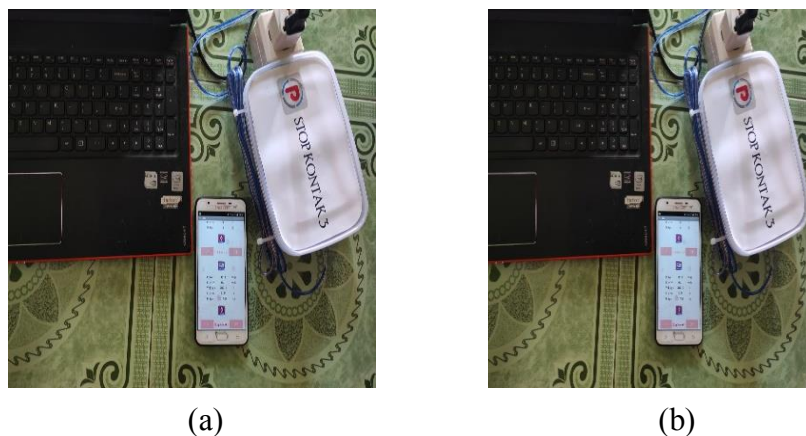
Ketika perintah *on* dikirimkan dari *smartphone* maka kondisi stop kontak 2 menyala. Ketika perintah *off* dikirimkan dari *smartphone* maka kondisi stop kontak 2 akan padam.

```
void loop() {
d=Firebase.getString("S2");
Serial.println(d);
if (d=="1"){ // perintah yang diberikan
Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
digitalWrite(S2,HIGH); //Mengeluarkan perintah
delay(5000);
}
else{
Serial.println("Lampu MATI");
digitalWrite(S2,LOW);} //Device1 if OFF
```



c. Pengujian stop kontak 3

Setelah dilakukan pengujian maka kondisi stop kontak 3 yang didapatkan dapat dilihat pada gambar 4.8 berikut.



Gambar 4. 9 Kondisi stop kontak 3 menyala (b) Kondisi stop kontak 3 padam

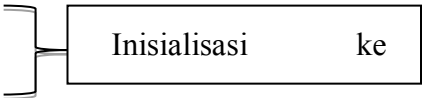
Adapun kondisi stop kontak 3 yang telah diuji akan ditampilkan dalam bentuk tabel 4.5 berikut ini.

Tabel 4.5 Tabel pengujian stop kontak 3

NO	Perintah	Kondisi Stop kontak
1	<i>ON</i>	Menyala
2	<i>OFF</i>	Tidak Menyala

Ketika perintah *on* dikirimkan dari *smartphone* maka kondisi stop kontak 2 menyala. Ketika perintah *off* dikirimkan dari *smartphone* maka kondisi stop kontak 2 akan padam.

```
void loop() {
e=Firebase.getString("S3");
Serial.println (e);
if (e=="1"){ // perintah yang diberikan
Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
digitalWrite(S3,HIGH); //Mengeluarkan perintah
delay(5000);
}
else{
Serial.println("Lampu MATI");
digitalWrite(S3,LOW); //Device1 if OFF
}
}
```



d. Analisa dan kesimpulan

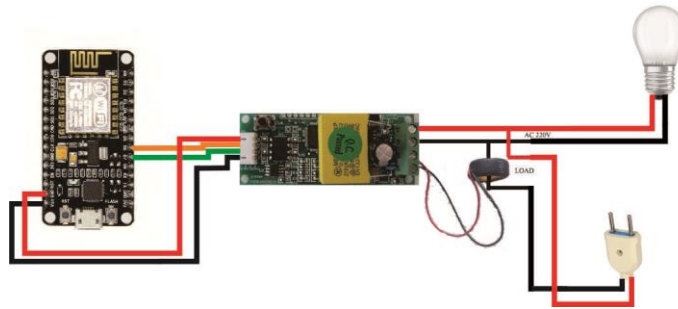
Pada hasil pengujian kontrol stop kontak 1, stop kontak 2, dan stop kontak 3 diketahui bahwa perangkat bekerja sesuai dengan perintah yang diberikan pada *smartphone*. Dengan begitu diketahui bahwa komponen *solid state relay* dalam keadaan baik dan komponen layak untuk digunakan.

**4.5 Pengujian Monitoring**

Pengujian pada fitting lampu dan stop kontak tidak hanya pengujian kontrol saja. Pada fitting lampu dan stop kontak juga terdapat data yang dapat di *monitoring* melalui *smartphone*. Tujuan dari pengujian *monitoring* ini untuk memastikan bahwa komponen yang digunakan dapat berfungsi dengan baik. Pada tahap pengujian *monitoring* yang akan diuji adalah sensor PZEM-004T yang akan

berfungsi untuk membaca data tegangan, arus dan daya. Pengujian ini dilakukan dengan menghubungkan pin DC PZEM-004T pada pin NodeMCU dan pin AC langsung ke sumber dan perangkat menggunakan kabel jumper.

#### 4.5.1 Pengujian *monitoring* pada fitting lampu



Gambar 4.10 Sistematis pemasangan pin PZEM dan NodeMCU pada fitting lampu

Pada pengujian ini dilakukan dengan menekan saklar *on* dan *off* untuk lampu di aplikasi yang dibuat pada *smartphone*. Pengguna dapat masuk pada aplikasi yang telah dibuat dan pilih perangkat lampu untuk mengontrol lampu 1 atau lampu 2. Saat perangkat dalam keadaan *on*, maka hasil dari sensor PZEM-004T akan tampil pada aplikasi yang dibuat.

Untuk mengetahui fungsi dari PZEM-004T ini apakah telah dapat bekerja dengan baik yaitu dengan menampilkan *power*, *current*, *voltage*, dan biaya pada *serial monitor* di NodeMCU ESP8266.

```
void loop()
{
float  voltage = pzem.voltage();
float current = pzem.current();
float power = pzem.power();
if (isnan(voltage)){
voltage = 0;
}
if (isnan(current) ) {
current = 0;
}
if (isnan(power) ) {
power = 0;
}
}
```

Membaca data sensor

Jika gagal membaca sensor

```

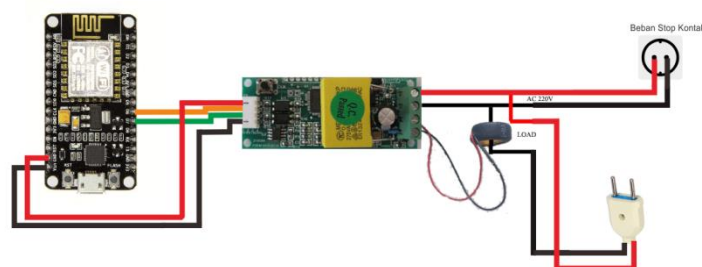
}
if (isnan(energy) ) {
    energy = 0;
}
float totalHarga = energy * harga_KWh;
Firebase.setString("volt4", String(voltage));
Firebase.setString("current4", String(current, 2));
Firebase.setFloat("watt4", power);
Firebase.setString("kwh4", String(energy, 1));
Firebase.setString("rp4", String(totalHarga, 3));
Serial.print("Voltage = ");
Serial.print(voltage);
Serial.println("v");
Serial.print("Current = ");
Serial.print(current, 2);
Serial.println("A");
Serial.print("Watt = ");
Serial.print(power);
Serial.println("watt");
Serial.print("KWh = ");
Serial.print(energy, 1);
Serial.println("KWh");
Serial.print("Harga = Rp ");
Serial.print(totalHarga, 3);

```

Mengirim data sensor ke database firebase

Menampilkan data sensor ke serial monitor dan *smartphone*

#### 4.5.2 Pengujian *monitoring* pada stop kontak



Gambar 4.11 Sistematis pemasangan pin PZEM dan NodeMCU pada stop kontak

Pada pengujian ini dilakukan dengan menekan saklar *on* dan *off* untuk stop kontak di aplikasi yang dibuat pada *smartphone*. Pengguna dapat masuk pada

aplikasi yang telah dibuat dan pilih perangkat stop kontak untuk mengontrol stop kontak 1, stop kontak 2, atau stop kontak 3. Saat perangkat dalam keadaan *on*, maka hasil dari sensor PZEM-004T akan tampil pada aplikasi yang dibuat.

Untuk mengetahui fungsi dari PZEM-004T ini apakah telah dapat bekerja dengan baik yaitu dengan menampilkan *power*, *current*, *voltage*, dan biaya pada *serial monitor* di NodeMCU ESP8266.

```

void loop()
{
float  voltage = pzem.voltage();
float current = pzem.current();
float power = pzem.power();
if (isnan(voltage)){
voltage = 0;
}
if (isnan(current) ) {
current = 0;
}
if (isnan(power) ) {
power = 0;
}
if (isnan(energy) ) {
energy = 0;
}

float totalHarga = energy * harga_KWh;
Firebase.setString("volt4", String(voltage));
Firebase.setString("current4", String(current, 2));
Firebase.setFloat("watt4", power);
Firebase.setString("kwh4", String(energy, 1));
Firebase.setString("rp4", String(totalHarga, 3));
Serial.print("Voltage = ");
Serial.print(voltage);
Serial.println("v");
Serial.print("Current = ");
Serial.print(current, 2);
Serial.println("A");
Serial.print("Watt = ");

```

```

Serial.print(power);
Serial.println("watt");
Serial.print("KWh = ");
Serial.print(energy, 1);
Serial.println("KWh");
Serial.print("Harga = Rp ");
Serial.print(totalHarga, 3);

```

Menampilkan data sensor ke serial monitor dan *smartphone*

#### 4.6 Pengujian Sensor Banding dengan Alat Ukur

Untuk menguji keakuratan alat yang dibuat dalam mengukur nilai arus, tegangan, dan daya yang digunakan, dibandingkan dengan alat ukur multimeter. Berikut hasil pengujian yang dilakukan pada fitting lampu dan stop kontak:

a. Hasil pengujian lampu 1 (11 watt)

Pengujian sensor PZEM-004T dilakukan dengan menggunakan beban lampu sebesar 11 watt. Berikut hasil pengujian dapat dilihat pada tabel 4.6 berikut:

Tabel 4.6 Data pengujian lampu 1

Beban	Percobaan Jam ke-	Tegangan (V)			Arus (A)		
		Vuk	Va	PaV %	Iuk	Ia	PaI %
Lampu 11 Watt	1	209,4	209,8	0,19	0,148	0,15	1,33
	2	210,1	210,6	0,23	0,148	0,15	1,33
	3	209,6	210,2	0,28	0,148	0,15	1,33
	4	209,4	209,8	0,19	0,148	0,15	1,33
	5	209	209,3	0,14	0,148	0,15	1,33
	6	209,6	210	0,19	0,148	0,15	1,33
	7	209,6	210	0,19	0,148	0,15	1,33
	8	209,6	210	0,19	0,148	0,15	1,33
Rata-rata		209,5	209,96	0,21	0,148	0,15	1,33

Keterangan:

Vuk = nilai tegangan yang terbaca pada alat ukur

- Va = nilai tegangan yang terbaca pada alat yang dibuat  
 Iuk = nilai arus yang terbaca pada alat ukur  
 Ia = nilai arus yang terbaca pada alat yang dibuat  
 PaV = perbandingan tegangan antara alat ukur dan alat yang dibuat  
 PaI = perbandingan arus antara alat ukur dan alat yang dibuat

Tabel 4. 7 Hasil data daya lampu 1

Beban	Percobaan Jam ke-	Daya (W)		
		Wuk	Wa	PaW
Lampu 11 Watt	1	10,89	11,9	8,48
	2	10,89	11,8	7,7
	3	10,89	11,7	6,9
	4	10,89	11,6	6,1
	5	10,89	11,5	5,3
	6	10,89	11,5	5,3
	7	10,89	11,5	5,3
	8	10,89	11,5	5,3
Rata-rata		10,89	11,6	6,29

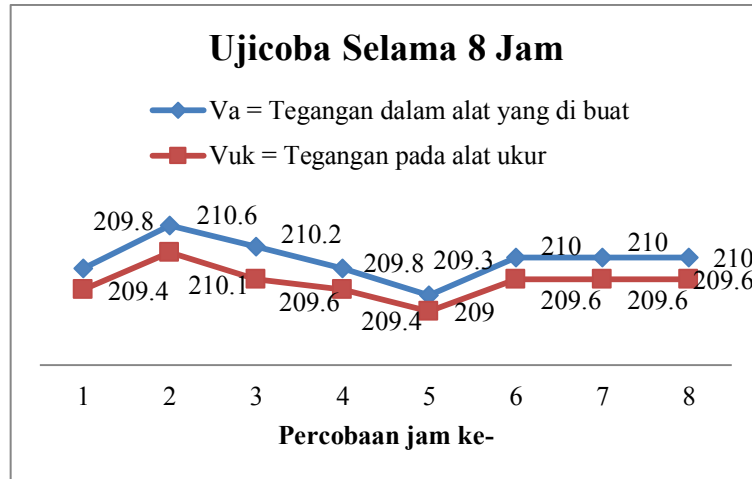
Keterangan:

- Wuk = nilai daya yang terbaca pada alat ukur  
 Wa = nilai daya yang terbaca pada alat yang dibuat  
 PaW = perbandingan daya antara perhitungan dan alat yang dibuat

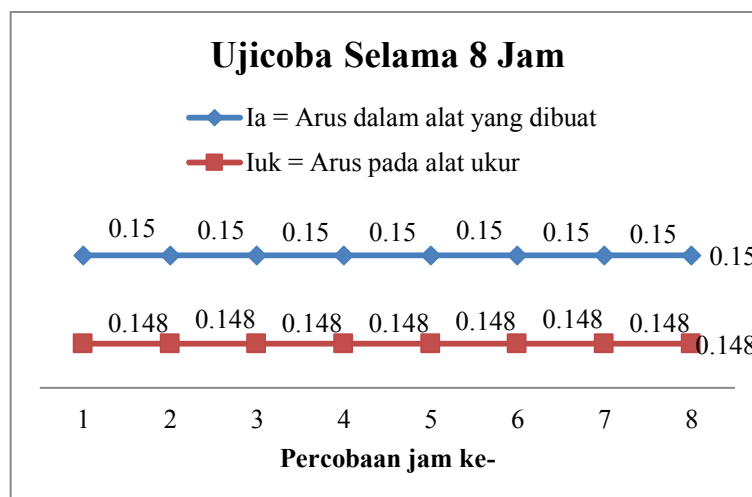
Setelah pengujian data tegangan, arus dan daya didapatkan rata-rata persentase perbandingan tegangan yang didapatkan pada persamaan (2.6), (2.7) dan (2.8) antara alat ukur dengan data yang didapatkan pada *smartphone* dengan beban lampu 11 watt dengan percobaan selama 8 jam berada pada 0,21% untuk tegangan, 1,33% untuk arus dan 6,29% untuk daya. Terdapat perbedaan daya yang diukur oleh sensor dan alat ukur. Namun dengan demikian alat ini dapat digunakan sebagaimana mestinya.



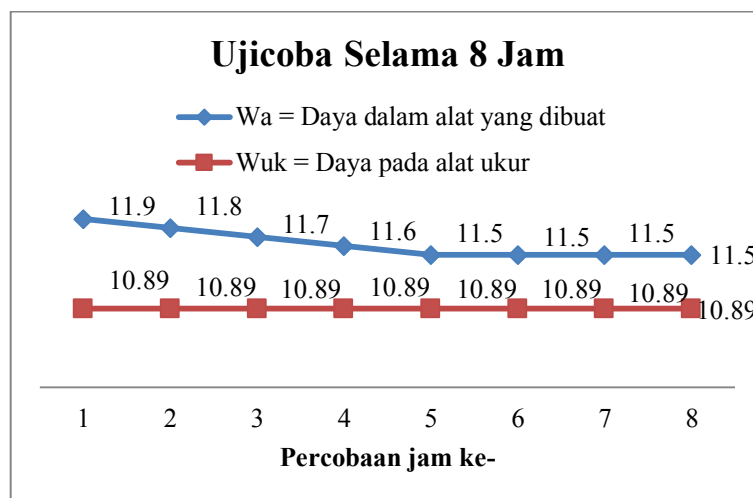
Setelah dilakukan ujicoba antara alat ukur dan alat yang dibuat, maka dibuatlah grafik perbandingan. Berikut grafik yang didapatkan ditunjukkan dibawah ini.



Gambar 4.12 Perbandingan nilai tegangan lampu 1



Gambar 4.13 Perbandingan nilai arus lampu 1



Gambar 4.14 Perbandingan nilai daya lampu 1

b. Hasil pengujian lampu 2 (7 watt)

Pengujian sensor PZEM-004T dilakukan dengan menggunakan beban lampu sebesar 7 watt. Berikut hasil pengujian dapat dilihat pada tabel 4.8 berikut:

Tabel 4.8 Data pengujian lampu 2

Beban	Percobaan Jam ke-	Tegangan (V)			Arus (A)		
		Vuk	Va	PaV %	Iuk	Ia	PaI %
Lampu 7 Watt	1	210,4	210,7	0,14	0,069	0,07	1,42
	2	209,7	210,1	0,19	0,069	0,07	1,42
	3	209,7	210,1	0,19	0,069	0,07	1,42
	4	209,9	210,3	0,14	0,069	0,07	1,42
	5	210,3	210,7	0,19	0,069	0,07	1,42
	6	210,3	210,7	0,19	0,069	0,07	1,42
	7	209,7	210,1	0,19	0,069	0,07	1,42
	8	209,3	209,6	0,14	0,069	0,07	1,42
	Rata-rata	209,9	210,28	0,17	0,069	0,07	1,42

Keterangan:

Vuk = nilai tegangan yang terbaca pada alat ukur

Va = nilai tegangan yang terbaca pada alat yang dibuat

- Iuk = nilai arus yang terbaca pada alat ukur  
 Ia = nilai arus yang terbaca pada alat yang dibuat  
 PaV = perbandingan tegangan antara alat ukur dan alat yang dibuat  
 PaI = perbandingan arus antara alat ukur dan alat yang dibuat

Tabel 4.9 Hasil data daya lampu 2

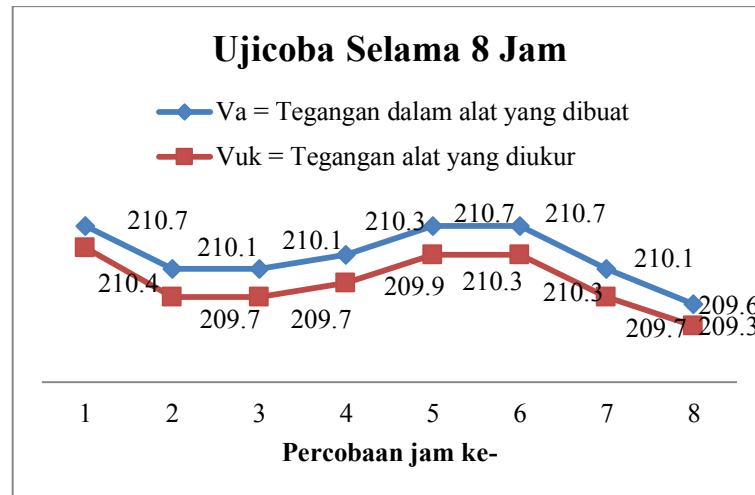
Beban	Percobaan		Daya (W)	
	Jam ke-	Wuk	Wa	PaW
Lampu 7 Watt	1	6,93	7,6	8,8
	2	6,93	7,6	8,8
	3	6,93	7,6	8,8
	4	6,93	7,6	8,8
	5	6,93	7,6	8,8
	6	6,93	7,5	7,6
	7	6,93	7,4	6,3
	8	6,93	7,5	7,6
Rata-rata		6,93	7,55	8,18

Keterangan:

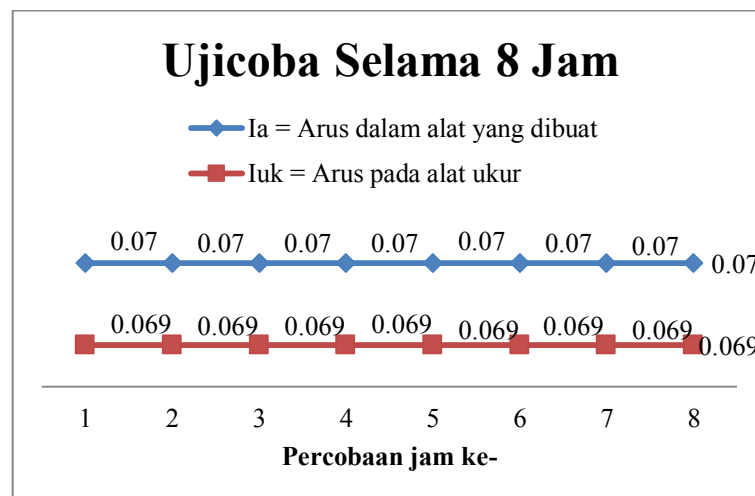
- Wuk = nilai daya yang terbaca pada alat ukur  
 Wa = nilai daya yang terbaca pada alat yang dibuat  
 PaW = perbandingan daya antara perhitungan dan alat yang dibuat

Setelah pengujian data tegangan, arus dan daya didapatkan rata-rata persentase perbandingan tegangan yang didapatkan pada persamaan (2.6), (2.7) dan (2.8) antara alat ukur dengan data yang didapatkan pada *smartphone* dengan beban lampu 7 watt dengan percobaan selama 8 jam berada pada 0,17% untuk tegangan, 1,42% untuk arus dan 8,18% untuk daya. Terdapat perbedaan daya yang diukur oleh sensor dan alat ukur. Namun dengan demikian alat ini dapat digunakan sebagaimana mestinya.

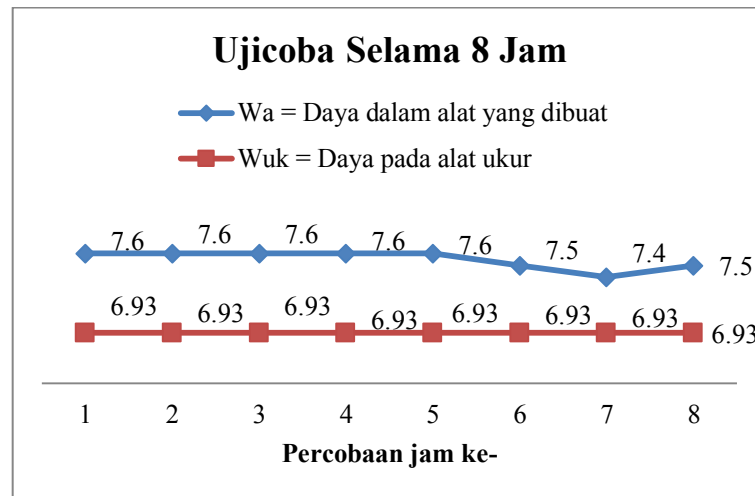
Setelah dilakukan ujicoba antara alat ukur dan alat yang dibuat, maka dibuatlah grafik perbandingan. Berikut grafik yang didapatkan ditunjukkan dibawah ini.



Gambar 4.15 Perbandingan nilai tegangan lampu 2



Gambar 4.16 Perbandingan nilai arus lampu 2



Gambar 4.17 Pebandingan nilai daya lampu 2

c. Hasil pengujian stop kontak 1

Pengujian sensor PZEM-004T dilakukan dengan menggunakan beban *charger* laptop. Berikut hasil pengujian dapat dilihat pada tabel 4.10 berikut:

Tabel 4.10 Tabel pengujian stop kontak 1

Beban	Percobaan Jam ke-	Tegangan (V)			Arus (A)		
		Vuk	Va	PaV %	Iuk	Ia	PaI %
<i>Charger</i> laptop	1	208	208,4	0,19	0,42	0,43	2,38
	2	208,4	208,7	0,14	0,42	0,43	2,38
	3	208,2	208,6	0,19	0,42	0,43	2,38
	4	208,2	208,5	0,14	0,42	0,43	2,38
	5	208,4	208,7	0,14	0,42	0,43	2,38
	6	208,1	208,4	0,14	0,42	0,43	2,38
	7	208,1	208,4	0,14	0,42	0,43	2,38
	8	207,9	208,2	0,14	0,42	0,43	2,38
	Rata-rata	208,1	210,28	0,15	0,42	0,43	2,38

Keterangan:

Vuk = nilai tegangan yang terbaca pada alat ukur

Va = nilai tegangan yang terbaca pada alat yang dibuat

Iuk = nilai arus yang terbaca pada alat ukur

- Ia = nilai arus yang terbaca pada alat yang dibuat  
 PaV = perbandingan tegangan antara alat ukur dan alat yang dibuat  
 PaI = perbandingan arus antara alat ukur dan alat yang dibuat

Tabel 4. 11 Hasil data daya stop kontak 1

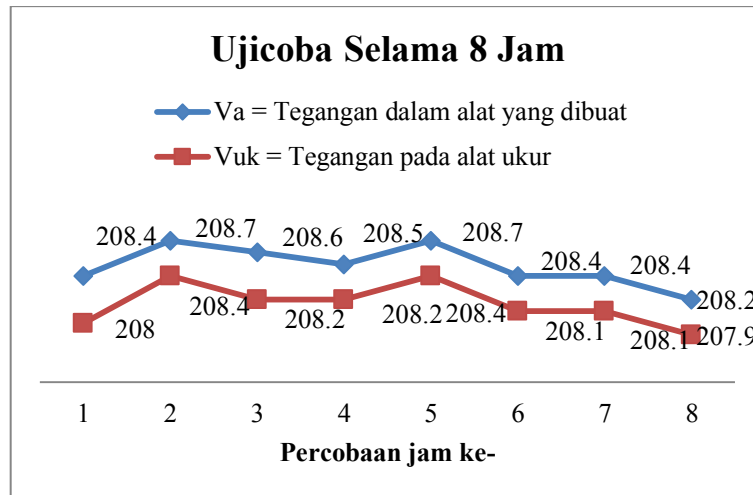
Beban	Percobaan		Daya (W)	
	Jam ke-	Wuk	Wa	PaW
<i>Charger</i> laptop	1	89,1	86	3,6
	2	89,1	86,3	3,2
	3	89,1	86,5	3
	4	89,1	86,5	3
	5	89,1	86,9	2,5
	6	89,1	86,7	2,7
	7	89,1	86,3	3,2
	8	89,1	86,7	2,7
Rata-rata		89,1	86,48	2,98

Keterangan:

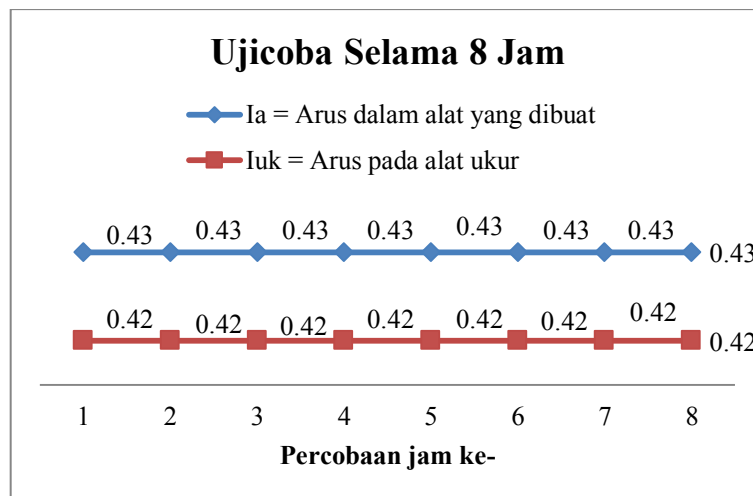
- Wuk = nilai daya pada alat ukur  
 Wa = nilai daya yang terbaca pada alat yang dibuat  
 PaW = perbandingan daya antara perhitungan dan alat yang dibuat

Setelah pengujian data tegangan, arus dan daya didapatkan rata-rata persentase perbandingan tegangan yang didapatkan pada persamaan (2.6), (2.7) dan (2.8) antara alat ukur dengan data yang didapatkan pada *smartphone* dengan beban *charger* laptop selama percobaan selama 8 jam berada pada 0,15% untuk tegangan, 2,38% untuk arus dan 2,98% untuk daya. Terdapat perbedaan daya yang diukur oleh sensor dan alat ukur. Namun dengan demikian alat ini dapat digunakan sebagaimana mestinya.

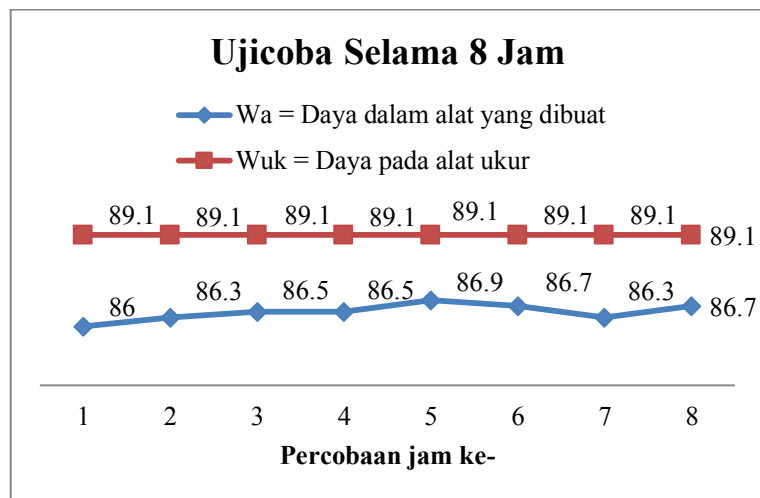
Setelah dilakukan ujicoba antara alat ukur dan alat yang dibuat, maka dibuatlah grafik perbandingan. Berikut grafik yang didapatkan ditunjukkan dibawah ini.



Gambar 4.18 Perbandingan nilai tegangan stop kontak 1



Gambar 4.19 Perbandingan nilai arus stop kontak 1



Gambar 4. 20 Perbandingan nilai daya stop kontak 1

d. Hasil pengujian stop kontak 2

Pengujian sensor PZEM-004T dilakukan dengan menggunakan beban lampu 7 watt. Berikut hasil pengujian dapat dilihat pada tabel 4.12 berikut:

Tabel 4. 12 Data pengujian stop kontak 2

Beban	Percobaan Jam ke-	Tegangan (V)			Arus (A)		
		Vuk	Va	PaV %	Iuk	Ia	PaI %
Lampu 7 watt	1	210,1	210,4	0,14	0,048	0,05	4
	2	210,4	210,7	0,14	0,048	0,05	4
	3	209,3	209,6	0,14	0,048	0,05	4
	4	209,3	209,6	0,14	0,048	0,05	4
	5	210,3	210,6	0,14	0,048	0,05	4
	6	208,7	209	0,19	0,048	0,05	4
	7	209,6	210	0,19	0,048	0,05	4
	8	209,2	209,6	0,19	0,048	0,05	4
Rata-rata		209,4	209,93	0,15	0,048	0,05	4

Keterangan:

Vuk = nilai tegangan yang terbaca pada alat ukur

Va = nilai tegangan yang terbaca pada alat yang dibuat

Iuk = nilai arus yang terbaca pada alat ukur

Ia = nilai arus yang terbaca pada alat yang dibuat



PaV = perbandingan tegangan antara alat ukur dan alat yang dibuat

PaI = perbandingan arus antara alat ukur dan alat yang dibuat

Tabel 4.13 Hasil data daya stop kontak 2

Beban	Percobaan Jam ke-	Daya (W)		
		Wuk	Wa	PaW
Lampu 7 watt	1	6,93	4,8	10
	2	6,93	4,9	10
	3	6,93	4,9	8,8
	4	6,93	4,9	8,8
	5	6,93	4,9	8,8
	6	6,93	4,8	8,8
	7	6,93	4,8	8,8
	8	6,93	4,8	8,8
Rata-rata		6,93	6,93	9,1

Keterangan:

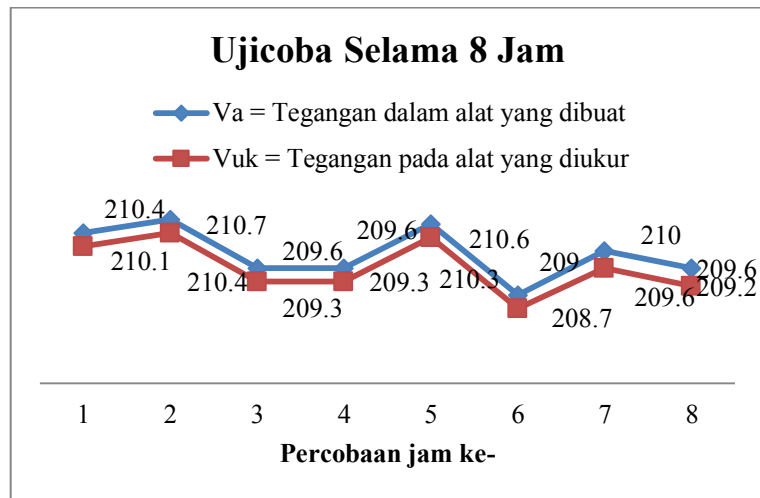
Wuk = nilai daya pada alat ukur

Wa = nilai daya yang terbaca pada alat yang dibuat

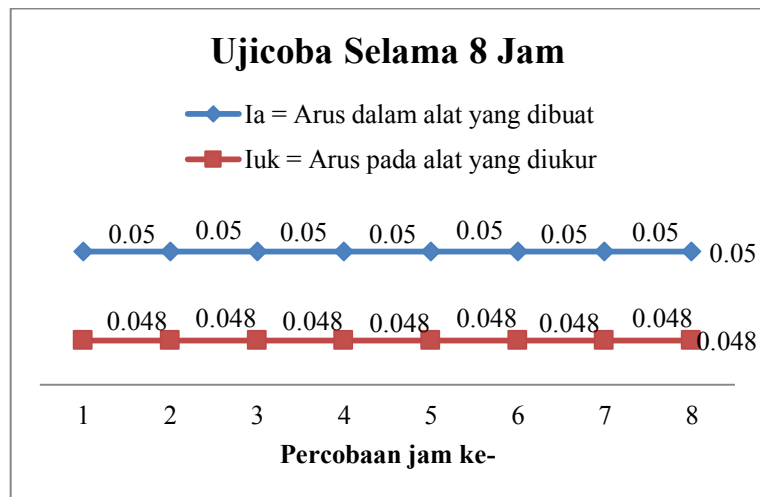
PaW = perbandingan daya antara perhitungan dan alat yang dibuat

Setelah pengujian data tegangan, arus dan daya didapatkan rata-rata persentase perbandingan tegangan yang didapatkan pada persamaan (2.6), (2.7) dan (2.8) antara alat ukur dengan data yang didapatkan pada *smartphone* dengan beban lampu 7 watt selama percobaan selama 8 jam berada pada 0,15% untuk tegangan, 4% untuk arus dan 9,1% untuk daya. Terdapat perbedaan daya yang diukur oleh sensor dan alat ukur. Namun dengan demikian alat ini dapat digunakan sebagaimana mestinya.

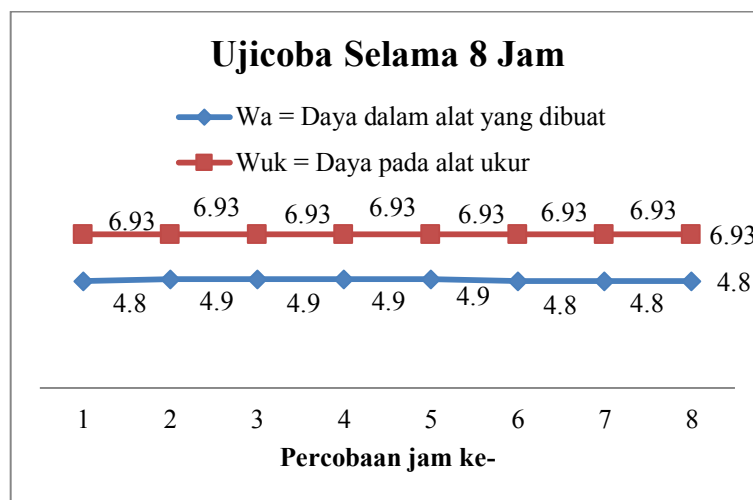
Setelah dilakukan ujicoba antara alat ukur dan alat yang dibuat, maka dibuatlah grafik perbandingan. Berikut grafik yang didapatkan ditunjukkan dibawah ini.



Gambar 4.21 Perbandingan nilai tegangan stop kontak 2



Gambar 4.22 Perbandingan nilai arus stop kontak 2



Gambar 4. 23 Perbandingan nilai daya stop kontak 2

e. Hasil pengujian stop kontak 3

Pengujian sensor PZEM-004T dilakukan dengan menggunakan beban *charger* laptop. Berikut hasil pengujian dapat dilihat pada tabel 4.14 berikut:

Tabel 4. 14 Data pengujian stop kontak 3

Beban	Percobaan Jam ke-	Tegangan (V)			Arus (A)		
		Vuk	Va	PaV %	Iuk	Ia	PaI %
<i>Charger</i> laptop	1	210,1	210,4	0,14	0,138	0,14	1,42
	2	209,5	209,9	0,19	0,138	0,15	8
	3	210	210,3	0,14	0,138	0,15	8
	4	209,1	209,4	0,14	0,138	0,14	1,42
	5	208,9	209,3	0,19	0,138	0,15	8
	6	209,6	209,9	0,14	0,138	0,15	8
	7	209,6	210	0,19	0,138	0,15	8
	8	209,6	210	0,19	0,138	0,14	1,42
	Rata-rata	209,55	209,9	0,165	0,138	0,146	5,5

Keterangan:

Vuk = nilai tegangan yang terbaca pada alat ukur

Va = nilai tegangan yang terbaca pada alat yang dibuat

- Iuk = nilai arus yang terbaca pada alat ukur  
 Ia = nilai arus yang terbaca pada alat yang dibuat  
 PaV = perbandingan tegangan antara alat ukur dan alat yang dibuat  
 PaI = perbandingan arus antara alat ukur dan alat yang dibuat

Tabel 4.15 Hasil data daya stop kontak 3

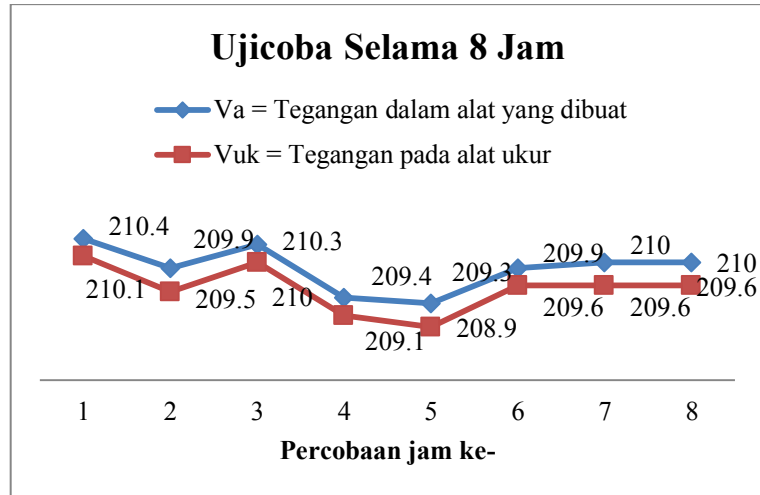
Beban	Percobaan	Daya (W)		
	Jam ke-	Wuk	Wa	PaW
<i>Charger</i> laptop	1	44,55	40,9	8,9
	2	44,55	39,9	11,65
	3	44,55	39,3	13,3
	4	44,55	40,5	10
	5	44,55	40,4	10,2
	6	44,55	40,5	10
	7	44,55	40,6	9,7
	8	44,55	40	11,3
	Rata-rata	44,55	40,2	10,8

Keterangan:

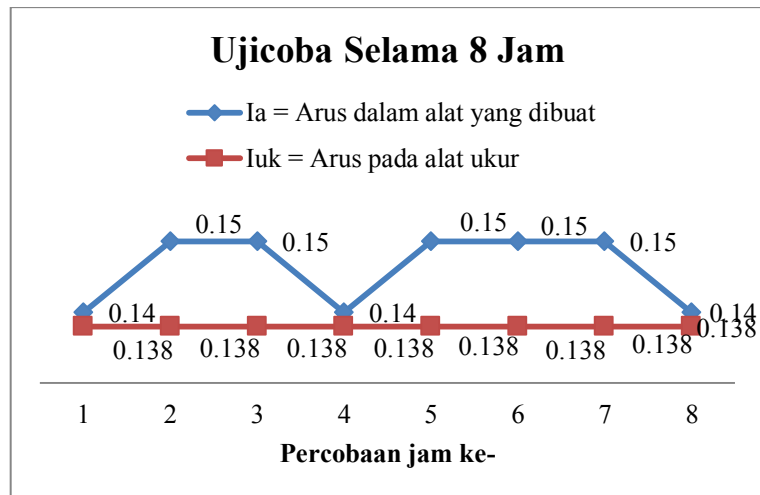
- Wuk = nilai daya dalam alat yang dibuat  
 Wa = nilai daya yang terbaca pada alat yang dibuat  
 PaW = perbandingan daya antara perhitungan dan alat yang dibuat

Setelah pengujian data tegangan, arus dan daya didapatkan rata-rata persentase perbandingan tegangan yang didapatkan pada persamaan (2.6), (2.7) dan (2.8) antara alat ukur dengan data yang didapatkan pada *smartphone* dengan beban *charger* laptop selama percobaan selama 8 jam berada pada 0,165% untuk tegangan, 5,5% untuk arus dan 10,8% untuk daya. Terdapat perbedaan daya yang diukur oleh sensor dan alat ukur. Namun dengan demikian alat ini dapat digunakan sebagaimana mestinya.

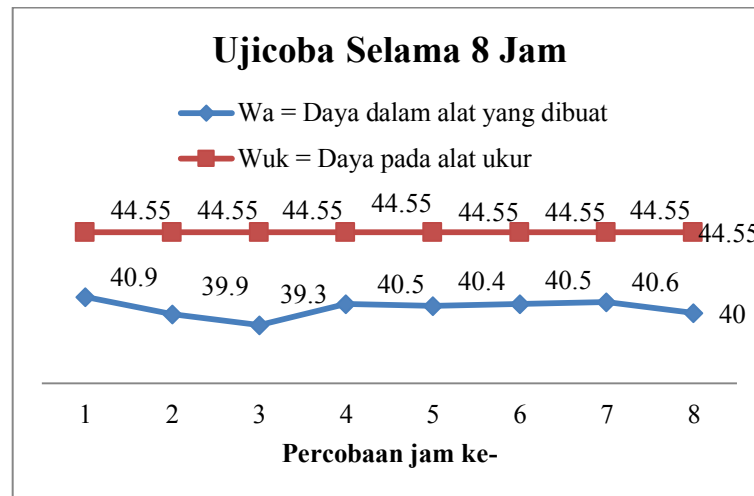
Setelah dilakukan ujicoba antara alat ukur dan alat yang dibuat, maka dibuatlah grafik perbandingan. Berikut grafik yang didapatkan ditunjukkan dibawah ini.



Gambar 4.24 Perbandingan nilai tegangan stop kontak 3



Gambar 4.25 Perbandingan nilai arus stop kontak 3



Gambar 4.26 Perbandingan nilai daya stop kontak 3

f. Analisa dan kesimpulan

Pada hasil pengujian *monitoring* perangkat selama 8 jam dengan aplikasi yang dibuat dan pada alat ukur pada perangkat stop kontak dan lampu memiliki hasil tegangan yang tidak stabil atau mengalami naik turun. Hal tersebut bisa terjadi karena faktor internal dan faktor eksternal. Faktor internal biasanya dikarenakan sensor yang digunakan, tetapi ini minim untuk terjadi. Lalu faktor eksternal penyebab paling umum adalah trafo distribusi sudah kelebihan beban, tegangan yang terlalu rendah juga bisa disebabkan oleh sambungan-sambungan listrik yang kendur. Selain itu, bisa juga akibat adanya arus listrik yang terlalu besar sehingga tegangan turun. Untuk nilai arus pada perangkat fitting lampu dan stop kontak memiliki nilai yang stabil. Untuk nilai daya tidak stabil tetapi tidak terlalu besar yakni naik turun sebesar 0,1-0,3. Hal ini terjadi terjadi dikarenakan beban yang ada dirumah terbagi ke perangkat elektronik lainnya, jarak dari rumah ke sumber pln terlalu jauh dan berada diperkampungan sehingga sumber beban akan terbagi ke rumah-rumah. Jadi kesimpulannya, sensor tersebut masih dapat digunakan.

**4.7 Data *Monitoring* Perangkat Kombinasi**

Data kombinasi ini berguna untuk melihat perbandingan antara beban pada perangkat lampu dengan beban pada perangkat stop kontak yang digunakan.

Tabel 4.16 Data tegangan dengan beban kombinasi

Percobaan Jam ke-		Perangkat									
		STOP KONTAK 1 (Charger laptop 89,1 watt)		STOP KONTAK 2 (Lampu 7 watt)		STOP KONTAK 3 (Charger laptop 44,55 watt)		LAMPU 1 (11 watt)		LAMPU 2 (7 watt)	
		VA	VUK	VA	VUK	VA	VUK	VA	VUK	VA	VUK
Tegangan(V)	1	210,7	210,4	210,7	210,4	210,7	210,4	210,8	210,4	210,8	210,4
	2	210	209,7	210,1	209,7	210,1	209,7	210	209,7	210	209,7
	3	209,7	209,5	209,7	209,5	209,7	209,5	209,6	209,5	209,6	209,5
	4	209,9	209,7	209,9	209,7	210	209,7	209,9	209,7	209,9	209,7
	5	209,6	209,4	209,6	209,4	209,6	209,4	209,6	209,4	209,5	209,4
	6	210	209,7	210	209,7	210	209,7	210	209,7	210	209,7
	7	210,2	210	210,1	210	210,1	210	210,1	210	210,1	210
	8	210,3	210	210,3	210	210,3	210	210,3	210	210,3	210
Rata-Rata		210,05	209,8	210,05	209,8	210,06	209,9	210,03	209,8	210,02	209,8

Tabel 4.17 Data arus dengan beban kombinasi

Percobaan Jam ke-		Perangkat									
		STOP KONTAK 1 (Charger laptop 89,1 watt)		STOP KONTAK 2 (Lampu 7 watt)		STOP KONTAK 3 (Charger laptop 44,55 watt)		LAMPU 1 (11 watt)		LAMPU 2 (7 watt)	
		IA	IUK	IA	IUK	IA	IUK	IA	IUK	IA	IUK
Arus (A)	1	0,46	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	2	0,46	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	3	0,46	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	4	0,45	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	5	0,45	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	6	0,46	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	7	0,45	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
	8	0,46	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058
Rata-Rata		0,456	0,45	0,06	0,058	0,18	0,17	0,09	0,089	0,06	0,058

Tabel 4.18 Data daya dengan beban kombinasi

Percobaan Jam ke-	Perangkat										
	STOP KONTAK 1 (Charger laptop 89,1 watt)		STOP KONTAK 2 (Lampu 7 watt)		STOP KONTAK 3 (Charger laptop 44,55 watt)		LAMPU 1 (11 watt)		LAMPU 2 (7 watt)		
	WA	WUK	WA	WUK	WA	WUK	WA	WUK	WA	WUK	
Daya (W)	1	90	89,1	7,2	6,93	40,5	44,55	11,1	10,89	7,2	6,93
	2	90	89,1	7,2	6,93	41	44,55	11,1	10,89	7,2	6,93
	3	89,9	89,1	7	6,93	41	44,55	11,1	10,89	7,3	6,93
	4	89,6	89,1	7,1	6,93	40,3	44,55	11	10,89	7,1	6,93
	5	89,1	89,1	7,2	6,93	40,5	44,55	11,2	10,89	7,2	6,93
	6	89,6	89,1	7,1	6,93	40	44,55	11,1	10,89	7	6,93
	7	90	89,1	7,1	6,93	40,2	44,55	11,2	10,89	7	6,93
	8	89,8	89,1	7	6,93	40,5	44,55	11	10,89	7,2	6,93
Rata-Rata	89,75	89,1	7,11	6,93	40,5	44,55	11,1	10,89	7,15	6,93	

#### 4.8 Pengujian Aplikasi

Pengujian aplikasi “Sistem Kontrol dan Monitoring Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*” dilakukan untuk mengetahui apakah alat yang dibuat sesuai dengan fungsi yang diinginkan. Pengujian dilakukan pada setiap menu yang dibuat, yaitu menghidupkan atau mematikan lampu, *monitoring* arus, tegangan, daya dan biaya yang digunakan. Adapun untuk melakukan pengujian pada aplikasi yang dibuat dapat dilakukan dengan langkah-langkah sebagai berikut:

1. Hidupkan *hotspot* pada perangkat Android.
2. Tunggu hingga perangkat terhubung (*connected*) pada Android.
3. Buka aplikasi “PRAKDISA”.
4. Masukkan *id* dan juga *password* pengguna yang dapat dilihat pada gambar 3.6 diatas.
5. Setelah login, pengguna dapat memilih perangkat mana yang akan di kontrol dan dimonitoring pada gambar 3.5. Pada menu tersebut, pengguna dapat mengontrol dan memonitoring perangkat yang hendak digunakan.

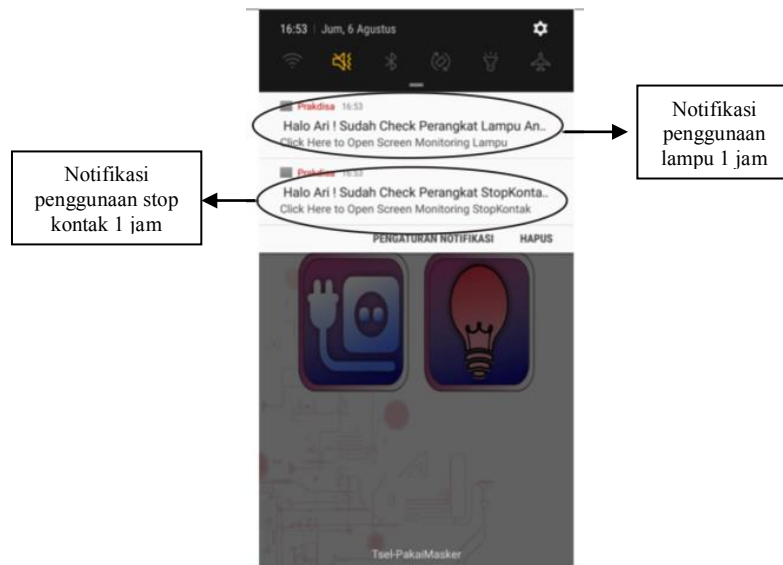


6. Jika pengguna ingin menambah perangkat, maka pengguna harus masuk ke menu *add* yang ditunjukkan pada gambar 3.9.
7. Jika pengguna ingin menghapus perangkat dapat masuk ke menu *edit* yang ditunjukkan pada gambar 3.10.
8. Setelah selesai menggunakan aplikasi, pengguna dapat logout dengan menekan garis tiga dibagian kiri atas, lalu pilih logout yang ditunjukkan pada gambar 4.26 dibawah ini.



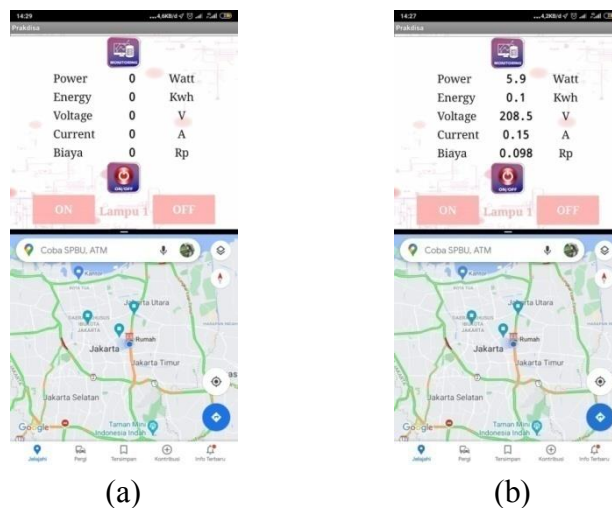
Gambar 4.27 Pilihan *logout*

Fiting lampu akan tetap menjalankan perintah yang telah diberikan oleh *user* melalui Android meskipun *user* keluar dari aplikasi “PRAKDISA”. Pada aplikasi ini telah disediakan indikator untuk mengetahui kondisi lampu pada saat ini. Indikator ini terdapat pada menu *on/off* pada aplikasi “PRAKDISA” dengan ciri berubah warna pada tulisan “Lampu 1” menjadi merah saat ditekan *on*, dan akan berubah warna menjadi pink saat ditekan *off*. Aplikasi tersebut juga menampilkan notifikasi pada Android jika penggunaan perangkat selama 1 jam. Berikut tampilan notifikasi pada aplikasi ditunjukkan pada gambar 4.27.



Gambar 4.28 Tampilan notifikasi aplikasi

Selain itu, tidak hanya pengujian dari jarak dekat saja aplikasi ini juga diuji oleh pengguna yang berada di beda pulau dengan perangkat. Berikut data-data responden yang diuji oleh pengguna yang berada di beda pulau ditunjukkan pada gambar dibawah ini.



Gambar 4.29 (a) Data responden lampu off (b) Data responden lampu on



(a)



(b)

Gambar 4. 30 (a) Data responden stop kontak *off* (b) Data responden stop kontak *on*

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan pengujian yang telah dilakukan pada masing-masing perangkat pada proyek akhir yang berjudul “Sistem Kontrol dan *Monitoring* Perangkat *Smart Home* Menggunakan IoT dengan Menerapkan Topologi *Star*” maka dapat ditarik kesimpulan sebagai berikut :

1. Data tegangan, arus, daya, dan biaya dapat ditampilkan pada aplikasi Android.
2. Semua perangkat dapat terhubung ke aplikasi secara bersamaan, namun memiliki waktu antrian 1-3 detik, tergantung koneksi *hotspot*.
3. Pengontrolan *on/off* perangkat pada fitting lampu dan stop kontak berfungsi dengan baik.
4. Pengujian perangkat dengan sistem IoT berfungsi dengan baik, yang mana perangkat dapat di akses dimana saja selagi memiliki jaringan internet.
5. Persentase perbandingan *error* daya pada fitting lampu 1 dengan beban 11 watt sebesar 6,29%, fitting lampu 2 dengan beban 7 watt sebesar 8,18%, stop kontak 1 dengan beban *charger* laptop 90 watt sebesar 2,98%, stop kontak 2 dengan beban lampu 7 watt sebesar 9,1% dan stop kontak 3 dengan beban *charger* laptop 45 watt sebesar 10,8% dikarenakan ukuran perbedaan daya saat pengambilan data dan pengukuran.

#### **5.2 Saran**

Apabila alat ini akan dikembangkan secara lanjut, maka perlunya beberapa saran agar alat berfungsi lebih baik lagi. Saran yang akan diberikan antara lain :

1. Membuat perangkat *hardware* fitting lampu dan stop kontak seminimalis mungkin.
2. Membuat aplikasi android yang lebih menarik dan multitasking.

3. Membuat *database online* khusus pengolahan data dan menyimpan data.
4. Perangkat *hardware* yang digunakan lebih banyak dengan variasi yang berbeda.
5. Dapat membatasi energi yang digunakan dalam satu hari.

## DAFTAR PUSTAKA

- [1] T. F. Yurnama and N. Azman, "Perancangan Software Aplikasi Pervasive Smart Home," in *Seminar Nasional Aplikasi Teknologi Informasi, Universitas Negeri Yogyakarta*, 2009, pp. E2–E5.
- [2] S. Astuti and M. Taufik, "Prototype Smarthome Dengan Konsep Internet of Thing (IoT) berbasis Website," Laporan Akhir Proyek Akhir, Politeknik Manufaktur Negeri Bangka Belitung, Sungailiat, 2018.
- [3] Y. Saputra and H. Alqadri, "Kontrol dan Monitoring Fiting Lampu," Laporan Akhir Proyek Akhir, Politeknik Manufaktur Negeri Bangka Belitung, Sungailiat, 2019.
- [4] A. Anki and Deswanto, "Integrasi Stop Kontak dan Fiting Lampu pada Smart Home menggunakan Smartphone," Laporan Akhir Proyek Akhir, Politeknik Manufaktur Negeri Bangka Belitung, Sungailiat, 2020.
- [5] M. Muslihudin, W. Renvilia, Taufiq, A. Andoyo, and F. Susanto, "Implementasi Aplikasi Rumah Pintar Berbasis Android Dengan Arduino Microcontroller," *J. Keteknikan dan Sains*, vol. 1, no. 1, pp. 23–31, 2018.
- [6] K. Asthon, "That ' Internet of Things ' Thing," *RFID J.*, 2010.
- [7] D. Bregman, "Smart home intelligence - The eHome that learns," *Int. J. Smart Home*, vol. 4, no. 4, pp. 35–46, 2010.
- [8] A. Isnaeni, "Rancang Bangun Smarhome Menggunakan Chat Bot Telegram Berbasis Arduino," Skripsi, Universitas Islam Negeri Alauddin Makassar, Makassar, 2018.
- [9] S. Halawa, "Perancangan Aplikasi Pembelajaran Topologi Jaringan Komputer Untuk Sekolah Menengah Kejuruan (SMK) Teknik Komputer Dan Jaringan (TKJ) Dengan Metode Computer Based Instruction," *JURIKOM (Jurnal Ris. Komputer)*, vol. 3, no. 1, pp. 66–71, 2016.
- [10] B. T. Cahya, "Keamanan Jaringan Dengan Packet Filtering Firewall (Studi KAasus: PT. Sukses Berkat Mandiri Jakarta)," *Katulistiwa Inform.*, vol. IV,

no. 2, pp. 37–39, 2016.

- [11] D. Rahmat Prasetya and N. Sukmawati, “Perencanaan dan Realisasi Prototipe Smart House System Menggunakan Smartphone Berbasis Android,” Tugas Akhir, Institut Teknologi Sepuluh November, Surabaya, 2016.
- [12] R. Indrianto, “Desain Dan Simulasi Monitoring Daya Sistem Tiga Fasa Menggunakan Matlab Simulink,” *J. Penelit.*, vol. 4, no. 4, pp. 52–60, 2020.
- [13] F. Simarsoit, “Sistem Pendeteksi pH Air Secara Wireless Menggunakan Wifi Berbasis NodeMCU,” Proyek Akhir, Universitas Sumatera Utara, Medan, 2020.
- [14] H. Silitonga, “Pengontrol Suhu Ruangan Otomatis Menggunakan NodeMCU V3 Lolin dan Sensor DHT11 Berbasis Internet,” Laporan Tugas Akhir, Universitas Sumatera Utara, Medan, 2019.
- [15] F. Nur and S. Setiawidayat, “Alat Monitoring Pemakaian Energi Listrik Berbasis Android Menggunakan Modul PZEM-004T,” vol. 01, no. 01, pp. 157–162, 2017.
- [16] S. Budi, “Power Supply,” vol. 1, no. July, pp. 1–4, 2019.
- [17] A. A’ruf, B. Cihur, N. Husna, and P. Srijayanti, “Power Supply,” Proyek Akhir, Universitas Negeri Medan, Medan, 2012.
- [18] M. Education, “Sejarah Android,” no. September, pp. 1–12, 2008.
- [19] I. Maulana Firman, “Penerapan Firebase Realtime Database pada Aplikasi E-Tilang,” *J. Resti (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1–5, no. 10, pp. 3–9, 2021.
- [20] I. K. G. Sudiartha, I. N. E. Indrayana, and I. W. Suasnawa, “Membangun Struktur Realtime Database Firebase Untuk Aplikasi Monitoring Pergerakan Group Wisatawan,” *Ilmu Komput.*, vol. XI, no. 2, pp. 1–7, 2017.
- [21] A. Kadir, “Pemrograman Arduino & Android Menggunakan Mit App Inventor,” in *PT Elex Media Komputino*, 2017, pp. 12–16.
- [22] F. Ardianto, B. Alfaresi, and R. A. Yuansyah, “Jaringan Hotspot Berbasis Mikrotik Menggunakan Metode Otentikasi Pengguna,” *J. Surya Energy*,

vol. 2, no. 2, p. 167, 2018.

- [23] R. Fakhruddin Lubis, S. Raharjo, and E. Sutanta, “Analisa Perbandingan Easyhotspot dan Mikrotik dalam Penerapan Hotspot Area dengan Sistem AAA,” *Jarkom*, vol. 1, no. 2, pp. 1–5, 2014.



# LAMPIRAN

# **LAMPIRAN 1**

## **Daftar Riwayat Hidup**

## DAFTAR RIWAYAT HIDUP

### I. Data Pribadi

Nama Lengkap : Ari Alamsyah  
Tempat, Tanggal Lahir : Sungailiat, 27 April 1999  
Alamat : Jalan Nelayan 2  
Telp : -  
HP : 082273782874



e-mail: wwindal80@gmail.com

Jenis kelamin : Laki-laki  
Agama : Islam

### II. Riwayat Pendidikan

Pendidikan	Tahun
SDN 1 SUNGAILIAT	2012
SMPN 2 SUNGAILIAT	2015
SMAN 1 SUNGAILIAT	2018

### III. Pendidikan Non Formal

-

Sungailiat, 6 Agustus 2021

## DAFTAR RIWAYAT HIDUP

### I. Data Pribadi

Nama Lengkap : Bulandari  
Tempat, Tanggal Lahir : Pemali, 10 Maret 2000  
Alamat : Jalan Limbang Jaya  
Telp : -  
HP : 0819 9352 3535



e-mail: bulan87878@gmail.com

Jenis kelamin : Perempuan

Agama : Islam

### II. Riwayat Pendidikan

Pendidikan	Tahun
SDN 22 KIMHIN	2011
SMPN 1 SUNGAILIAT	2014
SMKN 1 SUNGAILIAT	2017

### III. Pendidikan Non Formal

-

Sungailiat, 6 Agustus 2021

# **LAMPIRAN 2**

**PROGRAM NODEMCUESP8266**

## **PROGRAM *NodeMCUESP8266***

### **PROGRAM *ON/OFF***

```
void loop() {
  e=Firebase.getString("S3");
  Serial.println (e);
  if (e=="1"){                                // perintah yang diberikan
    Serial.println("Lampu MENYALA"); // menampilkan ke serial monitor
    digitalWrite(S3,HIGH); //Mengeluarkan perintah
    delay(5000);
  }
  else{
    Serial.println("Lampu MATI");
    digitalWrite(S3,LOW); //Device1 if OFF
  }
}
```

## **PROGRAM MONITORING**

```
{
float  voltage = pzem.voltage();
float  current = pzem.current();
float  power = pzem.power();
if (isnan(voltage)){
voltage = 0;
}
if (isnan(current) ) {
    current = 0;
}
if (isnan(power) ) {
    power = 0;
}
if (isnan(energy) ) {
    energy = 0;
}

float totalHarga = energy * harga_KWh;
Firebase.setString("volt4", String(voltage));
Firebase.setString("current4", String(current, 2));
Firebase.setFloat("watt4", power);
Firebase.setString("kwh4", String(energy, 1));
Firebase.setString("rp4", String(totalHarga, 3));
Serial.print("Voltage = ");
Serial.print(voltage);
Serial.println("v");
Serial.print("Current = ");
Serial.print(current, 2);
Serial.println("A");
Serial.print("Watt = ");
Serial.print(power);
Serial.println("watt");
Serial.print("KWh = ");
Serial.print(energy, 1);
Serial.println("KWh");
Serial.print("Harga = Rp ");
Serial.print(totalHarga, 3);}
```

# **LAMPIRAN 3**

**PROGRAM APLIKASI**



**PROGRAM APLIKASI**  
**PROGRAM ON/OFF STOP KONTAK**

```
when ON_BTN_SP1 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S1"
    valueToStore 1
  set Label1 . TextColor to [red]
```

```
when ON_BTN_SP3 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S3"
    valueToStore 1
  set Label3 . TextColor to [red]
```

```
when OFF_BTN_SP1 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S1"
    valueToStore 0
  set Label1 . TextColor to [pink]
```

```
when OFF_BTN_SP3 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S3"
    valueToStore 0
  set Label3 . TextColor to [pink]
```

```
when ON_BTN_SP2 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S2"
    valueToStore 1
  set Label2 . TextColor to [red]
```

```
when OFF_BTN_SP2 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S2"
    valueToStore 0
  set Label2 . TextColor to [pink]
```

## PROGRAM *ON/OFF* LAMPU

```
when ON_BTN_SP4 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S4"
    valueToStore 1
  set Label1 .TextColor to [red]
```

```
when ON_BTN_SP5 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S5"
    valueToStore 1
  set Label2 .TextColor to [red]
```

```
when OFF_BTN_SP4 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S4"
    valueToStore 0
  set Label1 .TextColor to [pink]
```

```
when OFF_BTN_SP5 .Click
do
  call FirebaseDB1 .StoreValue
    tag "S5"
    valueToStore 0
  set Label2 .TextColor to [pink]
```

## PROGRAM *MONITORING* STOP KONTAK

```
when Perangkat_StopKontak Initialize
do
  call FirebaseDatabase1 .GetValue
  tag current1
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag current2
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag current3
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag kwh1
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag kwh2
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag kwh3
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag rp1
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag rp2
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag rp3
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag volt1
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag volt2
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag volt3
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag watt1
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag watt2
  valueIfTagNotThere 0
  call FirebaseDatabase1 .GetValue
  tag watt3
  valueIfTagNotThere 0

when FirebaseDatabase1 .GoValue
do
  if
  then set Nisa_Current1 .Text to get value
  else if
  then set Nisa_C2 .Text to get value
  else if
  then set Nisa_C3 .Text to get value
  else if
  then set Nisa_Energy1 .Text to get value
  else if
  then set Nisa_E2 .Text to get value
  else if
  then set Nisa_E3 .Text to get value
  else if
  then set Nisa_Biaya1 .Text to get value
  else if
  then set Nisa_B2 .Text to get value
  else if
  then set Nisa_B3 .Text to get value
  else if
  then set Nisa_Voltage1 .Text to get value
  else if
  then set Nisa_V2 .Text to get value
  else if
  then set Nisa_V3 .Text to get value
  else if
  then set Nisa_Power1 .Text to get value
  else if
  then set Nisa_P2 .Text to get value
  else if
  then set Nisa_P3 .Text to get value

when FirebaseDatabase1 .DataChanged
do
  if
  then set Nisa_Current1 .Text to get value
  else if
  then set Nisa_C2 .Text to get value
  else if
  then set Nisa_C3 .Text to get value
  else if
  then set Nisa_Energy1 .Text to get value
  else if
  then set Nisa_E2 .Text to get value
  else if
  then set Nisa_E3 .Text to get value
  else if
  then set Nisa_Biaya1 .Text to get value
  else if
  then set Nisa_B2 .Text to get value
  else if
  then set Nisa_B3 .Text to get value
  else if
  then set Nisa_Voltage1 .Text to get value
  else if
  then set Nisa_V2 .Text to get value
  else if
  then set Nisa_V3 .Text to get value
  else if
  then set Nisa_Power1 .Text to get value
  else if
  then set Nisa_P2 .Text to get value
  else if
  then set Nisa_P3 .Text to get value
```

## PROGRAM *MONITORING* STOP KONTAK

```
when Perangkat_Lampu Initialize
do
  call FirebaseDatabase1.GetValue
  tag current4
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag current5
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag kwh4
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag kwh5
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag rp4
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag rp5
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag volt4
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag volt5
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag watt4
  valueIfTagNotThere 0
  call FirebaseDatabase1.GetValue
  tag watt5
  valueIfTagNotThere 0

when FirebaseDatabase1.GetValue
do
  if tag value
  then set Nilai_C4.Text to get value
  else if tag value
  then set Nilai_C5.Text to get value
  else if tag value
  then set Nilai_E4.Text to get value
  else if tag value
  then set Nilai_E5.Text to get value
  else if tag value
  then set Nilai_B4.Text to get value
  else if tag value
  then set Nilai_B5.Text to get value
  else if tag value
  then set Nilai_V4.Text to get value
  else if tag value
  then set Nilai_V5.Text to get value
  else if tag value
  then set Nilai_P4.Text to get value
  else if tag value
  then set Nilai_P5.Text to get value

when FirebaseDatabase1.DataChanged
do
  if tag value
  then set Nilai_C4.Text to get value
  else if tag value
  then set Nilai_C5.Text to get value
  else if tag value
  then set Nilai_E4.Text to get value
  else if tag value
  then set Nilai_E5.Text to get value
  else if tag value
  then set Nilai_B4.Text to get value
  else if tag value
  then set Nilai_B5.Text to get value
  else if tag value
  then set Nilai_V4.Text to get value
  else if tag value
  then set Nilai_V5.Text to get value
  else if tag value
  then set Nilai_P4.Text to get value
  else if tag value
  then set Nilai_P5.Text to get value
```

## PROGRAM MENU *EDIT*

```
initialize global Tampung_Database to create empty list

when Menu_Edit .Initialize
do call FirebaseDB1 .GetTagList

when FirebaseDB1 .TagList
value
do for each item in list get value
do call FirebaseDB1 .GetValue
tag get item
valueIfTagNotThere ""
call Clear

when ListView1 .AfterPicking
do call Tampilkan_Ditextbox

to Clear
do
set TextBox_NamaProject . Enabled to true
set TextBox_NamaProject . Text to ""
set TextBox_FirebaseHost . Text to ""
set TextBox_FirebaseAuth . Text to ""
call TextBox_NamaProject .RequestFocus
set BTN_OK . Enabled to true
set BTN_DELETE . Enabled to false
```

## PROGRAM MENU *LOGIN & SIGNUP*

```
when LoginBtn .Click
do
  call FirebaseDB1 .GetValue
  tag TextBox1 . Text
  valueIfTagNotThere " Not Value "

when SignUpBtn .Click
do
  call FirebaseDB1 .StoreValue
  tag TextBox1 . Text
  valueToStore PasswordTextBox1 . Text

when FirebaseDB1 .GotValue
  tag value
do
  if get tag = TextBox1 . Text
  then
    if get value = PasswordTextBox1 . Text
    then open another screen screenName " Menu_Utama "
    else set Label2 . Text to " ID & Password Salah "
```

The image displays three Scratch code blocks for a login and signup program. The first block, triggered by the 'LoginBtn' click, calls 'FirebaseDB1 .GetValue' with 'TextBox1 . Text' as the tag and 'Not Value' as the value if the tag is not found. The second block, triggered by the 'SignUpBtn' click, calls 'FirebaseDB1 .StoreValue' with 'TextBox1 . Text' as the tag and 'PasswordTextBox1 . Text' as the value to store. The third block, triggered by 'FirebaseDB1 .GotValue', checks if the tag is 'TextBox1 . Text'. If true, it checks if the value is 'PasswordTextBox1 . Text'. If both are true, it opens the 'Menu\_Utama' screen. If either is false, it sets 'Label2 . Text' to 'ID & Password Salah'.

## PROGRAM NOTIFIKASI

