

# **Smart House Electricity Monitoring Based on *Raspberry Pi***

## **Tugas Akhir**

Disampaikan Sebagai Bagian Dari Persyaratan Kelulusan Diploma 3  
Program Studi Teknik Komputer

**Oleh:**

13315003 Ika Monica Telaumbanua  
13315026 Christopher Mika Andrew Siahaan  
13315028 Richad Harianja



**Institut Teknologi Del  
2017/2018**

**Lembar Pengesahan Tugas Akhir**  
**Institut Teknologi Del**

**Smart House Electricity Monitoring Based on**  
***Raspberry Pi***

**Oleh:**

13315003 Ika Monica Telaumbanua  
13315026 Christopher Mika Andrew Siahaan  
13315028 Richad Harianja

Sitoluama, 07 Juni 2018

Pembimbing

Gerry Italiano Wowiling, S.Tr.Kom

**Dinyatakan memenuhi syarat dan karenanya disetujui dan disahkan**  
**Sebagai**  
**Laporan Tugas Akhir Diploma 3**  
**Program Studi Teknik Komputer**  
**Institut Teknologi Del**

## Prakata

Puji dan syukur kepada Tuhan Yang Maha Esa, atas rahmat yang diberikan kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik. Laporan Tugas Akhir ini bertujuan untuk memberikan informasi bagi pembaca mengenai *Smart House Electricity Monitoring based on Raspberry Pi*. Laporan Tugas Akhir ini merupakan syarat kelulusan Diploma III Institut Teknologi Del.

Penulis menyampaikan terimakasih kepada berbagai pihak, khususnya kepada:

1. Orang tua, saudara, dan teman dekat yang telah memberikan dukungan dalam pengerjaan tugas akhir.
2. Bapak Gerry Italiano Wowiling selaku dosen pembimbing tugas akhir yang telah memberikan saran dan arahan dalam menyelesaikan tugas akhir dan banyak meluangkan waktu dan tenaga untuk membimbing penulis selama penyusunan tugas akhir.
3. Bapak/Ibu dosen khususnya yang mengajar di Jurusan Teknik Komputer yang telah membekali penulis dengan beberapa disiplin ilmu yang berguna.
4. Teman-teman seperjuangan Mahasiswa Teknik Komputer Institut Teknologi Del Angkatan 2015 yang telah banyak berdiskusi dan bekerjasama dengan penulis selama masa pendidikan.

Penulis menyadari bahwa pada tugas akhir ini masih memiliki kekurangan dan kelemahan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun agar tugas akhir yang dibuat menjadi lebih baik kedepannya. Semoga keberadaan tugas akhir ini dapat bermanfaat dan menambah wawasan bagi semua pihak, khususnya tentang *Smart House Electricity Monitoring based on Raspberry Pi*. Akhir kata penulis mengucapkan terimakasih.

Sitoluama, 07 Juni 2018

Ika Monica Telaumbanua

Christopher Mika Andrew Siahaan

Richad Harianja

## Abstrak

Saat ini, banyaknya alat-alat rumah tangga yang membutuhkan sumber daya listrik membuat pemakaian listrik semakin meningkat. Namun dalam pemakaiannya, pengguna seringkali berlebihan dan tidak dapat mengontrolnya dengan baik. Misalnya, pengguna yang lupa atau malas mematikan alat-alat elektronik secara manual saat berada dalam rumah ataupun meninggalkan rumah. Pemakaian listrik yang berlebihan tersebut sangat merugikan bagi pengguna baik dalam segi daya listrik maupun biaya yang dikeluarkan.

Sistem monitoring peralatan listrik secara otomatis menjadi penyelesaian terbaik untuk mengatasi masalah penggunaan listrik secara berlebih. Dalam penelitian ini dibuat sebuah sistem monitoring penggunaan peralatan listrik secara otomatis dengan judul *Smart House Electricity Monitoring Based on Raspberry Pi* yang bertujuan untuk melakukan monitoring dan mengendalikan penggunaan peralatan listrik dalam rumah. Penelitian ini bermanfaat bagi pengguna yang dalam menangani pemakaian listrik secara berlebihan dan tidak tepat.

Metode monitoring penggunaan listrik yang digunakan pada penelitian ini adalah mendeteksi daya yang dikonsumsi perangkat listrik menggunakan sensor *Current Transformer* yang terhubung ke *Arduino* kemudian *Arduino* akan mengirimkan daya tersebut *Raspberry Pi* untuk disimpan dan diolah. *Raspberry Pi* akan mengirimkan data yang telah diolah tersebut ke database dan akan ditampilkan pada aplikasi berbasis website yang dibangun menggunakan *framework Codeigniter* agar pengguna dapat melakukan monitoring terhadap perangkat listrik dengan mudah.

Pada Tugas Akhir ini akan dibahas bagaimana cara mengimplementasikan *Smart House Electricity Monitoring Based on Raspberry Pi* dimana user dapat menggunakan perangkat listrik dalam rumah secara hemat dan tepat melalui aplikasi website yang dapat diakses melalui laptop atau *smartphone*.

Kata kunci : Sensor *Current Transformer*, *Arduino*, *Raspberry Pi*, *Relay* dan *Codeigniter*.

# Daftar Isi

|  |    |
|--|----|
| Prakata.....   | 2  |
| Abstrak.....   | 3  |
| Bab I Pendahuluan .....  | 9  |
| 1.1. Latar Belakang .....  | 9  |
| 1.2. Tujuan.....   | 10 |
| 1.3. Ruang Lingkup .....   | 10 |
| 1.4. Istilah, Defenisi dan Singkatan.....  | 10 |
| 1.5. Metodologi Penelitian .....   | 11 |
| 1.6. Sistematika Penyajian .....   | 12 |
| BAB II Tinjauan Pustaka .....  | 14 |
| 2.1. Arus Listrik .....  | 14 |
| 2.2. <i>Relay</i> .....  | 14 |
| 2.3. Sensor Current Transformer .....  | 15 |
| 2.4. <i>Raspberry Pi</i> .....   | 16 |
| 2.5. Aplikasi <i>Website</i> .....   | 16 |
| 2.6. <i>CodeIgniter</i> .....  | 17 |
| 2.7 <i>Arduino UNO</i> .....   | 19 |
| 2.8 LM358.....   | 21 |
| Bab III Perancangan Sistem.....  | 22 |
| 3.1 Perancangan Sistem Secara Umum .....   | 22 |
| 3.1.1 Perancangan Desain <i>Raspberry Pi</i> dan <i>Relay</i> untuk mengendalikan perangkat listrik .....                            | 24 |
| 3.1.2 Perancangan Desain <i>Arduino</i> dan Sensor CT untuk mengukur arus perangkat listrik .....                                    | 24 |
| 3.1.3 Perancangan Desain Komunikasi Serial antara <i>Raspberry Pi</i> dan <i>Arduino</i> .....                                       | 25 |
| 3.1.4 <i>Block diagram</i> perancangan perangkat keras .....   | 25 |
| 3.1.5 <i>Flowchart</i> diagram mendapatkan data daya arus listrik pada perangkat listrik dan ditampilkan pada aplikasi website ..... | 26 |
| 3.1.6 <i>Flowchart diagram</i> mengendalikan perangkat secara otomatis .....   | 27 |
| 3.2 Perancangan aplikasi.....  | 28 |
| 3.2.1 <i>Business Process</i> .....  | 28 |
| 3.2.2 <i>Use Case Diagram</i> .....  | 33 |
| 3.2.3 <i>Use Case Scenario</i> .....   | 33 |

|   |    |
|---|----|
| 3.3 Perangkat yang digunakan .....  | 38 |
| 3.3.1 Perangkat Keras ( <i>Hardware</i> ) yang Digunakan.....                                     | 39 |
| 3.3.2 Perangkat Lunak ( <i>Software</i> ) yang Digunakan .....                                    | 42 |
| Bab IV Implementasi .....   | 44 |
| 4.1. Implementasi <i>Prototype</i> Sistem .....   | 44 |
| 4.1.1. Implementasi <i>Raspberry Pi</i> dan <i>Relay</i> untuk mengendalikan perangkat listrik .. | 46 |
| 4.1.2. Implementasi <i>Arduino</i> dan Sensor CT untuk mengukur perangkat listrik ..              | 46 |
| 4.1.3. Komunikasi serial antara <i>Raspberry Pi</i> dan <i>Arduino</i> .....                      | 47 |
| 4.2. Implementasi Aplikasi.....   | 49 |
| 4.2.1. <i>Layer Login</i> .....   | 49 |
| 4.2.2. <i>Dashboard</i> .....   | 50 |
| 4.2.3. <i>Layer Log User</i> .....  | 52 |
| 4.2.4. <i>Layer Details Room</i> .....  | 52 |
| 4.2.5. <i>Layer Users</i> .....   | 54 |
| 4.2.6. <i>Layer Record</i> .....  | 55 |
| 4.2.7. <i>Layer Report</i> .....  | 56 |
| Bab V Pengujian dan Analisis .....  | 58 |
| 5.1. Pengujian.....   | 58 |
| 5.1.1. Pengujian <i>Prototype</i> .....   | 58 |
| 5.1.2. Pengujian Sensor <i>Current Transformer</i> .....  | 59 |
| 5.1.3. Pengujian Aplikasi .....   | 65 |
| 5.2. Analisis Hasil Pengujian .....   | 72 |
| Bab VI Kesimpulan dan Saran.....  | 73 |
| 6.1. Kesimpulan.....  | 73 |
| 6.2. Saran.....   | 73 |
| Daftar Pustaka.....   | 74 |
| Lampiran .....  | 77 |

## Daftar Tabel

|  |    |
|--|----|
| Tabel 1 Istilah dan Defenisi .....                                   | 11 |
| Tabel 2 Singkatan .....  | 11 |
| Tabel 3 spesifikasi data teknis pada <i>board Arduino UNO</i> .....  | 20 |
| Tabel 4 <i>Use case scenario login</i> .....                         | 33 |
| Tabel 5 <i>Use case scenario ganti password</i> .....                | 34 |
| Tabel 6 Menambah <i>user</i> baru .....                              | 35 |
| Tabel 7 <i>Use Case Scenario</i> Melihat <i>Detail</i> Ruangan ..... | 36 |
| Tabel 8 <i>Use Case Scenario</i> Melihat Pemakaian daya Listrik..... | 37 |
| Tabel 9 <i>Use Case Scenario</i> Mengontrol Listrik .....            | 38 |
| Tabel 10 Perangkat lunak yang digunakan .....                        | 43 |
| Tabel 11 Penggunaan Pin <i>Arduino Uno</i> .....                     | 48 |
| Tabel 12 Penggunaan Pin <i>Raspberry Pi</i> .....                    | 49 |
| Tabel 13 Skenario Pengujian Sensor CT .....                          | 60 |
| Tabel 14 Hasil Pengujian Sensor CT terhadap Lampu .....              | 63 |
| Tabel 15 Hasil Pengujian Sensor CT terhadap Ekstensi.....            | 64 |

## Daftar Gambar

|  |    |
|--|----|
| Gambar 1 Struktur <i>Relay</i> [6].....  | 15 |
| Gambar 2 Struktur CodeIgniter .....  | 18 |
| Gambar 3 Konfigurasi LM358 IC.....   | 21 |
| Gambar 4 Gambaran umum Smart House Electricity .....   | 22 |
| Gambar 5 Desain <i>Raspberry Pi</i> dan <i>Relay</i> .....   | 24 |
| Gambar 6 Desain <i>Arduino</i> dan <i>Sensor CT</i> .....  | 25 |
| Gambar 7 Komunikasi serial antara <i>Raspberry Pi</i> dan <i>Arduino</i> .....   | 25 |
| Gambar 8 Perancangan perangkat keras .....   | 25 |
| Gambar 9 <i>Flowchart</i> mendapatkan daya arus listrik pada perangkat listrik dan ditampilkan pada aplikasi website ..... | 26 |
| Gambar 10 <i>Flowchart</i> mengendalikan alat secara otomatis .....  | 27 |
| Gambar 11 <i>Business Process</i> sistem secara umum.....  | 29 |
| Gambar 12 <i>Business Process login</i> .....  | 30 |
| Gambar 13 <i>Business process</i> melihat <i>detail</i> ruangan .....  | 31 |
| Gambar 14 <i>Business Process</i> melihat pemakaian daya listrik.....  | 31 |
| Gambar 15 <i>Business Process</i> mengontrol pemakaian daya listrik .....  | 32 |
| Gambar 16 <i>Use case</i> diagram aplikasi.....  | 33 |
| Gambar 17 <i>Raspberry Pi</i> .....  | 40 |
| Gambar 18 <i>DI-Relay</i> 8.....   | 41 |
| Gambar 19 Current Transformer Sensor [17].....   | 41 |
| Gambar 20 <i>Arduino UNO</i> [18] .....  | 42 |
| Gambar 21 LM358 [19].....  | 42 |
| Gambar 22 Implementasi <i>Prototype</i> Rumah.....   | 44 |
| Gambar 23 Gambaran skematik secara umum .....  | 45 |
| Gambar 24 Implementasi <i>Raspberry Pi</i> dan <i>Relay</i> .....  | 46 |
| Gambar 25 Implementasi <i>Arduino</i> dan <i>Sensor CT</i> .....   | 47 |
| Gambar 26 Implementasi Komunikasi serial antara <i>Raspberry Pi</i> dan <i>Arduino</i> .....                               | 48 |
| Gambar 27 <i>Layer Login</i> .....   | 50 |
| Gambar 28 <i>Dashboard</i> untuk <i>client</i> .....   | 50 |
| Gambar 29 <i>Dashboard</i> untuk <i>admin</i> .....  | 51 |



|   |    |
|---|----|
| Gambar 30 Layer Log User .....                                | 52 |
| Gambar 31 Menu <i>Details Room</i> .....                      | 53 |
| Gambar 32 Menu <i>Item Management</i> .....                   | 53 |
| Gambar 33 Menu <i>View Details</i> .....                      | 54 |
| Gambar 34 Menu <i>User Management</i> .....                   | 55 |
| Gambar 35 menu <i>Add User</i> .....                          | 55 |
| Gambar 36 Layer <i>Record (per item)</i> .....                | 56 |
| Gambar 37 Layer Record (Per tanggal) .....                    | 56 |
| Gambar 38 Hasil Pengujian <i>Prototype</i> Rumah.....         | 59 |
| Gambar 39 Pengujian selama 15 menit .....                     | 62 |
| Gambar 40 Pengujian selama 30 menit .....                     | 62 |
| Gambar 41 Pengujian selama 60 menit .....                     | 63 |
| Gambar 42 Daya yang dikonsumsi perangkat melebihi batas ..... | 65 |
| Gambar 43 Tampilan aplikasi ketika pukul 23.59 .....          | 65 |
| Gambar 44 <i>Layer Details Room</i> .....                     | 66 |
| Gambar 45 Pengujian tombol <i>ON/OFF</i> pada aplikasi .....  | 67 |
| Gambar 46 Hasil Pengujian untuk mengedit perangkat .....      | 67 |
| Gambar 47 Menu <i>Add New User</i> .....                      | 68 |
| Gambar 48 Notifikasi jika user berhasil ditambahkan.....      | 68 |
| Gambar 49 Hasil Pengujian Layer Users.....                    | 69 |
| Gambar 50 <i>Layer Log User</i> .....                         | 69 |
| Gambar 51 Hasil Pengujian <i>Layer Log User</i> .....         | 70 |
| Gambar 52 <i>Layer Record</i> .....                           | 70 |
| Gambar 53 Biaya Tagihan pada 21 mei 2018.....                 | 71 |
| Gambar 54 Hasil Pengujian <i>Layer Record</i> .....           | 71 |

# Bab I

## Pendahuluan

Pada bab ini berisi penjelasan mengenai latar belakang, tujuan pelaksanaan, ruang lingkup, pendekatan yang dilakukan dan sistematika penyajian Tugas Akhir.

### 1.1. Latar Belakang

Listrik merupakan komponen yang sangat penting dalam membantu setiap pekerjaan umat manusia, mulai dari penerangan hingga pengaturan suhu ruangan dan lain sebagainya. Namun dalam penggunaannya sering kali tidak efisien, banyak orang yang membiarkan suatu perangkat listrik menyala padahal tidak digunakan. Contohnya lupa mematikan semua lampu atau perangkat lainnya saat pergi dari rumah. Hal ini tentu saja mengakibatkan pemborosan dalam penggunaan daya listrik.

Penelitian yang telah banyak dilakukan untuk melakukan pengontrolan alat-alat listrik pada dasarnya hanya sekedar mematikan dan menyalakan peralatan listrik saja, belum ada fitur lain [1]. Pada penelitian kali ini penulis membuat sebuah *prototype* sistem rumah pintar berbasis *Raspberry Pi* dengan user interface android yang akan melakukan komunikasi data melalui web service yakni *Smart House Electricity Monitoring based on Raspberry Pi*.

*Smart home* atau rumah pintar adalah sistem yang telah diprogram dan dapat bekerja dengan bantuan komputer untuk mengintegrasikan dan mengendalikan sebuah perangkat atau peralatan rumah secara otomatis dan efisien yang bertujuan untuk mempermudah penghematan daya energi, mendapatkan kenyamanan dan lain sebagainya [2]. *Smart Home* dapat digunakan untuk mengendalikan hampir semua perlengkapan dan peralatan di rumah, mulai dari pengaturan tata lampu hingga ke berbagai alat-alat rumah tangga, yang perintahnya dapat dilakukan dengan menggunakan suara, sinar merah infra, atau melalui kendali jarak jauh (*remote*).

Sistem kendali peralatan listrik rumah yang telah dibangun sebelumnya seperti sistem kendali peralatan listrik rumah menggunakan media pesan singkat (SMS), atau menggunakan *bluetooth* android maupun sistem kendali peralatan listrik rumah berbasis web tidak menyediakan data yang lengkap untuk user yang dapat membuat user mengetahui berapa besar daya listrik yang telah digunakan [3].

Dengan *Smart House Electricity Monitoring based on Raspberry Pi* masalah tersebut akan terpecahkan. Sistem ini membantu pengguna dalam melakukan *monitoring* terhadap pemakaian daya yang dikonsumsi alat-alat listrik dalam rumah.

## **1.2. Tujuan**

Tujuan dari Tugas Akhir ini adalah untuk membangun sebuah sistem yang digunakan untuk:

1. Melakukan monitoring terhadap penggunaan daya listrik dalam rumah dengan Sensor *Current Transformer* yang dapat menghitung besarnya daya listrik yang digunakan
2. Menghidupkan atau mematikan alat-alat listrik seperti lampu dan ekstensi yang ada dalam rumah menggunakan aplikasi berbasis *website*.

## **1.3. Ruang Lingkup**

Ruang lingkup dari pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. *Smart House Electricity Monitoring Based on Raspberry Pi* digunakan untuk sistem monitoring dan pengendalian pemakaian perangkat listrik dalam rumah.
2. Sistem diterapkan dalam bentuk *prototype/miniature* rumah.
3. Membangun sebuah aplikasi berbasis *website* sebagai *remote* pengontrol sistem yang hanya dapat diakses menggunakan jaringan lokal.
4. Ada dua User yang dapat mengakses *website* yaitu Admin dan Client.
5. User dapat mematikan dan menghidupkan perangkat listrik seperti lampu dan ekstensi melalui *website*.
6. Penambahan dan pengurangan perangkat listrik dalam rumah hanya dapat dilakukan oleh *developer*/penyedia layanan agar daya dari perangkat tersebut dapat diukur.

## **1.4. Istilah, Defenisi dan Singkatan**

Daftar istilah, defenisi dan singkatan yang terdapat dalam dokumen ini dapat dilihat pada **Tabel 1** dan **Tabel 2**.

**Tabel 1 Istilah dan Defenisi**

| <b>Istilah</b>   | <b>Defenisi</b>   |
|------------------|---|
| <i>User</i>      | Pengguna aplikasi.                                      |
| <i>Prototype</i> | Model produk sederhana                                  |
| <i>Testing</i>   | Melakukan pengecekan bahwa program bekerja dengan benar |
| <i>Hardware</i>  | Perangkat keras yang digunakan                          |
| <i>Software</i>  | Perangkat lunak yang digunakan                          |

**Tabel 2 Singkatan**

| <b>Singkatan</b> | <b>Keterangan</b>              |
|------------------|--------------------------------|
| CT               | <i>Current Transformer</i>     |
| SBC              | <i>Single-board circuit</i>    |
| MVC              | <i>Model, View, Controller</i> |
| CI               | <i>CodeIgniter</i>             |
| SoC              | <i>System on chip</i>          |
| RPi/ Raspi       | <i>Raspberry Pi</i>            |

### **1.5. Metodologi Penelitian**

Metodologi penelitian yang dilakukan dalam pengerjaan Tugas Akhir ini adalah sebagai berikut:

a. Tahap Observasi

Pada tahap ini berisi tentang tahap yang digunakan untuk mendukung perancangan dan pembangunan sistem yang dibangun yakni berupa:

1. Pencarian referensi dan sumber-sumber yang berhubungan dengan monitoring listrik menggunakan *Raspberry Pi*.
2. Memahami serta mempelajari perangkat yang digunakan baik perangkat lunak maupun perangkat keras.

b. Tahap Perancangan Sistem dan Implementasi

Pada tahap ini dirancang desain sistem dan pengimplementasian sistem. Desain meliputi perancangan terhadap sistem yang ada seperti, penentuan sistem operasi

yang tepat untuk dikonfigurasi dengan *Raspberry Pi*, serta penentuan aplikasi/*software* pendukung yang tepat.

c. Tahap Analisis dan Pengujian

Setelah sistem dirancang dan diimplementasikan, maka tahap selanjutnya adalah tahap analisa dan pengujian yang bertujuan mengetahui apakah sistem monitoring listrik dengan *Raspberry Pi* bekerja sesuai dengan tujuan pembuatan produk.

d. Tahap Pembuatan Laporan

Pada tahap ini dilakukan pembuatan atau penyusunan laporan akhir sesuai dengan ketentuan atau sistematika dokumen laporan yang telah ditetapkan oleh Institut Teknologi Del.

e. Tahap Kesimpulan dan Saran

Kesimpulan dan saran diuraikan pada tahap ini. Kesimpulan diambil dari analisis sistem, serta hasil dari implementasi dan pengujian yang dilakukan terhadap sistem. Sedangkan pada bagian saran berisi kajian terhadap penelitian untuk perbaikan maupun pengembangan sistem ke depannya.

## 1.6. Sistematika Penyajian

Adapun sistematika penulisan laporan Tugas Akhir ini adalah sebagai berikut:

### 1. BAB I PENDAHULUAN

Pada bab ini diuraikan Latar Belakang pemilihan judul, tujuan penelitian, lingkup pembahasan, pendekatan yang dilakukan dan sistematika penyajian sehingga permasalahan tersebut memiliki titik fokus dan tidak mengambang dari judul yang telah dibuat.

### 2. BAB II TINJAUAN PUSTAKA

Pada bab ini diuraikan informasi yang didapatkan dari berbagai pustaka yang berhubungan dengan topik kajian dalam Tugas Akhir untuk membantu dan memperluas pemahaman mengenai topik Tugas Akhir.

### 3. BAB III PERANCANGAN SISTEM

Pada bab ini dijelaskan analisis data yang dikumpulkan dan tahapan-tahapan yang dirancang berdasarkan analisis data yang telah dilakukan. Bab ini menjelaskan mengenai perancangan dokumen.

#### 4. BAB IV IMPLEMENTASI

Pada bab ini dilakukan implementasi terhadap sistem dan perangkat lunak. Pada implementasi berisi penjelasan bagaimana implementasi sistem/perangkat lunak dilakukan dan hasil yang didapatkan.

#### 5. BAB V PENGUJIAN DAN ANALISIS

Pada bab ini berisi pengujian yang dilakukan terhadap sistem yang dibangun dan analisis mengenai pengujian yang telah dilakukan.

#### 6. BAB VI KESIMPULAN DAN SARAN

Pada bab ini diberikan kesimpulan dan saran mengenai Tugas Akhir yang sudah dilakukan dan saran-saran untuk di masa yang akan datang.

## **BAB II**

### **Tinjauan Pustaka**

Pada bab ini membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik Tugas Akhir yang dikerjakan dan hal-hal yang berguna dalam proses perancangan dan implementasi sistem.

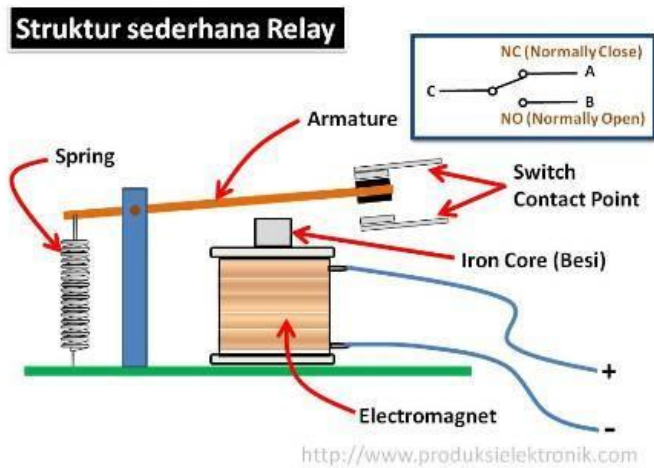
#### **2.1. Arus Listrik**

Arus listrik adalah banyaknya muatan listrik yang disebabkan dari pergerakan elektron-elektron, mengalir melalui suatu titik dalam sirkuit listrik tiap satuan waktu yang diukur dalam satuan *coloumb*/detik atau *Ampere* [4] . Arus listrik merupakan suatu hal yang penting bagi kehidupan manusia karena dapat membantu pekerjaan manusia menjadi lebih efektif dan efisien. Hampir semua peralatan rumah tangga menggunakan arus listrik untuk dapat bekerja. Contoh arus listrik dalam kehidupan sehari-hari berkisar dari yang sangat lemah dalam satuan mikroAmpere seperti di dalam jaringan tubuh hingga arus yang sangat kuat 1-200 kiloAmpere seperti yang terjadi pada petir.

#### **2.2. Relay**

*Relay* adalah Saklar yang dioperasikan secara listrik dan merupakan komponen Elektromekanikal yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak Saklar/Switch) [5] . *Relay* menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi.

*Relay* memiliki 4 buah bagian penting yakni *Electromagnet (Coil)*, *Armature*, *Switch Contact Point (Saklar)*, dan *Spring*. Besi (*Iron Core*) yang dililit oleh kumparan *Coil*, berfungsi untuk mengendalikan Besi tersebut. Apabila Kumparan *Coil* dialiri arus listrik, maka akan muncul gaya elektromagnetik yang dapat menarik *Armature* sehingga dapat berpindah dari posisi sebelumnya yakni tertutup (NC) menjadi posisi baru yakni terbuka (NO). Dalam posisi (NO) saklar dapat menghantarkan arus listrik. Pada saat tidak dialiri arus listrik, *Armature* akan kembali ke posisi awal (NC). Sedangkan *Coil* yang digunakan oleh *Relay* untuk menarik *Contact Poin* ke posisi *close* hanya membutuhkan arus listrik yang relatif cukup kecil.



Gambar 1 Struktur Relay [6]

### 2.3. Sensor Current Transformer

*Current transformer* (CT) atau Trafo Arus adalah alat yang digunakan untuk mengukur besaran arus pada instalasi tenaga listrik disisi primer (TET, TT dan TM) yang berskala besar dengan melakukan transformasi dari besaran arus yang besar menjadi besaran arus yang kecil sehingga dapat melakukan pengukuran atau metering dan menentukan batasan proteksi alat secara akurat dan teliti [7].

Trafo arus pengukuran untuk metering memiliki ketelitian tinggi pada daerah kerja (daerah pengenalnya) 5% - 120% arus nominalnya tergantung dari kelasnya dan tingkat kejenuhan yang relatif rendah dibandingkan trafo arus untuk proteksi. Sedangkan trafo arus untuk proteksi, memiliki ketelitian tinggi pada saat terjadi gangguan dimana arus yang mengalir beberapa kali dari arus pengenalnya dan tingkat kejenuhan cukup tinggi. Sensor CT akan dililitkan pada kabel di kutub positif untuk dapat mengukur besar arus listrik yang mengalir melalui kabel tersebut. Rumus perhitungan penggunaan daya yang dihasilkan oleh Sensor CT dan dikalkulasikan menjadi satuan KW adalah sebagai berikut [8] :

**Rumus mencari daya :**

$$I = \text{analogRead}(pin) * \left( \frac{1.1}{10^{-5}} \right);$$

$$P = I . V$$

$$J = P . 24 \text{ (hours)}$$

$$KW = \frac{J}{1000}$$



Keterangan:

$P = \text{Power (W)}$

$I = \text{Arus listrik (A)}$

$V = \text{Tegangan (V)} = 220 \text{ V}$

$J = \text{Energi}$

$KW = \text{Daya yang dikonsumsi}$

$\text{analogRead(pin)} = \text{input analog arduino}$

## 2.4. Raspberry Pi

*Raspberry Pi* yang disingkat dengan nama Raspi merupakan suatu perangkat keras yang berukuran kecil layaknya kartu kredit yang dapat digunakan untuk menjalankan berbagai program yang diinginkan pengguna [9]. *Raspberry Pi* yang dapat digunakan untuk menjalankan program perkantoran, permainan komputer, dan sebagai pemutar media hingga video beresolusi tinggi. Beberapa fungsi *Raspberry Pi* adalah sebagai berikut:

1. Sebagai Komputer Desktop Mini
2. Sebagai File Server
3. Sebagai Download Server
4. Sebagai Access Point
5. Sebagai Server DNS
6. Sebagai Multimedia Player

Pada tugas akhir ini *Raspberry Pi* berfungsi sebagai file server untuk menjadi media penyimpanan data. Data yang disimpan pada *Raspberry pi* akan dikirim ke aplikasi website dan ditampilkan ke database.

## 2.5. Aplikasi Website

*Website* adalah suatu kumpulan halaman yang menampilkan berbagai macam informasi teks, data, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya, baik itu yang bersifat statis maupun yang dinamis, yang membentuk satu rangkaian yang saling berkaitan [10]. Sebuah situs *web* biasanya ditempatkan setidaknya pada sebuah *server web* yang dapat diakses melalui jaringan seperti Internet, ataupun jaringan wilayah lokal (LAN) melalui alamat Internet yang dikenali sebagai URL.

Gabungan atas semua situs yang dapat diakses publik di Internet disebut pula sebagai *World Wide Web* atau lebih dikenal dengan singkatan WWW. Meskipun setidaknya halaman beranda situs Internet umumnya dapat diakses publik secara bebas, namun tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya. Beberapa situs *web* mewajibkan pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi anggota untuk dapat mengakses isi yang terdapat dalam situs web tersebut, misalnya situs-situs berita, layanan surel (*e-mail*), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, privasi, atau karena tujuan komersial tertentu.

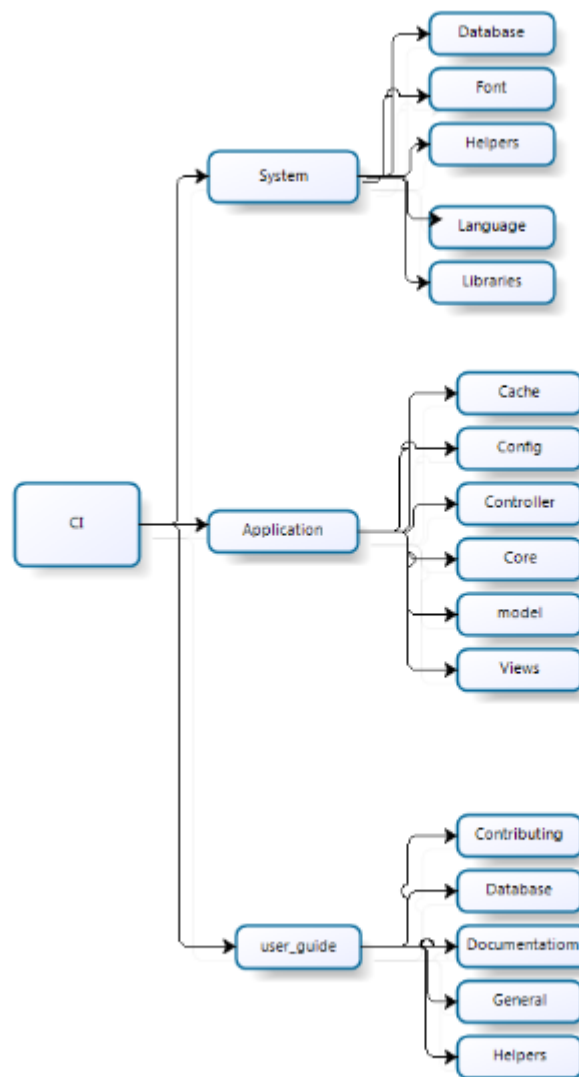
## 2.6. *CodeIgniter*

*CodeIgniter* menjadi sebuah framework PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web [11]. Selain ringan dan cepat, *CodeIgniter* juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya.

Ada beberapa kelebihan *CodeIgniter* (CI) dibandingkan dengan framework PHP lain,

- Performa sangat cepat: *Codeigniter* memiliki performa yang sangat cepat dibanding framework yang lain.
- Konfigurasi yang sangat minim: tentu saja untuk menyesuaikan dengan database dan keleluasaan routing tetap diizinkan melakukan konfigurasi dengan mengubah beberapa *file* konfigurasi seperti database, php atau autoload. Namun untuk menggunakan *codeigniter* dengan *setting standard*, hanya perlu mengubah sedikit saja *file* pada *folder config*.
- Banyak komunitas: dengan banyaknya komunitas CI, memudahkan pengguna untuk berinteraksi dengan yang lain.
- Dokumentasi yang sangat lengkap: Setiap paket instalasi *codeigniter* sudah disertai *user guide* yang sangat bagus dan lengkap untuk dijadikan permulaan. Bahasa yang digunakan juga mudah untuk dipahami.

Berikut ini adalah struktur folder dari *framework CodeIgniter* [14]:



Gambar 2 Struktur CodeIgniter

- *Folder Application*, merupakan *folder* yang pada dasarnya menyimpan aplikasi yang sedang kita buat.
- *Folder Cache*, merupakan *folder* yang menyimpan semua *cache* yang dibuat oleh *cache library*.
- *Folder Config*, merupakan *folder* yang menyimpan informasi mengenai konfigurasi aplikasi seperti *autoload*, *database*, *routes* dan lainnya.
- *Folder Controller*, merupakan *folder* menyimpan *controller-controller* aplikasi yang dapat digunakan untuk menyusun aktivitas program.
- *Folder Core*, adalah *folder* untuk memperluas *class-class* inti *codeigniter*.
- *Folder Helpers*, merupakan *folder* untuk menyimpan *helpers*.

- *Folder Hooks*, merupakan *folder* untuk menyimpan *hooks* untuk mengubah alur fungsi dari *core Codeigniter*
- *Folder Language*, merupakan *folder* untuk menyimpan bahasa-bahasa yang digunakan.
- *Folder Libraries*, merupakan *folder* untuk menyimpan *library*.
- *Folder Logs*, merupakan *folder* untuk menyimpan semua *error log* apabila *error log* diaktifkan.
- *Folder Models*, merupakan *folder* untuk menyimpan *models* yang akan mendefinisikan tabel dari database yang dapat kita gunakan oleh Controller yang kita buat untuk mengakses database.
- *Folder Third\_party*, merupakan *folder* untuk menyimpan fungsi fungsi tambahan dalam cara kerja *codeigniter*.
- *Folder Views*, merupakan *folder* untuk menyimpan tampilan dari aplikasi yang dibuat.
- *Folder System*, merupakan *folder* untuk menyimpan sistem inti dari *Codeigniter*.

## 2.7 Arduino UNO

*Arduino UNO* merupakan sebuah board mikrokontroler yang didasarkan pada ATmega328 (datasheet) yang mempunyai 14 pin digital input/output yaitu 6 input analog, sebuah osilator kristal 16 MHz, sebuah koneksi USB, sebuah power jack, sebuah ICSP header, sebuah tombol reset dan 6 diantaranya dapat digunakan sebagai output PWM [13]. *Arduino UNO* berbeda dari semua *board Arduino* sebelumnya, *Arduino UNO* tidak menggunakan *chip driver FTDI USB-to-serial*. *Board Arduino Uno* mempunyai sebuah resistor yang menarik garis 8U2 HWB ke *ground*, yang membuatnya lebih mudah untuk diletakkan ke dalam DFU mode. *Board Arduino UNO* memiliki fitur-fitur baru sebagai berikut:

- Pinout 1.0 ditambah pin SDA dan SCL yang dekat dengan pin AREF juga dua pin baru lainnya yang diletakkan dekat dengan pin RESET, IOREF yang memungkinkan *shield-shield* untuk menyesuaikan tegangan yang disediakan dari board.
- Sirkuit RESET yang lebih kuat

- Atmega 16U2 menggantikan 8U2

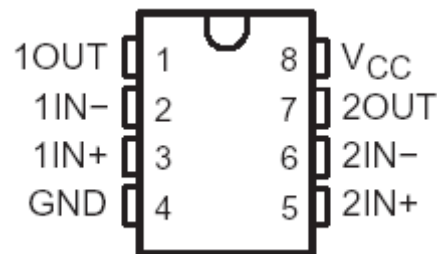
*Arduino UNO* dapat disuplai melalui koneksi USB atau dengan sebuah *power supply* eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (*non-USB*) dapat diperoleh dari sebuah *adaptor* AC ke DC atau *battery*. *Adaptor* dapat dihubungkan dengan mencolokkan sebuah *center-positive plug* yang panjangnya 2,1mm (milimeter) ke *power jack* dari *board*. Kabel *lead* dari sebuah *battery* dapat dimasukkan dalam *header/kepala pin* *Ground* (Gnd) dan pin *Vin* dari konektor *power*. *Board Arduino UNO* dapat beroperasi pada sebuah suplai eksternal 6 sampai 20 Volt. Jika disuplai dengan yang lebih kecil dari 7 Volt, kiranya pin 5 Volt mungkin mensuplai lebih kecil dari 5 Volt dan *board Arduino UNO* bisa menjadi tidak stabil. Jika menggunakan suplai yang lebih dari besar 12 Volt, *voltage regulator* bisa kelebihan panas dan membahayakan *board Arduino UNO*. *Range* yang direkomendasikan adalah 7 sampai 12 Volt. Adapun spesifikasi data teknis yang terdapat pada *board Arduino UNO R3* adalah sebagai berikut:

**Tabel 3 spesifikasi data teknis pada *board Arduino UNO***

|                                     |  |
|-------------------------------------|--|
| <b>Mikrokontroler</b>               | ATmega328  |
| <b>Tegangan Operasi</b>             | 5 V  |
| <b>Tegangan Input (recommended)</b> | 7-12 V   |
| <b>Tegangan Input (limit)</b>       | 6-20 V   |
| <b>Pin digital I/O</b>              | 14 (6 diantaranya pin PWM)                       |
| <b>Pin Analog input</b>             | 6 input pin 21                                   |
| <b>Pin PWM Digital I/O</b>          | 6  |
| <b>Arus DC per pin I/O</b>          | 40 mA  |
| <b>Arus DC untuk pin 3.3 V</b>      | 150 mA   |
| <b>Flash Memory</b>                 | 32 KB dengan 0.5 KB digunakan sebagai bootloader |
| <b>SRAM</b>                         | 2 KB   |
| <b>EEPROM</b>                       | 1 KB   |
| <b>Clock Speed</b>                  | 16 Mhz   |
| <b>Panjang</b>                      | 68.6 mm  |
| <b>Lebar</b>                        | 53.4 mm  |
| <b>Berat</b>                        | 25 g   |

## 2.8 LM358

LM358 adalah IC penguat operasional ganda (*dual operational amplifiers / Op-Amps*) yang terdiri atas dua penguat operasional *high-gain* dengan kompensator frekuensi yang independen [14] . LM358 dirancang untuk beroperasi cukup dari satu catu daya tunggal dengan rentang tegangan yang lebar untuk fleksibilitas penuh dalam menerapkan rancangan rangkaian elektronika. LM358 juga dapat menggunakan catu daya terpisah selama perbedaan tegangan antara kedua catu daya antara 3V hingga 32V dan  $V_{cc}$  setidaknya 1,5 volt lebih tinggi dibanding tegangan masukan moda-bersama (*input common-mode voltage*). Tarikan dari arus pasokan rendah (*low supply current drain*) bersifat independen dari besarnya tegangan catu daya. Pin Konfigurasi LM358 IC:



Gambar 3 Konfigurasi LM358 IC

Diagram pin LM358 IC terdiri dari 8 pin, di mana:

- Pin-1 dan pin-8 adalah o / p dari komparator
- Pin-2 dan pin-6 adalah pembalik i / id
- Pin-3 dan pin-5 adalah non inverting i / id
- Pin-4 adalah terminal GND
- Pin-8 adalah VCC +
- LM358 IC Pin Configuration
- LM358 IC Pin Configuration
- Fitur LM358 IC

Contoh aplikasi meliputi penguat transduser, blok penguat DC dan berbagai macam sirkuit penguat operasional konvensional lainnya. Implementasi rangkaian menjadi lebih mudah dengan sistem catu daya tunggal, sebagai contoh IC ini dapat beroperasi secara langsung dari catu daya standar 5V yang banyak digunakan di sistem digital dan mampu menyediakan antarmuka elektronika tanpa tambahan catu daya  $\pm 5$ -V maupun pembalik tegangan / voltage inverter.

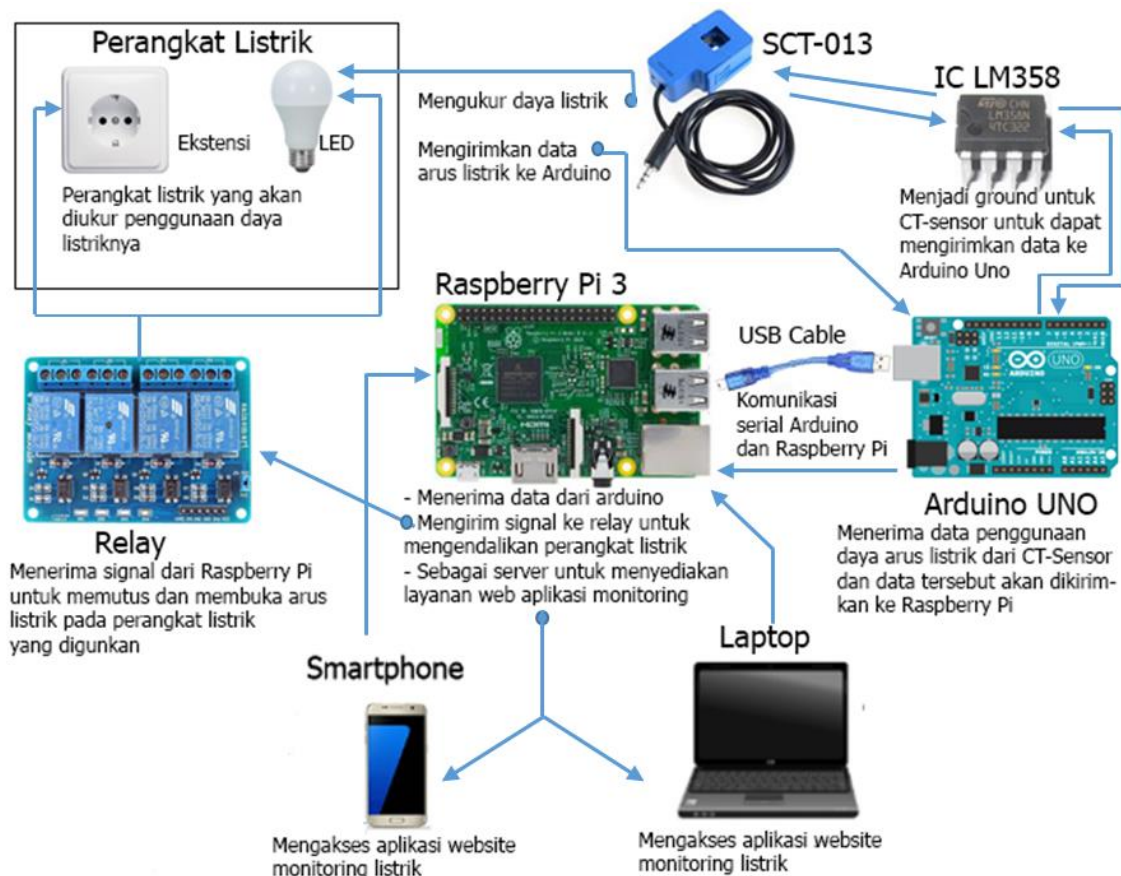
## Bab III

### Perancangan Sistem

Pada bab ini akan dijelaskan perancangan sistem yang dilakukan saat proses implementasi pada pengerjaan Tugas Akhir ini.

#### 3.1 Perancangan Sistem Secara Umum

Berikut ini ditampilkan gambaran sistem kendali listrik secara umum.



Gambar 4 Gambaran umum Smart House Electricity

Pada Gambar 4 ditampilkan gambaran umum sistem yang dibangun. Perangkat listrik yang digunakan adalah lampu dan ekstensi. Penggunaan daya dari perangkat listrik tersebut diukur oleh Sensor CT. Relay digunakan untuk memutus atau menyambung aliran listrik ketika menerima signal dari *Raspberry pi*. Relay dapat berkerja karena adanya sebuah kumparan yang berisi besi dan jika kumparan tersebut dialiri listrik maka kumparan tersebut menjadi magnet yang menarik plat sehingga terjadi aliran listrik.

Sensor CT berfungsi sebagai alat untuk mengukur arus listrik yang dikonsumsi dari tiap-tiap perangkat listrik yang terhubung langsung dengan Sensor CT dengan cara dijepitkan. *Lm358* berfungsi sebagai penyedia *ground* bagi Sensor CT. Sensor CT hanya memiliki kabel. Kabel pertama berfungsi untuk menyediakan daya bagi Sensor CT dan kabel lainnya berfungsi untuk menyediakan pengiriman data pada *Arduino UNO* melalui *analog* yang terdapat pada *board Arduino UNO*. Daya arus listrik yang didapat akan dikirimkan ke *Arduino UNO*. *Arduino UNO* berfungsi untuk menerima data berupa daya arus listrik dari Sensor CT. Data yang diterima oleh *Arduino UNO* tersebut akan dikirimkan ke *Raspberry Pi* untuk selanjutnya data tersebut disimpan dan diolah dalam sistem operasi *Raspberry Pi*. Data dari *Arduino UNO* tersebut dapat terkirim ke *Raspberry Pi* dengan menggunakan kabel USB sebagai komunikasi serial atau penghubung komunikasi antara *Raspberry Pi* dan *Arduino*. Data daya yang didapat oleh *Arduino* berupa daya arus listrik dalam satuan *Ampere* yang telah dikalkulasikan menjadi satuan KW(*KiloWatt*) yang hanya dapat dilihat ketika *user* mengakses aplikasi *monitoring*.

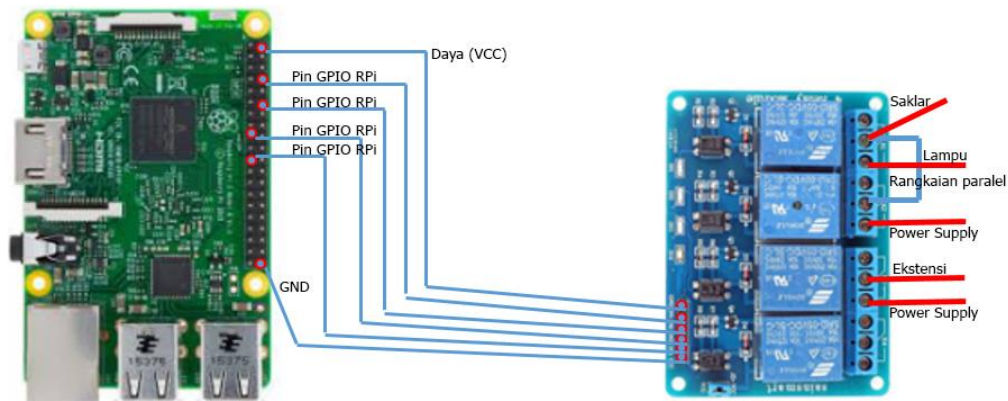
*Raspberry pi* berperan dalam mengolah data yang dikirim oleh Sensor CT untuk selanjutnya ditampilkan kedalam aplikasi monitoring berbasis website. Selanjutnya *Raspberry pi* mengirimkan data ke database untuk ditampilkan pada aplikasi berbasis website. Aplikasi yang dibangun memiliki *one way authentication* sebagai proses untuk menentukan apakah *user* yang ingin masuk ke dalam sistem merupakan *valid user* atau tidak. Aplikasi ini hanya digunakan oleh dua *user* yang bertindak sebagai admin dan client.

Proses autentikasi saat ini umumnya hanya menggunakan *username* dan *password*. Ketika *user* berhasil *login* maka *user* dapat melihat *dashboard* aplikasi dan menggunakan aplikasi tersebut untuk memantau dan mengontrol penggunaan daya listrik dalam rumah. Apabila *user* melakukan *login* kedalam sistem ini maka dengan secara otomatis sistem akan mencatat *log/ history* *user* untuk mengetahui kapan saja aplikasi tersebut digunakan yang dilakukan pada sistem yang digunakan. Aplikasi dibangun dengan tujuan mempermudah *user* dalam mengontrol penggunaan listrik dalam rumah.



### 3.1.1 Perancangan Desain *Raspberry Pi* dan *Relay* untuk mengendalikan perangkat listrik

Berikut ini ditampilkan gambaran desain *Raspberry Pi* dan *Relay* untuk mengukur daya perangkat listrik.

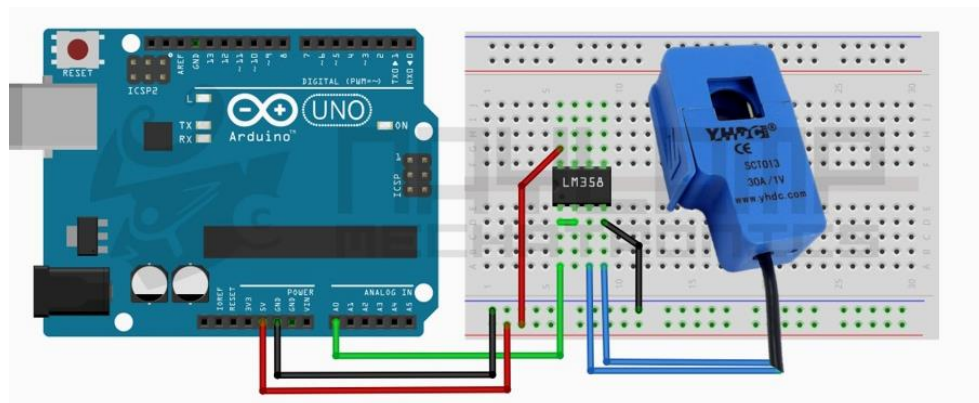


Gambar 5 Desain *Raspberry Pi* dan *Relay*

Pada Gambar 5 ditampilkan desain rangkaian *Raspberry Pi* dan *Relay* dalam mengendalikan perangkat listrik berupa lampu dan ekstensi. Perangkat dirangkai secara seri maupun paralel pada *port module Relay*. Setelah dirangkai pada *module Relay* selanjutnya baik *input*, *ground* dan daya (VCC) pada *Relay* akan dihubungkan menggunakan kabel *jumper* ke pin GPIO yang terdapat pada *Raspberry Pi*. Sehingga ketika *Relay* menerima sinyal dari *Raspberry Pi* maka *Relay* akan memutuskan arus listrik yang mengalir pada tiap perangkat baik secara *id* maupun secara keseluruhan.

### 3.1.2 Perancangan Desain *Arduino* dan *Sensor CT* untuk mengukur arus perangkat listrik

Berikut ini ditampilkan perancangan desain *Arduino* dan *Sensor CT* untuk mengukur daya perangkat listrik.



Gambar 6 Desain *Arduino* dan *Sensor CT*

Gambar 6 merupakan desain rangkaian *Arduino* dan *Sensor CT* untuk mengukur daya pada perangkat listrik yang digunakan. Rangkaian tersebut di rangkai diatas *motherboard* (papan rangkaian). *Sensor CT* dan *Arduino* dihubungkan dengan sebuah *LM358* agar *Sensor CT* dapat mengirimkan daya arus listrik ke *Arduino*.

### 3.1.3 Perancangan Desain Komunikasi Serial antara *Raspberry Pi* dan *Arduino*

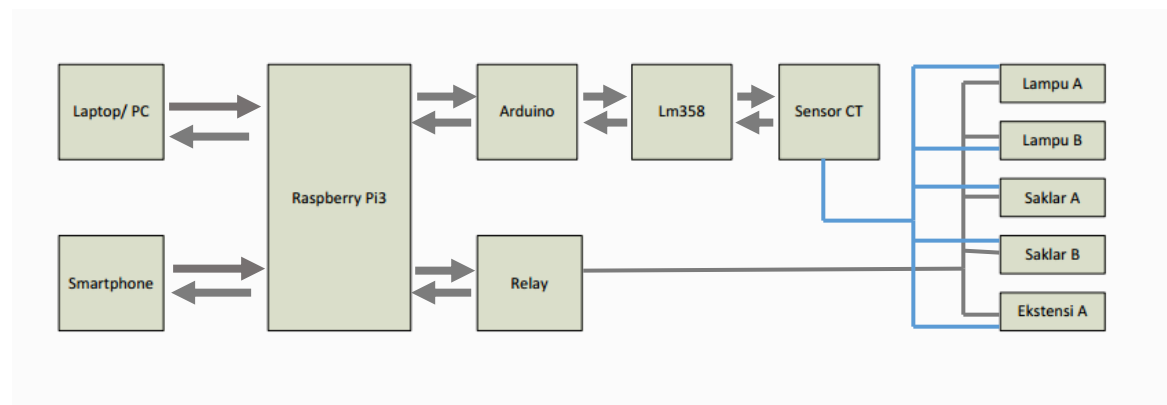
Berikut ini ditampilkan perancangan desain untuk komunikasi serial antara *Raspberry Pi* dan *Arduino*.



Gambar 7 Komunikasi serial antara *Raspberry Pi* dan *Arduino*

Pada Gambar 7 ditampilkan desain komunikasi serial antara *Raspberry Pi* dan *Arduino* dalam proses pengiriman data arus listrik dari *Raspberry Pi* dan *Arduino*. Komunikasi serial tersebut menggunakan sebuah kabel USB yang dihubungkan pada *port Arduino* dan *port USB* pada *Raspberry Pi*.

### 3.1.4 *Block diagram* perancangan perangkat keras

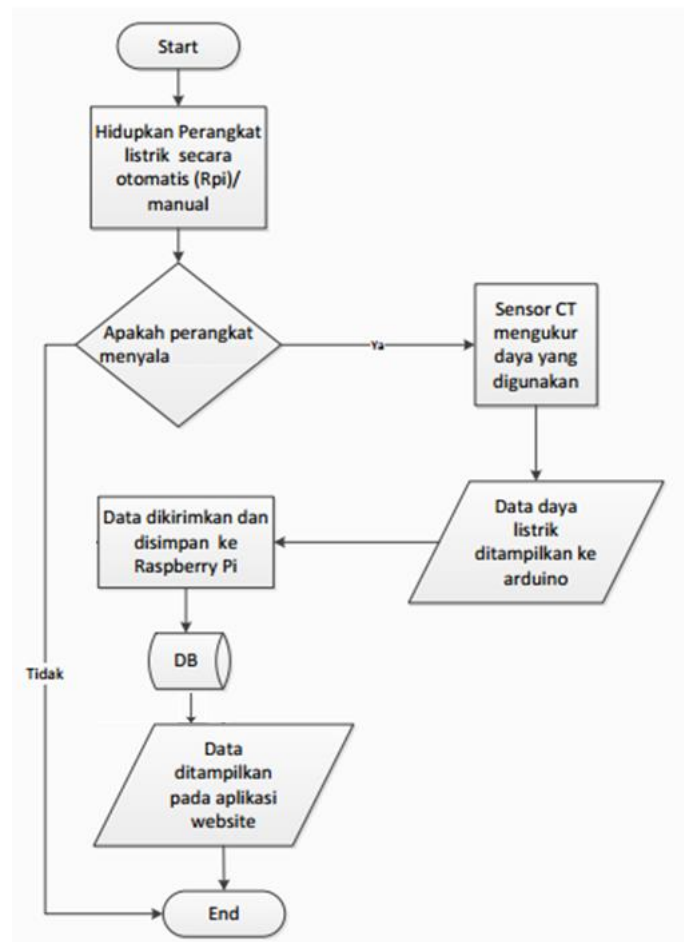


Gambar 8 Perancangan perangkat keras

Gambar 8 merupakan perancangan perangkat keras dalam pembangunan sistem kendali listrik. Laptop/PC mengambil data dari *Raspberry Pi*. *Raspberry Pi* mendapatkan data dari Sensor CT yang dijepitkan pada kabel perangkat listrik melalui *Arduino*. Sensor CT dan *Arduino* terhubung dengan komponen *lm358* yang dimana berfungsi untuk membantu Sensor CT dalam mengirimkan data ke *Raspberry pi*. Sensor CT mengambil data arus listrik yang digunakan pada lampu atau *outlet*. Arus pada lampu, saklar dan ekstensi dapat diputus menggunakan *Relay* yang terhubung dengan *Raspberry Pi*.

### 3.1.5 Flowchart diagram mendapatkan data daya arus listrik pada perangkat listrik dan ditampilkan pada aplikasi website

Berikut ditampilkan *flowchart* cara mendapatkan data daya arus listrik pada perangkat listrik dan ditampilkan pada aplikasi website.

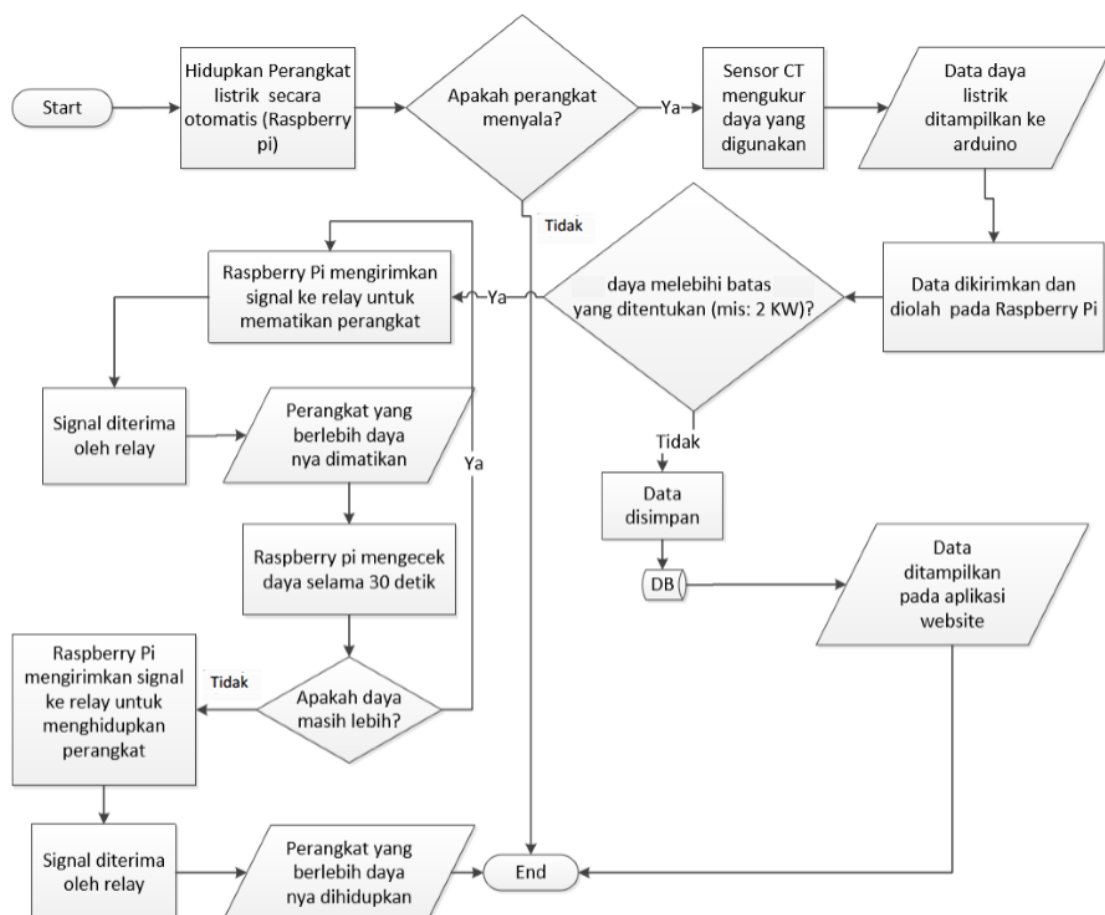


Gambar 9 *Flowchart* mendapatkan daya arus listrik pada perangkat listrik dan ditampilkan pada aplikasi website

Gambar 9 menunjukkan *flowchart* dari sistem yang dibangun. Perangkat dapat dihidupkan secara manual atau otomatis menggunakan *Raspberry Pi*. Ketika saklar dinyalakan lampu akan menyala dengan kondisi *Raspberry Pi* dan *Relay* dalam keadaan selalu menyala. Ketika ekstensi di hubungkan pada *power supply* atau sumber tegangan maka ekstensi dapat menyala. Arus listrik yang mengalir akan dideteksi oleh Sensor CT dan daya arus listrik tersebut akan dikirimkan ke *Raspberry Pi* melalui komunikasi serial antara *Arduino* dan *Raspberry Pi*. Begitu juga dengan perangkat listrik lainnya. Ketika perangkat lainnya dihidupkan secara manual atau secara otomatis maka perangkat dapat menyala. Arus listrik yang mengalir dideteksi oleh Sensor CT dan arus listrik tersebut akan dikirimkan ke *Arduino* dan selanjutnya akan dikirimkan lagi ke *Raspberry Pi*.

### 3.1.6 Flowchart diagram mengendalikan perangkat secara otomatis

Berikut ditampilkan *flowchart* mematikan arus menggunakan *Relay*.



Gambar 10 *Flowchart* mengendalikan alat secara otomatis

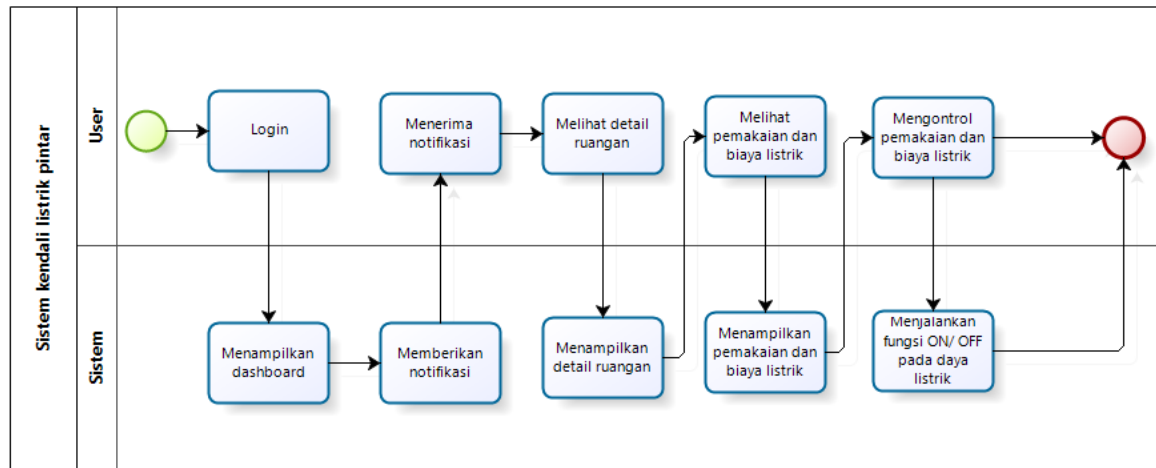
Gambar 10 menunjukkan proses kerja sistem dalam mengendalikan perangkat arus listrik secara otomatis menggunakan *Raspberry Pi* dan *Relay*. *Raspberry Pi* berfungsi untuk menyalakan lampu secara otomatis dan *Relay* dapat memutus arus listrik maupun membuka aliran arus listrik pada setiap perangkat ketika mendapat *signal*/perintah dari *Raspberry Pi*. Ketika perangkat dinyalakan secara otomatis, Sensor CT mengukur daya arus listrik yang digunakan pada tiap perangkat yang menggunakan daya arus listrik. Daya yang didapat Sensor CT akan dikirim ke *Arduino* dan akan dikirimkan lagi ke *Raspberry Pi*. Data tersebut dibandingkan dengan batas daya yang ditentukan oleh *user* pada aplikasi web. Ketika daya telah melebihi batas daya konsumsi yang ditetapkan maka *Raspberry pi* akan mengirimkan *signal* ke *Relay* untuk mematikan perangkat yang penggunaan dayanya berlebih tersebut. Perangkat yang dimatikan tersebut akan hidup kembali setelah 30 detik. *Raspberry Pi* akan mengecek daya perangkat, jika daya tidak lagi melebihi batas maka perangkat akan tetap hidup namun jika daya masih melebihi batas yang ditentukan maka perangkat akan mati kembali. Ketika daya tersebut tidak melebihi batas daya yang ditentukan maka data tersebut akan ditampung dalam sebuah *array* dan akan dimasukkan kedalam database untuk selanjutnya ditampilkan pada aplikasi monitoring berbasis website.

### **3.2 Perancangan aplikasi**

Pada sub bab ini akan dijelaskan perancangan aplikasi yang akan dibangun. Hal yang dibahas pada subbab ini antara lain adalah *business process*, *use case diagram*, *use case scenario* dan *sequence diagram*.

#### **3.2.1 Business Process**

Pada *business process* berikut menampilkan proses sistem secara umum

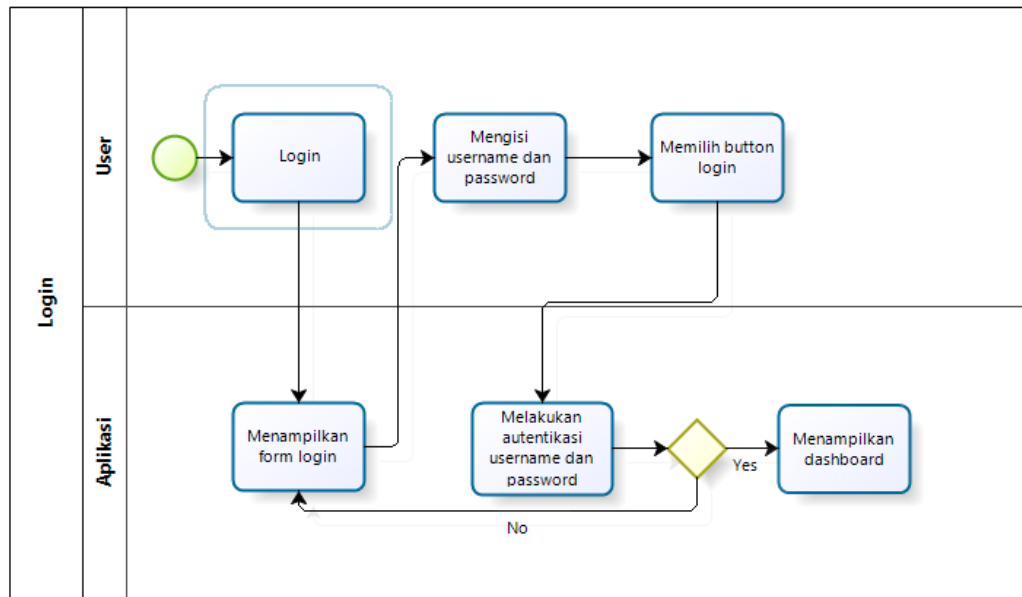


Gambar 11 *Business Process* sistem secara umum

Gambar 11 merupakan *business process* dari sistem *Smart House Electricity* yang dibangun. Untuk dapat menggunakan aplikasi *website* tersebut, *user* harus melakukan login terlebih dahulu. Aplikasi akan memeriksa apakah *user* yang melakukan proses login adalah *user* yang *valid* atau tidak *valid*. Jika *valid* maka *user* dapat masuk kedalam *dashboard* aplikasi. *User* dapat melihat penggunaan daya listrik dalam rumah yang telah habis digunakan dalam satuan KiloWatt. *User* dapat memantau penggunaan listrik dalam rumah dalam jarak yang jauh dari rumah atau keberadaan *user* sedang tidak didalam rumah. Dengan aplikasi *website* ini *user* dapat mematikan atau menghidupkan lampu dan alat-alat listrik lainnya dalam rumah.

### 3.2.1.1 *Business Process login*

Pada *business process* berikut menampilkan proses untuk *login*.

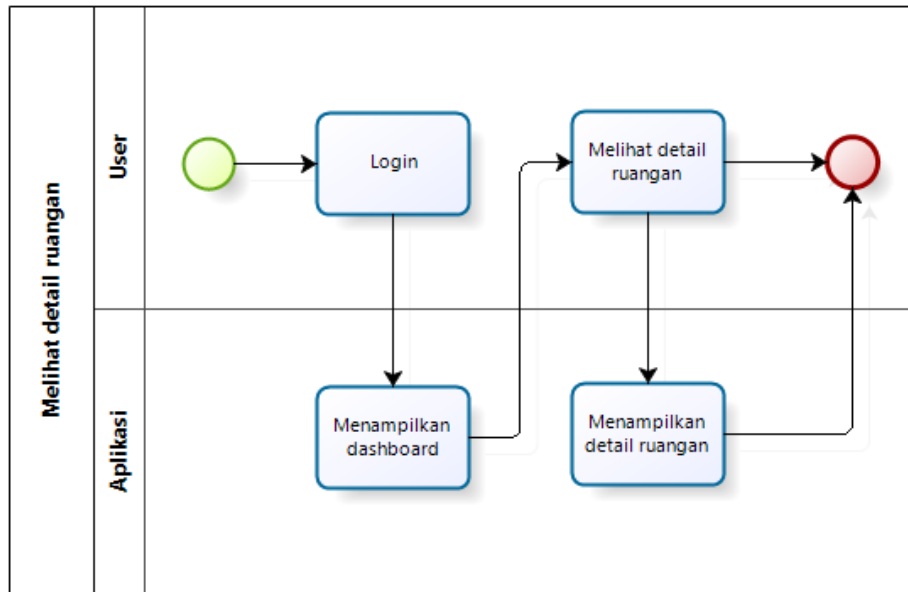


Gambar 12 *Business Process login*

Gambar 12 merupakan *business process login*. Aplikasi menampilkan form login, kemudian *user* mengisi *username* dan *password* yang sesuai. Setelah itu, *user* mengklik tombol login. Aplikasi melakukan autentikasi terhadap *username* dan *password* yang telah dimasukkan. Jika *username* dan *password* yang diberikan sesuai, maka aplikasi akan menampilkan dashboard.

### 3.2.1.2 *Business Process melihat detail ruangan*

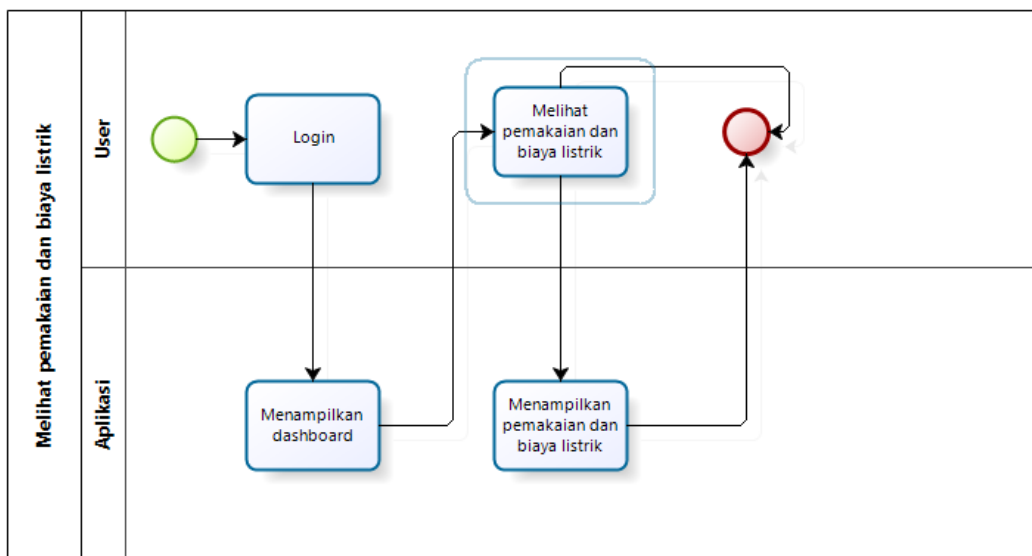
*Business process* melihat *detail* ruangan dapat dilihat pada Gambar 13. *User* harus *login* terlebih dahulu. Kemudian aplikasi menampilkan *dashboard*. Setelah itu *user* dapat melihat detail ruangan. Aplikasi menampilkan detail ruangan.



Gambar 13 *Business process* melihat detail ruangan

### 3.2.1.3 *Business Process* melihat pemakaian daya listrik

Pada *business process* berikut menampilkan proses untuk melihat pemakaian daya listrik.



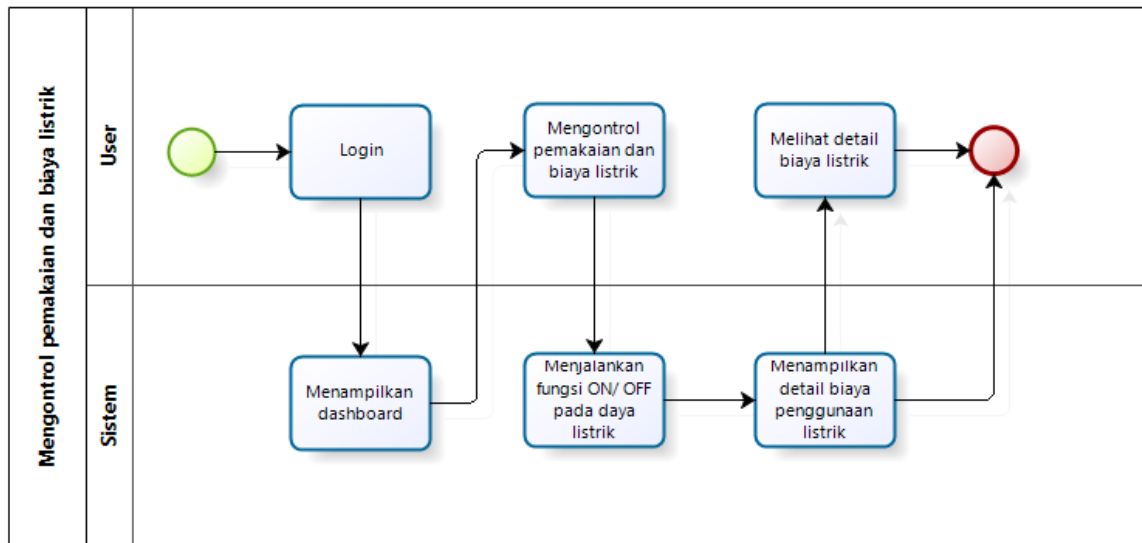
Gambar 14 *Business Process* melihat pemakaian daya listrik

Gambar 14 merupakan *business process* melihat pemakaian daya listrik. *User* harus *login* terlebih dahulu. Aplikasi menampilkan *dashboard*. *User* dapat melihat pemakaian daya listrik. Aplikasi menampilkan rincian pemakaian daya listrik.



### 3.2.1.4 Business Process mengontrol pemakaian daya listrik

Pada *business process* berikut menampilkan proses untuk mengontrol pemakaian daya listrik.

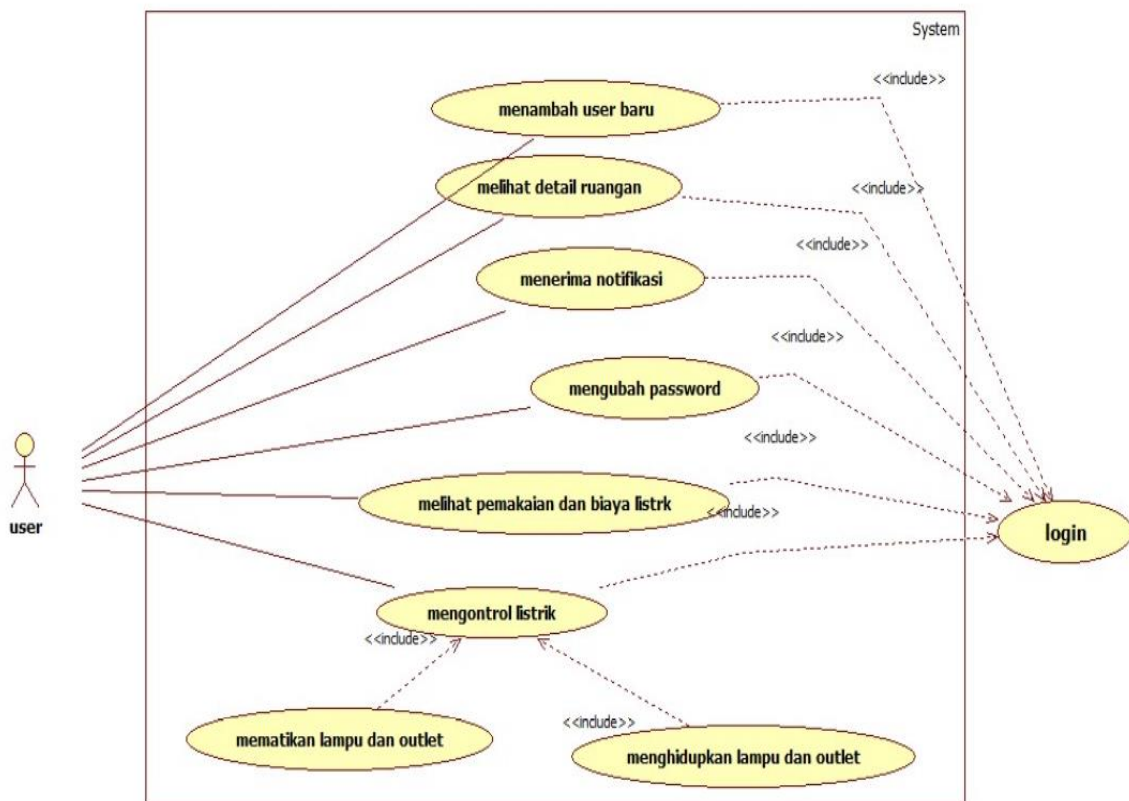


Gambar 15 *Business Process* mengontrol pemakaian daya listrik

Gambar 15 merupakan *business process* mengontrol pemakaian daya listrik. *User* harus *login* terlebih dahulu. Setelah itu, sistem menampilkan *dashboard*. *User* melakukan kontrol terhadap pemakaian daya listrik. Sistem dapat mengontrol pemakaian daya listrik seperti mematikan dan menghidupkan lampu atau ekstensi.

### 3.2.2 Use Case Diagram

Deskripsi fungsional aplikasi digambarkan pada Gambar 16.



Gambar 16 Use case diagram aplikasi

### 3.2.3 Use Case Scenario

Pada subbab ini dijelaskan *scenario* dari setiap *use case* yang meliputi 7 tabel. *Use Case Scenario* yang diberikan dapat dilihat pada tabel berikut.

Tabel 4 Use case scenario login

| Use case ID Number     | UCS_01 Login  |                                  |
|------------------------|---|----------------------------------|
| Use case Name          | Authentikasi Pengguna   |                                  |
| Use case Description   | Use case berikut mendeskripsikan prosedur login   |                                  |
| Actor                  | Admin   |                                  |
| Precondition           | <ul style="list-style-type: none"> <li>- User sudah mengakses/membuka system</li> <li>- User masuk ke dalam system</li> </ul> |                                  |
| Primary Flow of Events | User Action   | System Response                  |
|                        | 1. User mengklik menu login   | 2. Sistem menampilkan form login |
|                        |   |                                  |

| <i>Use case ID Number</i>       | <b>UCS_01 Login</b>   |  |
|---------------------------------|---|--|
|                                 | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i>  |  |
|                                 | 4. Klik <i>button login</i>   | 5. Sistem melakukan validasi terhadap <i>username</i> dan <i>password</i> pada tabel <i>database</i> |
|                                 |   | 6. Sistem menampilkan halaman dashboard aplikasi   |
| <i>Alternate Flow of Events</i> | <i>User Action</i>  | <i>System Response</i>   |
|                                 | 5a. <i>Username</i> dan <i>password</i> yang dimasukkan tidak sesuai (salah)  | 1. Sistem menampilkan notifikasi ketidaksesuaian <i>username</i> dan <i>password</i>                 |
|                                 | 5b. <i>Username</i> yang dimasukkan kosong  | 2. Sistem menampilkan notifikasi   |
|                                 |   | 3. Sistem kembali ke <i>form login</i>   |
| <i>Error Flow of Events</i>     | -   | -  |
| <i>Post Condition</i>           | <i>User</i> berhasil <i>login</i> ke sistem dan halaman utama ditampilkan dan dapat membuka halaman dari menu-menu yang disediakan dan menggunakan aplikasi |  |

**Tabel 5 Use case scenario ganti password**

| <i>Use case ID Number</i>     | <b>UCS_02 Ganti Password</b>   |   |
|-------------------------------|--|---|
| <i>Use case Name</i>          | Ganti password   |   |
| <i>Use case Description</i>   | <i>Use case</i> berikut mendeskripsikan bagaimana user dapat mengganti password yang lama ke password yang baru                                    |   |
| <i>Actor</i>                  | Admin, user  |   |
| <i>Precondition</i>           | <ul style="list-style-type: none"> <li>- <i>User</i> sudah mengakses/membuka <i>sistem</i></li> <li>- <i>User</i> masuk ke dalam system</li> </ul> |   |
| <i>Primary Flow of Events</i> | <i>User Action</i>   | <i>System Response</i>                  |
|                               | 1. <i>User</i> mengklik menu <i>login</i>  | 2. Sistem menampilkan <i>form login</i> |

| <i>Use case ID Number</i>   | <b>UCS_02 Ganti Password</b>   |  |
|-----------------------------|--|--|
|                             | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i> |  |
|                             | 4. Memilih menu change password  | 5. Akan muncul form pengisian form password yang lama dan baru |
|                             |  | 6. System akan mengkonfirmasi apakah account benar valid       |
|                             | 7. <i>User</i> sudah memiliki password baru  |  |
| <i>Error Flow of Events</i> | -  | 8. system akan kembali ke menu login untuk memastikan kembali  |
| <i>Post Condition</i>       | <i>User</i> dapat mengganti password yang lama ke password yang baru                 |  |

**Tabel 6 Menambah *user* baru**

| <i>Use case ID Number</i>     | <b>UCS-03 Membuat akun baru</b>   |  |
|-------------------------------|---|--|
| <i>Use case Name</i>          | Membuat akun baru   |  |
| <i>Use case Description</i>   | <i>Use case</i> berikut mendeskripsikan bagaimana <i>user</i> dapat membuat akun baru.  |  |
| <i>Actor</i>                  | Admin, <i>user</i>  |  |
| <i>Precondition</i>           | <ul style="list-style-type: none"> <li>- <i>User</i> sudah mengakses/membuka <i>sistem</i></li> <li>- <i>User</i> masuk ke dalam <i>system</i></li> </ul> |  |
| <i>Primary Flow of Events</i> | <i>User Action</i>  | <i>Primary Flow of Events</i>                          |
|                               | 1. <i>User</i> mengklik menu <i>login</i>   | 2.Sistem menampilkan <i>form login</i>                 |
|                               | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i>  |  |
|                               | 4.Memilih menu <i>create account</i>  | 5.Akan muncul form pengisian form penambahan akun baru |
|                               |   | 6. System akan mengkonfirmasi apakah account benar     |
|                               |   |  |

|                             |  |                             |
|-----------------------------|--|-----------------------------|
| <i>Use case ID Number</i>   | <b>UCS-03 Membuat akun baru</b>            |                             |
|                             | 7. <i>User</i> sudah memiliki account baru |                             |
| <i>Error Flow of Events</i> | -  | <i>Error Flow of Events</i> |
| <i>Post Condition</i>       | <i>User</i> dapat membuat akun tambahan    |                             |

**Tabel 7 Use Case Scenario Melihat Detail Ruangan**

|                               |  |  |
|-------------------------------|--|--|
| <i>Use case ID Number</i>     | <b>UCS_05 Melihat Detail Ruangan</b>   |  |
| <i>Use case Name</i>          | Melihat Detail Ruangan   |  |
| <i>Use case Description</i>   | <i>Use case</i> berikut mendeskripsikan bagaimana user dapat melihat detail ruangan  |  |
| <i>Actor</i>                  | Admin  |  |
| <i>Precondition</i>           | <ul style="list-style-type: none"> <li>- <i>User</i> sudah mengakses/membuka <i>sistem</i></li> <li>- <i>User</i> login ke dalam system</li> </ul> |  |
| <i>Primary Flow of Events</i> | <i>User Action</i>   | <i>System Response</i>   |
|                               | 1. <i>User</i> mengklik menu <i>login</i>  |  |
|                               |  | 2. Sistem menampilkan <i>form login</i>  |
|                               | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i>   |  |
|                               | 4. Klik <i>button login</i>  | 5. Sistem melakukan validasi terhadap <i>username</i> dan <i>password</i> pada tabel <i>database</i> |
|                               |  | 6. Sistem menampilkan halaman dashboard aplikasi   |
|                               |  | 7. Aplikasi menampilkan detail ruangan   |
|                               | 8. <i>User</i> dapat melihat detail ruangan  |  |
| <i>Error Flow of Events</i>   | -  | -  |
| <i>Post Condition</i>         | <i>User</i> dapat melihat detail ruangan   |  |

**Tabel 8 Use Case Scenario Melihat Pemakaian daya Listrik**

|                               |  |  |
|-------------------------------|--|--|
| <i>Use case ID Number</i>     | <b>UCS_06 Melihat Pemakaian daya Listrik</b>   |  |
| <i>Use case Name</i>          | Melihat Pemakaian Daya Listrik   |  |
| <i>Use case Description</i>   | <i>Use case</i> berikut mendeskripsikan bagaimana user dapat melihat pemakaian daya listrik  |  |
| <i>Actor</i>                  | Admin  |  |
| <i>Precondition</i>           | <ul style="list-style-type: none"> <li>- <i>User</i> sudah mengakses/membuka <i>sistem</i></li> <li>- <i>User</i> login ke dalam system</li> </ul> |  |
| <i>Primary Flow of Events</i> | <i>User Action</i>   | <i>System Response</i>   |
|                               | 1. <i>User</i> mengklik menu <i>login</i>  |  |
|                               |  | 2. Sistem menampilkan <i>form login</i>  |
|                               | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i>   |  |
|                               | 4. Klik <i>button login</i>  | 5. Sistem melakukan validasi terhadap <i>username</i> dan <i>password</i> pada tabel <i>database</i> |
|                               |  | 6. Sistem menampilkan halaman dashboard aplikasi   |
|                               |  | 7. Aplikasi menampilkan rincian pemakaian daya listrik   |
|                               | 14. <i>User</i> dapat melihat rincian pemakaian daya listrik   |  |
| <i>Error Flow of Events</i>   | -  | -  |
| <i>Post Condition</i>         | <i>User</i> dapat melihat rincian pemakaian daya listrik   |  |

**Tabel 9 Use Case Scenario Mengontrol Listrik**

|                               |  |  |
|-------------------------------|--|--|
| <i>Use case ID Number</i>     | <b>UCS_07 Mengontrol Listrik</b>   |  |
| <i>Use case Name</i>          | Mengontrol Listrik   |  |
| <i>Use case Description</i>   | <i>Use case</i> berikut mendeskripsikan bagaimana user dapat mengontrol listrik  |  |
| <i>Actor</i>                  | Admin  |  |
| <i>Precondition</i>           | <ul style="list-style-type: none"> <li>- <i>User</i> sudah mengakses/membuka <i>sistem</i></li> <li>- <i>User</i> login ke dalam system</li> </ul> |  |
| <i>Primary Flow of Events</i> | <i>User Action</i>   | <i>System Response</i>   |
|                               | 1. <i>User</i> mengklik menu <i>login</i>  |  |
|                               |  | 2. Sistem menampilkan <i>form login</i>  |
|                               | 3. <i>User</i> memasukkan <i>username</i> dan <i>password</i> pada <i>form login</i>   |  |
|                               | 4. Klik <i>button login</i>  | 5. Sistem melakukan validasi terhadap <i>username</i> dan <i>password</i> pada tabel <i>database</i> |
|                               |  | 6. Sistem menampilkan halaman dashboard aplikasi   |
|                               | 7. <i>User</i> dapat melakukan control terhadap listrik seperti mematikan / menghidupkan lampu dan outlet  |  |
| <i>Error Flow of Events</i>   | -  | -  |
| <i>Post Condition</i>         | <i>User</i> dapat mengontrol listrik   |  |

### 3.3 Perangkat yang digunakan

Pada subbab ini, dijelaskan perangkat yang digunakan selama proses pengerjaan Tugas Akhir ini. Adapun perangkat tersebut mencakup perangkat lunak (*software*) dan perangkat keras (*hardware*).

### 3.3.1 Perangkat Keras (*Hardware*) yang Digunakan

Perangkat keras yang digunakan selama pengerjaan Tugas Akhir ini dijelaskan lebih rinci sebagai berikut:

#### 1. Laptop

Laptop merupakan sebuah komputer dengan ukuran yang relatif lebih kecil. Laptop merupakan gabungan dari semua perangkat yang dibutuhkan pada komputer secara umum seperti *monitor*, *keyboard*, *monitor*, *mouse* yang pada laptop menggunakan *touchpad*. Pada Tugas Akhir ini laptop digunakan sebagai media penyimpanan, *interface*, dan media yang digunakan untuk menjalankan aplikasi yang telah dibangun. Adapun spesifikasi laptop yang digunakan adalah laptop dengan *processor* Intel(R) Core(TM) i5-4200M CPU @ 2.50Ghz 2.50 Ghz, dengan kapasitas RAM 4,00 GB dan *hard driver* 500 GB.

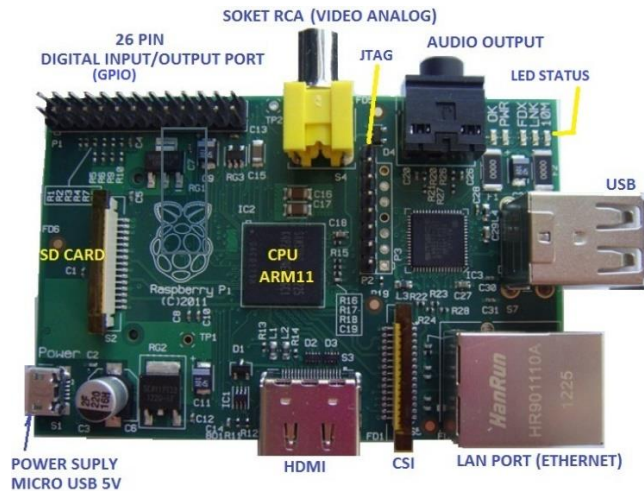
#### 2. Monitor

Monitor komputer adalah salah satu jenis *soft-copy device*, karena keluarannya adalah berupa sinyal elektronik, dalam hal ini berupa gambar yang tampil di layar monitor. Gambar yang tampil adalah hasil pemrosesan data ataupun informasi masukan. Pada Tugas Akhir ini, monitor yang digunakan adalah LCD Monitor 19 Inch Wide Screen.

#### 3. *Raspberry Pi*

Pada Tugas Akhir ini, *Raspberry Pi* yang digunakan adalah *Raspberry Pi 3* Tipe B yang dilengkapi dengan Wi-Fi 802.11n dan Bluetooth versi 4.1. Sehingga pengguna yang ingin *Raspberry Pi* mereka terhubung dengan jaringan Internet tidak perlu lagi menambahkan modul atau menggunakan slot RJ45. Dilihat dari segi prosesor, *Raspberry Pi 3* diklaim lebih cepat. *Raspberry Pi 3* menggunakan prosesor ARM Cortex-A53 dari *Broadcom*, dengan spesifikasi 64-bit *Quad-Core* dan berkecepatan 1,2 Ghz. Kapasitas RAM menggunakan RAM 1 GB. *Board Raspberry Pi* dapat dilihat pada Gambar 17 [15] .





Gambar 17 Raspberry Pi

#### 4. Relay

Pada tugas akhir ini, *Relay* yang digunakan adalah *DI-Relay 8*. *DI-Relay 8* adalah modul *Relay SPDT (Single Pole Double Throw)* yang memiliki ketahanan yang baik terhadap arus dan tegangan yang besar. *DI-Relay 8* merupakan modul 8 buah *Relay* dalam 1 board, sehingga pengguna lebih hemat dari sisi biaya maupun dari sisi dimensi. Spesifikasi *DI-Relay 8*:

1. Menggunakan 8 buah *Relay* HKE HRS4H-S-DC5V
2. Menggunakan tegangan rendah, 5V, sehingga dapat langsung dihubungkan pada sistem *mikrokontroler*.
3. Tipe *Relay* adalah SPDT (*Single Pole Double Throw*): 1 COMMON, 1 NC (*Normally Close*), dan 1 NO (*Normally Open*).
4. Memiliki daya tahan sampai dengan 10A.
5. Dapat langsung dihubungkan pada *DI-Smart AVR System* atau *DI-Smart 51 System*.
6. Dilengkapi rangkaian penggerak (*driver*) *Relay* dengan level tegangan TTL sehingga dapat langsung dikendalikan oleh *mikrokontroler*.
7. *Driver* bertipe *active high* atau kumparan *Relay* akan aktif saat pin pengendali diberi logika 1.
8. *Driver* dilengkapi rangkaian peredam GGL induksi sehingga tidak akan membuat *reset* sistem *mikrokontroler*.
9. Dilengkapi rangkaian sumber tegangan yang dapat digunakan sebagai tegangan sumber *Relay*, sehingga tidak mengganggu sumber tegangan dari

rangkaian penggeraknya (*mikrokontroler*). *DI-Relay* 8 dapat dilihat pada Gambar 18 [16] .



Gambar 18 *DI-Relay* 8

#### 5. Sensor Current Transformer (CT)

Pada tugas akhir ini, sensor CT yang digunakan adalah CT 100A SCT-013 dengan spesifikasi sebagai berikut:

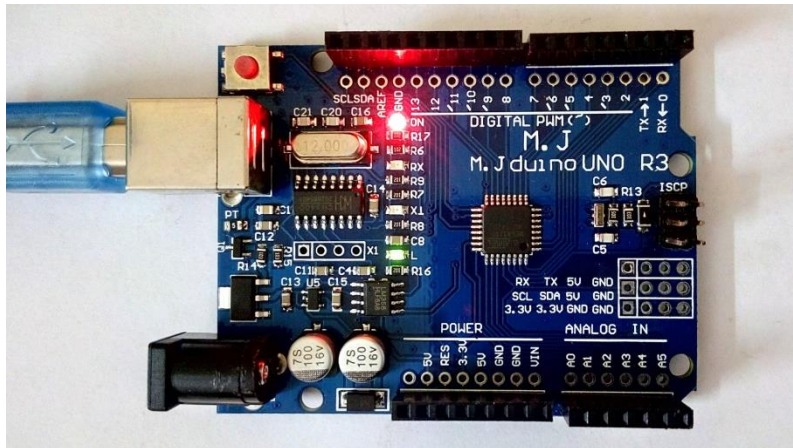
1. Input Current: 0~100A AC
2. Output Mode: 0~50mA
3. Turn Ratio: 100A:0.05A
4. Work Temperature: -25C ~ 70C



Gambar 19 Current Transformer Sensor [17]

#### 7. *Arduino UNO*

Pada tugas akhir ini, *Arduino* berinteraksi dengan Sensor CT untuk menerima data berupa daya listrik yang digunakan oleh perangkat listrik. Data yang telah diterima akan dikirimkan *Raspberry Pi* untuk selanjutnya diolah dan ditampilkan pada aplikasi berbasis *website*.



Gambar 20 Arduino UNO [18]

## 8. LM358

Pada tugas akhir ini LM358 digunakan sebagai ground pada Sensor CT karena Sensor CT hanya memiliki kabel untuk tegangan (*volt*) dan GPIO. Spesifikasi LM358 yang digunakan:

- *Using LM358 chips*
- *100 times gain circuit design*
- *10K ohm adjustable resistance to amplify*
- *LED power indicator*
- *Working voltage: 5 - 12 volt DC*
- *Board size: 32.7mm x 13.3mm*



Gambar 21 LM358 [19]

### 3.3.2 Perangkat Lunak (*Software*) yang Digunakan

Perangkat lunak yang digunakan selama pengerjaan Tugas Akhir ini dapat dilihat pada **Tabel 10**.

**Tabel 10 Perangkat lunak yang digunakan**

| <i>Software</i>             |                       | <b>Keterangan</b>   |
|-----------------------------|-----------------------|---|
| <i>Text Editor</i>          | <b>Sublime Text 3</b> | Merupakan text editor yang digunakan untuk membangun aplikasi.                              |
|                             | <b>Bizagi</b>         | Digunakan untuk mendesain <i>flowchart</i> .  |
| <i>Database</i>             | <b>MySql</b>          | Digunakan untuk mendesain <i>database</i> .   |
| <i>Paket Office</i>         | <b>Ms Word</b>        | Digunakan untuk membuat dokumentasi dari proses pembangunan aplikasi.                       |
| <i>Operating system</i>     | <b>Windows 10</b>     | Sistem operasi yang digunakan selama proses pembangunan aplikasi.                           |
|                             | <b>Raspbian</b>       | Sistem operasi pada <i>Raspberry Pi 3</i> yang digunakan selama proses pembangunan aplikasi |
| <i>Programming Language</i> | <b>PHP</b>            | Bahasa pemrograman yang digunakan untuk membangun aplikasi.                                 |
| <i>Web server</i>           | <b>XAMPP</b>          | Web server yang digunakan untuk menjalankan aplikasi web based                              |

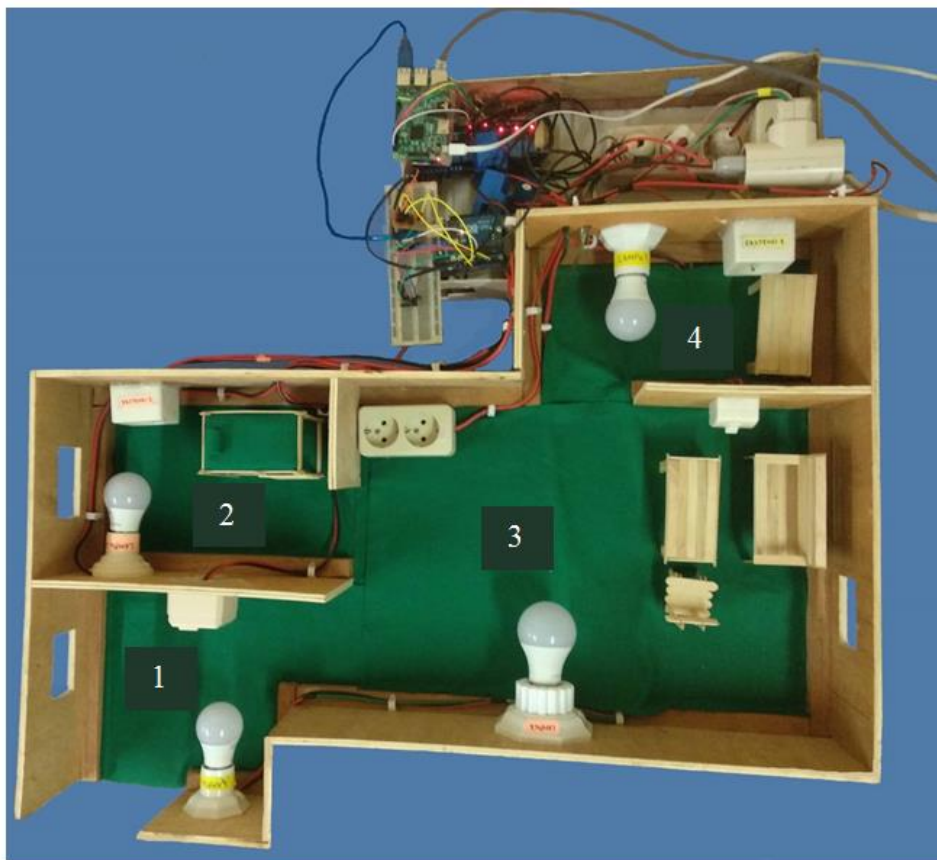
## Bab IV

### Implementasi

Pada bab ini dijelaskan implementasi yang dilakukan saat proses pengerjaan Tugas Akhir. Proses implementasi yang dilakukan berupa pembuatan *prototype* rumah, perancangan alat, implementasi Sensor *Current Transformer (CT)* dan pembuatan aplikasi.

#### 4.1. Implementasi *Prototype* Sistem

Pada subbab ini diuraikan mengenai implementasi yang dilakukan dalam pembuatan *prototype*. Implementasi *prototype* dapat dilihat pada Gambar 22.

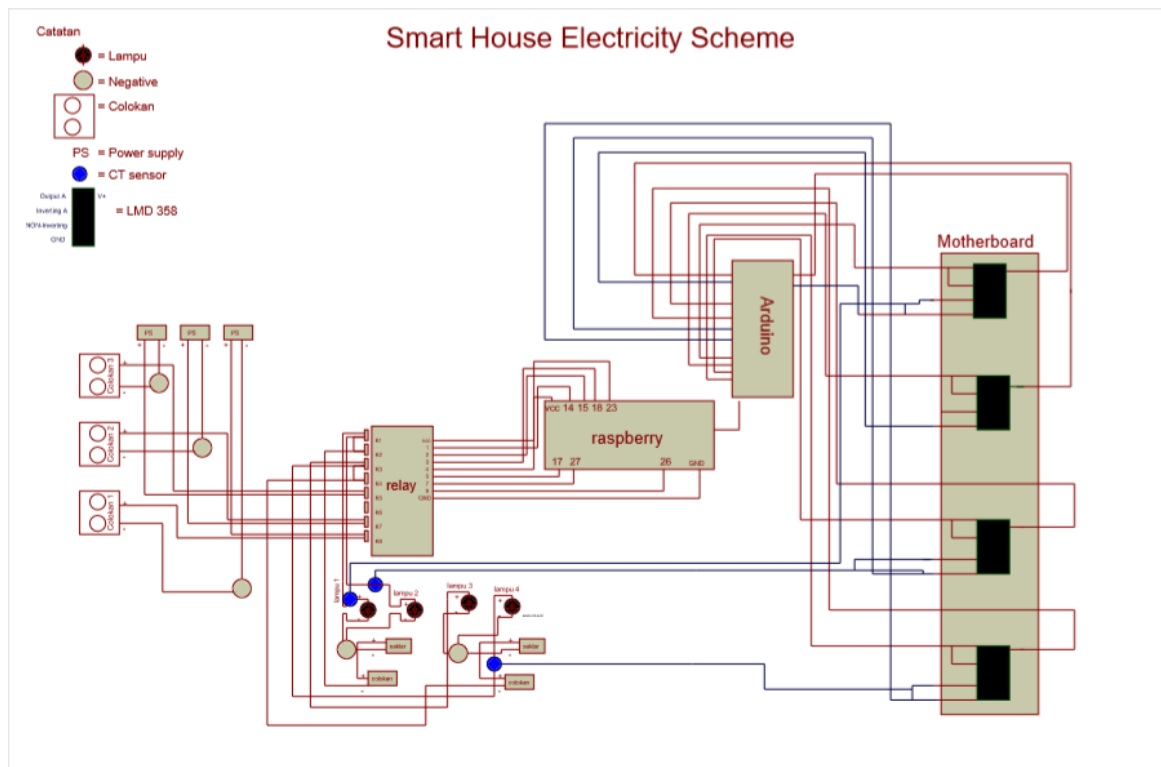


Gambar 22 Implementasi *Prototype* Rumah

Keterangan:

- 1: Ruang Tamu
- 2: Kamar
- 3: Ruang Tengah
- 4: Dapur

Pada ruang tamu dipasang satu lampu dengan daya sebesar 3Watt dan satu saklar. Pada kamar dan dapur dipasang satu lampu dengan daya sebesar 3Watt dan satu ekstensi. Sedangkan, pada ruang tengah dipasang satu lampu dengan daya sebesar 5 Watt, satu saklar dan dua ekstensi. Lampu pada ruang tengah memiliki besar daya yang berbeda dari ruangan lainnya dikarenakan ruangan ini memiliki luas yang cukup besar juga untuk mendapatkan data yang berbeda antara satu lampu dengan lampu lainnya.



Gambar 23 Gambaran skematik secara umum

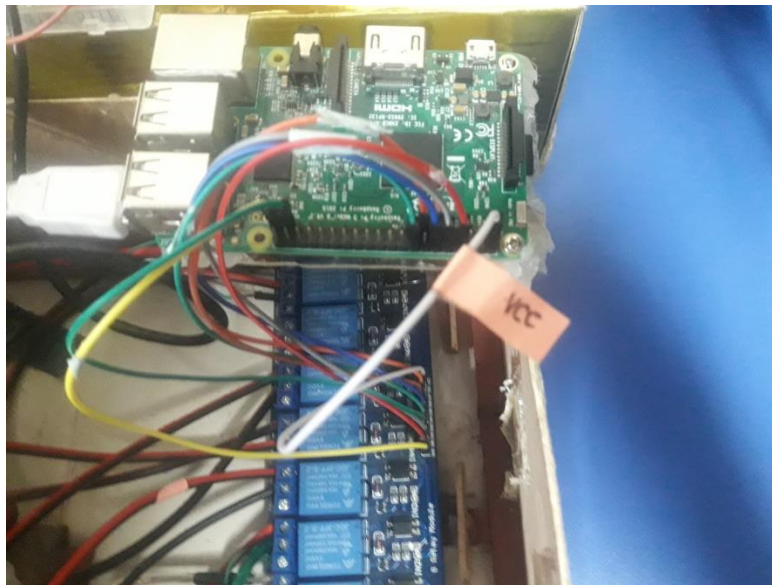
Pada Gambar 23 ditampilkan rancangan skematik secara umum dan mencakup semua aspek komponen dalam sistem yang sedang di bangun. Komponen yang digunakan adalah sebuah *Relay*, *Raspberry Pi*, *Arduino*, enam *Sensor Current Transformer (CT)* dan empat *LM358*. Sensor CT dikaitkan ke tiap kabel positif (+) perangkat. Sensor CT tidak dapat di hubungkan ke *Arduino* secara langsung, karena Sensor CT hanya menyediakan 2 *output* yaitu data dan *output vacancy*. Sensor CT membutuhkan tegangan listrik namun tidak memiliki kabel inputan untuk daya masuk. Oleh karena itu, pada tugas akhir ini digunakan alternatif untuk dapat menjalankan Sensor CT yaitu dengan cara menambahkan sebuah *LM358* sebagai penampung daya (*VCC*).



Sensor CT akan mendeteksi data berupa daya listrik dari setiap perangkat listrik yang digunakan, data tersebut akan di simpan dan di tampilkan pada GUI dari sistem *Arduino*. Setelah data berhasil ditampung oleh *Arduino*, *Arduino* langsung mengirimkan data tersebut ke *Raspberry Pi* yang akan mengolah dan menyimpan data yang telah di peroleh tersebut kemudian akan ditampilkan pada aplikasi *monitoring* berbasis *website*.

#### 4.1.1. Implementasi *Raspberry Pi* dan *Relay* untuk mengendalikan perangkat listrik

Relay bertindak sebagai pemutus daya dari *Power Supply* atau bertindak sebagai pengganti fungsi saklar. Pada *Relay*, modul yang diperlukan untuk merangkai lampu dan saklar secara *parallel* dibutuhkan sebanyak dua. Kutub positif pada satu modul *Relay* dipasang secara *parallel* ke kutub positif modul *Relay* lainnya menggunakan *Jumper*. Kutub positif lampu 1 dan 2 di hubungkan ke tiap-tiap kutub positif yang terdapat pada 2 modul *Relay* yang digunakan. Kutub negatif antara lampu 1, 2 dan saklar dihubungkan secara *parallel* atau disambungkan langsung. *Power* dari saklar harus selalu menyala agar *Relay* dapat berfungsi sebagai pemutus daya.

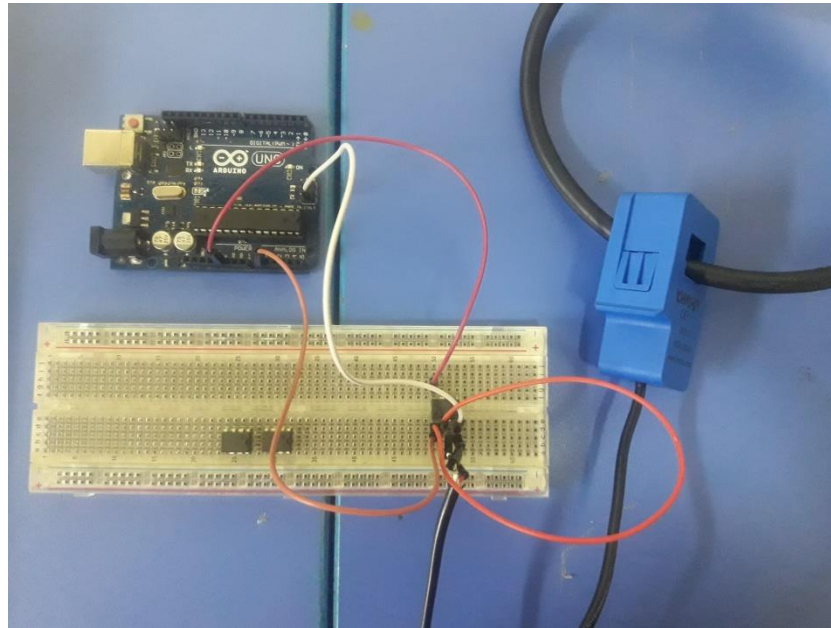


Gambar 24 Implementasi *Raspberry Pi* dan *Relay*

#### 4.1.2. Implementasi *Arduino* dan Sensor CT untuk mengukur perangkat listrik

Pada tugas akhir ini, Sensor *Current Transformer* (CT) yang digunakan pada *prototype* berjumlah 6 yang dijepitkan pada setiap kabel perangkat yang ada di semua ruangan. Sensor CT berfungsi untuk mengetahui daya listrik yang dikonsumsi oleh setiap perangkat. Sensor CT terhubung dengan *Arduino* agar dapat menghasilkan data berupa daya

yang dikonsumsi oleh setiap perangkat. Setelah data didapatkan, *Arduino* mengirimkan data tersebut ke *Raspberry Pi*. *Raspberry Pi* akan mengolah data tersebut dan mengirimnya ke database yang kemudian akan ditampilkan pada aplikasi.



Gambar 25 Implementasi *Arduino* dan Sensor CT

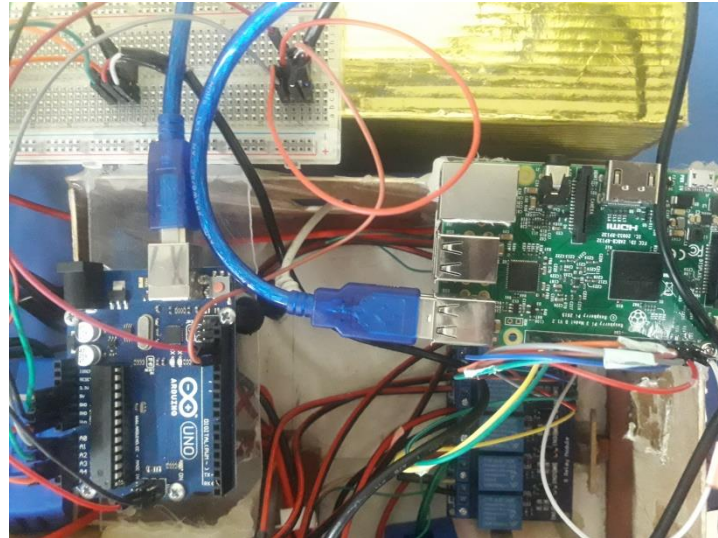
Sensor CT tidak dapat mengirim data pengukuran arus listrik ke *Arduino* secara langsung, karena Sensor CT tidak memiliki *output* untuk daya (VCC) dan *ground*. *Output* yang terdapat pada Sensor CT hanya untuk input data dan *vacancy* (*output* kosong). Sehingga untuk dapat mengirimkan data ke *Arduino* diperlukan perangkat tambahan untuk menyediakan daya (VCC) dan *ground* bagi Sensor CT. Perangkat tambahan tersebut adalah LM358 yang memiliki kaki 6 yang berfungsi untuk menyediakan daya (VCC) dan *ground* bagi Sensor CT. *Arduino* akan terhubung ke LM358 dan Sensor CT pada rangkaian di *motherboard*. Setelah rangkaian dibentuk, disusun ataupun dirangkai dengan benar, Sensor CT dijepitkan pada kabel positif perangkat listrik dan data arus listrik dapat dikirimkan Sensor CT ke *Arduino*.

#### 4.1.3. Komunikasi serial antara *Raspberry Pi* dan *Arduino*

Komunikasi serial pada *Arduino* dan *Raspberry Pi* bertujuan untuk proses pengiriman data daya arus listrik dari *Arduino* ke *Raspberry Pi*. Komunikasi serial tersebut menggunakan sebuah kabel USB yang dihungkan antara *port* USB *Arduino* dan *port* USB pada *Raspberry Pi*. Komunikasi tersebut memungkinkan *Arduino* untuk mengirimkan data ke *Raspberry Pi* secara *real time*. Cara menjalankan komunikasi tersebut adalah ketika



program *Arduino* berjalan maka *Raspberry Pi* tidak boleh mengeksekusi program pada *Arduino*. Begitu juga sebaliknya ketika program pada *Raspberry Pi* dieksekusi maka program *Arduino* harus diakhiri atau bisa dikatakan bahwa keduanya tidak boleh berjalan dalam waktu yang bersamaan.



Gambar 26 Implementasi Komunikasi serial antara *Raspberry Pi* dan *Arduino*

Penggunaan Pin pada *Arduino Uno* dan *Raspberry Pi* dapat dilihat pada **Tabel 11** dan **Tabel 12**.

**Tabel 11 Penggunaan Pin *Arduino Uno***

| Pin        | Fungsi  |
|------------|---|
| 5V         | Berfungsi sebagai power supply  |
| AREF       | Berfungsi sebagai alternative sumber daya pada <i>Arduino</i>                                   |
| VCC (ICSP) | Sebagai alternative daya eksternal yang berada pada <i>Arduino</i>                              |
| 3,3 V      | Berfungsi sebagai power supply  |
| GND        | Berfungsi sebagai pencegahan terjadinya kontak antara pengguna dengan tegangan listrik langsung |
| GND (ICSP) | Berfungsi sebagai pencegahan terjadinya kontak antara pengguna dengan tegangan listrik langsung |
| A0         | Membaca nilai analog pada Sensor CT 1   |
| A1         | Membaca nilai analog pada Sensor CT 2   |
| A2         | Membaca nilai analog pada Sensor CT 3   |
| A3         | Membaca nilai analog pada Sensor CT 4   |

**Tabel 12 Penggunaan Pin Raspberry Pi**

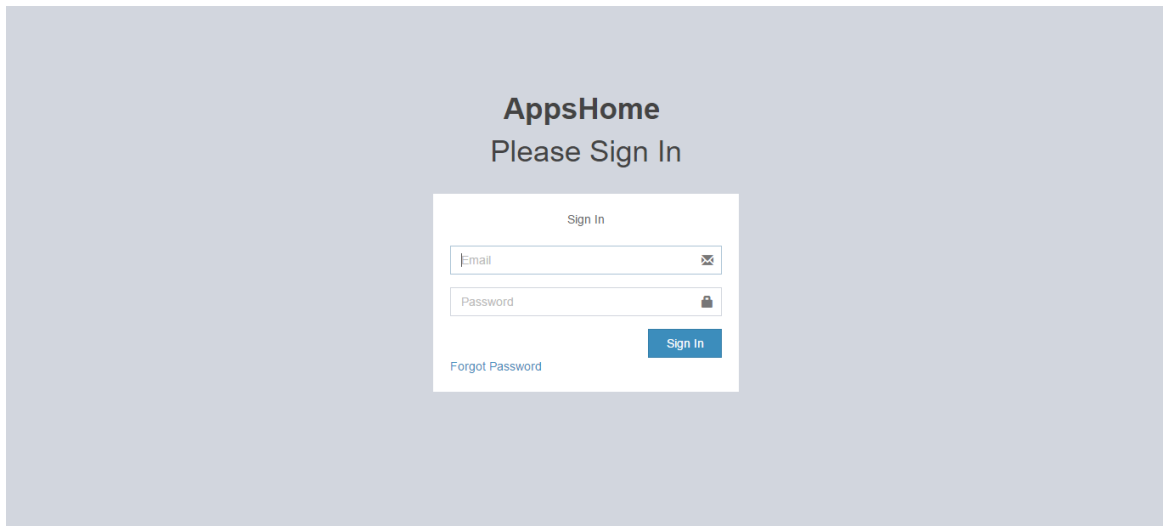
| Pin      | Fungsi  |
|----------|---|
| 2 (5V)   | Berfungsi sebagai <i>power supply</i>   |
| 39 (GND) | Berfungsi sebagai pencegahan terjadinya kontak antara pengguna dengan tegangan listrik langsung |
| GPIO 15  | Sebagai input masuk untuk Lampu 1   |
| GPIO 17  | Sebagai input masuk untuk lampu 2   |
| GPIO 27  | Sebagai input masuk untuk lampu 3   |
| GPIO 14  | Sebagai input masuk untuk lampu 4   |
| GPIO 26  | Sebagai input masuk untuk ekstensi 1  |
| GPIO 23  | Sebagai input masuk untuk ekstensi 2  |
| GPIO 18  | Sebagai input masuk untuk ekstensi 3  |

## **4.2. Implementasi Aplikasi**

Pada subbab ini diuraikan mengenai implementasi yang dilakukan dalam pembangunan Aplikasi. Aplikasi dibangun dengan *framework CodeIgniter*. Aplikasi terdiri dari 7 layer yakni *login*, *dashboard*, *log users*, *details room*, *users*, *record* dan *report*. *Layer-layer* tersebut dapat dilihat pada subbab berikut.

### **4.2.1. Layer Login**

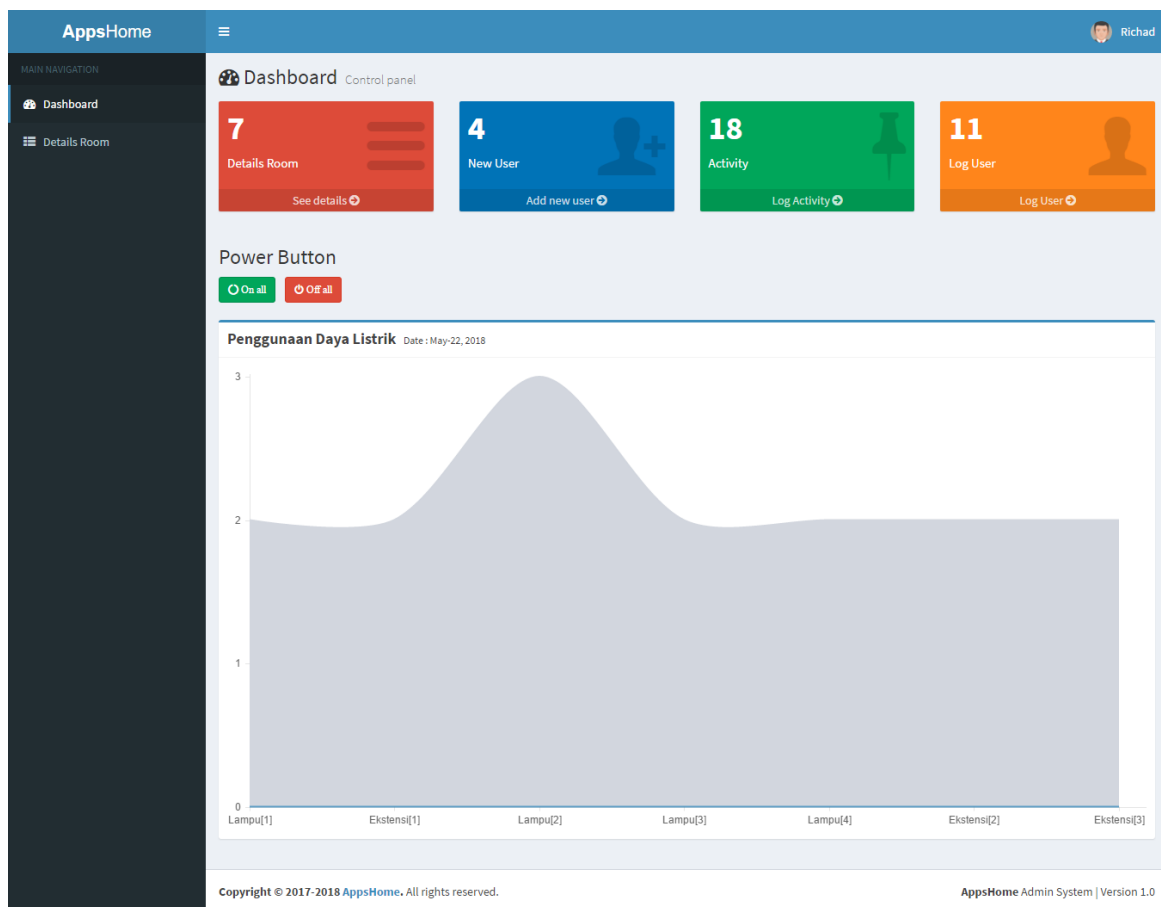
*Layer login* merupakan *layer* yang pertama kali muncul saat aplikasi diakses. Pada *layer login*, pengguna harus memasukkan *username* dan *password* yang valid yang telah didaftarkan terlebih dahulu pada admin. Kemudian pilih tombol *sign in* untuk masuk ke aplikasi. Jika *username* dan *password* yang dimasukkan tidak sesuai maka *user* tidak dapat login untuk mengakses *layer* selanjutnya. Aplikasi menampilkan *layer login* kembali jika *user* memasukkan *username* dan *password* yang salah. *Layer login* dapat dilihat pada Gambar 27.



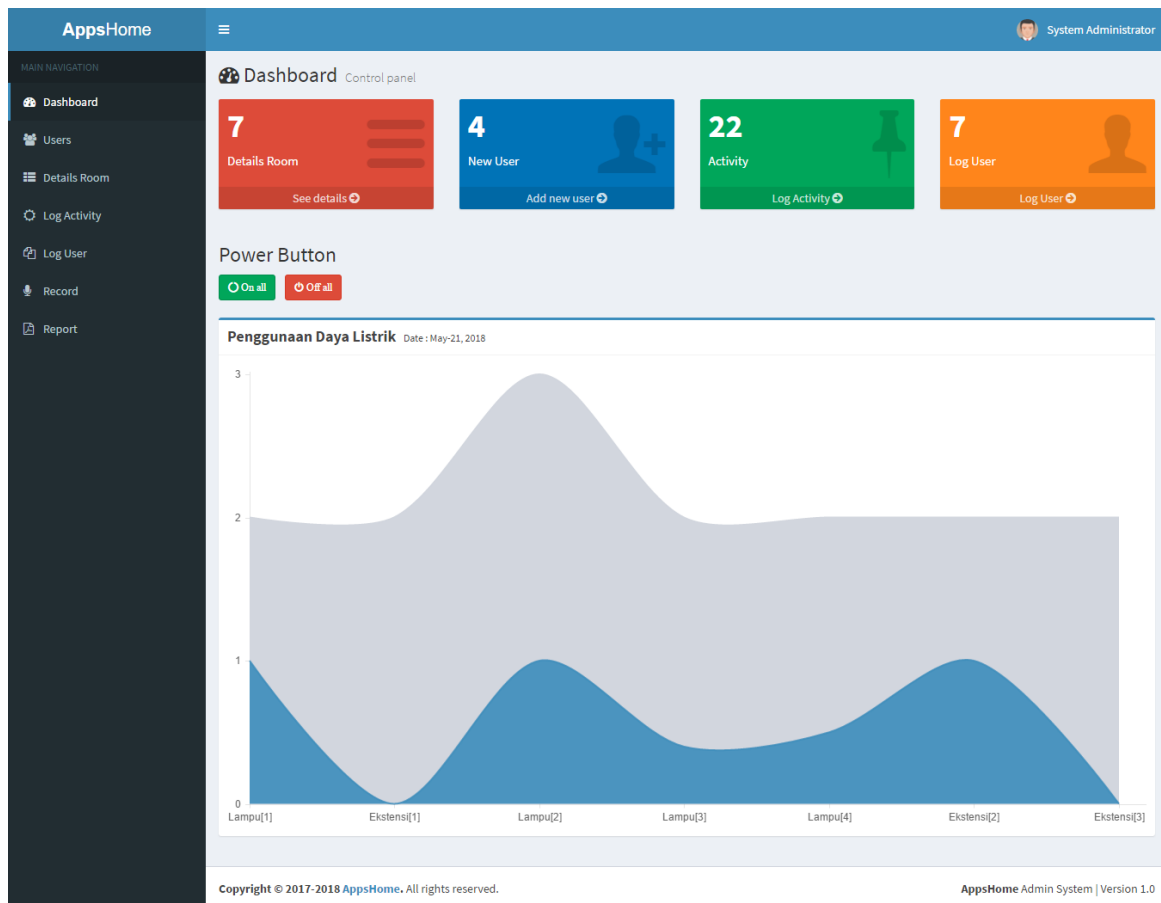
Gambar 27 Layer Login

#### 4.2.2. Dashboard

*Dashboard* merupakan *layer* yang tampil setelah *user* melakukan *login* dengan cara memasukkan *username* dan *password* yang valid. Terdapat dua tampilan dashboard yakni dashboard untuk admin dan dashboard untuk *client*.



Gambar 28 Dashboard untuk client



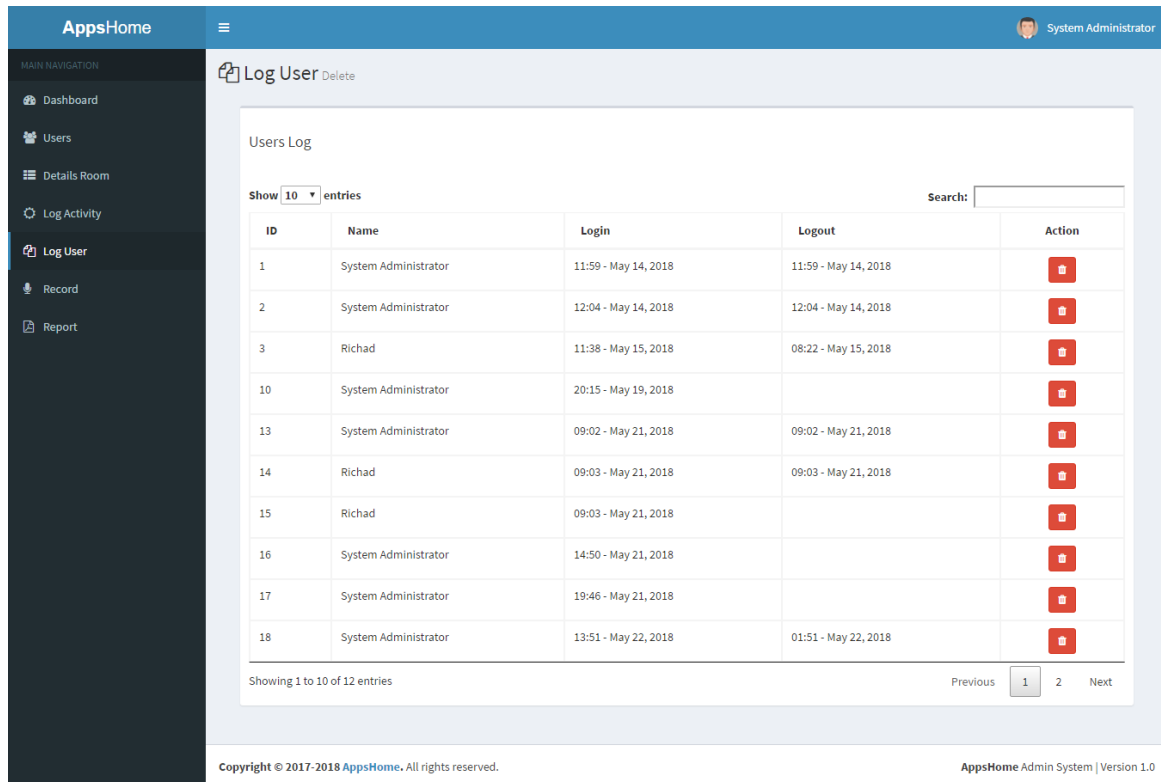
Gambar 29 Dashboard untuk admin

Pada bagian *sidebar dashboard* untuk *admin* terdapat menu *Users*, *Details Room*, *Log Activity*, *Log User*, *Record* dan *Report*. Sedangkan pada bagian *sidebar dashboard* untuk *client* hanya terdapat satu menu yakni menu *Details Room*. *Dashboard* menampilkan grafik penggunaan daya listrik dalam satu hari dalam satuan *KiloWatt (KW)*.

Pada bagian atas *dashboard* terdapat empat menu yakni *details room*, *new user*, *log activity* dan *log user*. Menu *details room* merupakan menu yang berisi rincian mengenai perangkat yang digunakan pada ruangan, pada menu inilah *user* dapat mengontrol perangkat pada setiap ruangan. Menu *New User* merupakan menu untuk menambah *user* baru. *User* yang dapat mengakses aplikasi ini dapat berjumlah lebih dari satu orang. Menu *Activity* merupakan menu yang berisi catatan mengenai waktu penggunaan perangkat sedangkan menu *Log User* merupakan menu untuk melihat data *user*. Admin dapat mengakses keempat menu tersebut tetapi *client* hanya bisa mengakses menu *Details Room*.

#### 4.2.3. Layer Log User

*Layer Log User* berisi data tentang *user* yakni ID, *email*, nama, hak akses dan keterangan mengenai kapan *user* aktif menggunakan aplikasi. *Layer Log User* hanya dapat diakses oleh admin. Tampilan *layer Log User* dapat dilihat pada Gambar 30.



| ID | Name                 | Login                | Logout               | Action       |
|----|----------------------|----------------------|----------------------|--------------|
| 1  | System Administrator | 11:59 - May 14, 2018 | 11:59 - May 14, 2018 | [Red Square] |
| 2  | System Administrator | 12:04 - May 14, 2018 | 12:04 - May 14, 2018 | [Red Square] |
| 3  | Richad               | 11:38 - May 15, 2018 | 08:22 - May 15, 2018 | [Red Square] |
| 10 | System Administrator | 20:15 - May 19, 2018 |                      | [Red Square] |
| 13 | System Administrator | 09:02 - May 21, 2018 | 09:02 - May 21, 2018 | [Red Square] |
| 14 | Richad               | 09:03 - May 21, 2018 | 09:03 - May 21, 2018 | [Red Square] |
| 15 | Richad               | 09:03 - May 21, 2018 |                      | [Red Square] |
| 16 | System Administrator | 14:50 - May 21, 2018 |                      | [Red Square] |
| 17 | System Administrator | 19:46 - May 21, 2018 |                      | [Red Square] |
| 18 | System Administrator | 13:51 - May 22, 2018 | 01:51 - May 22, 2018 | [Red Square] |

Gambar 30 Layer Log User

#### 4.2.4. Layer Details Room

*Layer details room* menampilkan deskripsi mengenai perangkat yang digunakan didalam rumah. Deskripsi barang meliputi nama perangkat, batas daya, status, *action* dan *control*. *Layer Details Room* dapat diakses oleh admin dan client. Pada tabel *action*, admin dapat mengedit perangkat jika terjadi perubahan atau menghapus perangkat jika tidak digunakan lagi oleh *client*. Pada tabel *control*, *client* dapat menyalakan atau mematikan perangkat tersebut hanya dengan menekan tombol *on* atau tombol berwarna hijau untuk menyalakan perangkat dan tombol *off* atau tombol berwarna merah untuk mematikan perangkat. Tampilan *Layer Details Room* dapat dilihat pada Gambar 31.

Copyright © 2017-2018 AppsHome. All rights reserved. AppsHome Admin System | Version 1.0

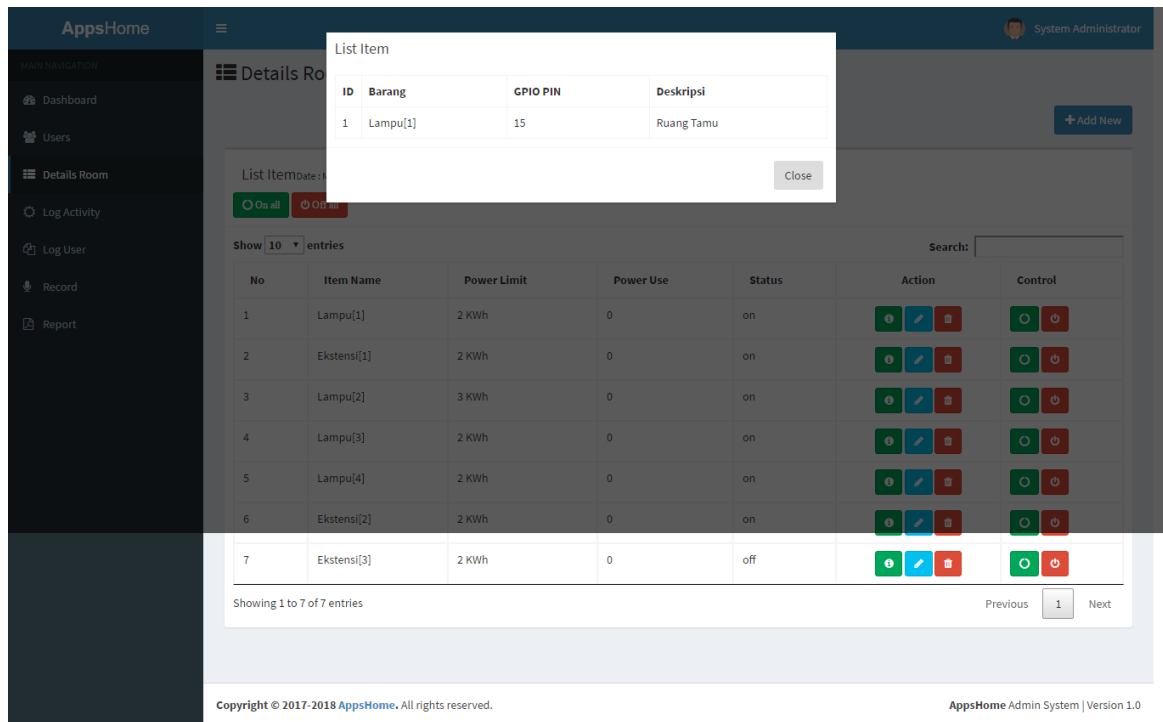
Gambar 31 Menu *Details Room*

Pada layer *Details Room* juga terdapat menu yang dapat menghidupkan atau mematikan semua perangkat sekaligus yakni tombol *On all* untuk menghidupkan semua perangkat atau *Off all* untuk mematikan semua perangkat. Selain itu, terdapat menu *add new* yang berguna untuk menambah perangkat yang baru beserta deskripsi lengkapnya. Pada menu *Add New*, admin harus mengisi kotak yang disediakan berupa nama perangkat, batas daya dan deskripsi. Menu *Add New* pada layer *Details Room* dapat dilihat pada Gambar 32.

Copyright © 2017-2018 AppsHome. All rights reserved. AppsHome Admin System | Version 1.0

Gambar 32 Menu *Item Management*

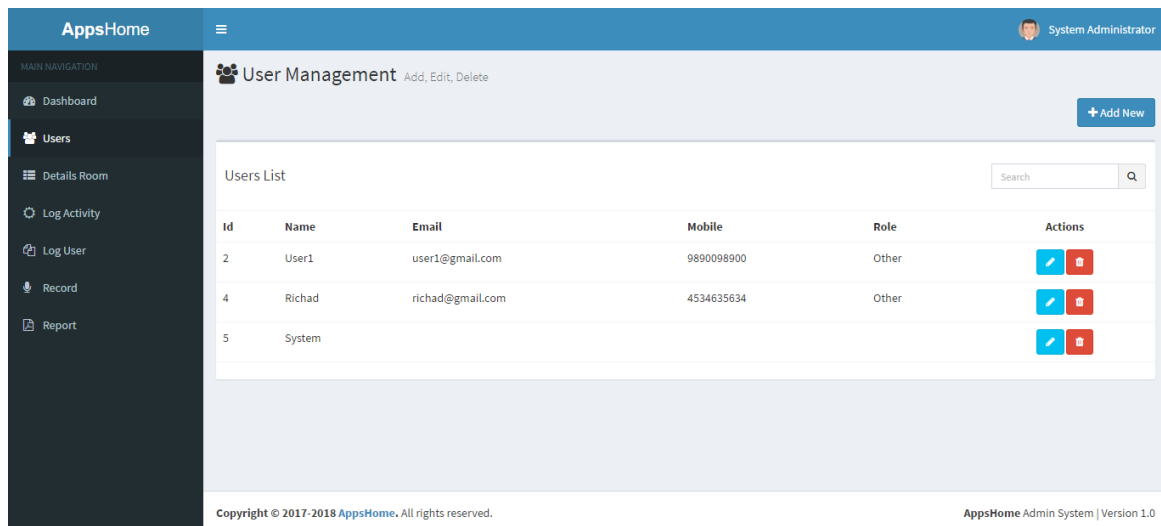
Menu lain yang terdapat pada *layer Details Room* adalah *View Details*, menu ini berisi mengenai letak setiap perangkat yang digunakan oleh pengguna. Pada *prototype*, perangkat yang digunakan sebanyak tujuh yakni 4 lampu dan 3 ekstensi. Menu *View Details* dapat dilihat pada Gambar 33.



Gambar 33 Menu *View Details*

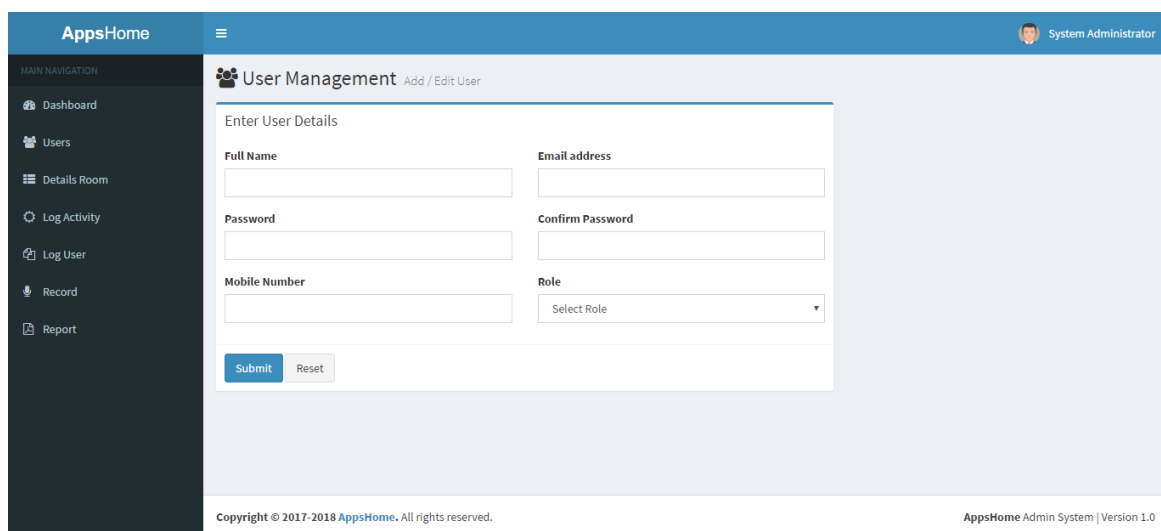
#### 4.2.5. Layer Users

Layer *users* hanya dapat diakses oleh admin. Pada layer *users* ditampilkan daftar pemilik rumah yang dapat menggunakan aplikasi ini untuk mengontrol penggunaan perangkat listrik dalam rumah. Pengguna aplikasi dideskripsikan dengan id, nama, email, nomor handphone dan peran. Dalam menu ini juga terdapat menu edit atau add new dan hapus. Menu *users* dapat dilihat pada Gambar 34.



Gambar 34 Menu *User Management*

Pada *layer users* terdapat menu *add new* yang digunakan untuk menambah pengguna yang dapat mengakses aplikasi. Menu *add new* menampilkan deskripsi yang harus diisi oleh *admin* berupa nama lengkap pengguna, *password*, nomor *handphone*, alamat *email* dan peran.



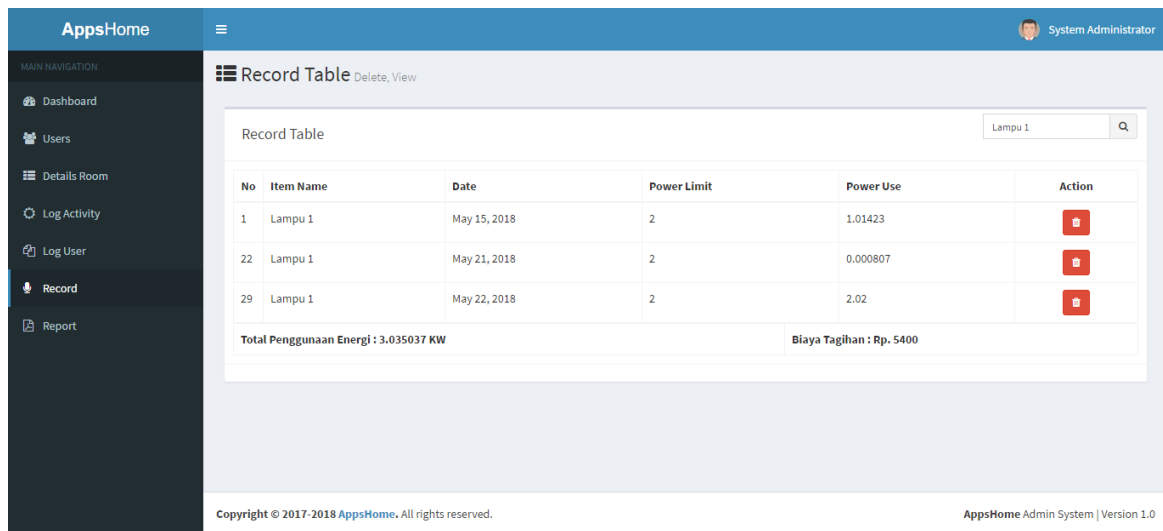
Gambar 35 menu *Add User*

#### 4.2.6. Layer Record

*Layer Record* menampilkan penggunaan daya setiap perangkat yang dihidupkan. Layer ini berisi deskripsi mengenai nama perangkat, tanggal, batas daya yang dapat dikonsumsi setiap perangkat dan penggunaan daya dari setiap perangkat yang dihidupkan. Jumlah penggunaan daya akan terus meningkat selama perangkat dihidupkan dan digunakan. Jumlah penggunaan daya akan berhenti jika perangkat dimatikan. Perangkat



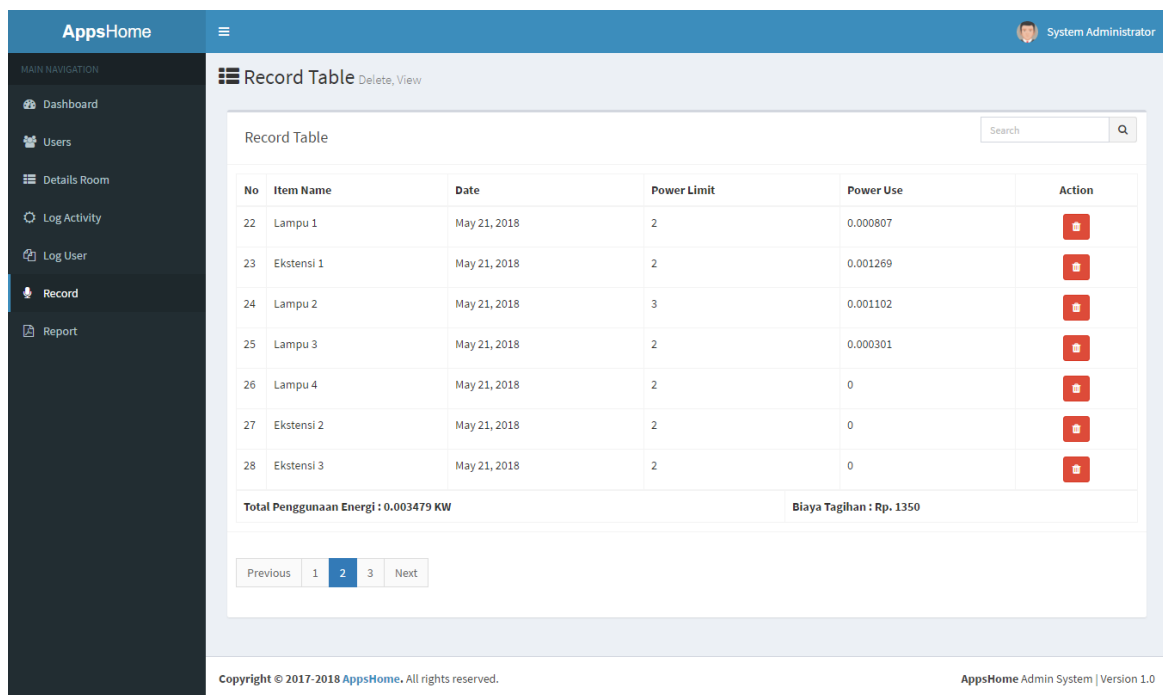
akan mati otomatis jika sudah melebihi batas daya yang ditentukan. Tampilan layer Record dapat dilihat pada Gambar 36 dan Gambar 37.



| No | Item Name | Date         | Power Limit | Power Use | Action |
|----|-----------|--------------|-------------|-----------|--------|
| 1  | Lampu 1   | May 15, 2018 | 2           | 1.01423   |        |
| 22 | Lampu 1   | May 21, 2018 | 2           | 0.000807  |        |
| 29 | Lampu 1   | May 22, 2018 | 2           | 2.02      |        |

Total Penggunaan Energi : 3.035037 KW      Biaya Tagihan : Rp. 5400

Gambar 36 Layer *Record* (per item)



| No | Item Name  | Date         | Power Limit | Power Use | Action |
|----|------------|--------------|-------------|-----------|--------|
| 22 | Lampu 1    | May 21, 2018 | 2           | 0.000807  |        |
| 23 | Ekstensi 1 | May 21, 2018 | 2           | 0.001269  |        |
| 24 | Lampu 2    | May 21, 2018 | 3           | 0.001102  |        |
| 25 | Lampu 3    | May 21, 2018 | 2           | 0.000301  |        |
| 26 | Lampu 4    | May 21, 2018 | 2           | 0         |        |
| 27 | Ekstensi 2 | May 21, 2018 | 2           | 0         |        |
| 28 | Ekstensi 3 | May 21, 2018 | 2           | 0         |        |

Total Penggunaan Energi : 0.003479 KW      Biaya Tagihan : Rp. 1350

Previous 1 2 3 Next

Gambar 37 Layer Record (Per tanggal)

#### 4.2.7. Layer Report

*Layer Report* menampilkan keseluruhan penggunaan daya yang telah digunakan selama satu hari ke dalam sebuah PDF dan setiap harinya data yang di tampilkan berbeda berdasarkan penggunaan daya yang terpakai.

Report Power Used in House

1 / 1

Laporan AppsHome

Record Penggunaan Daya Arus Listrik

by D3TK-03

Report Power Used in House

| Month | Date       | Item Name   | Power Used |          |
|-------|------------|-------------|------------|----------|
| Jun   | 2018-06-02 | Lampu 1     | 0.00594    |          |
|       |            | Ekstensi 1  | 0.003657   |          |
|       |            | Lampu 2     | 0.000246   |          |
|       |            | Lampu 3     | 0.002007   |          |
|       |            | Lampu 4     | 0          |          |
|       |            | Ekstensi 2  | 0          |          |
|       |            | Ekstensi 3  | 0          |          |
|       |            | total       | 0.01185    |          |
|       | 2018-06-03 | Ekstensi 3  | 0          |          |
|       |            | Lampu 1     | 0.001671   |          |
|       |            | Ekstensi 1  | 0.001419   |          |
|       |            | Lampu 2     | 0.005533   |          |
|       |            | Lampu 3     | 0.003555   |          |
|       |            | Lampu 4     | 0          |          |
|       |            | Ekstensi 2  | 0          |          |
|       |            | total       | 0.012178   |          |
|       |            | Grand Total |            | 0.024028 |

Gambar 30 Menu *Layer Report*

## **Bab V**

### **Pengujian dan Analisis**

#### **5.1. Pengujian**

Pengujian dilakukan untuk melihat dan memastikan bahwa sistem yang dibangun dapat berjalan dengan baik. Pengujian yang dilakukan meliputi pengujian *prototype*, Sensor CT dan aplikasi yang dibangun.

##### **5.1.1. Pengujian *Prototype***

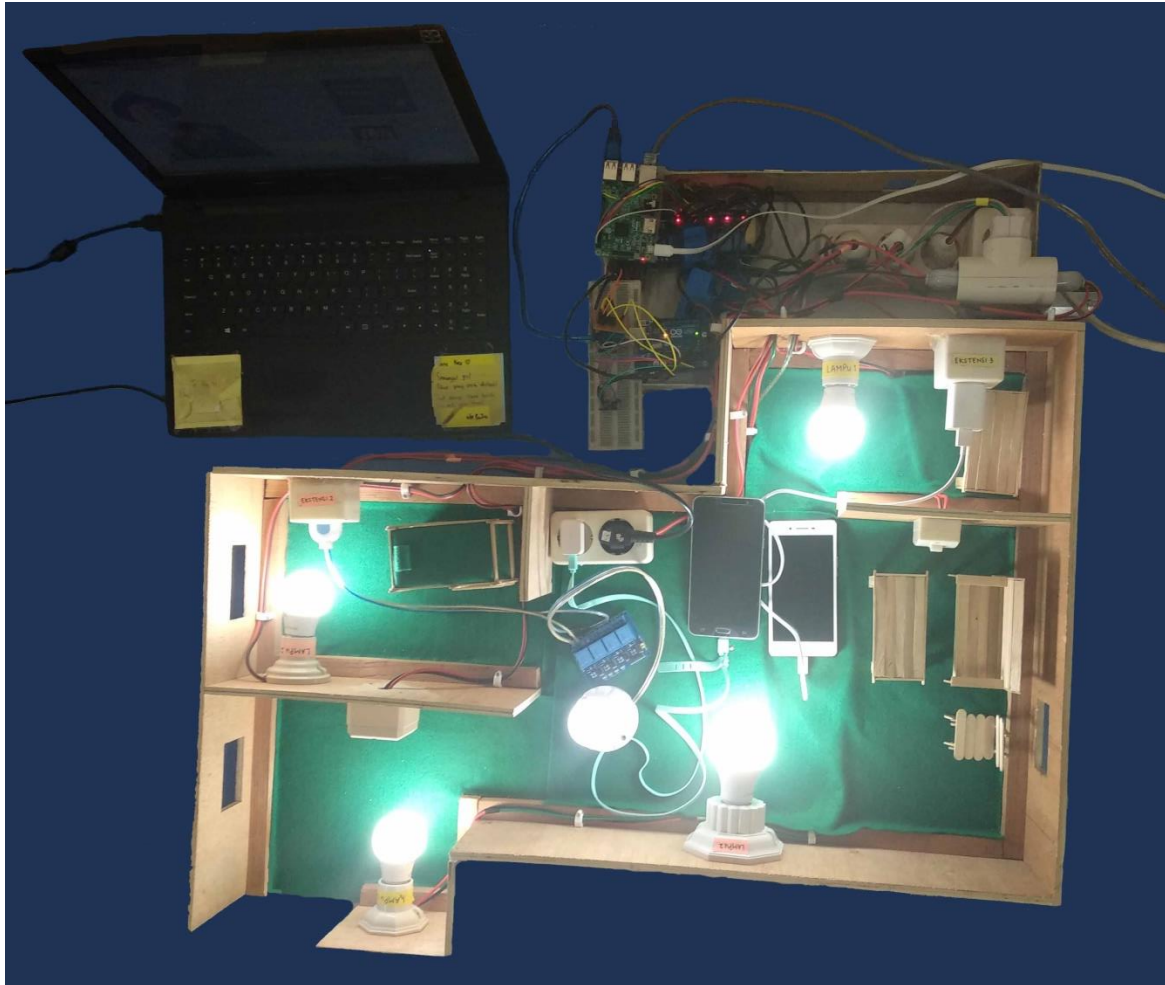
Pada subbab ini dijelaskan mengenai skenario pengujian dan hasil pengujian *prototype Smart House Electricity Monitoring Based on Raspberry Pi*.

##### **5.1.1.1. Skenario Pengujian *Prototype* Rumah**

Pengujian ini dilakukan untuk menguji semua perangkat yang ada pada *prototype* rumah secara manual. Pengujian ini bertujuan untuk memastikan apakah semua perangkat dapat menyala dan berfungsi dengan baik. Perangkat yang diuji berupa 4 lampu, 3 ekstensi dan 2 *switch*. Jika sudah dipastikan semua perangkat dapat menyala atau berfungsi dengan baik maka langkah selanjutnya adalah pengujian *prototype* menggunakan aplikasi.

##### **5.1.1.2. Hasil Pengujian**

Dari pengujian yang telah dilakukan, maka didapatkan hasil seperti Gambar 38. Semua perangkat dapat menyala dan berfungsi dengan baik. Pengujian dilakukan dengan menyalakan semua lampu dan mencolokkan charger pada ekstensi. Ekstensi pada kamar digunakan untuk menyalakan lampu, ekstensi pada ruang tengah digunakan untuk mengisi daya *handphone* dan laptop, sedangkan ekstensi pada dapur digunakan untuk mengisi daya *handphone*.



Gambar 38 Hasil Pengujian *Prototype* Rumah

### 5.1.2. Pengujian Sensor *Current Transformer*

Pada subbab ini dijelaskan mengenai skenario dan hasil pengujian Sensor *Current Transformer*.

#### 5.1.2.1. Skenario Pengujian Sensor *Current Transformer*

Skenario pengujian Sensor CT dilakukan untuk melihat apakah Sensor CT dapat berfungsi dengan baik atau tidak. Skenario Sensor CT dilakukan dengan 3 cara yakni mengukur daya yang di konsumsi oleh setiap perangkat selama kurun waktu tertentu, melihat apakah yang akan terjadi apabila daya yang dikonsumsi melebihi batas daya yang ditentukan untuk setiap perangkat dan mematikan semua perangkat pada pukul 23.59 dan mereset data pada aplikasi website kembali ke nol. Skenario pengujian Sensor *Current Transformer* dapat dilihat pada **Tabel 13**.

**Tabel 13 Skenario Pengujian Sensor CT**

| Pengujian          | Perangkat  | Batas Daya (KiloWatt) | Beban  | Skenario Pengujian   | Hasil yang diharapkan   |
|--------------------|------------|-----------------------|--|--|---|
| <b>Pengujian 1</b> | Lampu 1    | 2                     | -  | Perangkat dihidupkan/ dinyalakan untuk melihat daya yang dikonsumsi selama:<br>- 15 menit<br>- 30 menit<br>- 60 menit/ 1 jam                         | Sistem dapat mengukur besar daya yang dikonsumsi oleh perangkat selama kurun waktu yang telah ditentukan pada skenario dengan tujuan untuk membandingkan daya yang dikonsumsi pada jangka waktu dan perangkat yang berbeda. |
|                    | Lampu 2    | 2                     | -  |  |   |
|                    | Lampu 3    | 3                     | -  |  |   |
|                    | Lampu 4    | 2                     | -  |  |   |
|                    | Ekstensi 1 | 2                     | Charger Laptop Lenovo & Charger HP Samsung G7 2016 |  |   |
|                    | Ekstensi 2 | 2                     | Charger HP Samsung                                 |  |   |
|                    | Ekstensi 3 | 2                     | Charger HP Nokia 105                               |  |   |
| <b>Pengujian 2</b> | Lampu 1    | 2                     | -  | Daya yang dikonsumsi perangkat melebihi batas daya yang dapat di konsumsi oleh Perangkat   | Perangkat mati secara otomatis dan sistem berhenti menghitung ataupun menampilkan data berupa daya listrik.   |
|                    | Ekstensi 1 | 2                     | Charger Hp Samsung G7 2016                         |  |   |
|                    | Lampu 2    | 3                     | -  |  |   |
|                    | Lampu 3    | 2                     | -  |  |   |
| <b>Pengujian 3</b> | Lampu 1    | 2                     | -  | Aplikasi website akan mereset data kembali ke nol pada pukul 23.59, perangkat akan mati secara otomatis dan sistem berhenti menampilkan daya listrik | aplikasi website mereset data kembali ke nol pada pukul 23.59, sistem mati secara otomatis dan akan berfungsi kembali setelah jam 24.00   |
|                    | Lampu 2    | 2                     | -  |  |   |
|                    | Lampu 3    | 3                     | -  |  |   |
|                    | Lampu 4    | 2                     | -  |  |   |
|                    | Ekstensi 1 | 2                     | Charger Laptop Lenovo & Charger HP Samsung G7 2016 |  |   |
|                    | Ekstensi 2 | 2                     | Charger HP Samsung G7 2016                         |  |   |
|                    | Ekstensi 3 | 2                     | Charger HP Nokia 105                               |  |   |

Pengujian sensor *Current Transformer* (CT) dilakukan dengan menjepit sensor pada setiap kabel positif pada perangkat yang diukur. Pada pengujian ini langkah pertama yang dilakukan adalah memastikan data yang berupa daya listrik yang dikonsumsi perangkat dapat diambil menggunakan Sensor CT dan ditampilkan pada aplikasi. Pengujian ini dilakukan dengan menghidupkan semua perangkat yang ada di prototype rumah. Perangkat dihidupkan selama 15 menit, 30 menit dan 60 menit untuk melihat perbedaan besar daya yang dikonsumsi oleh kedua perangkat. Daya listrik yang telah didapatkan dari Sensor CT akan ditampilkan di layer record pada tabel penggunaan daya.

Setelah data didapatkan, langkah selanjutnya adalah menguji apakah perangkat mati otomatis jika melebihi batas daya yang ditentukan. Pengujian ini dilakukan dengan 4 perangkat yakni lampu 1, lampu 2, lampu 3 dan ekstensi 1. Ketiga lampu memiliki batas daya sebesar 2 KW, sedangkan ekstensi 1 memiliki batas daya sebesar 3 KWh. Jika suatu perangkat sudah melewati batas yang ditentukan, maka perangkat tersebut akan mati otomatis. Daya tidak akan mengalir dan data berhenti berjalan.

Pengujian selanjutnya adalah memastikan perangkat berhenti menyala apabila sudah pukul 23.59 dan daya berhenti mengalir. Aplikasi akan mereset daya kembali ke nol dan akan berfungsi kembali setelah pukul 24.00 jika perangkat dihidupkan. Pengujian dimaksudkan untuk melihat perbedaan daya yang dikonsumsi oleh perangkat setiap harinya. Daya yang dikonsumsi oleh semua perangkat dalam satu hari disimpan pada menu record, sehingga user bisa mengetahui jumlah daya yang dikonsumsi perhari.

#### **5.1.2.2. Hasil Pengujian**

Pengujian pertama berhasil dilakukan, semua Sensor CT dapat mengambil data dari setiap perangkat yang menggunakan arus listrik. Data yang berupa daya listrik tersebut diukur setiap detik. Jika Sensor CT dijepitkan pada kabel perangkat namun perangkat tidak dihidupkan, maka daya yang dikonsumsi perangkat masih nol (tidak ada daya listrik yang dikonsumsi). Data ditampilkan melalui program python dan di running menggunakan console di *Raspberry Pi*. Data yang ditampilkan merupakan hasil kalkulasi dari data yang didapatkan dari Sensor CT. Pada *console* tersebut ditampilkan nama perangkat, status dan daya yg dikonsumsi. Gambar 39 menunjukkan hasil pengukuran daya yang dikonsumsi perangkat selama 15 menit.

```

Daya Ekstensi 1
0.001264 KW
ON
Daya Ekstensi 2
0.000216 KW
ON
Daya Ekstensi 3
0.000167 KW
ON
Daya Lampu 1
0.001404 KW
ON
Daya Lampu 2
0.000833 KW
ON
Daya Lampu 3
0.001022 KW
ON
Daya Lampu 4
0.000585 KW
ON

```

Gambar 39 Pengujian selama 15 menit

Gambar 40 menunjukkan hasil pengukuran daya yang dikonsumsi perangkat selama 30 menit. Pada gambar tersebut dapat dilihat bahwa daya yang dikonsumsi selama 30 menit berjumlah lebih banyak dibanding data yang dikonsumsi selama 15 menit.

```

Daya Ekstensi 2
0.001104 KW
ON
Daya Ekstensi 3
0.000246 KW
ON
Daya Ekstensi 1
0.002356 KW
ON
Daya Lampu 1
0.002132 KW
ON
Daya Lampu 2
0.00151 KW
ON
Daya Lampu 3
0.001583 KW
ON
Daya Lampu 4
0.000899 KW
ON

```

Gambar 40 Pengujian selama 30 menit

Hasil pengukuran daya yang dikonsumsi perangkat selama 60 menit ditunjukkan pada Gambar 41. Pada gambar tersebut dapat dilihat bahwa daya yang dikonsumsi selama 60 menit berjumlah lebih banyak dibanding data yang dikonsumsi selama 15 menit dan 30 menit.

```

Daya Ekstensi 2
0.003628 KW
ON
Daya Ekstensi 3
0.000246 KW
ON
Daya Ekstensi 1
0.005892 KW
Daya Lampu 1
0.00416 KW
ON
Daya Lampu 4
0.001868 KW
ON
Daya Lampu 2
0.003182 KW
ON
Daya Lampu 3
0.003128 KW
ON

```

Gambar 41 Pengujian selama 60 menit

Keterangan lengkap mengenai hasil pengujian 1 untuk mengukur daya listrik yang digunakan semua perangkat dapat dilihat pada **Tabel 14** dan **Tabel 15**.

**Tabel 14 Hasil Pengujian Sensor CT terhadap Lampu**

| Sensor CT | Perangkat yang diukur | Besarnya Tegangan | Besarnya Daya | Status | Besarnya Daya yang dikonsumsi oleh perangkat (KW) |          |          |
|-----------|-----------------------|-------------------|---------------|--------|---|----------|----------|
|           |                       |                   |               |        | 15 menit  | 30 menit | 60 menit |
| CT 1      | Lampu 1               | 220 V             | 3 Watt        | ON     | 0.001404  | 0.002132 | 0.00416  |
| CT 2      | Lampu 2               | 220 V             | 5 Watt        | ON     | 0.000833  | 0.00151  | 0.001868 |
| CT 3      | Lampu 3               | 220 V             | 3 Watt        | ON     | 0.001022  | 0.001583 | 0.003182 |
| CT 4      | Lampu 4               | 220 V             | 3 Watt        | ON     | 0.000585  | 0.000899 | 0.003128 |



**Tabel 15 Hasil Pengujian Sensor CT terhadap Ekstensi**

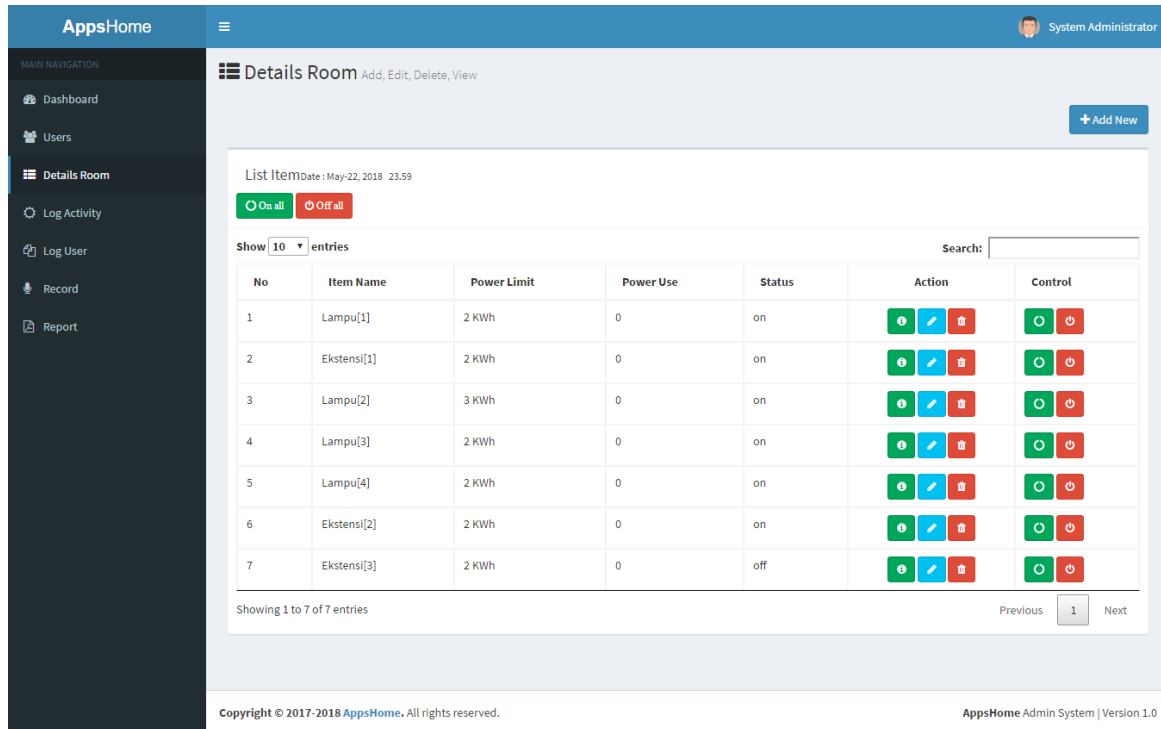
| Sensor CT | Perangkat yang diukur | Besarnya Tegangan | Beban yang dicolokkan pada perangkat                       | Status | Besarnya Daya yang dikonsumsi oleh perangkat (KWh) |          |          |
|-----------|-----------------------|-------------------|--|--------|--|----------|----------|
|           |                       |                   |  |        | 15 menit   | 30 menit | 60 menit |
| CT 5      | Ekstensi 1            | 220 V             | Charger Laptop lenovo & Charger HP Samsung Samsung G7 2016 | ON     | 0.001264   | 0.002356 | 0.005892 |
| CT 6      | Ekstensi 2            | 220 V             | Charger HP Samsung Samsung G7 2016                         | ON     | 0.000216   | 0.001104 | 0.003628 |
| CT 7      | Ekstensi 3            | 220 V             | Charger HP Nokia 105                                       | ON     | 0.000167   | 0.000246 | 0.000246 |




































Pengujian kedua didapatkan hasil seperti Gambar 42, perangkat berhenti menyala jika sudah melewati batas daya yang ditentukan. Aplikasi mengubah status perangkat yang tadinya on menjadi off. Pada Gambar 42 ditunjukkan bahwa lampu 1 dan lampu 3 telah melebihi batas yang ditentukan yaitu 2 Kwh dan status kedua perangkat tersebut menjadi off. Sementara pada ekstensi 1 dan lampu 2 daya tetap berjalan dan status kedua perangkat tersebut masih on karena daya yang dikonsumsi tidak melebihi batas daya yang ditentukan.

| No | Item Name   | Power Limit | Power Use | Status | Action | Control |
|----|-------------|-------------|-----------|--------|--------|---------|
| 1  | Lampu[1]    | 2 KWh       | 2.02      | off    |        |         |
| 2  | Ekstensi[1] | 2 KWh       | 1.368     | on     |        |         |
| 3  | Lampu[2]    | 3 KWh       | 2.015     | on     |        |         |
| 4  | Lampu[3]    | 2 KWh       | 2.0034    | off    |        |         |
| 5  | Lampu[4]    | 2 KWh       | 0         | on     |        |         |
| 6  | Ekstensi[2] | 2 KWh       | 0         | on     |        |         |
| 7  | Ekstensi[3] | 2 KWh       | 0         | on     |        |         |

Gambar 42 Daya yang dikonsumsi perangkat melebihi batas

Pengujian ketiga berhasil dilakukan, perangkat berhenti menyala pada pukul 23.59 dan daya berhenti mengalir. Aplikasi mereset daya kembali ke nol dan berfungsi kembali saat pukul 24.00. Hasil pengujian dapat dilihat pada Gambar 43.



| No | Item Name   | Power Limit | Power Use | Status | Action   | Control   |
|----|-------------|-------------|-----------|--------|--|---|
| 1  | Lampu[1]    | 2 KWh       | 0         | on     |       |     |
| 2  | Ekstensi[1] | 2 KWh       | 0         | on     |       |     |
| 3  | Lampu[2]    | 3 KWh       | 0         | on     |       |     |
| 4  | Lampu[3]    | 2 KWh       | 0         | on     |       |     |
| 5  | Lampu[4]    | 2 KWh       | 0         | on     |       |     |
| 6  | Ekstensi[2] | 2 KWh       | 0         | on     |       |     |
| 7  | Ekstensi[3] | 2 KWh       | 0         | off    |    |   |

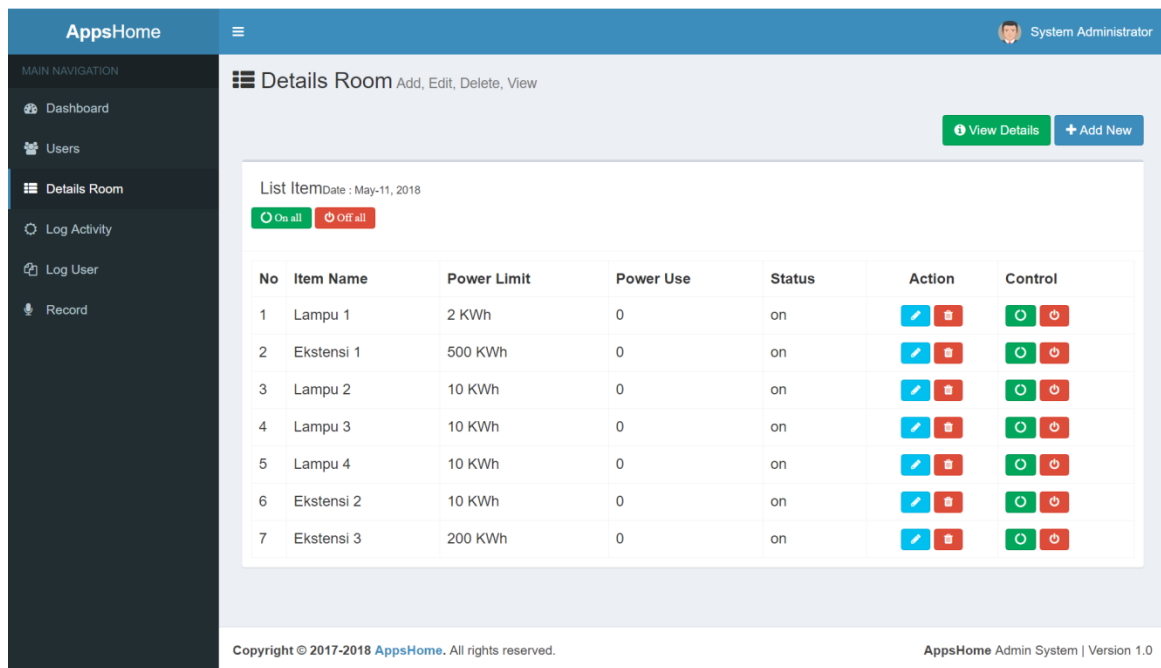
Gambar 43 Tampilan aplikasi ketika pukul 23.59

### 5.1.3. Pengujian Aplikasi

Aplikasi diuji untuk memastikan aplikasi yang dibangun dapat di implementasikan dengan baik.

#### 5.1.3.1. Skenario Pengujian *Layer Details Room*

Pengujian pertama yang dilakukan adalah menekan button atau tombol *ON/OFF* untuk setiap perangkat yang digunakan pada layer *details room* di tabel *control*. Tombol *ON* atau tombol yang berwarna hijau digunakan untuk menyalakan perangkat sedangkan tombol *OFF* atau tombol yang berwarna merah digunakan untuk mematikan perangkat. User juga dapat menekan tombol *On all* untuk menyalakan semua perangkat sekaligus dan tombol *Off all* untuk mematikan semua perangkat sekaligus.

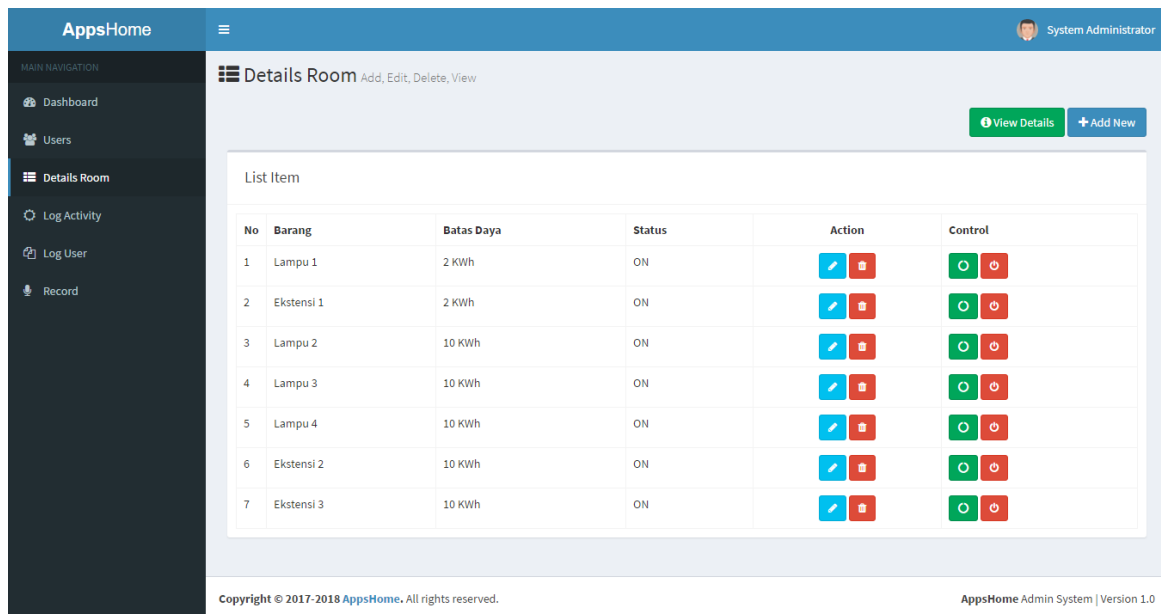


Gambar 44 *Layer Details Room*

Pengujian selanjutnya adalah mengedit perangkat yang digunakan pada layer *Details Room* ditabel *Action*. Pengujian dilakukan dengan mengubah batas daya pada ekstensi 3 yaitu 200 KW menjadi 100 KW.

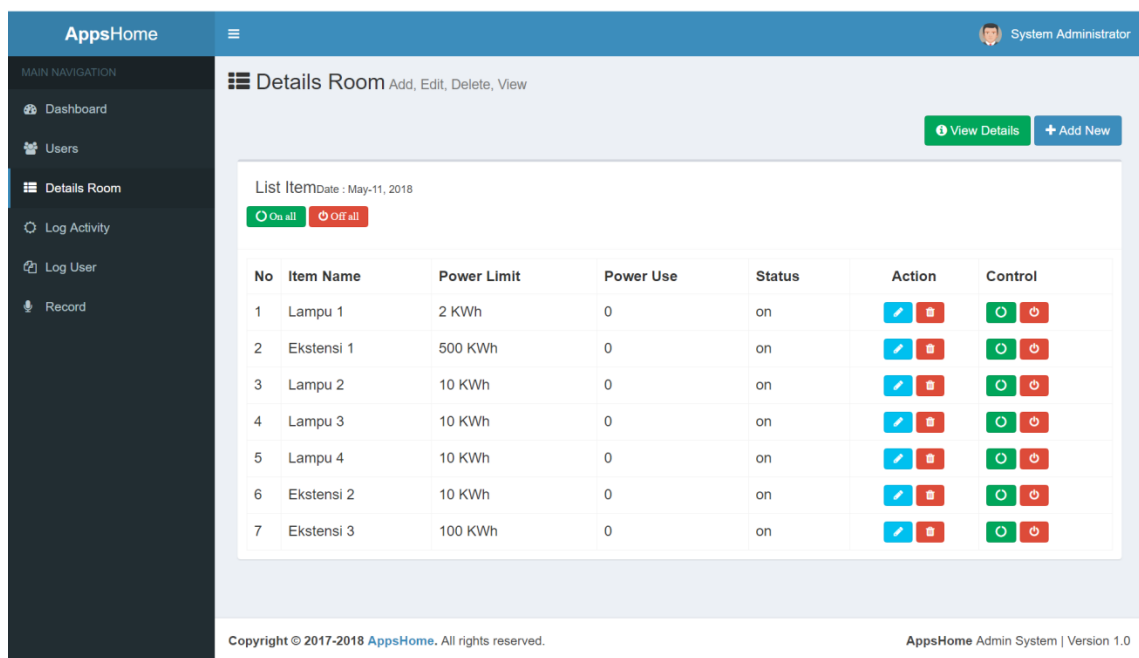
#### 5.1.3.2. Hasil Pengujian *Layer Details Room*

Pengujian pertama berhasil dilakukan, jika tombol ON ditekan maka perangkat menyala dan status perangkat berubah menjadi ON. Sebaliknya, jika tombol OFF ditekan maka perangkat berhenti menyala dan status perangkat akan berubah menjadi OFF. Perangkat yang menyala atau tidak menyala dapat diketahui dari tabel status pada layer *details room*. Dari pengujian pertama yang telah dilakukan maka didapatkan hasil seperti pada Gambar 45.



Gambar 45 Pengujian tombol *ON/OFF* pada aplikasi

Pengujian selanjutnya yaitu untuk mengedit perangkat berhasil dilakukan. Batas daya ekstensi yang sebelumnya adalah 200 KWh berubah menjadi 100 KWh. Hasil pengujian dapat dilihat pada Gambar 46.

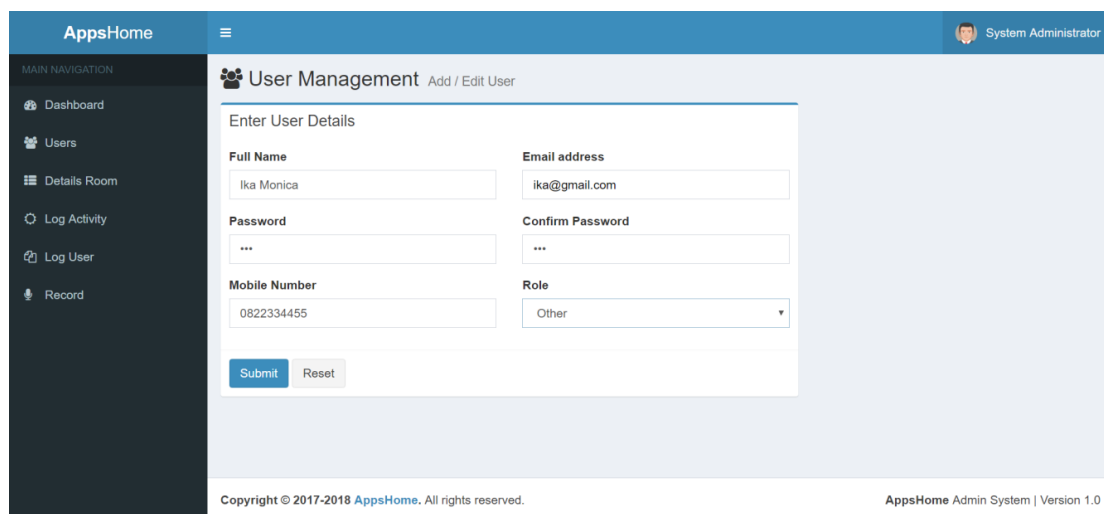


Gambar 46 Hasil Pengujian untuk mengedit perangkat

### 5.1.3.3. Skenario Pengujian *Layer Users*

Pengujian yang dilakukan adalah menambah user baru pada menu *Add New* di *layer Users*. Admin harus mengisi data diri user berupa nama, email, password, nomor handphone dan peran. Jika user berhasil ditambahkan, user dapat login ke aplikasi sebagai *client*

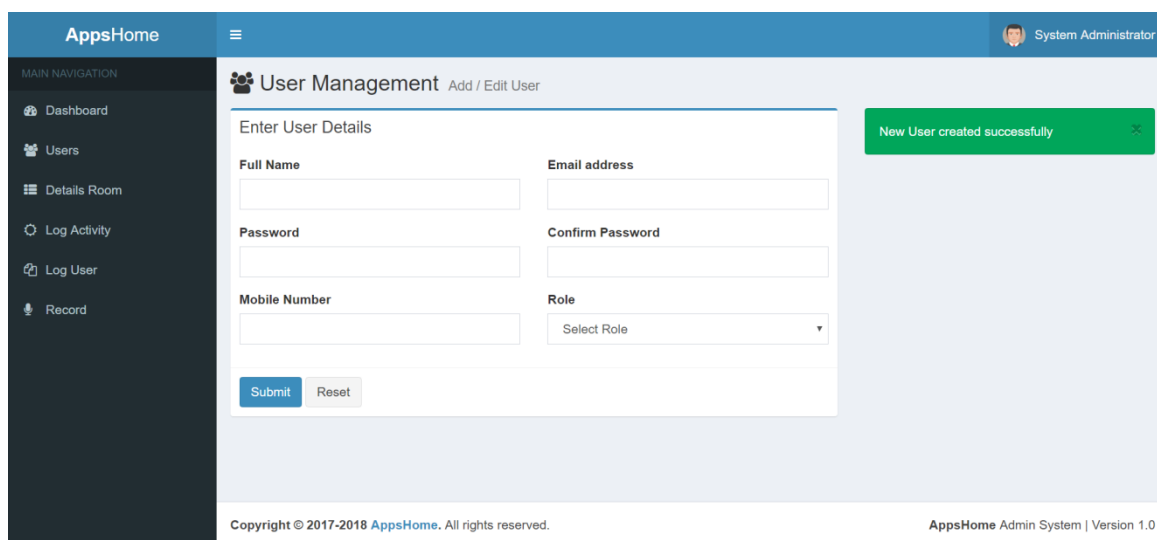
menggunakan username dan password yang telah dibuat sebelumnya. Menu *Add New* dapat dilihat pada Gambar 47.



Gambar 47 Menu *Add New User*

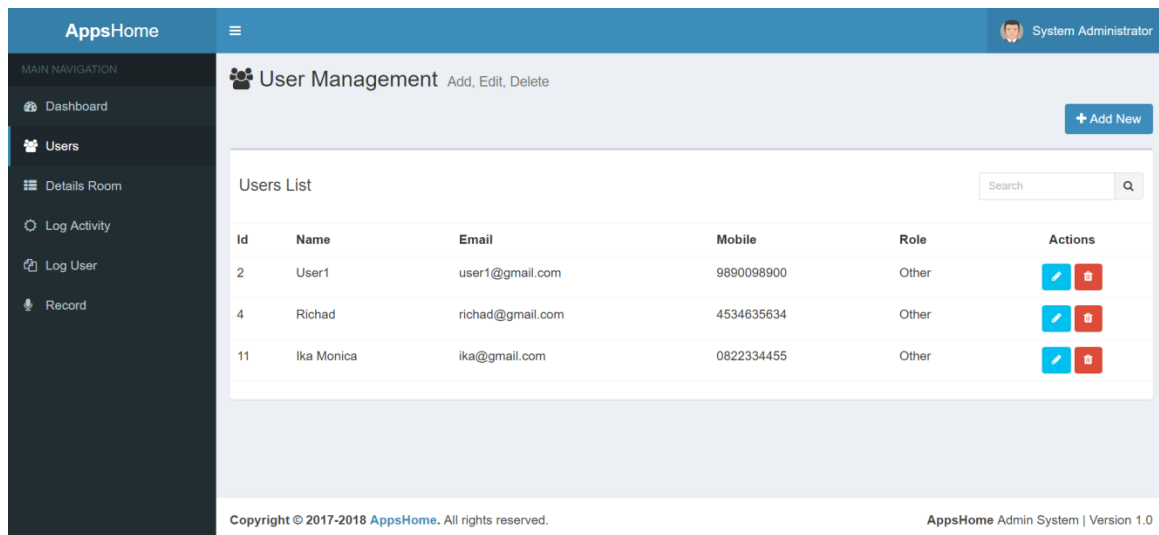
#### 5.1.3.4. Hasil Pengujian *Layer Users*

Pengujian yang dilakukan untuk menambah user berhasil dilakukan. Jika user berhasil ditambah, maka akan muncul notifikasi seperti Gambar 48. Notifikasi berada pada sebelah kanan atas aplikasi.



Gambar 48 Notifikasi jika user berhasil ditambahkan

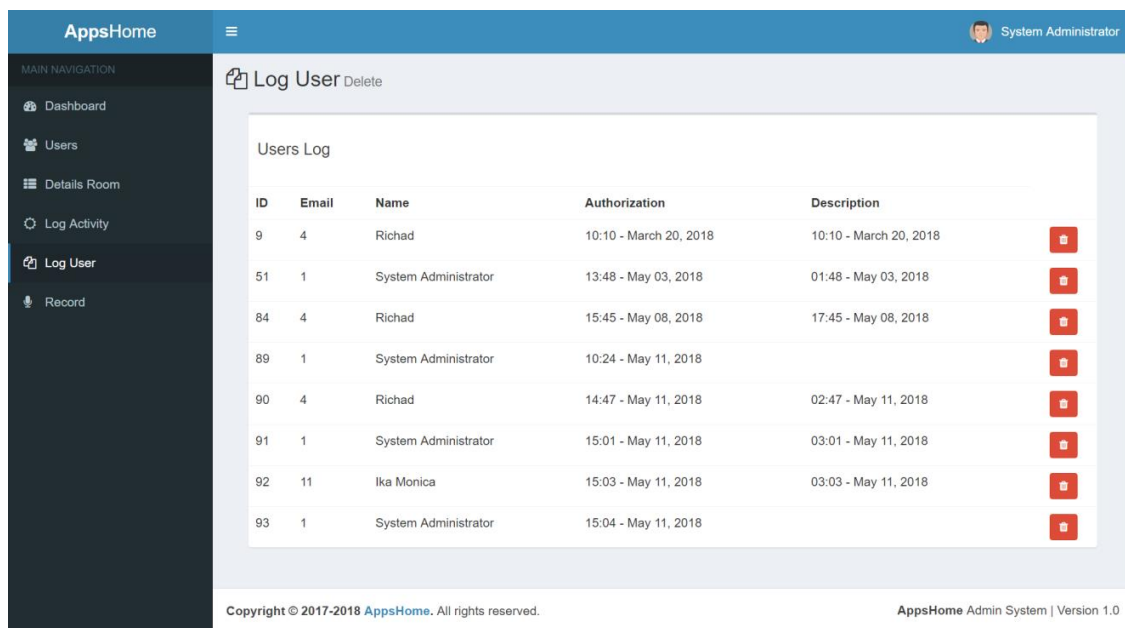
Setelah berhasil ditambahkan, user yang telah ditambah tersebut dapat dilihat pada user list seperti pada Gambar 49. User bernama Ika Monica berhasil ditambahkan dengan email ika@gmail.com.



Gambar 49 Hasil Pengujian Layer Users

#### 5.1.3.5. Skenario Pengujian *Layer Log User*

Pengujian yang dilakukan yaitu menghapus data kapan user menggunakan aplikasi pada layer log user. Pengujian dilakukan dengan menghapus data user Ika Monica yang memiliki ID 92.



Gambar 50 Layer Log User

#### 5.1.3.6. Hasil Pengujian *Layer Log User*

Pengujian untuk menghapus data user Ika Monica pada layer *Log User* berhasil dilakukan, user Ika Monica telah terhapus dan tidak ada lagi pada layer *Log User*. Hasil pengujian dapat dilihat pada Gambar 51.

AppsHome

System Administrator

MAIN NAVIGATION

Dashboard

Users

Details Room

Log Activity

Log User

Record

Log User

Delete

Users Log

| ID | Email | Name                 | Authorization          | Description            |  |
|----|-------|----------------------|------------------------|------------------------|--|
| 9  | 4     | Richad               | 10:10 - March 20, 2018 | 10:10 - March 20, 2018 |  |
| 51 | 1     | System Administrator | 13:48 - May 03, 2018   | 01:48 - May 03, 2018   |  |
| 84 | 4     | Richad               | 15:45 - May 08, 2018   | 17:45 - May 08, 2018   |  |
| 89 | 1     | System Administrator | 10:24 - May 11, 2018   |                        |  |
| 90 | 4     | Richad               | 14:47 - May 11, 2018   | 02:47 - May 11, 2018   |  |
| 91 | 1     | System Administrator | 15:01 - May 11, 2018   | 03:01 - May 11, 2018   |  |
| 93 | 1     | System Administrator | 15:04 - May 11, 2018   |                        |  |

Copyright © 2017-2018 AppsHome. All rights reserved.

AppsHome Admin System | Version 1.0

Gambar 51 Hasil Pengujian *Layer Log User*

#### 5.1.3.7. Skenario Pengujian *Layer Record*

Layer record menampilkan besar daya yang dikonsumsi setiap perangkat dalam satuan KW dan biaya. Biaya yang ditampilkan adalah jumlah dari biaya setiap perangkat yang digunakan dalam sehari. Pengujian yang dilakukan pada layer record adalah menghapus data perangkat pada tanggal 10 Mei 2018 yakni lampu 1.

AppsHome

System Administrator

MAIN NAVIGATION

Dashboard

Users

Details Room

Log Activity

Log User

Record

Record Table

Delete, View

Record Table

| No | Item Name  | Date         | Power Limit | Power Use | Action      |
|----|------------|--------------|-------------|-----------|-------------|
| 1  | Lampu 1    | May 10, 2018 | 2           | 0.005554  | <div></div> |
| 2  | Ekstensi 1 | May 10, 2018 | 500         | 0         | <div></div> |
| 3  | Lampu 2    | May 10, 2018 | 10          | 0         | <div></div> |
| 4  | Lampu 3    | May 10, 2018 | 10          | 0         | <div></div> |
| 5  | Lampu 4    | May 10, 2018 | 10          | 0         | <div></div> |
| 6  | Ekstensi 2 | May 10, 2018 | 10          | 0         | <div></div> |
| 7  | Ekstensi 3 | May 10, 2018 | 100         | 0         | <div></div> |
| 24 | Lampu 1    | May 11, 2018 | 2           | 0         | <div></div> |
| 25 | Ekstensi 1 | May 11, 2018 | 500         | 0         | <div></div> |
| 26 | Lampu 2    | May 11, 2018 | 10          | 0         | <div></div> |
| 27 | Lampu 3    | May 11, 2018 | 10          | 0         | <div></div> |

Gambar 52 *Layer Record*

### 5.1.3.8. Hasil Pengujian *Layer Record*

Pada Gambar 53, *User* dapat melihat biaya tagihan yang dihasilkan dari 4 perangkat yakni lampu 1, ekstensi 1, lampu 2 dan lampu 3 pada tanggal 21 Mei 2018.

The screenshot shows the 'Record Table' interface in the AppsHome system. The table lists power usage records for May 21, 2018. The records are as follows:

| No | Item Name  | Date         | Power Limit | Power Use | Action   |
|----|------------|--------------|-------------|-----------|----------|
| 22 | Lampu 1    | May 21, 2018 | 2           | 0.000807  | [Delete] |
| 23 | Ekstensi 1 | May 21, 2018 | 2           | 0.001269  | [Delete] |
| 24 | Lampu 2    | May 21, 2018 | 3           | 0.001102  | [Delete] |
| 25 | Lampu 3    | May 21, 2018 | 2           | 0.000301  | [Delete] |
| 26 | Lampu 4    | May 21, 2018 | 2           | 0         | [Delete] |
| 27 | Ekstensi 2 | May 21, 2018 | 2           | 0         | [Delete] |
| 28 | Ekstensi 3 | May 21, 2018 | 2           | 0         | [Delete] |

Summary: Total Penggunaan Energi : 0.003479 KW, Biaya Tagihan : Rp. 1350

Navigation: Previous, 1, 2 (selected), 3, Next

Gambar 53 Biaya Tagihan pada 21 mei 2018

Pengujian untuk menghapus data pemakaian perangkat berhasil dilakukan. Hasil pengujian dapat dilihat pada Gambar 54.

The screenshot shows the 'Record Table' interface in the AppsHome system. The table lists power usage records for May 10, 2018, and May 11, 2018. The records are as follows:

| No | Item Name  | Date         | Power Limit | Power Use | Action   |
|----|------------|--------------|-------------|-----------|----------|
| 2  | Ekstensi 1 | May 10, 2018 | 500         | 0         | [Delete] |
| 3  | Lampu 2    | May 10, 2018 | 10          | 0         | [Delete] |
| 4  | Lampu 3    | May 10, 2018 | 10          | 0         | [Delete] |
| 5  | Lampu 4    | May 10, 2018 | 10          | 0         | [Delete] |
| 6  | Ekstensi 2 | May 10, 2018 | 10          | 0         | [Delete] |
| 7  | Ekstensi 3 | May 10, 2018 | 100         | 0         | [Delete] |
| 24 | Lampu 1    | May 11, 2018 | 2           | 0         | [Delete] |
| 25 | Ekstensi 1 | May 11, 2018 | 500         | 0         | [Delete] |
| 26 | Lampu 2    | May 11, 2018 | 10          | 0         | [Delete] |
| 27 | Lampu 3    | May 11, 2018 | 10          | 0         | [Delete] |

Gambar 54 Hasil Pengujian *Layer Record*



## **5.2. Analisis Hasil Pengujian**

Dari berbagai pengujian yang telah dilakukan, maka didapatkan hasil bahwa aplikasi dapat berfungsi dengan baik dan data yang didapat dari sistem telah sesuai. Data yang didapat dari sistem dapat diolah pada aplikasi dan sistem bekerja dengan benar sesuai dengan kondisi data yang ditetapkan.

## **Bab VI**

### **Kesimpulan dan Saran**

#### **6.1. Kesimpulan**

Setelah melakukan perancangan dan pembuatan sistem kemudian dilakukan pengujian dan analisisnya, maka dapat diambil beberapa kesimpulan mengenai *Smart House Electricity Monitoring based on Raspberry Pi* antara lain:

1. Setelah dilakukan beberapa pengujian terhadap sistem, pendeteksian arus menggunakan Sensor CT tidak dapat dilakukan melalui *Raspberry Pi* sehingga dilakukan perubahan dengan menambah perangkat yakni *Arduino*. Dengan *Arduino* arus yang dikonsumsi setiap perangkat listrik dapat dideteksi oleh Sensor CT dengan baik. Namun pada Sensor CT hanya memiliki kabel untuk tegangan (*volt*) dan GPIO dan tidak terdapat kabel untuk ground. Sehingga ditambahkan perangkat berupa LM358 yang digunakan sebagai ground pada Sensor CT.
2. Sistem dapat melakukan monitoring terhadap penggunaan daya listrik dalam rumah juga dapat mengontrol perangkat listrik seperti menghidupkan dan mematikan perangkat listrik menggunakan aplikasi *website* yang dapat diakses melalui laptop atau *smartphone*.
3. Sistem yang dibangun sudah sesuai dengan ruang lingkup yang dijelaskan sebelumnya.

#### **6.2. Saran**

Dalam pengerjaan tugas akhir ini, sistem tidak dapat melakukan pengukuran menggunakan sensor CT secara *realtime* jika ada penambahan perangkat listrik seperti lampu, ekstensi dan lain-lain. Sistem juga tidak dapat digunakan pada jangkauan yang terlalu luas. Aplikasi website Appshome hanya dapat digunakan jika menggunakan jaringan yang sama dengan sistem. Untuk pengembangan kedepannya, disarankan agar pengukuran daya pada perangkat yang baru ditambahkan dapat dilakukan secara *realtime* dan Aplikasi website Appshome dapat digunakan dalam jangkauan luas.

## Daftar Pustaka

- [1] Kumar, S. and Sharma, A. (2017). *An Efficient Home Automation Approach using Raspberry Pi in Wireless Sensors with Smart Phone. International Journal of Computer Applications*, [online] 173(9), pp.38-43.
- [2] Elib.unikom.ac.id. (2018). *Digital library - Perpustakaan Pusat Unikom - Knowledge Center - WELCOME | Powered by GDL4.2 | ELIB UNIKOM*. [online] Available at: <http://elib.unikom.ac.id/gdl.php?mod=browse&op=read&id=jbptunikompp-gdl-putrimeili-35884> [Accessed 23 Jan. 2018].
- [3] Imron, H., Isnanto, R. and Widiyanto, E. (2016). *Perancangan Sistem Kendali pada Alat Listrik Rumah Tangga Menggunakan Media Pesan Singkat (SMS). Jurnal Teknologi dan Sistem Komputer*, 4(3), p.454.
- [4] Id.wikipedia.org. (2018). *Arus listrik*. [online] Available at: [https://id.wikipedia.org/wiki/Arus\\_listrik](https://id.wikipedia.org/wiki/Arus_listrik) [Accessed 25 Jan. 2018].
- [5] Teknik Elektronika. (2018). *Pengertian Relay dan Fungsi Relay*. [online] Available at: <http://teknikelektronika.com/pengertian-relay-fungsi-relay/> [Accessed 25 Jan. 2018].
- [6] Immersa-lab.com. (2018). [online] Available at: [http://www.immersa-lab.com/wp-content/uploads/2018/03/Struktur-Sederhana-Relay\\_Featured-Image.jpg](http://www.immersa-lab.com/wp-content/uploads/2018/03/Struktur-Sederhana-Relay_Featured-Image.jpg) [Accessed 25 Jan 2018].
- [7] Muharram, A. (2018). *Trafo Arus (CT)*. [online] Bloglistrik.com. Available at: <http://www.bloglistrik.com/2016/08/trafo-arus-ct.html> [Accessed 26 Jan. 2018].
- [8] Materi Mafia Online. (2018). *Cara Menghitung Energi dan Daya Listrik*. [online] Available at: <https://mafia.mafiaol.com/2013/04/cara-menghitung-energi-dan-daya-listrik.html> [Accessed 5 Jun. 2018].

- [9] Id.wikipedia.org. (2018). *Raspberry Pi*. [online] Available at: [https://id.wikipedia.org/wiki/Raspberry\\_Pi](https://id.wikipedia.org/wiki/Raspberry_Pi) [Accessed 26 Jan. 2018].
- [10] N, S. (2018). *Mengetahui Pengertian Website Dan Jenisnya - Pengertian Apapun*. [online] Pengertian Apapun. Available at: <http://www.pengertianku.net/2014/09/mengetahui-pengertian-website-dan-jenisnya.html> [Accessed 26 Jan. 2018].
- [11] Id.wikipedia.org. (2018). *CodeIgniter*. [online] Available at: <https://id.wikipedia.org/wiki/CodeIgniter> [Accessed 30 Jan. 2018].
- [12] Pintarci.blogspot.co.id. (2018). *Pengenalan Struktur Codeigniter | Pintar Codeigniter*. [online] Available at: <http://pintarci.blogspot.co.id/2014/09/pengenalan-struktur-codeigniter.html> [Accessed 30 Jan. 2018].
- [13] Blog.langitrobotika.com. (2018). *Pengertian Arduino UNO | LangitRobotika.Com*. [online] Available at: <http://blog.langitrobotika.com/article/robotika/pengertian-arduino-uno/.html> [Accessed 3 Jun. 2018].
- [14] DIP-8, I. (2018). *IC LM358 Dual OpAmps (SMD/SOP-8 atau DIP-8)*. [online] Vcc2GND.com | Solusi Rekayasa Elektronika. Available at: [http://blog.vcc2gnd.com/2014/02/ic-lm358-dual-opamps-smdsop-8-atau-dip-8\\_10.html](http://blog.vcc2gnd.com/2014/02/ic-lm358-dual-opamps-smdsop-8-atau-dip-8_10.html) [Accessed 3 Jun. 2018].
- [15] DASAR KOMPUTER BUAT PEMULA. (2018). *Pengetahuan Dasar dan Pemrograman Raspberry Pi*. [online] Available at: <https://pccontrol.wordpress.com/2014/06/17/pengetahuan-dasar-dan-pemrograman-raspberry-pi/>. [Accessed 30 Jan 2018].
- [16] 14core.com. (2018). *Wiring 8 Channel Optocoupler Relay Module | 14core.com*. [online] Available at: <https://www.14core.com/wiring-8-channel-optocoupler-relay-module/> [Accessed 31 Jan 2018].

- [17] Learn.openenergymonitor.org. (2018). Learn | OpenEnergyMonitor. [online] Available at: <https://learn.openenergymonitor.org/electricity-monitoring/ct-sensors/introduction> [Accessed 31 Jan 2018].
- [18] Blog.langitrobotika.com. (2018). [online] Available at: <http://blog.langitrobotika.com/wp-content/uploads/2016/01/arduino-uno.jpg> [Accessed 3 Jun. 2018].
- [19] DIP-8), I. (2018). *IC LM358 Dual OpAmps (SMD/SOP-8 atau DIP-8)*. [online] Vcc2GND.com | Solusi Rekayasa Elektronika. Available at: [http://blog.vcc2gnd.com/2014/02/ic-lm358-dual-opamps-smdsop-8-atau-dip-8\\_10.html](http://blog.vcc2gnd.com/2014/02/ic-lm358-dual-opamps-smdsop-8-atau-dip-8_10.html) [Accessed 3 Jun. 2018].

# Lampiran

## 1. Kode Program

A. Kode program Sensor CT untuk mengambil data yang berupa daya listrik yang dikonsumsi perangkat.

```
#include <SPI.h>
#include <Ethernet.h>
#define NUM_CT 2

void setup() {
  Serial.begin(9600); // Serial Komunikasi Arduino
  analogReference(INTERNAL);
}

void loop() {
  int ct[] = {0,1};
  float d[NUM_CT];
  int n;
  for(n=0; n<NUM_CT; n++){
    //Serial.print(ct[n]);
    float Irms=get_corriente(ct[n]); // Menghitung arus masuk (A)
    float P=Irms*220.0; // P=IV (Watts) Menghitung daya masuk
    float J=P*24;
    float K=J/1000; // Menghitung Energi yang digunakan tiap jam
    d[n]=K;
  }

  int i;
  String strL;
  char tmp[10];
  strL = "n,";
  for(i=0;i<NUM_CT;i++){
    dtostrf(d[i],1,3, tmp);
```

```

    //str = str+", ";
    strL= strL + tmp;
    strL=strL+", ";
}
//dtostrf(b,1,2, &tmp[20]);
//Serial.print("\n");
Serial.print(strL);
Serial.print("\n");
    //Serial.print(K);
delay(1000);
}
float get_corriente(int pin)
{
    float voltajeSensor;
    float corriente=0;
    float Sumatoria=0;
    long tiempo=millis();
    int N=0;
    while(millis()-tiempo<500) //Melakukan perhitungan data yang masuk dari CT_Sensor
dan mendelay selama 0.5 second
    {
        voltajeSensor = analogRead(pin) * (1.1 / 1000000.0); //Menghitung jumlah tegangan yang
dihasilkan dan data analog yang didapat arduino dari CT_Sensor
        corriente=voltajeSensor*30.0;
        Sumatoria=Sumatoria+sq(corriente);
        N=N+1;
        delay(1);
    }
    Sumatoria=Sumatoria*2;
    corriente=sqrt((Sumatoria)/N);
    return(corriente);
}

```

B. Kode program untuk mengolah data yang didapat dari Sensor CT ke dalam database

```
import serial
import time
import datetime
import MySQLdb
import RPi.GPIO as GPIO
import time
import re
from threading import Lock, Thread

lock = Lock() # Fungsi Thread
# global data
data = "" # Mendeklarasikan data sebagai variabel global
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

# pin_relay = [14, 15, 17, 18, 27, 23, 24]
pin_relay = [14, 15] # Mendaftarkan pin GPIO yang digunakan oleh LED dan Ekstensi

GPIO.setup(pin_relay, GPIO.OUT)
#print "test1"
db = MySQLdb.connect(host="localhost", user="kelompok_13", passwd="semenholsim",
db="appshome") # Untuk koneksi ke database
curs= db.cursor()

curs= db.cursor()
#print "test2"
ser=serial.Serial('/dev/ttyACM0', 9600) # Komunikasi serial antara Arduino dan RPi
t=0
# num_ct=7
num_ct=2 # Banyak Pin GPIO yang digunakan
```



```

def threadProcess(id): # Proses Thread membagi setiap CT-Sensor untuk berjalan sendiri
sesuai dengan GPIO yang ditentukan
    while True:
        print "thread "+str(id)+" run"
        now = datetime.datetime.now().strftime("%Y-%m-%d")
        try:
            lock.acquire()
            sql = "SELECT tbl_record.power_used,tbl_items.power from
tbl_record INNER JOIN tbl_items ON tbl_record.itemId=tbl_items.itemId WHERE
tbl_record.itemId=%d AND tbl_record.record_date='%s'" % (id, now) # Proses relasi dan
INNER JOIN antara tbl_record dan tbl_items untuk mendapatkan nilai itemId, power,
power_used dan record_date
            curs.execute(sql)
            db.commit()
            res = curs.fetchall()
            print res[0][0]

            lock.release()
            if res[0][0] <= res[0][1]: # Melakukan pengecekan apakah daya yang
digunakan sudah melebihi batas yang ditentukan
                GPIO.output(pin_relay[id-1], GPIO.LOW) # Jika daya yang
digunakan belum melewati batas daya yang ditentukan maka Lampu/ Ekstensi akan tetap
menyala
                print ("LED ON")
                status=1
            else :
                GPIO.output(pin_relay[id-1], GPIO.HIGH)
                print ("LED OFF") # Jika daya yang digunakan sudah
melewati batas daya yang ditentukan maka Lampu/ Ekstensi akan mati
                status=0
                time.sleep(5)

```

```

        if status == 1: # Jika lampu menyala dan proses pembacaan daya
yang masuk masih berjalan

            time.sleep(0.1)
            lock.acquire()
            global data
            time.sleep(1)
            t = float(data[id])
            sql = "UPDATE tbl_record SET power_used =
power_used+%.f WHERE itemId=%d AND record_date='%s'" % (t,id,now) # Memperbarui
data power_used dengan menjumlahkan data yang sebelumnya dengan data yang masuk
melalui CT_Sensor

            curs.execute(sql)
            db.commit()
            lock.release()

    except IndexError:
        continue

sql = "SELECT itemId from tbl_items" # Memanggil data itemId dari tbl_items
curs.execute(sql)
db.commit()
res = curs.fetchall() # Menampung data itemId sementara
for r in res:
    t=t+1

itemId = [0]*t
t=0
for r in res:
    itemId[t]=r[0]
    t=t+1

```

```

threads = []

for i in range (num_ct):
    threads.append(Thread(target=threadProcess, args=(itemId[i], )))
lock.acquire()
for thread in threads:
    thread.start()
lock.release()
while True: # Menampung data daya dari CT_Sensor
    read_serial=ser.readline()
    # data=""
    global data
    data = re.split(',',read_serial)
    if data[0] != 'n' or len(data) != num_ct+2:
        continue

    print data

    #time.sleep(1)

```

C. Kode program untuk mengubah nilai dari tabel daya yang dikonsumsi per hari menjadi nol

```

import serial
import time
from datetime import datetime
import MySQLdb
import RPi.GPIO as GPIO
import time

db = MySQLdb.connect(host="localhost", user="kelompok_13", passwd="sementholism",
db="appshome") # Untuk koneksi ke database
curs= db.cursor()

```

```

t=0
sql = "SELECT itemId from tbl_items" # Memanggil data itemId dari tbl_items
curs.execute(sql)
db.commit()
res = curs.fetchall() # Menampung data itemId sementara
for r in res:
    t=t+1

itemId = [0]*t
t=0
for r in res:
    itemId[t]=r[0]
    t=t+1

now = datetime.now() # Mendeklarasikan type Date
formatted_date = now.strftime("%Y-%m-%d")
for i in range(len(itemId)):
    sql = "INSERT INTO tbl_record(itemId, record_date, power_used) VALUES
(%d,'%s', 0)" % (itemId[i], formatted_date) # Menginput data kedalam record_date dan
power_used
    curs.execute(sql)
    db.commit()

```

## 2. Database yang digunakan untuk membangun aplikasi website Appshome

