



TUGAS AKHIR - IF184802

“*PROTOTYPE* SISTEM CERDAS UNTUK PENGAWASAN DAN PENGENDALIAN PENGUNAAN DAYA LISTRIK DI RUMAH TINGGAL”

FARRAS RABBANI
NRP 05111640000155

Dosen Pembimbing
Ir. Muchammad Husni, M.Kom.
Royyana Muslim Ijtihadie, S.Kom., M.Kom., PhD.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020



TUGAS AKHIR - IF184802

**“*PROTOTYPE* SISTEM CERDAS UNTUK
PENGAWASAN DAN PENGENDALIAN
PENGUNAAN DAYA LISTRIK DI RUMAH
TINGGAL”**

FARRAS RABBANI
NRP 05111640000155

Dosen Pembimbing
Ir. Muchammad Husni, M.Kom.
Royyana Muslim Ijtihadie, S.Kom., M.Kom., PhD.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Ceras
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]



UNDERGRADUATE THESIS - IF184802

**“*PROTOTYPE* SISTEM CERDAS UNTUK
PENGAWASAN DAN PENGENDALIAN
PENGUNAAN DAYA LISTRIK DI RUMAH
TINGGAL”**

FARRAS RABBANI
NRP 05111640000155

Supervisor
Ir. Muchammad Husni, M.Kom.
Royyana Muslim Ijtihadie, S.Kom., M.Kom., PhD.

DEPARTMENT OF INFORMATICS
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

Prototype Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

TUGAS AKHIR

**Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Arsitektur dan Jaringan Komputer
Program Studi S-1 Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember**

Oleh:

**Farras Rabbani
NRP: 05111640000155**

Disetujui oleh Pembimbing Tugas Akhir:

1. Ir. Muchammad Husni, M.Kom.

(NIP. 196002211984031001)

.....
(Pembimbing 1)

2. Royyana Muslim Ijtihadie, S.Kom.,

M.Kom., PhD.

(NIP. 197708242006041001)

.....
(Pembimbing 2)

**SURABAYA
JUNI, 2020**

[Halaman ini sengaja dikosongkan]

”PROTOTYPE SISTEM CERDAS UNTUK PENGAWASAN DAN PENGENDALIAN PENGGUNAAN DAYA LISTRIK DI RUMAH TINGGAL”

Nama Mahasiswa : FARRAS RABBANI
NRP : 05111640000155
Jurusan : Informatika ITS
Dosen Pembimbing I : Ir. Muchammad Husni, M.Kom.
Dosen Pembimbing II : Royyana Muslim Ijtihadie, S.Kom., M.Kom,
PhD.

Abstrak

Kebutuhan akan energi listrik setiap tahunnya terus meningkat. Untuk mengimbangi hal ini, harus dilakukan penghematan energi listrik dengan manajemen yang baik. Salah satu upaya adalah menerapkan sebuah sistem yang dapat mengawasi dan mengendalikan energi tersebut. Dengan menggunakan Relay dan pengendalian melalui Web maka proses pengendalian akan semakin mudah. Untuk melakukan pengendalian digunakan relay yang terhubung ke Microcontroller Wemod D1 Mini sebagai switching on-off. Proses switching ini memiliki dua buah kondisi. Kondisi pertama digunakan untuk mematikan listrik perangkat elektronik, dan kondisi dua adalah menyalakan listrik perangkat elektronik. Untuk pengawasan beban listrik dari perangkat elektronik berupa tegangan, arus, faktor daya, Kwh, dan biaya.

Saat ini di negara kita telah dikembangkan listrik Prabayar disamping listrik pasca bayar yang telah berjalan, fungsi dari listrik Prabayar ini adalah untuk dapat mengendalikan penggunaan dari listrik itu sendiri. Namun, penggunaan listrik Prabayar tidak sepenuhnya digunakan secara optimal oleh pelanggan, dikarenakan pelanggan tidak mengetahui beban dari alat listrik yang digunakan sehingga tidak dapat melakukan pengendalian. Dari permasalahan itu menggunakan sensor PZEM-004T untuk pengambilan data yang akan diukur.

Kata kunci: Teknologi IoT, Arduino IDE, MySQL

SMART SYSTEM PROTOTYPE FOR CONTROLLING AND MONITORING HOUSEHOLD ELECTRICITY USAGE

Name : FARRAS RABBANI
NRP : 05111640000155
Major : Informatika ITS
Supervisor I : Ir. Muchammad Husni, M.Kom.
Supervisor II : Royyana Muslim Ijtihadie, S.Kom, M.Kom., PhD.

Abstract

The need for electricity every years to increase. To compensate for this, electrical energy savings must be made with good management. One effort is to implement a system that can monitor and control the energy. By using Relay and control via the Web, the control process will be easier. To control the relay is used which is connected to the WeMos D1 Mini Microcontroller as on-off switching. This switching process has two conditions. The first condition is used to turn off the electricity of electronic devices, and the second condition is to turn on the electricity of electronic devices. For the supervision of electrical loads from electronic devices in the form of voltage, current, power factor, Kwh, and cost.

Currently in our country prepaid electricity has been developed besides running postpaid electricity, the function of prepaid electricity is to be able to control the use of electricity itself. However, the use of prepaid electricity is not fully used optimally by the customer, because the customer does not know the load of the electrical equipment used so that it cannot control. From this problem using the PZEM-004T for taking of data to be measured.

Keywords: IoT Technology, Arduino IDE, MySQL

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji dan syukur kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga dapat di selesaikan tugas akhir ini yang berjudul “*Prototype Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik Di Rumah Tinggal Berbasis Web*”.

Dalam pelaksanaan Tugas Akhir ini tentunya tidak dapat diselesaikan tanpa bantuan dari pihak lain. Tanpa mengurangi rasa hormat, diberikan penghargaan serta ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT dan Nabi Muhammad SAW.
2. Kedua Orang Tua yang sudah melahirkan dan membesarkan serta memberi dukungan hingga saat ini.
3. Bapak Ir.Muchammad Husni, M.Kom. selaku dosen wali dan dosen pembimbing Tugas Akhir pertama yang telah membimbing, memotivasi dan memberikan banyak masukan dalam pengerjaan Tugas Akhir ini.
4. Bapak Royyana Muslim Ijtihadie, S.Kom., M.Kom., PhD. selaku dosen pembimbing Tugas Akhir kedua yang selalu memberikan koreksi serta masukan-masukan yang dapat dikembangkan pada Tugas Akhir ini.
5. Bapak dan Ibu dosen Departemen Informatika ITS yang telah mengajarkan banyak ilmu berharga.
6. Firda Alvina Putri Rahayu sebagai penyemangat dalam pengerjaan Tugas Akhir ini.
7. Teman-teman IoT yang saling menyayangi satu sama lain.
8. Lor Squad yang selalu kompak dan solid dalam memberikan semangat untuk mengerjakan Tugas Akhir ini.
9. Teman- teman discord yang selama ini menemani untuk mengerjakan Tugas Akhir.

10. Bapak dan Ibu karyawan Departemen Informatika ITS atas berbagai bantuan yang telah diberikan selama masa perkuliahan.
11. Teman-teman satu angkatan Informatika ITS 2016 yang saling menyemangati satu sama lain.
12. Pihak-pihak lain yang tidak bisa penulis sebutkan satu per satu.

Diharapkan bahwa apa yang dihasilkan dari Tugas Akhir ini bisa memberikan manfaat bagi semua pihak, khususnya bagi diri sendiri dan seluruh *civitas academica* Informatika ITS, serta bagi agama, bangsa, dan negara. Tidak ada manusia yang sempurna sekalipun berusaha sebaik mungkin dalam menyelesaikan tugas akhir ini. Karena itu, mohon maaf apabila terdapat kesalahan, kekurangan, maupun kelalaian yang telah dilakukan. Kritik dan saran yang membangun sangat diharapkan untuk dapat disampaikan guna perbaikan selanjutnya.

Surabaya, Juni 2020

Farras Rabbani

DAFTAR ISI

Abstrak	vii
Abstract	viii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL	xix
DAFTAR KODE SUMBER	xxi
1 BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Metodologi	3
1.6. Sistematika Penulisan.....	5
2 BAB II TINJAUAN PUSTAKA	7
2.1. Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal	7
2.2. Hardware Tools	7
2.2.1. WeMos D1 Mini.....	7
2.2.2. Relay	8
2.2.3. PZEM-004T	9

2.2.4.	LCD (Liquid Cristal Display).....	10
2.2.5.	Terminal Block.....	10
2.2.6.	Jumper Cable	11
2.3.	Software Tools	12
2.3.1.	Arduino IDE	12
2.3.2.	Library	12
2.3.3.	Visual Studio Code.....	13
2.3.4.	phpMyAdmin.....	13
3	BAB III PERANCANGAN SISTEM.....	15
3.1.	Ilustrasi Rancangan Pengawasan dan Pengendalian Penggunaan Daya Listrik	15
3.1.1.	Alur Mengendalikan Arus 1 dan Arus 2.....	18
3.1.2.	Alur Memeriksa Arus 1 dan Arus 2.....	19
3.1.3.	Alur Mengukur Daya Listrik	20
3.2.	Perancangan <i>Database</i>	21
3.3.	Perancangan Alat.....	23
3.3.1.	Prototype.....	23
3.3.2.	Perancangan Sistem Monitor.....	24
3.3.3.	Deskripsi Kasus Pengguna Monitor Sistem	25
7.	32
4	BAB IV IMPLEMENTASI.....	39
4.1.	Implementasi Database	39
4.1.1.	Implementasi Tabel	39
4.2.	Implementasi Perangkat	40

4.2.1.	Rangkaian Alat Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal.....	40
4.2.2.	Implementasi Arduino	45
4.3.	PHP.....	54
4.3.1.	Implementasi Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal	55
5	BAB V UJI COBA DAN EVALUASI.....	63
5.1.	Lingkungan Uji Coba.....	63
5.2.	Skenario Pengujian.....	63
5.2.1.	Skenario Pengendalian Alat Pada Maket atau Miniatur	64
5.2.2.	Skenario Melihat Kondisi Saat Ini.....	69
5.2.3.	Skenario Pengujian Melihat Pengukuran Listrik	69
5.2.4.	Skenario Pengujian Melihat Data Biaya Per Hari	70
5.2.5.	Skenario Pengujian Melihat Data Biaya Per Bulan	70
5.2.6.	Skenario Pengujian Melihat Grafik	71
5.2.7.	Skenario Pengujian Melihat Notifikasi.....	71
5.2.8.	Skenario Mengukur Dengan Avo Meter dan Tampilan di LCD	72
5.2.9.	Skenario Uji Coba Menghitung Dengan PZEM-004T	73

5.2.10.	Skenario Uji Coba Perbandingan Akurasi Lampu 3 Watt.....	76
5.2.11.	Skenario Uji Coba Perbandingan Akurasi Kipas Angin 20 Watt.....	76
5.2.12.	Skenario Uji Coba Perbandingan Akurasi Lampu 25 Watt.....	77
5.2.13.	Skenario Uji Coba Perbandingan Akurasi Lampu 100 Watt.....	78
6	BAB VI KESIMPULAN DAN SARAN.....	81
6.1.	Kesimpulan.....	81
6.2.	Saran.....	81
	DAFTAR PUSTAKA.....	83
	LAMPIRAN.....	85
	BIODATA PENULIS.....	93

DAFTAR GAMBAR

Gambar 2.1 <i>WeMos D1 Mini</i>	8
Gambar 2.2 Relay	9
Gambar 2.3 PZEM-004T [3].....	9
Gambar 2.4 LCD (Liquid Cristal Display)	10
Gambar 2.5 Terminal Block	11
Gambar 3.1 Ilustrasi Rancangan Penggunaan Daya Listrik di Rumah Tinggal	15
Gambar 3.2 Desain Rangkaian Implementasi.....	16
Gambar 3.3 Flowchart <i>Prototype</i> Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal.....	17
Gambar 3.4 Flowchart Mengendalikan Arus 1 dan Arus 2	18
Gambar 3.5 Flowchart Memeriksa Arus 1 dan Arus 2	19
Gambar 3.6 Flowchart Mengukur Daya Listrik.....	20
Gambar 3.7 <i>Use Case</i> Diagram Monitor Sistem	24
Gambar 3.8 Diagram Aktivitas Pengendalian Arus 1 dan Arus 2.....	27
Gambar 3.9 Diagram Aktivitas Melihat Kondisi Saat Ini..	29
Gambar 3.10 Diagram Aktivitas Melihat Tegangan, Arus, faktor daya, dan <i>watt second</i> (ws).....	31
Gambar 3.11 Diagram Aktivitas Melihat Biaya Per Har.	33
Gambar 3.12 Diagram Aktivitas Melihat Biaya Per Bulan	35
Gambar 3.13 Diagram Aktivitas Grafik Penggunaan Daya Listrik.....	37
Gambar 4.1 Rangkaian WeMos D1 Mini dengan Relay ...	41
Gambar 4.2 Rangkaian Wemos D1 Mini dengan LCD	42
Gambar 4.3 Rangkaian Wemos D1 Mini dengan PZEM-004T	43

Gambar 4.4 Rangkaian <i>Prototype</i> Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal.....	45
Gambar 4.5 Konfigurasi Database	54
Gambar 4.6 Fungsi Melakukan Pengendalian Arus 1 dan Arus 2.....	56
Gambar 4.7 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Mati dan Arus 2 Mati.....	57
Gambar 4.8 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Mati dan Arus 2 Nyala.....	58
Gambar 4.9 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Nyala dan Arus 2 Mati.....	59
Gambar 4.10 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Nyala dan Arus 2 Mati.....	60
Gambar 4.11 Fungsi Mengambil Data Kwh	61
Gambar 4.12 Fungsi Menerima Data Arduino	61
Gambar 4.13 Implementasi Menghitung Tegangan, Arus, Faktor Daya, dan <i>Watt Second (WS)</i> dan Biaya.....	62
Gambar 4.14 Fungsi Memasukkan Data.....	62
Gambar 5.1 Pengendalian Menyalakan Arus 1 dan Arus 2	64
Gambar 5.2 Pengendalian Mematikan Arus 1 dan Arus 2	64
Gambar 5.3 Arduino Serial Monitor Arus 1 Mati.....	65
Gambar 5.4 Keadaan Lampu Mati.....	65
Gambar 5.5 Arduino Serial Monitor Arus 1 Nyala.....	66
Gambar 5.6 Keadaan Lampu Nyala.....	66
Gambar 5.7 Arduino Serial Monitor Arus 2 Mati.....	67
Gambar 5.8 Keadaan Kipas Angin Mati.....	67
Gambar 5.9 Arduino Serial Monitor Arus 2 Nyala.....	68
Gambar 5.10 Keadaan Kipas Angin Menyala	68
Gambar 5.11 Kondisi Arus 1 dan Arus 2 Menyala.....	69
Gambar 5.12 Kondisi Arus 1 dan Arus 2 Mati	69

Gambar 5.13 Halaman Pengukuran Listrik	69
Gambar 5.14 Halaman Data Biaya Per Hari	70
Gambar 5.15 Halaman Data Biaya Per Bulan	71
Gambar 5.16 Halaman Grafik Penggunaan Daya Listrik ..	71
Gambar 5.17 Halaman Notifikasi	72
Gambar 5.18 Avo Meter Mengukur Tegangan.....	72
Gambar 5.19 Tampilan di LCD	73

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 3.1 Deskripsi Tabel <i>Database</i>	21
Table 3.2 Detail Tabel <i>data_kwh</i>	21
Table 3.3 Detail Tabel <i>Biaya per_hari</i>	22
Table 3.4 Tabel Deskripsi Kasus Penggunaan Sistem Monitor	25
Table 3.5 <i>Use Case</i> Skenario Melakukan Pengendalian Arus 1 dan Arus 2	25
Table 3.6 <i>Use Case</i> Skenario Melihat Kondisi Saat Ini.....	28
Table 3.7 <i>Use Case</i> Skenario Melihat Tegangan, faktor daya, dan <i>watt second (ws)</i>	30
Table 3.8 <i>Use Case</i> Skenario Melihat Biaya Per Hari.....	32
Table 3.9 <i>Use Case</i> Skenario Melihat Biaya Per Bulan.....	33
Table 3.10 <i>Use Case</i> Skenario Grafik Penggunaan Daya Listrik.....	36
Table 4.1 Rangkaian WeMos D1 Mini dengan Relay	41
Table 4.2 Rangkaian Wemos D1 Mini dengan LCD.....	42
Table 4.3 Rangkaian WeMos D1 Mini dengan PZEM-004T	44
Table 5.1 Lingkup Spesifikasi	63
Table 5.2 Hasil Pembacaan Sensor PZEM-004T Lampu 3 Watt.....	73
Table 5.3 Hasil Pembacaan Sensor PZEM-004T Kipas Angin 20 Watt.....	74
Table 5.4 Hasil Pembacaan Sensor PZEM-004T Lampu 25 Watt.....	75
Table 5.5 Hasil Pembacaan Sensor PZEM-004T Lampu 100 Watt.....	75
Table 5.6 Uji Coba Perbandingan Akurasi Lampu 3 Watt.....	76

Table 5.7 Uji Coba Perbandingan Akurasi Kipas Angin 20 Watt.....	77
Table 5.8 Uji Coba Perbandingan Akurasi Lampu 100 Watt	79

DAFTAR KODE SUMBER

Kode Sumber 4.1 MySql data_kwh	39
Kode Sumber 4.2 MySql biaya_hari.....	40
Kode Sumber 4.3 <i>Library</i> dan Konfigurasi <i>WiFi</i>	46
Kode Sumber 4.4 Deklarasi Pin.....	46
Kode Sumber 4.5 Implementasi Membaca File Json.....	47
Kode Sumber 4.6 Setup LCD	47
Kode Sumber 4.7 Fungsi Connect WiFi.....	48
Kode Sumber 4.8 Set Relay	48
Kode Sumber 4.9 Fungsi <i>Loop Connect Server</i>	49
Kode Sumber 4.10 Implementasi Lampu dan Kipas	50
Kode Sumber 4.11 Fungsi Menghitung Daya.....	50
Kode Sumber 4.12 Set <i>Cursor LCD</i>	51
Kode Sumber 4.13 Menampilkan Tegangan, Arus, Faktor Daya, dan <i>Watt Second (Ws)</i>	52
Kode Sumber 4.14 Menampilkan Data di Serial Monitor .	53
Kode Sumber 4.15 Fungsi Menghitung Voltage dan Kwh	53
Kode Sumber 4.16 Fungsi Menyiapkan Data ke Database	54
Kode Sumber 4.17 Fungsi httpCode.....	54

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Bab pendahuluan membahas garis besar penyusunan tugas akhir yang meliputi latar belakang, tujuan pembuatan, rumusan dan batasan permasalahan, metodologi penyusunan Tugas Akhir, dan sistematika penulisan.

1.1. Latar Belakang

Listrik merupakan kebutuhan manusia yang sangat penting dalam kehidupan sehari-hari, sehingga sangat berpengaruh terhadap kehidupan manusia saat ini. Dapat dikatakan bahwa listrik telah menjadi sumber energi utama dalam setiap kegiatan manusia. Seiring dengan perkembangan teknologi, kebutuhan akan sumber daya listrik juga semakin tinggi. Hal ini disebabkan oleh banyaknya penggunaan sumber daya listrik seperti lampu, kipas angin, AC dan lain-lainnya.

Namun, pemanfaatan sumber daya listrik tidak dapat dilakukan dengan bebas tanpa kendali karena berpengaruh terhadap ketersediaan sumber daya listrik itu sendiri. Untuk menghemat penggunaan sumber energi atau sumber daya, terutama sumber daya listrik, pemerintah dengan gencar mensosialisasikan kepada masyarakat untuk menggunakan listrik secara efektif dan juga dengan menaikkan tarif dasar listrik secara berskala. Sedangkan, upaya yang dilakukan masyarakat agar pemakaian listrik lebih efektif dan efisien, biasanya hanya terbatas pada mematikan peralatan elektronik yang tidak digunakan. Langkah ini dinilai dapat menghemat pemakaian listrik, tetapi masyarakat lebih memilih untuk mencari solusi praktis dalam mengatasi keterbatasan sumber daya listrik.

Untuk itu, perlu adanya sistem yang dapat membantu dalam mengendalikan dan memeriksa penggunaan sumber daya listrik, dan juga memeriksa daya arus listrik agar tahu pemakaian listrik di rumah tinggal dan tahu berapa biaya penggunaan rumah di

listrik perbulannya, sehingga penggunaan daya listrik dapat menjadi lebih efektif. Sistem tersebut dapat direalisasikan dengan mengimplementasikan prototype sistem cerdas untuk penggunaan sumber daya listrik.

1.2. Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut.

1. Bagaimana cara mengendalikan penggunaan peralatan elektronik dengan menggunakan konsep IoT?
2. Bagaimana cara mengetahui peralatan elektronik yang aktif dan tidak aktif di rumah tinggal?
3. Bagaimana cara mengetahui konsumsi listrik sehari-hari pada rumah tinggal? Sehingga bisa diprediksi biaya penggunaan listrik pada bulan berjalan.

1.3. Batasan Masalah

Permasalahan yang dibahas dalam Tugas Akhir memiliki beberapa batasan antara lain:

1. Sistem yang dibangun berupa *prototype*.
2. Sistem pengaturan berbasis *web*.
3. Untuk mengakses sistem ini memerlukan akses internet.
4. Sistem ini di aplikasikan pada pelanggan PLN pasca bayar.
5. Sistem cerdas ini dibatasi kemampuan sistem manajemen waktu.

1.4. Tujuan

Tujuan dari pengerjaan Tugas Akhir ini yaitu:

1. Sistem yang dibuat dapat digunakan untuk memeriksa peralatan elektronik.
2. Sistem yang dibuat dapat digunakan untuk mengendalikan peralatan elektronik.

3. Sistem yang dibuat dapat digunakan untuk memeriksa pemakaian peralatan elektronik dan memeriksa biaya pada bulan berjalan.

1.5. Metodologi

Tahap yang dilakukan untuk menyelesaikan Tugas Akhir ini adalah sebagai berikut:

1. Penyusunan Proposal Tugas Akhir

Proposal Tugas Akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan Tugas Akhir, rumusan masalah yang diangkat, batasan masalah untuk Tugas Akhir, tujuan dari pembuatan Tugas Akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan Tugas Akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan Tugas Akhir mulai dari penyusunan proposal hingga penyusunan buku Tugas Akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan Tugas Akhir.

2. Studi Literatur

Pada tahap ini, akan dipelajari beberapa jumlah referensi yang akan diperlukan untuk pengerjaan Tugas Akhir, yaitu Arduino. Informasi didapatkan dari buku, paper, jurnal dan materi-materi kuliah yang berhubungan dengan topik Tugas Akhir.

3. Implementasi

Perangkat yang akan digunakan kali ini diperuntukkan untuk membuat "*Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal". Perangkat yang digunakan adalah relay, Arduino esp8266, PZEM-004T

4. Uji Coba dan Evaluasi

Proses pengujian dilakukan dengan melalui dua cara yaitu:

1. Pengujian *hardware*

Pengujian ini dilakukan untuk menguji apakah “Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal” dengan menggunakan fungsional dari perangkat keras.

2. Pengujian *software*

Pengujian ini dilakukan untuk menguji apakah “Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal” dengan menggunakan fungsional dari perangkat lunak.

5. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam Tugas Akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku Tugas Akhir secara garis besar antara lain:

1. Pendahuluan
 - a. Latar Belakang
 - b. Rumusan Masalah
 - c. Batasan Tugas Akhir
 - d. Tujuan
 - e. Metodologi
 - f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

1.6. Sistematika Penulisan

Buku Tugas Akhir ini terdiri atas beberapa bab yang tersusun secara sistematis, yaitu sebagai berikut.

1. Bab I. Pendahuluan

Bab pendahuluan berisi penjelasan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan Tugas Akhir.

2. Bab II. Tinjauan Pustaka

Bab tinjauan pustaka berisi penjelasan mengenai dasar teori yang mendukung pengerjaan Tugas Akhir.

3. Bab III. Analisis dan Perancangan

Bab ini berisi tentang desain sistem, rancangan basis data, diagram kasus penggunaan, diagram aktivitas dan rancangan antarmuka pengguna.

4. Bab IV. Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa tampilan antarmuka yang telah dibuat dan dapat berfungsi untuk mengakomodir kebutuhan fungsional yang ada.

5. Bab V. Uji Coba dan Evaluasi

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

6. Bab VI. Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak berbasis IoT ke depannya.

[Halaman ini sengaja dikosongkan]

BAB II

TINJAUAN PUSTAKA

Bab tinjauan pustaka berisi mengenai penjelasan teori yang berkaitan dengan implementasi perangkat lunak. Penjelasan tersebut bertujuan untuk memberikan gambaran mengenai sistem yang akan dibangun dan berguna sebagai pendukung dalam pengembangan perangkat lunak.

2.1. Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

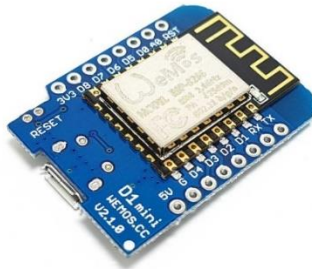
Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal merupakan salah satu solusi untuk mengatasi masalah energi dan bahkan sangat tepat digunakan di rumah tinggal, karena cara ini dapat meningkatkan stabilitas, keandalan, dan meningkatkan efisiensi.

Oleh karena itu diperlukan sebuah sistem yang mampu memantau perancangan dan implementasi *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal. Ini adalah *prototype* yang dirancang untuk membantu pengguna di rumah tinggal. Yang di antaranya dapat menyalakan atau mematikan peralatan elektronik melalui *website*, memantau kondisi arus listrik menyala atau mati, dan memantau total Kwh dan biaya perbulannya.

2.2. Hardware Tools

2.2.1. WeMos D1 Mini

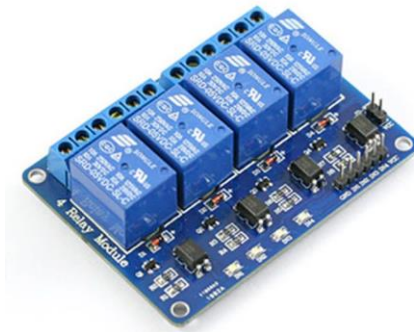
WeMos D1 Mini merupakan *module development board* yang berbasis *WiFi* dari keluarga ESP8266 yang dimana dapat diprogram menggunakan *software* IDE Arduino seperti halnya dengan NodeMCU, perangkat inilah yang nantinya akan menghubungkan perangkat dengan melalui *WiFi*. [1]



Gambar 2.1 *WeMos D1 Mini*

2.2.2. Relay

Relay merupakan suatu peranti yang bekerja berdasarkan elektromagnetik untuk menggerakkan sejumlah kontaktor yang tersusun atau sebuah saklar elektronis yang dapat dikendalikan dari rangkaian elektronik lainnya dengan memanfaatkan tenaga listrik sebagai sumber energinya. Kontaktor akan tertutup (menyala) atau terbuka (mati) karena efek induksi magnet yang dihasilkan kumparan (induktor) ketika dialiri arus listrik. Berbeda dengan saklar, pergerakan kontaktor (*on* atau *off*) dilakukan manual tanpa perlu arus listrik. [2]



Gambar 2.2 Relay

2.2.3. PZEM-004T

PZEM-004T adalah sensor yang dapat mengukur arus, tegangan, *power*, dan energi dari listrik AC. Sensor ini mengeluarkan *output* dengan komunikasi serial. Jika kita ingin menghubungkan dengan Arduino maka komunikasi yang digunakan adalah komunikasi serial. [3]



Gambar 2.3 PZEM-004T [3]

2.2.4. LCD (Liquid Cristal Display)

LCD (*Liquid Cristal Display*) berfungsi untuk menampilkan karakter angka, huruf ataupun simbol dengan lebih baik dan dengan konsumsi arus yang rendah. LCD (*Liquid Cristal Display*) dot matrik M1632 merupakan modul LCD buatan Hitachi. Modul LCD (*Liquid Cristal Display*) dot matrik M1632 terdiri dari bagian penampil karakter (LCD) yang berfungsi menampilkan karakter dan bagian sistem prosesor LCD dalam bentuk modul dengan mikrokontroller yang diletakkan dibagian belakang LCD tersebut yang berfungsi untuk mengatur tampilan LCD serta mengatur komunikasi antara LCD dengan mikrokontroller yang menggunakan modul LCD tersebut. [4]



Gambar 2.4 LCD (Liquid Cristal Display)

2.2.5. Terminal Block

Terminal Block adalah satu set dari dua atau lebih titik koneksi sekrup serupa. Anda bisa menghubungkan kabel pada titik-titik ini. Strip terminal menggabungkan banyak blok serupa dalam satu perangkat. Di sebuah strip, blok-blok itu saling terisolasi satu sama lain.



Gambar 2.5 Terminal Block

2.2.6. Jumper Cable

Kabel jumper adalah kabel yang digunakan untuk menghubungkan Arduino dengan sensor atau modul lainnya, terdapat tiga jenis kabel *jumper* yaitu *Female to Female*, *Male to Male*, dan *Female to Male*. Dimana masing-masing memiliki fungsi yang sama hanya berbeda pada ujung kabelnya saja.



Gambar 2.6 Jumper Cable

2.3. Software Tools

2.3.1. Arduino IDE

Arduino IDE merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Melalui *software* inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (*Sketch*) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler* Arduino dengan mikrokontroler. [5]

2.3.2. Library

Dalam pembuatan Tugas Akhir ini digunakan beberapa *library* untuk mendukung berjalannya pemrograman pada Arduino, beberapa *library* yang digunakan adalah:

Nama Library	Deskripsi
PZEM004Tv30.h	<i>Library</i> yang memungkinkan Arduino untuk menghitung arus listrik berupa tegangan, arus, dan faktor daya.
LiquidCrystal_I2C.h	<i>Library</i> yang digunakan untuk menampilkan angka, huruf ataupun simbol.
Arduino.Json.h	<i>Library</i> yang memungkinkan untuk menjalankan program Json.
ESP8266WiFi.h	<i>Library</i> yang digunakan untuk menghubungkan ESP8266 pada Arduino dengan <i>internet</i> .

ESP8266HTTPClient.h WiFiClient.h	<i>Library</i> yang memungkinkan untuk ESP8266 pada Arduino mengakses sebuah <i>Hypertext Transfer Protocol (HTTP)</i> .
-------------------------------------	--

2.3.3. Visual Studio Code

Visual Studio Code adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung Bahasa pemrograman JavaScript, Typescript, dan Node.js, serta Bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace* Visual Studio Code (seperti C++, C#, Python, Go, Java, dst). [6]

2.3.4. phpMyAdmin

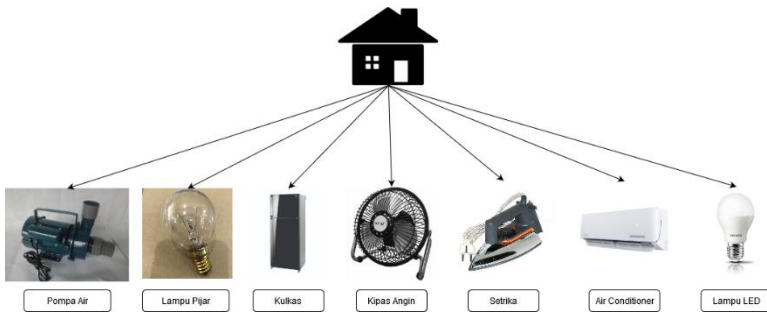
phpMyAdmin adalah aplikasi *web* untuk mengelola *database* MySQL dan database MariaDB dengan lebih mudah melalui antarmuka (*interface*) grafis. Aplikasi *web* ini ditulis menggunakan Bahasa pemrograman PHP. Sebagaimana aplikasi-aplikasi lain untuk lingkungan *web* (aplikasi yang dibuka atau dijalankan menggunakan *browser*), phpMyAdmin ditunjukkan untuk memudahkan pengolahan basis data MySQL dan MariaDB dengan penyajian antarmuka *web* yang lengkap dan menarik. [7]

[Halaman ini sengaja dikosongkan]

BAB III PERANCANGAN SISTEM

Bab ini menjelaskan tentang perancangan dan pengimplementasian *prototype* sistem cerdas penggunaan daya listrik yang dapat mengendalikan lampu dan kipas angin dengan menyalakan dan mematikannya, dapat memeriksa keadaan saat ini lampu dan kipas angin apakah menyala atau sedang mati, dan mengukur listrik agar mengetahui biaya per harinya berapa.

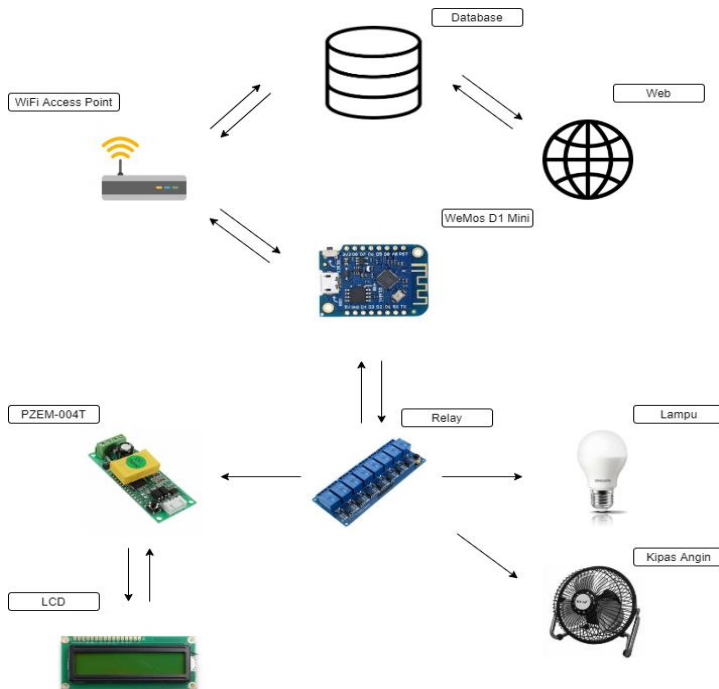
3.1. Ilustrasi Rancangan Pengawasan dan Pengendalian Penggunaan Daya Listrik



Gambar 3.1 Ilustrasi Rancangan Penggunaan Daya Listrik di Rumah Tinggal

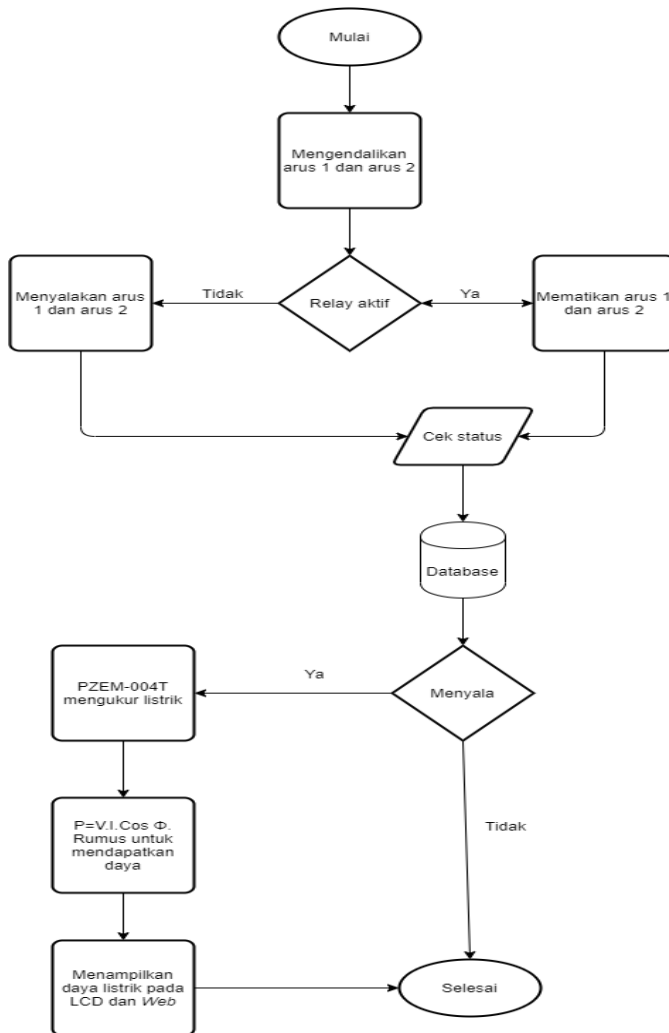
Pada gambar 3.1 adalah ilustrasi dari rancangan pengendalian dan pengawasan penggunaan sumber daya listrik pada rumah tinggal, namun pada rancangan implementasi untuk *prototype* pada Tugas Akhir ini. Penggunaan sumber daya listrik di rumah tinggal termasuk penggunaan pompa air, lampu pijar, kulkas, kipas angin, setrika, *Air Conditioner*, lampu LED dan lain-lain. Namun implemementasi pada rancangan untuk *prototype* yang dibuat menggunakan penggunaan lampu, dan kipas angin untuk mewakili motor listrik. *Flow* rangkaian akan terlihat sebagai berikut, perangkat-perangkat ini akan dihubungkan ke relay, kemudian

tegangan, arus dan faktor daya yang dihitung oleh *microcontroller*, dimana dayanya akan dihitung oleh WeMos D1 Mini. Dan WeMos D1 Mini mengirim daya ke *database* lalu ditampilkan di *web*. Lalu pada *web* dapat mengontrol perangkat dimana *microcontroller* membaca status perangkat pada *database*.



Gambar 3.2 Desain Rangkaian Implementasi

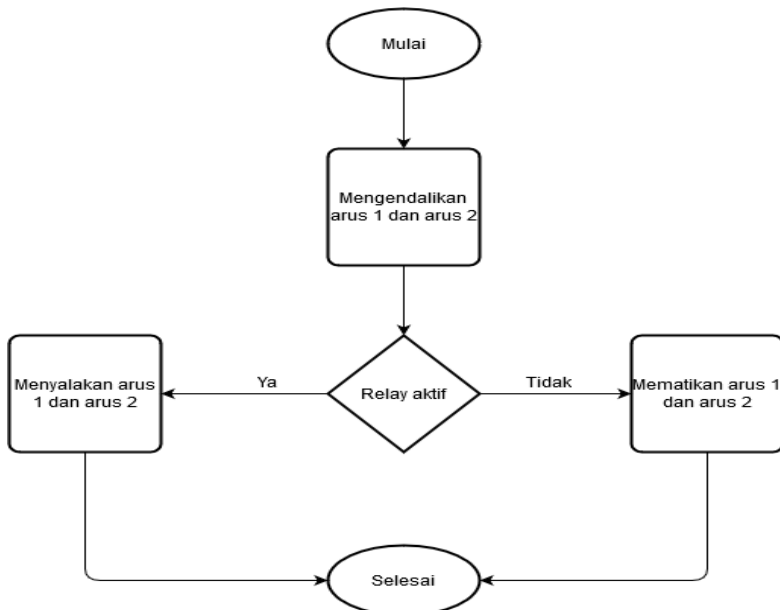
Proses Aplikasi *Prototype* Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal dapat dilihat pada gambar 3.2



Gambar 3.3 Flowchart *Prototype* Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

3.1.1. Alur Mengendalikan Arus 1 dan Arus 2

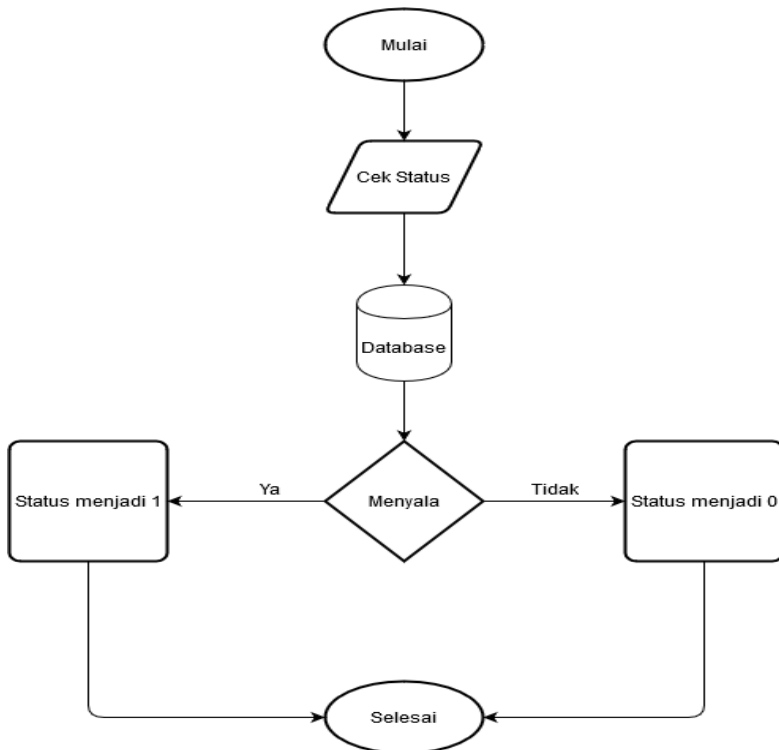
Proses mengendalikan mengaktifkan arus 1 dan arus 2, menon-aktifkan arus 1 dan arus 2 dapat dilihat pada gambar *flowchart* 3.3. Ketika sistem menyala, maka sistem langsung melihat status dari masing-masing *device*. Jika *user* menginginkan *device* tersebut untuk tidak aktif maka *user* menekan tombol tidak aktif, maka *user* tinggal menekan tombol tidak aktif (*off*) pada *device*. Sedangkan jika *user* menginginkan *device* tersebut untuk aktif maka *user* menekan tombol aktif (*on*) pada *device* yang memiliki status 0.



Gambar 3.4 Flowchart Mengendalikan Arus 1 dan Arus 2

3.1.2. Alur Memeriksa Arus 1 dan Arus 2

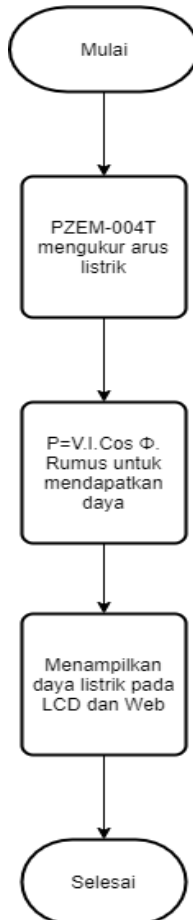
Proses memeriksa arus 1 dan arus 2 dapat dilihat pada gambar flowchart 3.4. Ketika sistem menyala, maka sistem langsung melihat status dari masing-masing *device*. Jika 1 maka *device* tersebut sedang aktif, sedangkan jika 0 maka *device* tersebut tidak aktif.



Gambar 3.5 Flowchart Memeriksa Arus 1 dan Arus 2

3.1.3. Alur Mengukur Daya Listrik

Proses mengukur listrik ini menggunakan sensor PZEM-004T. Dan untuk mencari daya digunakan rumus $P=V.I$. Setelah mendapatkan daya maka ditampilkan di LCD dan *Website*.



Gambar 3.6 Flowchart Mengukur Daya Listrik

3.2. Perancangan *Database*

Rancangan *Database* ini digunakan sebagai media media penyimpanan data oleh Arduino yang nantinya data akan berubah sesuai data yang dikirim oleh Arduino, yang akan ditampilkan melalui *Website*. *Database* akan berisi dua tabel yaitu:

Tabel 3.1 Deskripsi Tabel *Database*

Nama Tabel	Deskripsi
data_kwh	Tabel yang berisi tentang data waktu, tegangan, arus, faktor daya, watt second (ws)
biaya_hari	Tabel yang berisi hubungan antara data tanggal, data total Kwh, dan data biaya per hari

Table 3.2 Detail Tabel data_kwh

data_kwh		
Atribut	Data Type	Deskripsi
No	Bigint(20)	Atribut yang berisi nomer urut.
Tanggal	Date	Atribut yang berisi tanggal
Waktu	Time	Atribut yang berisi waktu .
Tegangan	Float	Atribut yang berisi tegangan dari PLN.
Arus1	Float	Atribut yang berisi arus 1 listrik.

Pf1	Float	Atribut yang berisi faktor daya.
Energy1	Float	Atribut yang berisi daya.
Arus2	Float	Atribut yang berisi arus 2 listrik.
Pf2	Float	Atribut yang berisi faktor daya.
Energy2	Float	Atribut yang berisi daya.
Kwh	Float	Atribut yang berisi Kwh.
Kwh_row	Float	Atribut yang berisi Kwh_row
Biaya	Float	Atribut yang berisi jumlah biaya per harinya

Table 3.3 Detail Tabel Biaya per_hari

biaya_hari		
Atribut	Data Type	Deskripsi
Id	int(11)	Atribut <i>primary key</i> pada table ini yang berfungsi sebagai ID <i>primary</i> pada tabel.
Tanggal	Date	Atribut nomor kartu pengguna.
Biaya	Time	Atribut yang berisi nama pengguna.

3.3. Perancangan Alat

3.3.1. Prototype

Perancangan *prototype* pada Tugas Akhir ini akan menggunakan beberapa komponen *hardware*, dengan WeMos D1 Mini sebagai *microcontroller*, di antaranya :

1. 8-Channel Relay 5V
2. Sensor PZEM-004T 30v
3. LCD (Liquid Cristal Display)
4. Terminal block
5. Stop kontak
6. Lampu
7. Kipas angin

Microcontroller akan dihubungkan dengan masing-masing komponen melalui 8-Channel Relay 5V dengan mengirimkan arus listrik kepada relay, dimana *microcontroller* akan mengirimkan sinyal *High* dan *Low* kepada relay.

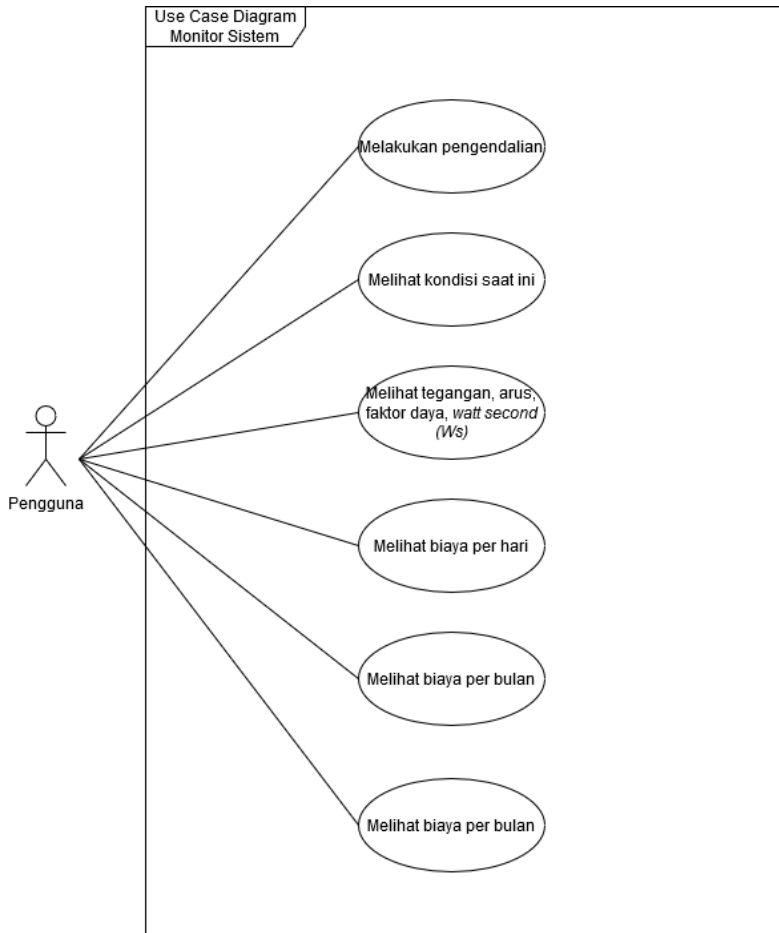
Ketika *microcontroller* mengirimkan sinyal *LOW* maka relay akan aktif dan saklar akan tertutup sehingga terminal *Normally Open* akan mendapatkan arus, sedangkan saat *microcontroller* mengirimkan sinyal *HIGH* maka relay akan mati dan saklar akan terbuka sehingga terminal *Normally Closed* akan mendapatkan arus.

Microcontroller yang terhubung pada 8-channel Relay 5V akan berfungsi sebagai berikut:

1. Mematikan lampu dengan mengirimkan sinyal *HIGH* saat status_lampu = 0 dan menyalakan lampu dengan mengirimkan sinyal *LOW* saat status_lampu = 1 kepada relay.
2. Mematikan kipas angin dengan mengirimkan sinyal *HIGH* saat status kipas_angin = 0 dan menyalakan lampu dengan mengirimkan sinyal *LOW* saat status_kipas = 1 kepada relay

3.3.2. Perancangan Sistem Monitor

Perancangan sistem monitor digambarkan melalui *Use Case Diagram* pada Gambar 3.5 *Use Case Diagram* Sistem Monitor.



Gambar 3.7 Use Case Diagram Monitor Sistem

3.3.3. Deskripsi Kasus Pengguna Monitor Sistem

Deskripsi kasus penggunaan yang dibutuhkan pada Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal ini disesuaikan dengan diagram kasus penggunaan sistem monitor. Deskripsi kasus penggunaan Arduino dapat dilihat pada tabel

Table 3.4 Tabel Deskripsi Kasus Penggunaan Sistem Monitor

Kode Kasus Penggunaan	Nama Kasus Penggunaan
UC-001	Melakukan pengendalian arus 1 dan arus 2
UC-002	Melihat kondisi saat ini
UC-003	Melihat tegangan, arus, faktor daya, <i>watt second</i> (ws)
UC-004	Melihat biaya per hari
UC-005	Melihat biaya per bulan
UC-006	Grafik penggunaan daya listrik

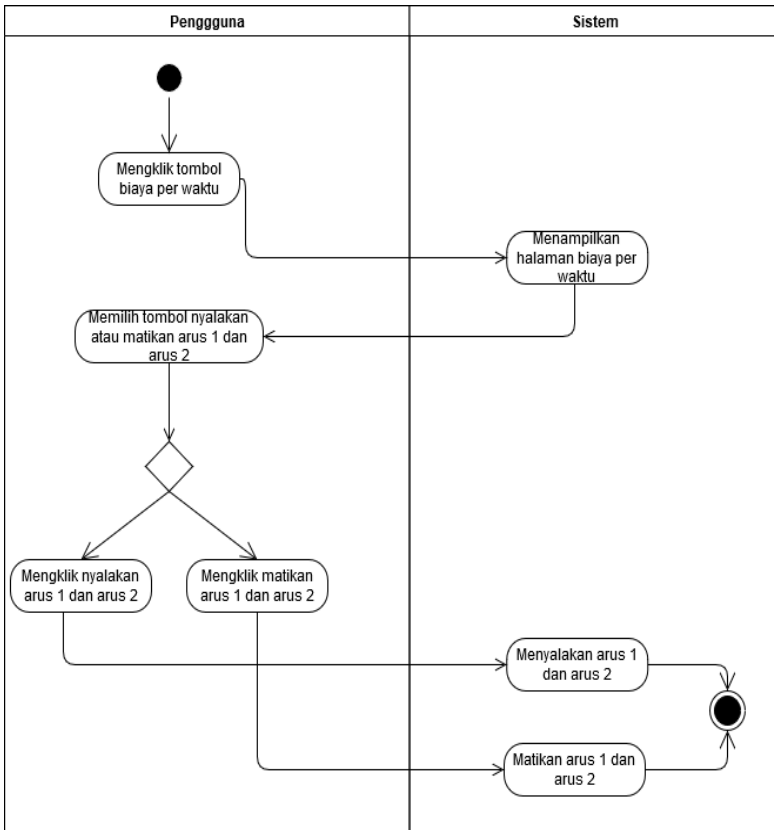
3.3.3.1. Fungsi 1 : Melakukan pengendalian arus 1 dan arus 2

Table 3.5 Use Case Skenario Melakukan Pengendalian Arus 1 dan Arus 2

Nama Kasus Penggunaan	Melakukan pengendalian arus 1 dan arus 2
Kode Use Case	UC-001
Aktor	Pengguna
Deskripsi	Pengguna menyalakan dan mematikan arus 1 dan arus 2

Relasi		-	
Trigger		-	
Kondisi Awal		Pengguna belum menyalakan dan mematikan arus 1 dan arus 2	
Alur Normal:		Sistem:	
1.	Pengguna mengklik tombol biaya per waktu		
		2.	Sistem menampilkan halaman biaya per waktu
3.	Pengguna memilih tombol nyalakan atau matikan arus 1 dan arus 2		
Alur Alternatif:			
4.	Pengguna mengklik nyalakan arus 1 dan 2 lalu submit		
		5.	Sistem menyalakan arus 1 dan arus 2
6.	Pengguna mengklik matikan arus 1 dan arus 2 lalu submit		
		7.	Sistem mematikan arus 1 dan arus 2
Kondisi Akhir		Pengguna sudah menyalakan dan mematikan arus 1 dan arus 2	
Eksepsi: -			

3.3.3.1.1. Diagram aktivitas : Melakukan pengendalian arus 1 dan arus 2



Gambar 3.8 Diagram Aktivitas Pengendalian Arus 1 dan Arus 2

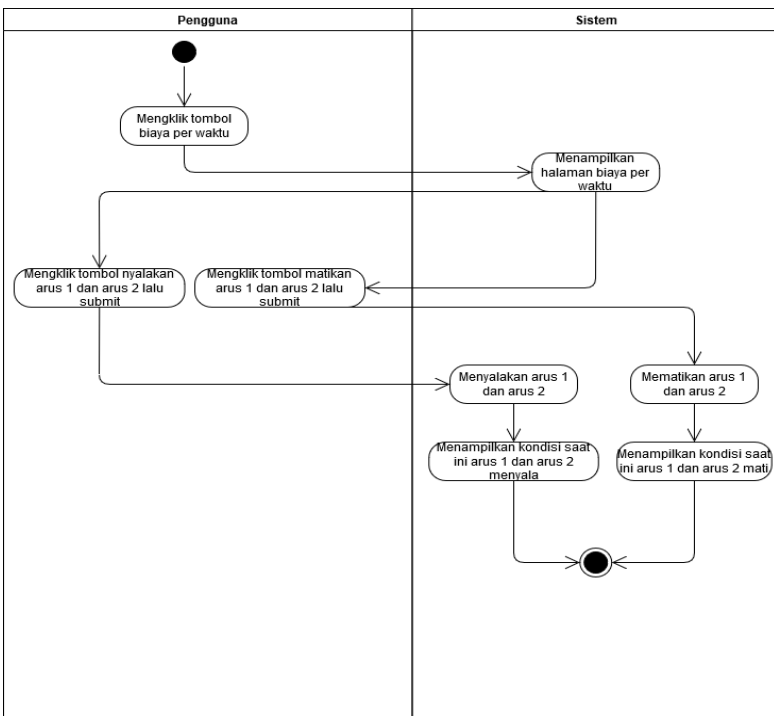
3.3.3.2. Fungsi 2: Melihat kondisi saat ini

Table 3.6 Use Case Skenario Melihat Kondisi Saat Ini

Nama Kasus Penggunaan		Melihat kondisi saat ini	
Kode Use Case		UC-002	
Aktor		Pengguna	
Deskripsi		Pengguna melihat kondisi arus 1 dan arus 2 yang menyala atau mati	
Relasi		-	
Trigger		-	
Kondisi Awal		Pengguna belum melihat kondisi saat ini	
Alur Normal:		Sistem:	
1.	Pengguna mengklik tombol biaya per waktu		
		2.	Sistem menampilkan halaman biaya per waktu
3.	Pengguna mengklik tombol nyalakan atau matikan arus 1 dan arus 2 lalu submit		
		4.	Sistem menyalakan arus 1 dan arus 2
		5.	Sistem menampilkan kondisi saat ini arus 1 dan arus 2 menyala
6.	Pengguna mengklik tombol matikan arus 1 dan arus 2 lalu submit		
		7.	Sistem mematikan arus 1 dan arus 2

		8.	Sistem menampilkan kondisi saat ini arus 1 dan arus 2 mati
Alur Alternatif			
Kondisi Akhir		Pengguna sudah melihat kondisi saat ini	
Eksepsi: -			

3.3.3.2.1. Diagram aktivitas : Melihat kondisi saat ini



Gambar 3.9 Diagram Aktivitas Melihat Kondisi Saat Ini

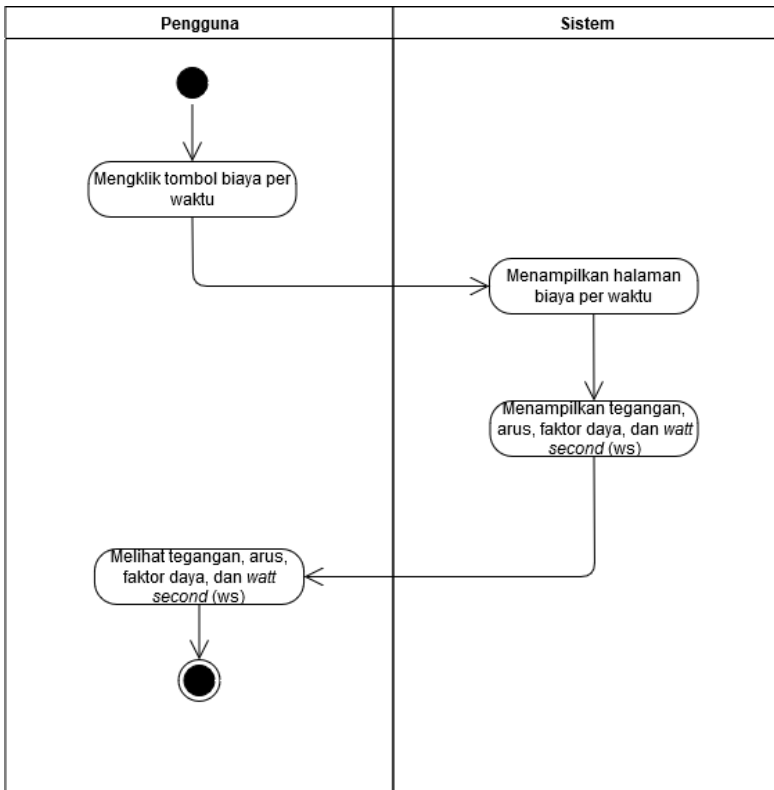
3.3.3.3. Fungsi 3 : Melihat tegangan, arus, faktor daya, dan *watt second* (ws)

Table 3.7 Use Case Skenario Melihat Tegangan, faktor daya, dan *watt second* (ws)

Nama Kasus Penggunaan		Melihat tegangan, arus, faktor daya, dan <i>watt second</i> (ws)	
Kode Use Case		UC-003	
Aktor		Pengguna	
Deskripsi		Pengguna melihat tegangan, arus, faktor daya, dan <i>watt second</i> (ws)	
Relasi		-	
Trigger		-	
Kondisi Awal		Pengguna belum melihat tegangan, arus, faktor daya, dan <i>watt second</i> (ws)	
Alur Normal:		Sistem:	
1.	Pengguna mengklik tombol biaya per waktu		
		2.	Sistem menampilkan halaman biaya per waktu
		3.	Sistem menampilkan tegangan, arus, faktor daya, dan <i>watt second</i> (ws)
4.	Pengguna melihat tegangan, arus, faktor daya, dan <i>watt second</i> (ws)		
Alur Alternatif:			

Kondisi Akhir	Pengguna sudah melihat tegangan, arus, faktor daya, dan <i>watt second</i> (ws)
Eksepsi: -	

3.3.3.3.1. Diagram Aktivitas : Melihat tegangan, arus, faktor daya, dan *watt second* (ws)



Gambar 3.10 Diagram Aktivitas Melihat Tegangan, Arus, faktor daya, dan *watt second* (ws)

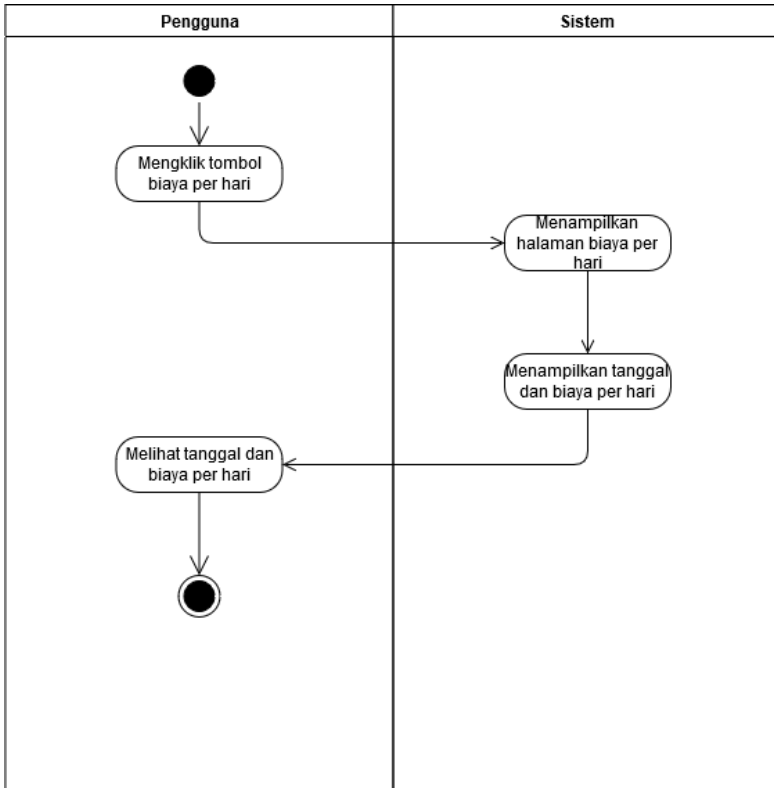
3.3.3.4. Fungsi 4: Melihat biaya per hari

Table 3.8 Use Case Skenario Melihat Biaya Per Hari

Nama Kasus Penggunaan		Melihat biaya per hari	
Kode Use Case		UC-004	
Aktor		Pegguna	
Deskripsi		Pegguna dapat melihat biaya per harinya	
Relasi		-	
Trigger		-	
Kondisi Awal		Pegguna belum melihat biaya per hari	
Alur Normal:		Sistem:	
1.	Pegguna mengklik tombol biaya per hari		
		2.	Sistem menampilkan halaman biaya per hari
		3.	Sistem menampilkan tanggal dan biaya per hari
4.	Pegguna melihat tanggal dan biaya per hari		
Alur Alternatif:			
Kondisi Akhir		Pegguna sudah melihat biaya per hari	
Eksepsi: -			

7.

3.3.3.4.1. Diagram Aktivitas : Melihat biaya per hari



Gambar 3.11 Diagram Aktivitas Melihat Biaya Per Har

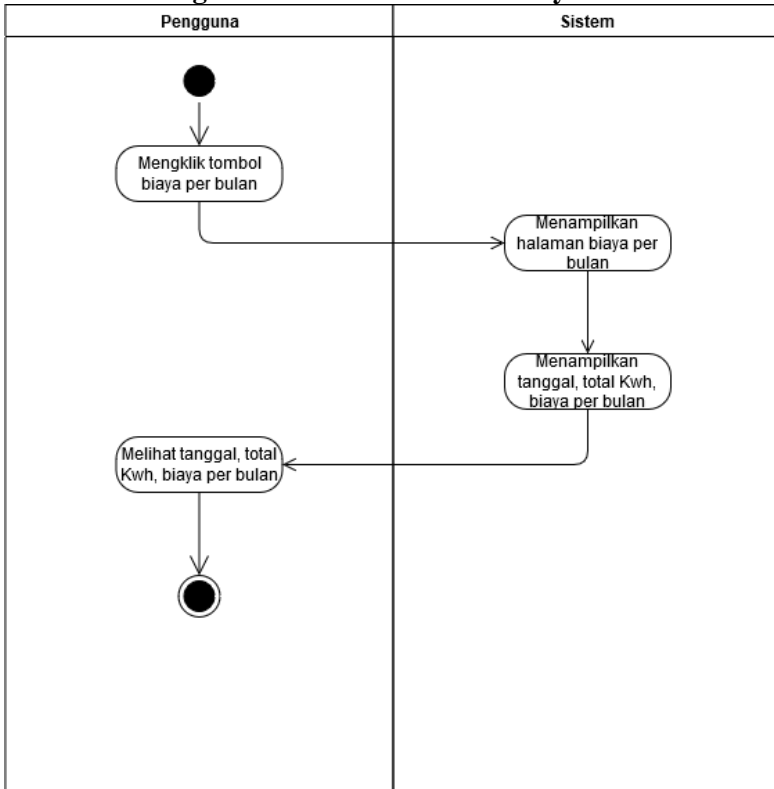
3.3.3.5. Fungsi 5: Melihat biaya per bulan

Table 3.9 Use Case Skenario Melihat Biaya Per Bulan

Nama Kasus Penggunaan	Melihat biaya per hari
Kode Use Case	UC-005
Aktor	Pengguna

Deskripsi		Pengguna dapat melihat biaya per bulannya	
Relasi		-	
Trigger		-	
Kondisi Awal		Pengguna belum melihat biaya per bulan	
Alur Normal:		Sistem:	
1.	Pengguna mengklik tombol biaya per bulan		
		2.	Sistem menampilkan halaman biaya per bulan
		3.	Sistem menampilkan tanggal dan biaya per bulan
4.	Pengguna melihat tanggal dan biaya per bulan		
Alur Alternatif			
Kondisi Akhir		Pengguna sudah melihat biaya per bulan	
Eksepsi: -			

3.3.3.5.1. Diagram Aktivitas : Melihat Biaya Per Bulan



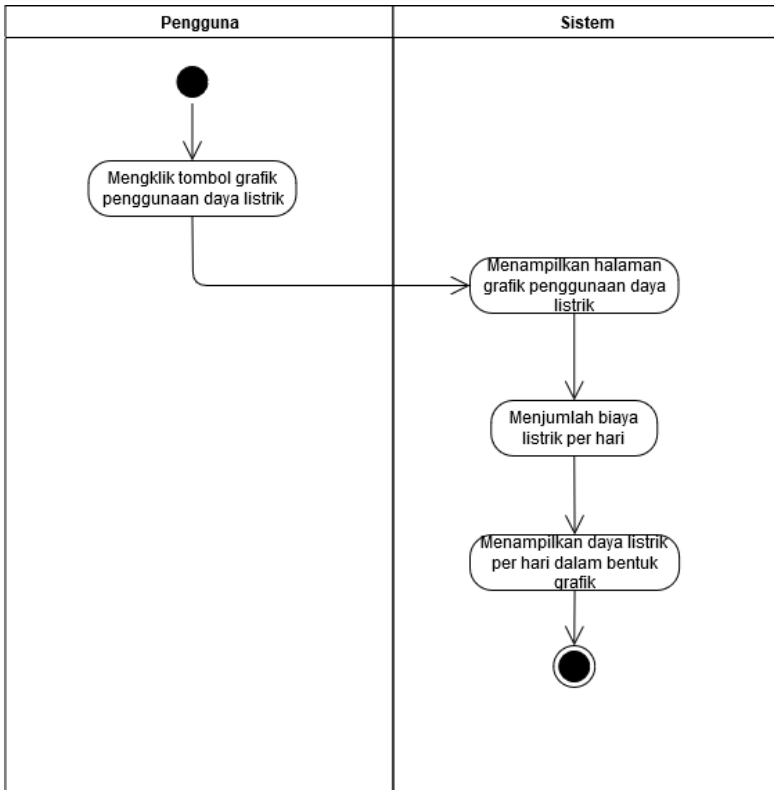
Gambar 3.12 Diagram Aktivitas Melihat Biaya Per Bulan

3.3.3.6. Fungsi 6: Grafik Penggunaan Daya Listrik

Table 3.10 Use Case Skenario Grafik Penggunaan Daya Listrik

Nama Kasus Penggunaan		Grafik Penggunaan Daya Listrik	
Kode Use Case		UC-006	
Aktor		Pengguna	
Deskripsi		Pengguna dapat melihat grafik penggunaan daya listrik	
Relasi		-	
Trigger		-	
Kondisi Awal		Pengguna belum melihat penggunaan daya listrik	
Alur Normal:		Sistem:	
1.	Pengguna mengklik tombol grafik penggunaan daya listrik		
		2.	Sistem menampilkan halaman grafik penggunaan daya listrik
		3.	Sistem menjumlah biaya listrik per hari
		4.	Sistem menampilkan daya listrik per hari dalam bentuk grafik
Alur Alternatif:			
Kondisi Akhir		Pengguna sudah melihat grafik penggunaan daya listrik	
Eksepsi: -			

3.3.3.6.1. Diagram Aktivitas : Grafik Penggunaan Daya Listrik



Gambar 3.13 Diagram Aktivitas Grafik Penggunaan Daya Listrik

[Halaman ini sengaja dikosongkan]

BAB IV IMPLEMENTASI

Pada bab ini akan dijelaskan tentang implementasi yang dilakukan berdasarkan perancangan pada bab sebelumnya.

4.1. Implementasi Database

Pada implementasi *database* ini, penulis menggunakan *hosting* dari DomaiNesia, sebagai penyimpanan *database MySQL* serta *Web Server*, akan dibuat 2 tabel pada *database* seperti yang telah dijelaskan pada bab perancangan.

4.1.1. Implementasi Tabel

4.1.1.1. Implementasi Tabel data_kwh

Pada impelementasi tabel *data_kwh* akan menjalankan *query* seperti pada Kode Sumber 4.1 pada *phpMyAdmin*.

```

1. CREATE TABLE `data_kwh` (
2.   `No` bigint(20) NOT NULL,
3.   `Tanggal` date NOT NULL,
4.   `Waktu` time NOT NULL,
5.   `Tegangan` float NOT NULL,
6.   `Arus1` float NOT NULL,
7.   `Arus2` float NOT NULL,
8.   `Kwh` float NOT NULL,
9.   `Kwh_row` float NOT NUL,
10.  `Biaya` float NOT NULL
11.) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

```

Kode Sumber 4.1 MySql data_kwh

4.1.1.2. Implementasi Tabel biaya_hari

Pada implemementasi tabel biaya_hari akan menjalankan *query* seperti pada Kode Sumber 4.2 pada phpMyAdmin.

```
1. CREATE TABLE `ht_hotel` (  
2. CREATE TABLE `biaya_hari` (  
3.   `id` int(11) NOT NULL,  
4.   `tanggal` date NOT NULL,  
5.   `biaya` float NOT NULL  
6. ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

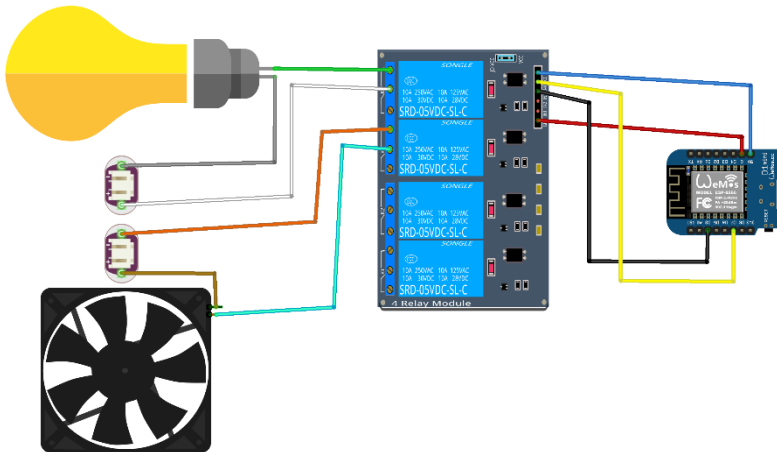
Kode Sumber 4.2 MySql biaya_hari

4.2. Implementasi Perangkat

4.2.1. Rangkaian Alat Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

Implementasi pada alat ini diperlukan beberapa alat yang dihubungkan pada Arduino seperti penjelasan pada bab perancangan, PZEM-004T dan LCD dihubungkan pada Arduino, dimana sensor PZEM-004T nantinya akan berfungsi untuk mengukur tegangan, arus dan kwh. Sedangkan LCD untuk menampilkan tegangan, arus dan kwh yang sudah dihitung.

4.2.1.1. Rangkaian Wemos D1 Mini dengan Relay

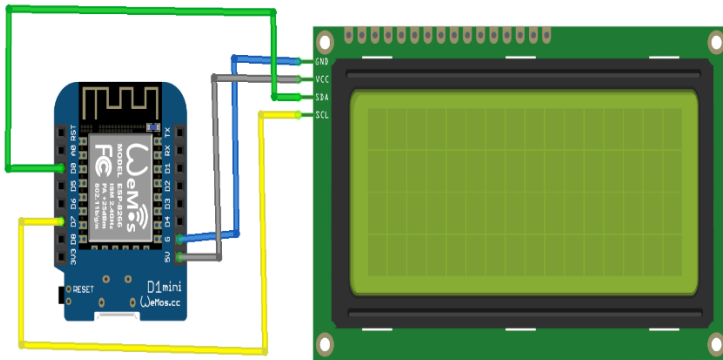


Gambar 4.1 Rangkaian WeMos D1 Mini dengan Relay

Table 4.1 Rangkaian WeMos D1 Mini dengan Relay

PIN	
Relay	Wemos D1 R1
VCC	5V
IN4	D7
IN3	D0
GND	GND

4.2.1.2. Rangkaian Wemos D1 Mini dengan LCD

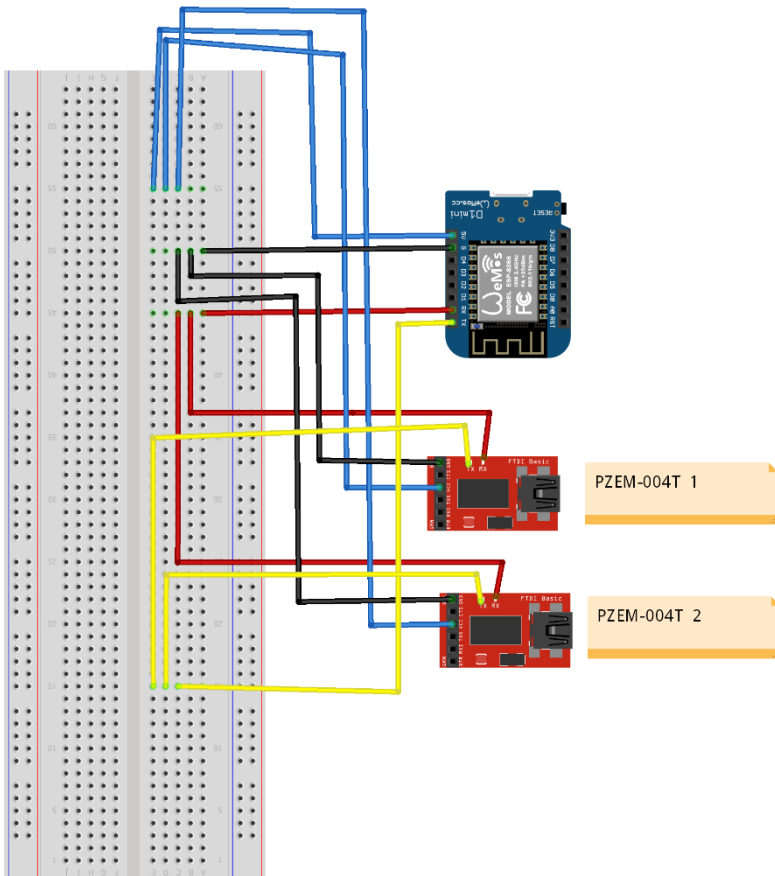


Gambar 4.2 Rangkaian Wemos D1 Mini dengan LCD

Table 4.2 Rangkaian Wemos D1 Mini dengan LCD

PIN	
Wemos D1 Mini	LCD
GND	GND
5V	VCC
SDA	D0
SCL	D7

4.2.1.3. Rangkaian Wemos D1 Mini dengan PZEM-004T

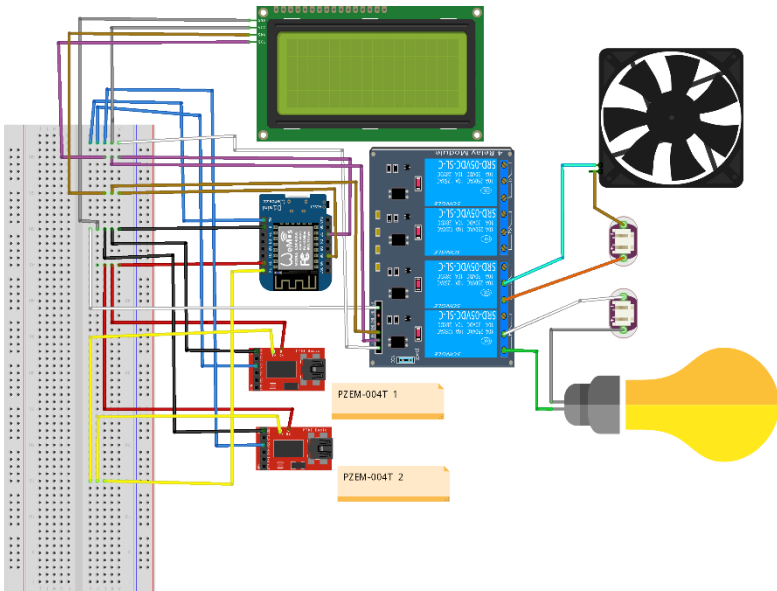


Gambar 4.3 Rangkaian Wemos D1 Mini dengan PZEM-004T

Table 4.3 Rangkaian WeMos D1 Mini dengan PZEM-004T

PIN		
Wemos D1 Mini	PZEM-004T 1	PZEM-004T 2
GND	GND	GND
5V	VCC	VCC
RX	RX	RX
TX	TX	TX

4.2.1.4. Rangkaian Prototype Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal



Gambar 4.4 Rangkaian *Prototype* Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

4.2.2. Implementasi Arduino

Rangkaian sistem dapat dijalankan dengan memasukkan kode program ke dalam Arduino melalui Arduino IDE.

4.2.2.1. *Library* dan Konfigurasi *WiFi*

Pada bagian ini berisi semua *library* yang digunakan dan juga konfigurasi untuk menyambungkan *microcontroller* ke *WiFi*.

```

1. #include <PZEM004Tv30.h>
2. #include <ESP8266WiFi.h>
3. #include <WiFiClient.h>
4. #include <ESP8266HTTPClient.h>
5. #include <ArduinoJson.h>
6. #include <LiquidCrystal_I2C.h>
7. const char* wifiName = "Farras";
8. const char* wifiPass = "12345678";
9. const          char          *host          =
   "http://tugasakhirinformatika.xyz/";

```

Kode Sumber 4.3 *Library* dan Konfigurasi *WiFi*

4.2.2.2. Deklarasi Pin

Pada bagian ini berisi deklarasi pin yang akan digunakan oleh relay, sensor PZEM-004T, dan LCD.

```

1. #define kipas      D7
2. #define lampu     D0
3. #define indikator D4
4.
5. LiquidCrystal_I2C lcd(0x27, 20, 4);
6. PZEM004Tv30 pzem(D3, D4);
7. PZEM004Tv30 pzem2(D5, D6);

```

Kode Sumber 4.4 Deklarasi Pin

4.2.2.3. Implementasi Membaca File Json

Pada bagian ini berisi paket data fungsi dari *web server* untuk membaca file json.

```

1. void packData(){
2.     token = strtok(input, ",");
3.     kondlampu = atoi(token);
4.     Serial.print("LAMPU: ");
5.     Serial.println(kondlampu);
6.     token = strtok(0, ",");
7.     kondkipas = atoi(token);
8.     Serial.println("KIPAS: ");
9.     Serial.println(kondkipas);
10. }
```

Kode Sumber 4.5 Implementasi Membaca File Json

4.2.2.4. Setup Arduino

Pada bagian ini fungsi *setup* Arduino ini berfungsi untuk *setup* LCD, fungsi ini akan menjalankan fungsi *connect WiFi* yang berisi kode program untuk menghubungkan Arduino dengan *WiFi* dan *Web Server*.

```

1. void setup()
2. {
3.     pinMode(indikator,OUTPUT);
4.     digitalWrite(indikator,LOW);
5.     lcd.begin();
6.     lcd.backlight();
7.     lcd.clear();
8.     Serial.begin(115200);
```

Kode Sumber 4.6 Setup LCD

```

1. Serial.print("Connecting to ");
2. Serial.println(wifiName);
3. WiFi.begin(wifiName, wifiPass);
4. while (WiFi.status() != WL_CONNECTED)
5. {
6.     delay(500);
7.     Serial.print(".");
8. }
9.
10. Serial.println("");
11. Serial.println("WiFi connected");
12. Serial.println("IP address: ");
13. Serial.println(WiFi.localIP());

```

Kode Sumber 4.7 Fungsi Connect WiFi

Setelah *connect WiFi* maka bisa mengendalikan relay, dan *set* relay menjadi *HIGH*.

```

1. pinMode(lampu,OUTPUT);
2. pinMode(kipas,OUTPUT);
3. digitalWrite(lampu,HIGH);
4. digitalWrite(kipas,HIGH);

```

Kode Sumber 4.8 Set Relay

4.2.2.5. Implementasi Loop Program

Setelah menjalankan fungsi *void setup*, Arduino akan melanjutkan dengan membaca fungsi *void loop* yaitu dengan *connect server*.

```

1. void loop()
2. {
3.     HTTPClient http;

```



```

4.     Serial.print("Request Link:");
5.
   Serial.println("http://tugasakhirinformatika.xyz/ge
t_status.php");
6.
   http.begin("http://tugasakhirinformatika.xyz/get_st
atus.php");
7.
8.     httpCode = http.GET();
9.     payload = http.getString();
10.
11.         Serial.print("Response Code:");
12.         Serial.println(httpCode);
13.
14.         Serial.print("Returned     data     from
Server:");
15.         Serial.println(payload);

```

Kode Sumber 4.9 Fungsi *Loop Connect Server*

Bagian ini adalah implementasi dari lampu dan kipas. Dimana data yang sudah dibaca oleh json dari *server*. Setelah itu *print* kondisi lampu pada json, apakah kondisi saat ini menyala atau mati. Jika 1 adalah menyala dan jika 0 maka mati.

```

1. if(httpCode == 200)
2.   {
3.     payload.toCharArray(input, sizeof(input));
4.     Serial.print("Returned data from Server:");
5.     packData();
6.     Serial.println("////////////////////////////////////
////////////////////////////////////");
7.     Serial.println(kondlampu);
8.     Serial.println(kondkipas);

```

```

9. Serial.println("////////////////////////////////////
   //////////////////////////////////////");
10.         if(kondlampu==0           &&
   kondkipas==0){digitalWrite(lampu,HIGH);digitalWrite
   (kipas,HIGH);}
11.         else           if(kondlampu==0           &&
   kondkipas==1){digitalWrite(lampu,HIGH);digitalWrite
   (kipas,LOW);}
12.         else           if(kondlampu==1           &&
   kondkipas==0){digitalWrite(lampu,LOW);digitalWrite(
   kipas,HIGH);}
13.         else           if(kondlampu==1           &&
   kondkipas==1){digitalWrite(lampu,LOW);digitalWrite(
   kipas,LOW);}
14.         }

```

Kode Sumber 4.10 Implementasi Lampu dan Kipas

Pada bagian ini berisikan fungsi untuk menghitung daya yang dikeluarkan. Dan rumus untuk mencari daya adalah $P=V.I.\cos \Phi$.

```

1. voltage1 = pzem.voltage();
2.   current1 = pzem.current();
3.   pf1 = pzem.pf();
4.
5.   voltage2 = pzem2.voltage();
6.   current2 = pzem2.current();
7.   pf2 = pzem2.pf();
8.
9.   energy1 = 3*voltage1*current1*pf1;
10.  energy2 = 3*voltage2*current2*pf2;

```

Kode Sumber 4.11 Fungsi Menghitung Daya

Pada bagian ini mengatur *cursor* LCD untuk peletakan daya yang sudah dihitung nantinya.

1. `lcd.setCursor(0,0);lcd.print(" ");`
2. `lcd.setCursor(0,1);lcd.print(" ");`
3. `lcd.setCursor(0,2);lcd.print(" ");`
4. `lcd.setCursor(0,3);lcd.print(" ");`

Kode Sumber 4.12 Set *Cursor* LCD

Pada bagian ini adalah untuk menampilkan hasil tegangan, arus, faktor daya dan *watt second* (Ws) di layar LCD.

1. `if(!isnan(voltage1))`
`{lcd.setCursor(0,0);lcd.print(voltage1,2);lcd.setCursor(7,0);lcd.print("V");}`
2. `else`
`{lcd.setCursor(0,0);lcd.print("0");lcd.setCursor(7,0);lcd.print("V");}`
- 3.
4. `if(!isnan(current1))`
`{lcd.setCursor(0,1);lcd.print(current1,4);lcd.setCursor(7,1);lcd.print("A");}`
5. `else`
`{lcd.setCursor(0,1);lcd.print("0");lcd.setCursor(7,1);lcd.print("A");}`
- 6.
7. `if(!isnan(pf1))`
`{lcd.setCursor(0,2);lcd.print(pf1,4);lcd.setCursor(7,2);lcd.print("");}`
8. `else`
`{lcd.setCursor(0,2);lcd.print("0");lcd.setCursor(7,2);lcd.print("");}`
- 9.
10. `if(!isnan(energy1))`
`{lcd.setCursor(0,3);lcd.print(energy1/3);lcd.setCursor(7,3);lcd.print("Ws");}`

```

11.         else
            {lcd.setCursor(0,3);lcd.print("0");lcd.setCursor(7,
              3);lcd.print("Ws");}
12.
13.         if(!isnan(voltage2))
            {lcd.setCursor(10,0);lcd.print(voltage2,2);lcd.setC
              ursor(17,0);lcd.print("V");}
14.         else
            {lcd.setCursor(10,0);lcd.print("0");lcd.setCursor(1
              7,0);lcd.print("V");}
15.
16.         if(!isnan(current2))
            {lcd.setCursor(10,1);lcd.print(current2,4);lcd.setC
              ursor(17,1);lcd.print("A");}
17.         else
            {lcd.setCursor(10,1);lcd.print("0");lcd.setCursor(1
              7,1);lcd.print("A");}
18.
19.         if(!isnan(pf2))
            {lcd.setCursor(10,2);lcd.print(pf2,4);lcd.setCursor
              (17,2);lcd.print("");}
20.         else
            {lcd.setCursor(10,2);lcd.print("0");lcd.setCursor(1
              7,2);lcd.print("");}
21.
22.         if(!isnan(energy2))
            {lcd.setCursor(10,3);lcd.print(energy2/3);lcd.setCu
              rsor(17,3);lcd.print("Ws");}
23.         else
            {lcd.setCursor(10,3);lcd.print("0");lcd.setCursor(1
              7,3);lcd.print("Ws");}

```

Kode Sumber 4.13 Menampilkan Tegangan, Arus, Faktor Daya, dan *Watt Second* (Ws)

Pada bagian ini untuk menampilkan data di serial monitor Arduino IDE.

1. `Serial.print("arus 1 =");`
2. `Serial.println(current1);`
3. `Serial.print("arus 2 =");`
4. `Serial.println(current2);`

Kode Sumber 4.14 Menampilkan Data di Serial Monitor

Pada bagian adalah berisikan fungsi untuk menghitung *voltage* pada arus 1 dan arus 2. Lalu menjumlahkan kwh di arus 1 dan arus 2.

1. `voltage3 = (voltage1+voltage2)/2;`
2. `kwh = (energy1+energy2);`

Kode Sumber 4.15 Fungsi Menghitung Voltage dan Kwh

Pada bagian ini adalah menyiapkan data yang akan dikirim ke *database* melalui API, untuk string adalah tipe data yang akan dikirim ke *database*. Lalu menyiapkan *link* yang akan dikirim ke API.

1. `senddata = "?t=" + String(voltage3,2) + "&a1=" + String(current1,2)+ "&a2=" + String(current2,2)+ "&kwh=" + String(kwh,3);`
2. `linksend = "http://tugasakhirinformatika.xyz/simpan.php" + senddata;`

Kode Sumber 4.16 Fungsi Menyiapkan Data ke Database

Pada bagian ini `httpCode` menjelaskan tentang dari *server* apakah berhasil atau *error*, lalu menampilkan hasil dari *server* apakah berhasil memasukkan data ke *database*.

```
1. httpCode = http.GET();
2.     payload = http.getString();
3.
4.     Serial.println(httpCode);
5.     Serial.println(payload);
```

Kode Sumber 4.17 Fungsi `httpCode`

4.3. PHP

Pada implementasi perancangan *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik penulis menggunakan PHP dalam pembuatannya. Perlu dilakukan konfigurasi *database* agar PHP dapat terhubung dengan MySQL, seperti pada gambar 4.5 Konfigurasi Database.

```
$servername = "localhost";
$dbname = "tugasak6_listrik";
$username = "tugasak6_farras";
$password = "farrasrabbani";
```

Gambar 4.5 Konfigurasi Database

4.3.1. Implementasi Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal

Prototype Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal akan diimplementasikan sesuai fungsi yang telah dibuat. Pada implementasi sistem ini penulis menggunakan empat fungsi yaitu sebagai berikut:

1. Biaya_waktu.php

Fungsi PHP ini digunakan untuk melakukan pengendalian dan pengawasan untuk arus 1 dan arus 2.

2. Biaya_hari.php

Fungsi PHP ini digunakan untuk menampilkan biaya per hari.

3. Simpan.php

Fungsi digunakan untuk menghitung daya listrik.

4. Chart.php

Fungsi PHP ini digunakan untuk menampilkan grafik penggunaan daya listrik.

4.3.1.1. Fungsi Melakukan Pengendalian Arus 1 dan Arus 2

Pada kasus ini pengguna melakukan pengendalian arus 1 dan arus 2 yaitu untuk menyalakan atau mematikan arus 1 dan arus 2.

```
if(isset($_GET['pilihLampu']))
{
    $lampu=$_GET['lampu'];
    if($lampu == "lampunya")
    {
        $fsta = file_get_contents("sta.json");
        $sisiLampu = json_decode($fsta,true);
        $sisiLampu['statusLampu']="1";

        file_put_contents('sta.json',json_encode($sisiLampu));
    }
    else if($lampu == "lampumati")
    {
        $fsta = file_get_contents("sta.json");
        $sisiLampu = json_decode($fsta,true);
        $sisiLampu['statusLampu']="0";

        file_put_contents('sta.json',json_encode($sisiLampu));
    }
} file_get_contents("sta.json");
    $sisiLampu = json_decode($fsta,true);
    $sisiLampu['statusLampu']="0";

    file_put_contents('sta.json',json_encode($sisiLampu));
}
```

Gambar 4.6 Fungsi Melakukan Pengendalian Arus 1 dan Arus 2

4.3.1.2. Fungsi Melihat Kondisi Saat Ini

Pada kasus ini pengguna dapat melihat *status* atau kondisi saat ini yaitu berupa menyala atau mati. Dimana jika pengguna menyalakan arus maka kondisi saat ini nyala, sedangkan jika pengguna mematikan arus maka kondisi saat ini mati. Dibawah terdapat 4 macam gambar kondisi saat ini yang berbeda yaitu gambar 4.7, 4.8, 4.9, dan 4.10.

```

if($isiLampu['statusLampu']== "0" &&
  $isiKipas['statusKipas']== "0")
{
    ?>
    <div class="container" style=" margin-
bottom:70px;">
    <div class="row">

        <div class="col text-right">
            <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
            <?php echo "Arus 1 Mati"; ?>
        </div>
        <div class="col text-right" >
            <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
            <?php echo "Arus 2 Mati"; ?>
        </div>
    </div>
    <?php
}

```

Gambar 4.7 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Mati dan Arus 2 Mati

```

else if($isiLampu['statusLampu']==="0" &&
$isiKipas['statusKipas']==="1")
{
    ?>
    <div class="container" style=" margin-
bottom:70px;">
    <div class="row">

        <div class="col text-right">
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 1 Menyala"; ?>
        </div>
        <div class="col text-right" >
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 2 Mati"; ?>
        </div>
    </div>
</div>
<?php
}

```

Gambar 4.8 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Mati dan Arus 2 Nyala

```

else if($isiLampu['statusLampu']==="1" &&
$isiKipas['statusKipas']==="0")
{
    ?>
    <div class="container" style=" margin-
bottom:70px;">
    <div class="row">

        <div class="col text-right">
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 1 Mati"; ?>
        </div>
        <div class="col text-right" >
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 2 Menyala"; ?>
        </div>
    </div>
</div>
<?php
}

```

Gambar 4.9 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Nyala dan Arus 2 Mati

```

else if($isiLampu['statusLampu']==="1" &&
$isiKipas['statusKipas']==="1")
{
    ?>
    <div class="container" style=" margin-
bottom:70px;">
    <div class="row">

        <div class="col text-right">
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 1 Mati"; ?>
        </div>
        <div class="col text-right" >
        <?php echo "Kondisi Saat Ini:"; ?>
        </div>
        <div class="col">
        <?php echo "Arus 2 Menyala"; ?>
        </div>
    </div>
</div>
<?php
}

```

Gambar 4.10 Fungsi Melihat Kondisi Saat Ini Yaitu Arus 1 Nyala dan Arus 2 Mati

4.3.1.3. Fungsi Mengambil Data Kwh

Pada fungsi ini \$sql2 untuk mengambil data kwh dari database, lalu disimpan di \$kwh.

```
$sql2 = mysqli_query($conn, "SELECT * FROM data_kwh  
ORDER BY Kwh DESC LIMIT 1");  
$kwh= mysqli_fetch_row($sql2);
```

Gambar 4.11 Fungsi Mengambil Data Kwh

4.3.1.4. Fungsi Menerima Data Arduino

Pada fungsi ini data dari Arduino akan diterima yang akan di implementasikan.

```
if(!empty($_GET['t']) && !empty($_GET['a1']) &&  
!empty($_GET['a2']) && !empty($_GET['kwh']))
```

Gambar 4.12 Fungsi Menerima Data Arduino

4.3.1.5. Implementasi Menghitung Tegangan, Arus, faktor daya, watt second (ws) , dan Biaya

Menghitung tegangan, arus dan biaya. Yang nantinya akan dihitung untuk mencari Kwh dan biaya per harinya. Daya yang sudah dihitung di Arduino dijadikan Kwh. Kwh dihitung menggunakan rumus jumlah total Kwh X 1467.28 untuk mendapatkan biaya per hari nya.

```
{
    $tegangan=$_GET['t'];
    $arus1=$_GET['a1'];
    $arus2=$_GET['a2'];
    $a=$_GET['kwh'];
    $pf1=$_GET['pf1'];
    $pf2=$_GET['pf2'];
    $energy1=$_GET['energy1'];
    $energy2=$_GET['energy2'];
    $jumlah=($a*2.77778e-7)+$kwh[6];
    $kwh_row=($a*2.77778e-7);
    $biaya=$jumlah*1467.28;
```

Gambar 4.13 Implementasi Menghitung Tegangan, Arus, Faktor Daya, dan Watt Second (WS) dan Biaya

4.3.1.6. Fungsi Memasukkan Data

Fungsi ini digunakan untuk memasukkan data yang diterima dari Arduino ke dalam *database sql*.

```
$sql = "INSERT INTO data_kwh (Tanggal, Waktu, Tegangan,
Arus1, pf1, energy1, Arus2, pf2, energy2, Kwh, Kwh_row, Biaya)
VALUES ('. $tanggal .', '. $waktu .', '.
$tegangan .', '. $arus1 .', '. $pf1 .', '. $energy1 .', '. $arus2
.', '. $pf2 .', '. $energy2 .', '. $jumlah .', '. $kwh_row .', '.
$biaya .')";
```

Gambar 4.14 Fungsi Memasukkan Data

BAB V UJI COBA DAN EVALUASI

Bab ini membahas tentang uji coba dan analisis dari implementasi yang telah dilakukan pada Tugas Akhir ini.

5.1. Lingkungan Uji Coba

Lingkungan uji coba penulis menggunakan konektivitas jaringan rumah “*Home Network*” untuk melakukan uji coba *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal dijalankan pada laptop dengan spesifikasi sebagai berikut.

Table 5.1 Lingkup Spesifikasi

Komponen	Spesifikasi
CPU	Intel® Core™ I7 7700HQ Processor @2.8 GHz – 3.2 GHz
Sistem Operasi	Windows 10 Home
Memori	Up to 32 GB SDRAM, 2x SO-DIMM socket
Penyimpanan	1 TB

5.2. Skenario Pengujian

Pada bagian ini akan dibahas mengenai proses uji coba yang digunakan. Pada proses uji coba, pengujian dilakukan dengan menjalankan serangkaian perintah terhadap sistem dimana pengujian dilakukan dalam bentuk kasus penggunaan pada *prototype* yang telah dibuat. Kasus pengujian ini berkolerasi dengan *flowchart* dan kasus-kasus penggunaan yang sebelumnya sudah dirancang dan dijelaskan pada BAB III.

5.2.1. Skenario Pengendalian Alat Pada Maket atau Miniatur

Pada bagian ini adalah hasil uji coba dari skenario pengendalian perangkat pada *Prototype* Sistem Cerdas Untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik di Rumah Tinggal. Dan untuk tampilan di *Web* bisa dilihat pada gambar 5.1 dan gambar 5.2.

DATA BIAYA PER WAKTU

<input type="button" value="NYALAKAN ARUS 1"/>	<input type="button" value="NYALAKAN ARUS 2"/>
<input type="button" value="Submit"/>	<input type="button" value="Submit"/>

Gambar 5.1 Pengendalian Menyalakan Arus 1 dan Arus 2

DATA BIAYA PER WAKTU

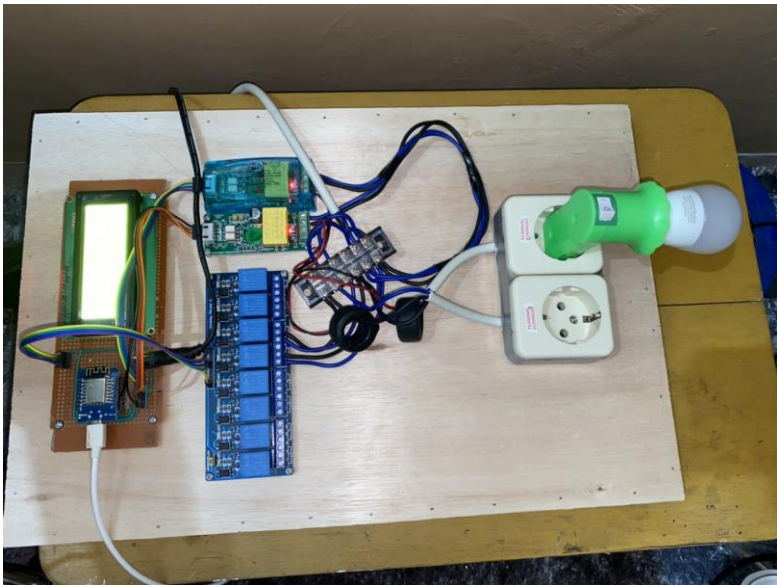
<input type="button" value="MATIKAN ARUS 1"/>	<input type="button" value="MATIKAN ARUS 2"/>
<input type="button" value="Submit"/>	<input type="button" value="Submit"/>

Gambar 5.2 Pengendalian Mematikan Arus 1 dan Arus 2

Gambar di bawah adalah serial monitor dari Arduino IDE yang akan menampilkan kondisi lampu dan kipas saat pengguna melakukan pengendalian perangkat. Pada gambar 5.3 dan gambar 5.5 adalah dimana server memberikan *response* status lampu dimana 0 adalah status pada saat mati, dan 1 adalah nyala. Dan server memberikan *response* arus 0 adalah relay sedang mati, dan arus 1 adalah saat relay menyala.


```
Request Link:http://tugasakhirinformatika.xyz/get_status.php
Response Code:200
Returned data from Server:0,0
Returned data from Server:Arus 2: 0
Arus 1:
0
////////////////////////////////////
0
0
////////////////////////////////////
```

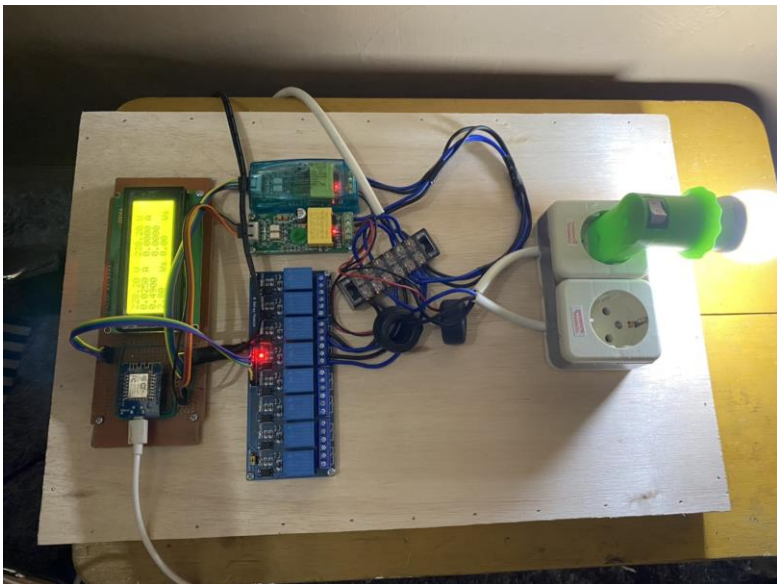
Gambar 5.3 Arduino Serial Monitor Arus 1 Mati



Gambar 5.4 Keadaan Lampu Mati

```
Request Link:http://tugasakhirinformatika.xyz/get_status.php
Response Code:200
Returned data from Server:0,1
Returned data from Server:Arus 2: 0
Arus 1:
1
////////////////////////////////////
0
1
```

Gambar 5.5 Arduino Serial Monitor Arus 1 Nyala

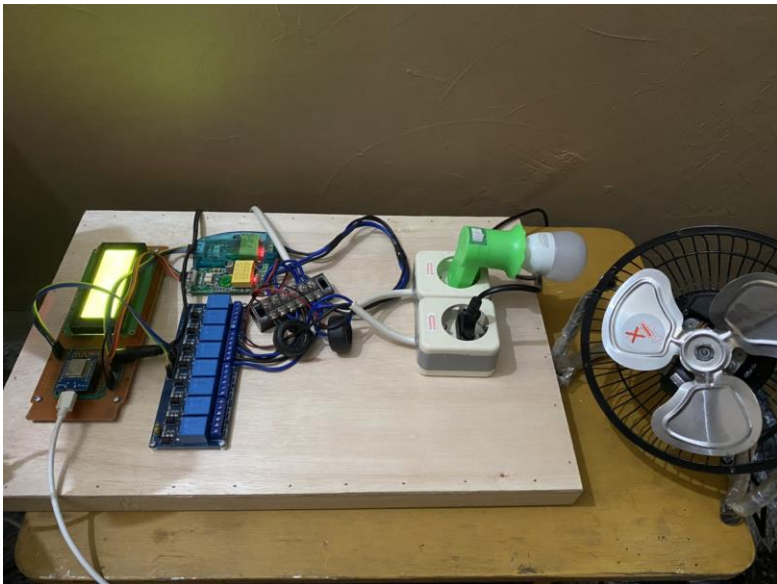


Gambar 5.6 Keadaan Lampu Nyala

Pada gambar 5.7 dan gambar 5.9 adalah dimana server memberikan *response* status kipas angin dimana 0 adalah status pada saat mati, dan 1 adalah nyala. Dan server memberikan *response* arus 0 adalah relay sedang mati, dan arus 1 adalah saat relay menyala.

```
Request Link:http://tugasakhirinformatika.xyz/get_status.php
Response Code:200
Returned data from Server:0,0
Returned data from Server:Arus 2: 0
Arus 1:
0
////////////////////////////////////
0
0
////////////////////////////////////
```

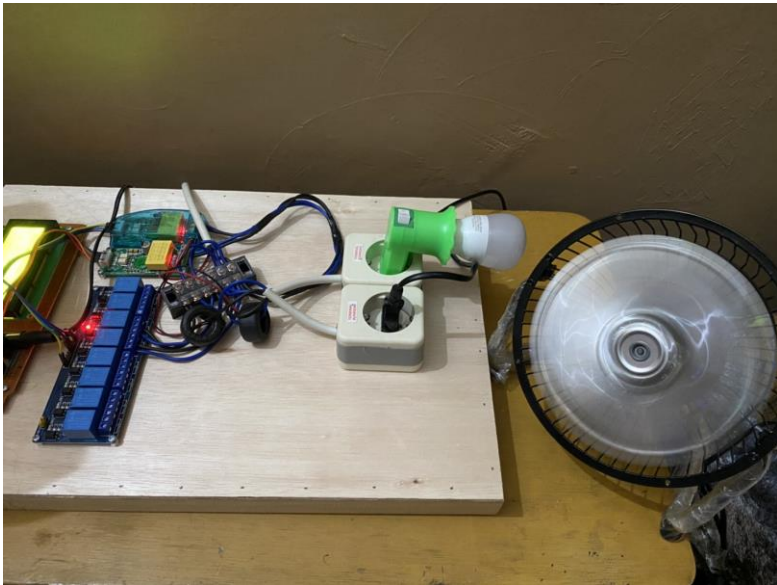
Gambar 5.7 Arduino Serial Monitor Arus 2 Mati



Gambar 5.8 Keadaan Kipas Angin Mati

```
Request Link:http://tugasakhirinformatika.xyz/get_status.php
Response Code:200
Returned data from Server:1,0
Returned data from Server:Arus 2: 1
Arus 1:
0
////////////////////////////////////
1
0
////////////////////////////////////
```

Gambar 5.9 Arduino Serial Monitor Arus 2 Nyala



Gambar 5.10 Keadaan Kipas Angin Menyala

5.2.2. Skenario Melihat Kondisi Saat Ini

Pada bagian ini adalah dilakukan untuk menguji ketika alat yang sudah di nyalakan maka kondisi saat ini menyala, dan apabila ketika alat dimatikan maka kondisi saat ini mati.

Kondisi Saat Ini: Arus 1 Menyala

Kondisi Saat Ini: Arus 2 Menyala

Gambar 5.11 Kondisi Arus 1 dan Arus 2 Menyala

Kondisi Saat Ini: Arus 1 Mati

Kondisi Saat Ini: Arus 2 Mati

Gambar 5.12 Kondisi Arus 1 dan Arus 2 Mati

5.2.3. Skenario Pengujian Melihat Pengukuran Listrik

Pada skenario ini dilakukan untuk menguji apakah pengguna dapat memantau pengukuran listrik. Pengukuran listrik dapat dilihat berupa tegangan, arus, faktor daya dan watt.

Fitur Robotik
 GEMAWA2020

DATA BIAYA PER WAKTU

Kondisi Saat Ini: Arus 1 Mati Kondisi Saat Ini: Arus 2 Mati

No	Tanggal	Waktu	Tegangan	Arus 1	FDayat	Wf1	Arus 2	FDayat	Wf2
1	2020-06-17	12:23:36	239	0.02	0	0	0	0	0
2	2020-06-17	12:23:38	239	0.02	0	0	0	0	0
3	2020-06-17	12:23:43	239	0.02	0	0	0	0	0
4	2020-06-17	12:23:48	232	0.02	0	0	0	0	0
5	2020-06-17	12:23:50	233	0.02	0	0	0	0	0
6	2020-06-17	12:23:54	233	0.02	0	0	0	0	0
7	2020-06-17	12:23:57	234	0.02	0	0	0	0	0
8	2020-06-17	12:24:01	234	0.02	0	0	0	0	0
9	2020-06-17	12:24:05	233	0.02	0	0	0	0	0
10	2020-06-17	12:24:08	232.9	0.02	0	0	0	0	0
11	2020-06-17	12:24:12	232.75	0.02	0	0	0	0	0
12	2020-06-17	12:24:15	232.7	0.02	0	0	0	0	0
13	2020-06-17	12:24:18	232.4	0.02	0	0	0	0	0
14	2020-06-17	12:24:23	232.05	0.02	0	0	0	0	0
15	2020-06-17	12:24:28	232.9	0.02	0	0	0	0	0

Gambar 5.13 Halaman Pengukuran Listrik

5.2.4. Skenario Pengujian Melihat Data Biaya Per Hari

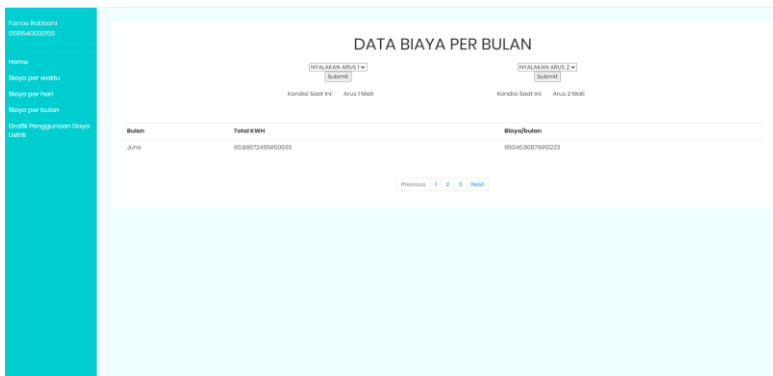
Pada skenario ini dilakukan untuk menguji apakah pengguna dapat memantau biaya per harinya. Dimana yang nantinya total daya yang sudah didapat pada per hari akan di jumlah dan dikalikan biaya per harinya. Untuk mendapat data kwh dari data daya yang sudah dihitung, lalu di konversi menjadi kwh.

Tanggal	Total KWH	Biaya/hari
2020-06-17	9378897678478056	9793394444875
2020-06-18	2726322934464382	400322796278792
2020-06-19	0.908007954638496	929.390798083984
2020-06-20	9.29089332647826	9898.3958287378
2020-06-21	937356449329846	9764.71847935223
2020-06-22	0.007987936889234462	1847929346679128
2020-06-23	0.0739893646328914	93294027942834
2020-06-24	0.0348893084882077	4.6089922884738

Gambar 5.14 Halaman Data Biaya Per Hari

5.2.5. Skenario Pengujian Melihat Data Biaya Per Bulan

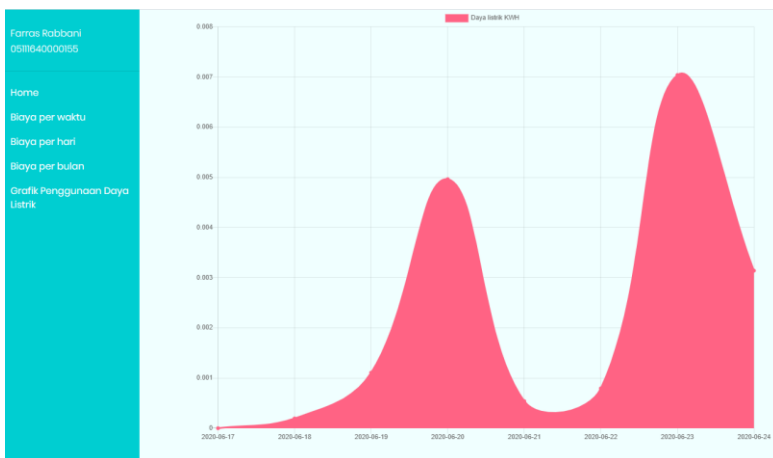
Pada skenario ini dilakukan untuk menguji apakah pengguna dapat memantau biaya per bulannya. Dimana yang nantinya data biaya per hari akan dijumlah pada per bulannya dan total kwh.



Gambar 5.15 Halaman Data Biaya Per Bulan

5.2.6. Skenario Pengujian Melihat Grafik

Pada skenario ini pengguna dapat melihat grafik per harinya. Data yang dilihat yaitu berupa tanggal dan total kwh.

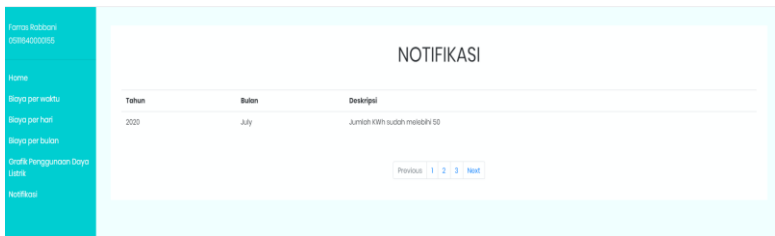


Gambar 5.16 Halaman Grafik Penggunaan Daya Listrik

5.2.7. Skenario Pengujian Melihat Notifikasi

Pada skenario ini pengguna dapat melihat notifikasi berupa tahun, bulan dan deskripsi. Ini adalah simulasi jika

kwh melebihi 50 Kwh dan yang nantinya akan mendapatkan notifikasi dari *website*.

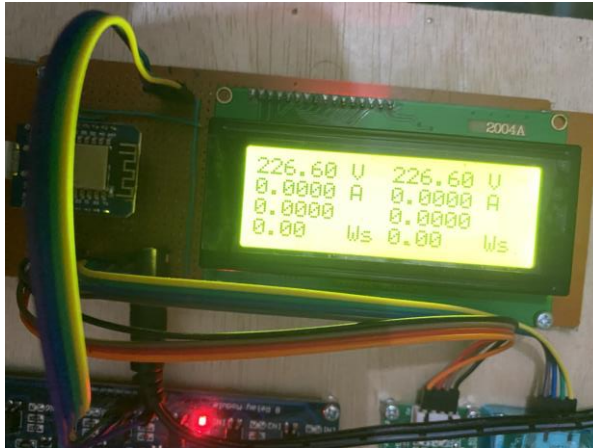


Gambar 5.17 Halaman Notifikasi

5.2.8. Skenario Mengukur Dengan Avo Meter dan Tampilan di LCD



Gambar 5.18 Avo Meter Mengukur Tegangan



Gambar 5.19 Tampilan di LCD

5.2.9. Skenario Uji Coba Menghitung Dengan PZEM-004T

Pada skenario uji coba ini penulis mengukur lampu dengan menggunakan sensor PZEM-004T. Penulis menggunakan rumus $P=V.I.\cos \Phi$. Pengujian lampu pijar dengan 3 watt.

Table 5.2 Hasil Pembacaan Sensor PZEM-004T Lampu 3 Watt

Tegangan (pembacaan)	Arus (pembacaan)	Faktor Daya (Pembacaan)	Daya (Perhitungan)
226.05 V	0.03 A	0.48	2.71 W
226.15 V	0.03 A	0.48	2.71 W
226.1 V	0.03 A	0.48	2.71 W
236.1 V	0.03 A	0.48	2.71 W
226.05 V	0.03 A	0.48	2.71 W
225.8 V	0.03 A	0.48	2.71 W
225.85 V	0.03 A	0.48	2.71 W
225.9 V	0.03 A	0.48	2.71 W
225.95 V	0.03 A	0.48	2.71 W

225.75 V	0.03 A	0.5	2.82 W
----------	--------	-----	--------

Pada skenario uji coba ini penulis mengukur kipas angin dengan menggunakan sensor PZEM-004T. Penulis menggunakan rumus $P=V.I.\cos \Phi$. Pengujian kipas angin dengan 20 watt.

Table 5.3 Hasil Pembacaan Sensor PZEM-004T Kipas Angin 20 Watt

Tegangan (Pembacaan)	Arus (Pembacaan)	Faktor Daya (Pembacaan)	Daya (Perhitungan)
226.8 V	0.11 A	0.54	12.49 W
226.75 V	0.11 A	0.54	12.49 W
227 V	0.11 A	0.54	12.63 W
226.9 V	0.11 A	0.54	12.5 W
226.65 V	0.11 A	0.54	12.48 W
226.55 V	0.11 A	0.54	12.48 W
226.65 V	0.11 A	0.54	12.48 W
226.55 V	0.11 A	0.55	12.71 W
226.35 V	0.11 A	0.54	12.46 W
226.45 V	0.11 A	0.54	12.47 W

Pada skenario uji coba ini penulis mengukur lampu dengan menggunakan sensor PZEM-004T. Penulis menggunakan rumus $P=V.I.Cos \Phi$. Pengujian lampu dengan 25 watt.

Table 5.4 Hasil Pembacaan Sensor PZEM-004T Lampu 25 Watt

Tegangan (pembacaan)	Arus (pembacaan)	Faktor Daya (Pembacaan)	Daya (Perhitungan)
226.55 V	0.11 A	1	25.82 W
226.4 V	0.11 A	1	25.57 W
226.1 V	0.11 A	1	25.54 W
226.05 V	0.11 A	1	25.54 W
226.25 V	0.11 A	1	25.56 W
227.15V	0.11 A	0.99	25.63 W
227 V	0.11 A	1	25.64 W
227.55 V	0.11A	0.99	25.68 W
227.35 V	0.11 A	1	25.91 W
227.4 V	0.11 A	1	25.91 W

Pada skenario uji coba ini penulis mengukur lampu dengan menggunakan sensor PZEM-004T. Penulis menggunakan rumus $P=V.I.Cos \Phi$. Pengujian lampu dengan 100 watt.

Table 5.5 Hasil Pembacaan Sensor PZEM-004T Lampu 100 Watt

Tegangan (pembacaan)	Arus (pembacaan)	Faktor Daya (Pembacaan)	Daya (Perhitungan)
226.05 V	0.44 A	1	99.03 W
226.15 V	0.44 A	1	99.08 W
226.1 V	0.44 A	1	99.03 W
236.1 V	0.44 A	1	99.03 W

226.05 V	0.44 A	1	99.03 W
225.8 V	0.44 A	1	98.9 W
225.85 V	0.44 A	1	98.94 W
225.9 V	0.44 A	1	98.94 W
225.95 V	0.44 A	1	98.99 W
225.75	0.44 A	1	98.9 W

5.2.10. Skenario Uji Coba Perbandingan Akurasi Lampu 3 Watt

Pada uji coba ini menghitung keberhasilan dan tingkat *error* dari PZEM-004T dan lampu 3 Watt. Dari hasil uji coba perbandingan akurasi dapat disimpulkan bahwa akurasi memiliki rata-rata 90.4%.

Table 5.6 Uji Coba Perbandingan Akurasi Lampu 3 Watt

Daya		
PZEM-004T	Lampu	Akurasi
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.71 W	3 W	90 %
2.82 W	3 W	94 %

5.2.11. Skenario Uji Coba Perbandingan Akurasi Kipas Angin 20 Watt

Pada uji coba ini menghitung keberhasilan dan tingkat *error* dari PZEM-004T dan kipas angin 20 watt. Dari

hasil uji coba perbandingan akurasi dapat disimpulkan bahwa akurasi memiliki rata-rata 62.2%.

Table 5.7 Uji Coba Perbandingan Akurasi Kipas Angin 20 Watt

Daya		
PZEM-004T	Kipas Angin	Akurasi
12.49 W	20 W	62%
12.49 W	20 W	62%
12.63 W	20 W	63%
12.5 W	20 W	62%
12.48 W	20 W	62%
12.48 W	20 W	62%
12.48 W	20 W	62%
12.71 W	20 W	63%
12.46 W	20 W	62%
12.47 W	20 W	62%

5.2.12. Skenario Uji Coba Perbandingan Akurasi Lampu 25 Watt

Pada uji coba ini menghitung keberhasilan dan tingkat *error* dari PZEM-004T dan lampu 25 watt. Dari hasil uji coba perbandingan akurasi dapat disimpulkan bahwa akurasi memiliki rata-rata 97.7%.

Daya		
PZEM-004T	Lampu	Akurasi
25.82 W	25 W	97%
25.57 W	25 W	98%
25.54 W	25 W	98%
25.54 W	25 W	98%
25.56 W	25 W	98%
25.63 W	25 W	98%
25.64 W	25 W	98%
25.68 W	25 W	98%
25.91 W	25 W	97%
25.91 W	25 W	97%

5.2.13. Skenario Uji Coba Perbandingan Akurasi Lampu 100 Watt

Pada uji coba ini menghitung keberhasilan dan tingkat *error* dari PZEM-004T dan lampu 100 watt. Dari hasil uji coba perbandingan akurasi dapat disimpulkan bahwa akurasi memiliki rata-rata 98.5%.

Table 5.8 Uji Coba Perbandingan Akurasi Lampu 100 Watt

Daya		
PZEM-004T	Lampu	Akurasi
99.03 W	100 W	99%
99.08 W	100 W	99%
99.03 W	100 W	99%
99.03W	100 W	99%
99.03 W	100 W	99%
98.9 W	100 W	98%
98.94 W	100 W	98%
98.94 W	100 W	98%
98.99 W	100 W	98%
98.8 W	100 W	98%

[Halaman ini sengaja dikosongkan]

BAB VI

KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan yang diperoleh selama pengerjaan Tugas Akhir ini berdasarkan hasil pengujian dan hal lainnya yang telah dilakukan. Selain itu, juga terdapat beberapa saran terhadap tugas akhir ini untuk pengembangan ke depannya.

6.1. Kesimpulan

Berdasarkan pengerjaan Tugas Akhir ini dan hasil yang didapatkan pada tahap uji coba, dapat diambil beberapa kesimpulan sebagai berikut.

1. *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik yang sudah dihubungkan dengan *microcontroller* bisa diterapkan dengan mengendalikan nyala atau matinya arus atau listrik sesuai dengan yang diinginkan.
2. *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik bisa diterapkan dengan melakukan pengawasan pada peralatan elektronik yang aktif dan tidak aktif dengan yang diinginkan.
3. *Prototype* Sistem Cerdas untuk Pengawasan dan Pengendalian Penggunaan Daya Listrik bisa diterapkan dengan melakukan pengawasan pada *website* dengan mencatat daya, kwh, biaya per hari, dan biaya per bulannya dengan yang diinginkan.

6.2. Saran

Terdapat beberapa saran terkait Tugas Akhir ini yang diharapkan bisa membuat Tugas Akhir ini menjadi lebih baik. Saran-saran tersebut antara lain:

1. Dapat dilakukan pengembangan terhadap fitur-fitur yang sudah ada.

2. Untuk memaksimalkan penggunaan daya listrik, perlu dilakukan perlakuan khusus untuk menaikkan faktor daya menjadi minimal 0.85, dan maksimal 1.

DAFTAR PUSTAKA

- [1] A. Faudin, "Pengenalan tentang WEMOS D1 MINI," [Online]. Available: <https://www.nyebarilmu.com/pengenalan-tentang-wemos-d1-mini/>.
- [2] "Relay | industri3601," [Online]. Available: <https://industri3601.wordpress.com/relay/>.
- [3] J. Sitepu, "Membaca Sensor PZEM-004t dengan nodemcu Arduino," [Online]. Available: <https://mikroavr.com/sensor-pzem-004t-arduino/>.
- [4] Elektronika Dasar, "LCD (Liquid Cristal Display) Dot Matrix 2x16 M1632," [Online]. Available: <https://elektronika-dasar.web.id/lcd-liquid-cristal-display-dot-matrix-2x16-m1632/>.
- [5] A. "Mengenal Software Arduino (IDE)," [Online]. Available: <http://sk.raharja.ac.id/2017/05/mengenal-software-arduino-ide/>.
- [6] R. Y. W, "Ekstensi dan Tema Visual Studio Code yang Saya Gunakan," [Online]. Available: <https://medium.com/kode-dan-kodean/ekstensi-dan-tema-visual-studio-code-yang-saya-gunakan-6c3555762816>.
- [7] phpMyAdmin, "Mengenal phpMyAdmin Dan Kegunaannya," [Online]. Available: <https://www.termasmedia.com/database/phpmyadmin/131-mengenal-phpmyadmin-dan-kegunaannya.html>.

- [8] Andriana, Z. and H. Baehaqi, "Sistem kWh Meter Digital Menggunakan Modul PZEM-004T," *Jurnal TIARSIE*, vol. 16, pp. 29-34, 2019.
- [9] K. Chooruang and K. Meekul, "Design of an IoT Energy Monitoring System," *International Conference on ICT and Knowledge Engineering*, vol. 2018, pp. 48-51, 2019.
- [10] D. Handarly and J. Lianda, "Sistem Monitoring Daya Listrik Berbasis IoT (Internet of Thing)," *Journal of Electrical Electronic Control and Automotive Engineering (JEECAE)*, pp. 205-208, 2018.
- [11] M. Maksimović, "IOT CONCEPT APPLICATION IN EDUCATIONAL SECTOR USING COLLABORATION," *Teaching, Learning and Teacher Education*, vol. 1, pp. 137-150, 2017.
- [12] H. L. Sadewa, H. Sujaini and R. . D. Nyoto, "Implementasi Mikrokontroler pada Sistem Kontrol Peralatan Listrik dan Monitoring Rumah Berbasis Website," *Jurnal Edukasi dan Penelitian Informatika*, pp. 99-105, 2015.
- [13] T. Syamsuri, H. Buwono and R. Amalia, "APLIKASI MIKROKONTROLER DALAM SISTEM KONTROL DAN MONITORING ENERGI LISTRIK," *JURNAL ELTEK*, pp. 107-119, 2019.

LAMPIRAN

```
#include <PZEM004Tv30.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266HTTPClient.h>
#include <ArduinoJson.h>
#include <LiquidCrystal_I2C.h>

#define kipas    D7
#define lampu    D0
#define indikator D4

#define INPUT_SIZE 3

const char* wifiName = "FARRAS";
const char* wifiPass = "sigitpo64";
const char *host = "http://tugasakhirinformatika.xyz/";

LiquidCrystal_I2C lcd(0x27, 20, 4);
PZEM004Tv30 pzem(D3, D4); // (RX,TX)connect to
TX,RX of PZEM
PZEM004Tv30 pzem2(D5, D6); // (RX,TX) connect to
TX,RX of PZEM

int httpCode;
String payload;
String senddata;
String linksend;

float kwh=0;
float voltage1=0;
```

```

float current1=0;
float energy1=0;
float pf1=0;
float voltage2=0;
float current2=0;
float energy2=0;
float pf2=0;
float voltage3=0;
float energy3=0;

//TAMBAHAN
// Get next command from Serial (add 1 for final 0)
char input[INPUT_SIZE + 1];
byte size = Serial.readBytes(input, INPUT_SIZE);
// Add the final 0 to end the C string
int kondlampu, kondkipas;
char *token;
const char s[2] = ",";

// ----- Function Pack Data From Web Server
void packData(){
    token = strtok(input, ",");
    kondlampu = atoi(token);
    Serial.print("Arus 2: ");
    Serial.println(kondlampu);
    token = strtok(0, ",");
    kondkipas = atoi(token);
    Serial.println("Arus 1: ");
    Serial.println(kondkipas);
}

void setup()

```

```
{
  pinMode(indikator,OUTPUT);
  digitalWrite(indikator,LOW);
  lcd.begin();
  lcd.backlight();
  lcd.clear();
  Serial.begin(115200);
  delay(10);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(wifiName);
  WiFi.begin(wifiName, wifiPass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  pinMode(lampu,OUTPUT);
  pinMode(kipas,OUTPUT);
  digitalWrite(lampu,HIGH);
  digitalWrite(kipas,HIGH);
  digitalWrite(indikator,HIGH);

  kondlampu = 0;
  kondkipas = 0;
}
```

```

void loop()
{
  HTTPClient http;
  Serial.print("Request Link:");

  Serial.println("http://tugasakhirinformatika.xyz/get_status
.php");

  http.begin("http://tugasakhirinformatika.xyz/get_status.ph
p");

  httpCode = http.GET();
  payload = http.getString();

  Serial.print("Response Code:");
  Serial.println(httpCode);

  Serial.print("Returned data from Server:");
  Serial.println(payload);

  if(httpCode == 200)
  {
    payload.toCharArray(input, sizeof(input));
    Serial.print("Returned data from Server:");
    packData();

    Serial.println("////////////////////////////////////////");
    Serial.println(kondlampu);
    Serial.println(kondkipas);

    Serial.println("////////////////////////////////////////");

```



```
    if(kondlampu==0 &&
kondkipas==0){ digitalWrite(lampu,HIGH);digitalWrite(k
ipas,HIGH);}
    else if(kondlampu==0 &&
kondkipas==1){ digitalWrite(lampu,HIGH);digitalWrite(k
ipas,LOW);}
    else if(kondlampu==1 &&
kondkipas==0){ digitalWrite(lampu,LOW);digitalWrite(ki
pas,HIGH);}
    else if(kondlampu==1 &&
kondkipas==1){ digitalWrite(lampu,LOW);digitalWrite(ki
pas,LOW);}
}

else
{
    Serial.println("Error in response");
}
http.end();
delay(1500);

voltage1 = pzem.voltage();
current1 = pzem.current();
pf1 = pzem.pf();

voltage2 = pzem2.voltage();
current2 = pzem2.current();
pf2 = pzem2.pf();

energy1 = 3*voltage1*current1*pf1;
energy2 = 3*voltage2*current2*pf2;
```

```
lcd.setCursor(0,0);lcd.print("          ");
lcd.setCursor(0,1);lcd.print("          ");
lcd.setCursor(0,2);lcd.print("          ");
lcd.setCursor(0,3);lcd.print("          ");

if(!isnan(voltage1))
{lcd.setCursor(0,0);lcd.print(voltage1,2);lcd.setCursor(7,
0);lcd.print("V");}
else
{lcd.setCursor(0,0);lcd.print("0");lcd.setCursor(7,0);lcd.p
rint("V");}

if(!isnan(current1))
{lcd.setCursor(0,1);lcd.print(current1,4);lcd.setCursor(7,1
);lcd.print("A");}
else
{lcd.setCursor(0,1);lcd.print("0");lcd.setCursor(7,1);lcd.p
rint("A");}

if(!isnan(pf1))
{lcd.setCursor(0,2);lcd.print(pf1,4);lcd.setCursor(7,2);lcd
.print("");}
else
{lcd.setCursor(0,2);lcd.print("0");lcd.setCursor(7,2);lcd.p
rint("");}

if(!isnan(energy1))
{lcd.setCursor(0,3);lcd.print(energy1/3);lcd.setCursor(7,3
);lcd.print("Ws");}
```

```
else
{lcd.setCursor(0,3);lcd.print("0");lcd.setCursor(7,3);lcd.p
rint("Ws");}

    if(!isnan(voltage2))
{lcd.setCursor(10,0);lcd.print(voltage2,2);lcd.setCursor(1
7,0);lcd.print("V");}
    else
{lcd.setCursor(10,0);lcd.print("0");lcd.setCursor(17,0);lcd
.print("V");}

    if(!isnan(current2))
{lcd.setCursor(10,1);lcd.print(current2,4);lcd.setCursor(1
7,1);lcd.print("A");}
    else
{lcd.setCursor(10,1);lcd.print("0");lcd.setCursor(17,1);lcd
.print("A");}

    if(!isnan(pf2))
{lcd.setCursor(10,2);lcd.print(pf2,4);lcd.setCursor(17,2);l
cd.print("");}
    else
{lcd.setCursor(10,2);lcd.print("0");lcd.setCursor(17,2);lcd
.print("");}

    if(!isnan(energy2))
{lcd.setCursor(10,3);lcd.print(energy2/3);lcd.setCursor(1
7,3);lcd.print("Ws");}
    else
{lcd.setCursor(10,3);lcd.print("0");lcd.setCursor(17,3);lcd
.print("Ws");}
```

```
Serial.print("arus 1 =");
  Serial.println(current1);
  Serial.print("arus 2 =");
  Serial.println(current2);
  voltage3 = (voltage1+voltage2)/2;
  kwh = (energy1+energy2);

  senddata = "?t=" + String(voltage3,2) + "&a1=" +
String(current1,2) + "&pf1=" + String(pf1,2) +
"&energy1=" + String(energy1/3,2) + "&a2=" +
String(current2,2) + "&pf2=" + String(pf2,2)+
"&energy2=" + String(energy2/3,2) + "&kwh=" +
String(kwh,3);

  linksend =
"http://tugasakhirinformatika.xyz/simpan.php" +
senddata;

//http://tugasakhirinformatika.xyz/simpan.php?t=1&a1=2
&pf1=3&energy1=4&a2=5&pf2=6&energy2=7&kwh=8

  http.begin(linksend);

  httpCode = http.GET();
  payload = http.getString();

  Serial.println(httpCode);
  Serial.println(payload);
  http.end();
  delay(1500);
}
```

BIODATA PENULIS



Penulis lahir di Gresik, 23 Juni 1997. Penulis telah menempuh pendidikan dasar di SDN Sidokumpul 2 Gresik, kemudian untuk pendidikan menengah pertama di SMPN 2 Gresik dan di jenjang menengah atas di SMAN 1 Kebomas. Sejak kecil, penulis suka dengan hal-hal terkait perkembangan teknologi dan perkembangan komputer. Hal tersebut juga yang mendasari penulis melanjutkan pendidikan sarjana S1 di Jurusan Informatika, Fakultas Teknologi

Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Selama kuliah, penulis aktif berorganisasi menjadi staf Ahli Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) ITS 2018/2019, staf Perkab Schematics ITS 2017/2018, staf Minat Bakat 2018/2019 dan staf Ahli Perkab Schematics ITS 2018/2019.

Penulis dalam menyelesaikan pendidikan S1 mengambil rumpun mata kuliah (RMK) Arsitektur Jaringan Komputer. Untuk komunikasi, penulis dapat dihubungi melalui surel: farrasabbani1@gmail.com.

