

HASIL PENELITIAN

MODEL SISTEM SURVEILAN OTOMATIS TERINTEGRASI DAN DINAMIS UNTUK OPERASIONAL PERANGKAT-PERANGKAT IOT KELISTRIKAN

***INTEGRATED AND DYNAMIC AUTOMATIC SURVEILLANCE
SYSTEM MODEL FOR OPERATIONS OF
ELECTRICAL IOT DEVICES***

MULIADI

NIM : D 053171001



**PROGRAM STUDI S3 TEKNIK ELEKTRO
DEPARTEMEN TEKNIK ELEKTRO
UNIVERSITAS HASANUDDIN
MAKASSAR
2022**

HALAMAN PENGESAHAN

MODEL SISTEM SURVEILAN OTOMATIS TERINTEGRASI DAN DINAMIS UNTUK OPERASIONAL PERANGKAT-PERANGKAT IOT KELISTRIKAN

Diajukan oleh:

MULIADI
NIM : D053171001

*Telah diperiksa dan dinyatakan memenuhi syarat untuk melaksanakan
seminar hasil penelitian*

Menyetujui,

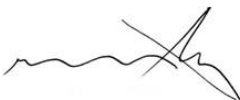
Tim Promotor



Prof. Dr. Ir. H. Andani Achmad, MT

Promotor

Tanggal : 27 Juli 2022



Dr. Elyas Palantei, ST., M.Eng

KoPromotor

Tanggal : 27 Juli 2022



Dr. Intan Sari Areni, ST., MT

KoPromotor

Tanggal : 27 Juli 2022

Ketua Pprogram Studi
S3 Teknik Elektro



Prof. Dr. Ir. H. Andani Achmad, MT

NIP. 196012311987031002

PRAKATA

Puji syukur penulis panjatkan atas kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya berupa nikmat kesehatan dan kekuatan sehingga penulis dapat menyelesaikan penelitian dan disertasi dengan judul “ Model Sistem Surveilans Otomatis Terintegrasi dan Dinamis untuk Operasional Perangkat-Perangkat IoT Kelistrikan “, sebagai salah satu syarat penyelesaian Pendidikan pada Program Doktor di Departemen Teknik elektro Fakultas Teknik Universitas Hasanuddin. Banyak kendala yang dihadapi penulis selama dalam penelitian maupun dalam penyusunan disertasi ini.

Alhamdulillah dengan izin Allah serta bantuan dari berbagai pihak, maka penelitian dan disertasi ini dapat diselesaikan dan dirampungkan. Dalam kesempatan ini penulis dengan tulus menyampaikan terima kasih kepada:

1. Prof. Dr. Ir. H. Andani Achmad, MT. selaku Promotor yang telah banyak meluangkan waktunya dalam membimbing dan mengarahkan penulis dari awal penelitian hingga selesainya disertasi ini. Terima kasih atas segala jasa dan keikhlasan yang tak ternilai, semoga Allah SWT memberikan imbalan pahala berupa amal jariyah.
2. Dr. Elyas Palantei, S.T., M.Eng dan Dr. Eng. Intan Sari Areni, ST.,MT. selaku Co-Promotor yang telah banyak meluangkan waktunya dalam membimbing dan mengarahkan penulis dalam menyelesaikan kendala yang ada dengan memberikan saran – saran, arahan, petunjuk dari awal penelitian hingga selesainya disertasi ini. Terima kasih atas segala jasa dan keikhlasan yang tak ternilai, semoga Allah SWT memberikan imbalan pahala berupa amal jariyah.
3. Dr. Ir. H. Rhiza S. Sadjad, MSEE, Dr., A. Ejah Umraeni, ST., M.T., Prof. Dr. Ing. Faisal Arya Samman, S.T., M.T., dan Dr. Indar Chaerah Gunadin, S.T., M.T selaku Tim Penguji. Terima kasih atas segala

saran dan masukan, serta kesempatan meluangkan waktu mendiskusikan segala hal yang menjadi keterbatasan kami selama penyusunan disertasi ini. Semoga Allah SWT membalas dengan keberkahan yang melimpah.

4. Bapak Rektor Universitas Hasanuddin, Dekan Fakultas Teknik beserta wakil dekan dan seluruh staf. Ketua program Studi S3 Teknik Elektro Prof. Dr. Ir. H. Andani Achmad, MT. dan seluruh staf. Terima kasih atas segala bantuan waktu dalam memberikan pelayanan terbaik.
5. Bapak Rektor Universitas Negeri Makassar (UNM) yang telah berkenan memberikan izin kepada penulis untuk melanjutkan studi. Dekan beserta wakil dekan, rekan sejawat dan Dosen dan staf Fakultas Teknik UNM atas doa dan dukungannya selama ini.
6. Para sahabat, teman seangkatan maupun sesama mahasiswa S3 Teknik Elektro UNHAS yang tidak sempat saya sebut satu persatu. Terima kasih atas supportnya selama ini, semoga kebersamaan kita tetap terjalin dan terjaga.

Ucapan terima kasih yang tak terhingga kepada orang tua tercinta Subereng yang telah mendidik dan membesarkan kami dan teriring doa kepada Ayahanda saya Alm Uba Dg Makkati serta mertua Alm. Andi Tappu dan Alm . Hj. St. Saodah semoga mendapat tempat yang layak disisiNya.

Akhir kata penulis sangat berharap semoga penelitian ini dapat berkontribusi pada perkembangan ilmu pengetahuan dan teknologi di bidang teknologi monitoring dan pengontrolan peralatan kelistrikan IoT

Makassar, Juli 2022

Muliadi

ABSTRAK

MULIADI, *Model Sistem Surveilan Otomatis Terintegrasi Dan Dinamis Untuk Operasional Perangkat-Perangkat Iot Kelistrikan*

(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Penelitian tentang Teknik Fault detection and diagnostics (FDD) dapat digunakan untuk memantau sistem kelistrikan bangunan serta mendeteksi dan mendiagnosa kerusakan peralatan kelistrikan. (Hannan et al., 2018). FDD adalah proses secara otomatis mengetahui dan mengisolasi kesalahan dalam bangunan untuk perlindungan sistem dari kerusakan. Beberapa aplikasi FDD dari bangunan dikembangkan dan diteliti berdasarkan hubungan antara suhu, tekanan, dan termodinamika untuk deteksi dan diagnosis kesalahan. Penelitian dalam disertasi ini menyajikan tentang Sistem Pemantauan dan Pengendalian Pengoperasian Perangkat Kelistrikan Berbasis Internet of Things (IoT) secara terpusat. Beberapa Perangkat yang dirancang dan dikembangkan dengan pemodelan sedemikian rupa agar memudahkan dalam pengoperasian, bersifat reconfigurable dan dimanis. Pemantauan, pengontrolan, pendeteksian kerusakan pada perangkat kelistrikan dapat diakses pada sebuah sistem yang terpusat dan terintegrasi. Pengujian prinsip kerja sistem sebanyak 10 kali pada setiap sistem dilakukan dan sistem dapat bekerja dengan baik sesuai dengan prinsip kerja. Pengujian respon time dilakukan sebanyak 25 kali pengiriman data control ke server dengan nilai rata-rata respon time secara keseluruhan sebesar 15 detik. Setelah melakukan pengujian Aplikasi android MACSys menggunakan Firebase Test Lab menggunakan device virtual dengan spesifikasi pixel 5e dan API level 30, orientasi portrait dan dinyatakan lulus/berhasil uji dan dapat berjalan dengan baik pada perangkat Samsung, vivo dan redmi.

Kata kunci: internet of things, Failure Detection system, fire base test lab, automation system, system

DAFTAR ISI

Halaman

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
PRAKATA	ii
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI	iv
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xii
DAFTAR SIMBOL DAN SINGKATAN	xiii
BAB 1 PENDAHULUAN UMUM	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Tujuan Penelitian	5
1.4. Kegunaan Penelitian	5
BAB 2 Pemantauan Konsumsi Energi Untuk Smart Home	6
2.1. Abstrak	6
2.2. Pendahuluan	7
2.3. Metode	9
2.4. Hasil dan Pembahasan	14
2.5. Kesimpulan	16
2.6. Daftar Pustaka	16
BAB 3 Pendeteksian Gangguan Dan Diagnosis Sistem IoT Kelistrikan	18
3.1. Abstrak	19
3.2. Pendahuluan	19

3.3.	Metode	23
3.4.	Hasil dan Pembahasan	32
3.5.	Kesimpulan	39
3.6.	Daftar Pustaka	39
BAB 4	Kendali Perangkat Kelistrikan Pintar	42
4.1.	Abstrak	42
4.2.	Pendahuluan	43
4.3.	Metode	44
4.4.	Hasil dan Pembahasan	49
4.5.	Kesimpulan	54
4.6.	Daftar Pustaka	55
BAB 5	Monitoring Konsumsi Dan Rugi Energi Berbasis IoT	58
5.1.	Abstrak	58
5.2.	Pendahuluan	58
5.3.	Metode	60
5.4.	Hasil dan Pembahasan	63
5.5.	Kesimpulan	66
5.6.	Daftar Pustaka	67
BAB 6	Sensor Jamak IoT untuk Smart Home	70
6.1.	Abstrak	70
6.2.	Pendahuluan	70
6.3.	Metode	73
6.4.	Hasil dan Pembahasan	78
6.5.	Kesimpulan	82
6.6.	Daftar Pustaka	83
BAB 7	Sistem Pemantauan Dan Pengendalian Terintegrasi Untuk Pengoperasian Perangkat Kelistrikan IoT	85
7.1.	Abstrak	85
7.2.	Pendahuluan	86
7.3.	Metode	89
7.4.	Hasil dan Pembahasan	93
7.5.	Kesimpulan	104
BAB 8	PEMBAHASAN UMUM	106

BAB 9	KESIMPULAN UMUM	114
DAFTAR PUSTAKA		118
LAMPIRAN		122

DAFTAR TABEL

Tabel 3.1	Hasil monitoring konsumsi arus penerangan (Lampu), penyejuk ruangan (AC), Arus Stok kontak dan arus panel menggunakan sensor arus ACS 712	33
Tabel 3.2	Hasil monitoring konsumsi arus penerangan (Lampu), penyejuk ruangan (AC), Arus Stok kontak dan arus panel menggunakan Clamp meter	34
Tabel 3.3	Hasil monitoring kondisi cahaya penerangan, suhu ruangan, dan suhu thermal kabel menggunakan sensor cahaya, DHT22 dan LM35	35
Tabel 4.1	Testing result of electrical devices control	52
Tabel 4.2	Testing Results of Sensor Reading.....	52
Tabel 6.1	Testing of principle work	80
Tabel 7.1	Hasil Pengujian balckbox untuk hardware	96
Tabel 7.2	Hasil Pengujian aplikasi android MACSys	98

DAFTAR GAMBAR

Gambar 3.1	Denah laboratorium telekomunikasi radio dan gelombang pendek	23
Gambar 3.2	Rangkaian power supply output +5V	24
Gambar 3.3	Pemasangan sensor arus ACS712, sensor cahaya, dan sensor suhu DHT22 pada modul wemos D1 board ESP8266	25
Gambar 3.4	Flowchart sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan	25
Gambar 3.5	Hasil pemasangan sensor arus, sensor cahaya sensor suhu dan sensor panas	32
Gambar 3.6	Pemasangan sistem pengontrolan untuk mengaktifkan penyejuk udara (AC), penerangan (Lampu), dan stok kontak menggunakan relay pada modul Wemos D1 board ESP8266	32
Gambar 3.7	Hasil perancangan web service firebase monitoring arus lampu, penyejuk ruangan, stok kontak dan deteksi cahaya lampu, suhu ruangan dan panas stok kontak	33
Gambar 3.8	Grafik konsumsi arus menggunakan sensor arus ACS712 (a) Lamp (b) Air Conditioner (AC) (c) Stok Kontak	35
Gambar 3.9	Grafik konsumsi arus menggunakan <i>clamp meter</i> (a) Lamp (b) Air Conditioner (AC) (c) Stok Kontak (<i>socket</i>).....	36
Gambar 3.10	Grafik kondisi suhu ruangan menggunakan sensor DHT22	37
Gambar 3.11	Grafik kondisi suhu thermal (panas) Kabel menggunakan sensor LM35	37
Gambar 4.1	Architecture System	46
Gambar 4.2	Hardware schematic	46
Gambar 4.3	Flowchart system	48

Gambar 4.4	System of prototype	50
Gambar 4.5	Sensor placements.....	50
Gambar 4.6	Application interface	51
Gambar 4.7	Grapph of DHT 22 and LM35 Sensor Accuracy	53
Gambar 4.8	Grapph of PZEM-004T Sensor Accuracy	53
Gambar 4.9	Grapph of respon time	54
Gambar 5.1	Architecture system	61
Gambar 5.2	Schematic hardware	61
Gambar 5.3	Flowchart of design system	62
Gambar 5.4	Prototype system	64
Gambar 5.5	(a) Dashboard dan (b) Overview Interface on the website application for monitoring consumption and lost electrical energy	65
Gambar 5.6	(a) Energy Consumption Graph using Three Measurement Device with average value is 16,11 kWh in seven days (b) Error Rate Percentage Graph with average value is 2,31%	65
Gambar 5.7	Lost Energy consumption graph using PZEM-004T modul sensor	66
Gambar 6.1	Architecture system	73
Gambar 6.2	Algorithm flow of smart lamp control	76
Gambar 6.3	Algorithm flow of smart AC control	77
Gambar 6.4	Algorithm flow of smart electrical socket control	78
Gambar 6.5	Prototype of air conditioner controller	79
Gambar 6.6	Prototype of light controller	79
Gambar 6.7	Prototype of electrical socket controller	79
Gambar 6.8	Result of response time	82

Gambar 7.1	Arsitektur Sistem	90
Gambar 7.2	Flowchart monitoring and Controlling Sistem (MACSys) Sistem	91
Gambar 7.3	Modul Air Conditioner Sub-System	93
Gambar 7.4	Modul Lamp Sub-System	94
Gambar 7.5	Modul Electrical Socket Sub-System	94
Gambar 7.6	Modul Panel Sub-System	95
Gambar 7.7	Respon time MACSys dalam kondisi normal	101
Gambar 7.8	Respon time MACSys dalam kondisi deteksi kesalahan	102
Gambar 7.9	Rata-rata Respon time MACSys dalam kondisi deteksi kesalahan	102
Gambar 7.10	Hasil Pengujian CPU	103
Gambar 7.11	Hasil Pengujian Memory	103
Gambar 7.12	Hasil Pengujian Network	103
Gambar 7.13	Hasil Pengujian Performance Samsung SM-105FN (API29)	103
Gambar 7.14	Hasil Pengujian Performance Vivo1610 (API 23) ...	103
Gambar 7.15	Hasil Pengujian Performance Redmi 6A (API 27)	104
Gambar 8.1	Penelitian terkait	106

BAB I

PENDAHULUAN UMUM

1.1. Latar belakang masalah

Pembangunan ekonomi berkelanjutan sangat dipengaruhi oleh konsep ketahanan energi (Harkouss et al., 2018). Badan Energi Internasional memperkirakan permintaan listrik global akan mengalami peningkatan secara signifikan pada tahun 2035 (Hannan et al., 2018).

Peningkatan efisiensi energi dapat mengurangi permintaan kelistrikan Negara, sehingga memberikan penghematan moneter, gas rumah kaca dan pengurangan polutan lainnya, dan peningkatan keamanan energi. Penggunaan energi secara efisien dapat menurunkan permintaan listrik dan mengurangi resiko kekurangan listrik yang berdampak pada pertumbuhan ekonomi. Permintaan energi puncak di Indonesia diperkirakan mencapai tiga kali lipat antara tahun 2020 dan 2030, untuk konsumsi pengkondisian udara (Air Conditioner) (McAllister et al., 2017)

PT. PLN (Persero) merupakan satu-satunya Badan Usaha Milik Negara yang bergerak di bidang penyediaan sampai penyaluran jasa tenaga listrik. Seiring berkembangnya teknologi, industri dan meningkatnya kebutuhan akan energi listrik, maka dibutuhkan pasokan energi listrik dan penyaluran yang andal. Keandalan akan pasokan energi listrik adalah merupakan kepuasan pelanggan. PT. PLN (Persero) menjaga keandalan sistem penyaluran tenaga listrik menggunakan sistem pengoperasian yang mempunyai tingkat keandalan tinggi, dikarenakan PT. PLN (Persero) mempunyai visi diakui sebagai perusahaan kelas dunia yang dinilai dari SAIDI (*System Average Interruption Duration Index*) dan SAIFI (*System Average Interruption Frequency Index*). Untuk itu diperlukan sistem operasi yang mempunyai keandalan baik, sehingga dapat meminimalisir pemadaman untuk menjaga kepuasan pelanggan (Soleh, 2014).

Bangunan modern terdiri dari berbagai peralatan seperti pemanas, ventilasi dan pendingin udara (HAVAC) dan penerangan. Setiap peralatan terdiri dari beberapa komponen seperti pompa, kipas, saluran, sensor, lampu, kabel dan lain-lain. Semua peralatan dapat dipantau dan dikelola oleh system manajemen gedung (BMS). Semua komponen ini dapat mengalami kerusakan yang dapat disebabkan karena aus karena pemakaian dalam waktu yang lama, kerusakan konfigurasi, dan masalah komunikasi. Kerusakan peralatan memengaruhi biaya perawatan dan khususnya efisiensi energi (Mills, 2011)(Kim et al., 2018)

Beban peralatan seperti *Heating, Ventilation, and Air-Conditioning* (HVAC) bervariasi tergantung pada jenis dan lokasi bangunan, tetapi kerusakan pada salah satu peralatan dapat membuat hingga 50% dari total konsumsi energi sehingga dapat menyebabkan kehilangan energi yang besar (Kim et al., 2018)(Pe, 2008). Penelitian menunjukkan bahwa penghematan energi antara 20% hingga 30% dapat dicapai dengan cara *re-commissioning malfungsi* sistem HVAC (Agency, 2005). Sistem HVAC sering disesuaikan untuk bangunan spesifik sehingga kurangnya integrasi sistem yang berkualitas (Yu, Woradechjumroen and Yu, 2014).

Sektor gedung saat ini merupakan konsumen energi terbesar di dunia (Jain et al., 2016). Sehingga, manajemen energi dalam gedung telah menjadi tujuan internasional teknologi modern (Li et al., 2013). Penelitian menunjukkan bahwa sekitar 40% dari total energi di dunia dikonsumsi oleh gedung, yang merupakan sepertiga dari emisi *Green House Gas* (GHG) (Huang et al., 2017) (Mytafides et al., 2017). Penelitian lain menunjukkan bahwa 49% dari total energi dikonsumsi oleh gedung pada tahun 2014 (Wu et al., 2020) dan 60% dari konsumsi energi gedung karena tujuan pemanasan dan pendinginan (Lizana et al., 2017). Karena itu, konsumsi energi dan pengaruhnya terhadap perubahan iklim paling banyak menjadi masalah yang perlu diselesaikan di sektor pemanfaatan energi pada gedung (Santamouris, 2016) (Sun et al., 2018) (Cao et al., 2016). *Building Energy Manajemen System* (BEMS) dengan menggunakan *Internet of Energy* (IoE)

dapat memaksimalkan efisiensi energi dengan meminimalkan kerugian dan dampak lingkungan (Hannan et al., 2018).

Penggunaan peralatan kelistrikan pada gedung umumnya belum memiliki sistem manajemen energi yang pemakaiannya dapat dipantau, begitupun pengendalian peralatan listrik juga masih bersifat konvensional. Kekurangan sistem yang ada saat ini adalah terkadang peralatan-peralatan listrik lupa dinonaktifkan padahal sudah tidak digunakan lagi.

Sensor array dengan keuntungan secara simultan mendeteksi beberapa parameter yang berbeda digunakan secara luas dalam banyak aplikasi teknik seperti teknik lingkungan, teknik kimia, teknik biologi, dan teknik aerospace (Chen et al., 2016) Sebagai sumber perolehan informasi, keakuratan dan keandalan output array sensor sangat penting untuk kinerja sistem pengukuran dan kontrol.

Teknik *Fault detection and diagnostics* (FDD) dapat digunakan untuk memantau sistem kelistrikan bangunan serta mendeteksi dan mendiagnosa kerusakan peralatan kelistrikan. (Hannan et al., 2018) FDD adalah proses secara otomatis mengetahui dan mengisolasi kesalahan dalam bangunan untuk perlindungan sistem dari kerusakan. Beberapa aplikasi FDD dari bangunan dikembangkan dan diteliti berdasarkan hubungan antara suhu, tekanan, dan termodinamika untuk deteksi dan diagnosis kesalahan

Sistem otomasi dengan menerapkan integrasi antara modul dengan lingkungan yang berbeda merupakan sarana yang efisien untuk menyediakan komunikasi otomatis diantara perangkat (Guilherme Mussi, Leonardo Barreto Campos, Carlos Eduardo Cugnasca, 2017)

Hasil verifikasi model menunjukkan bahwa model manufacture pintas dari industry konstruksi dapat mewujudkan integrasi masyarakat manusia dan sistem fisik dan mencapai tujuan manajemen waktu nyata dan pengendalian personal, mesin, peralatan dan infrastruktur dalam seluruh jaringan (Lingji Kong dan Biao Ma, 2019)

Tren dalam elektronik terintegrasi adalah rasio kinerja terhadap biaya yang lebih baik, radio berdaya rendah dan sensor mikroelektromekanis

dengan pemroses sinyal, komunikasi nirkabel, sumber daya dan sinkronisasi, semua dikemas dalam perangkat miniature murah (J. Agree, L. Clare, 2000)

Pada suatu sistem tenaga listrik dibutuhkan suatu sistem pengamanan yang handal yang mampu mendeteksi gangguan dan kesalahan yang terjadi pada sistem tersebut. Hal tersebut diperlukan agar suplai energi listrik dapat dipertahankan dengan baik. Agar sistem yang dimiliki mampu menjaga pelayanan suplai tenaga listrik dengan baik, maka sistem pengamanan harus memenuhi syarat *reliable*, selektif, sensitif, memiliki waktu operasi yang cepat, ekonomis dan sederhana.

Sistem yang dibangun dapat memonitoring energi yang digunakan oleh peralatan kelistrikan seperti AC, peralatan pencahayaan, dan peralatan kelistrikan yang terkoneksi dengan stok kontak yang ada seperti computer PC atau adaptor Laptop. Pengontrolan juga dengan mudah dapat dilakukan karena sistem terkoneksi dengan jaringan internet atau terintegrasi dengan Internet of Things (IoT). Pengontrolan dilakukan menggunakan *smartphone* dengan aplikasi android. Sistem ini juga dilengkapi dengan peringatan dini terhadap gangguan yang terjadi sehingga teknisi dengan cepat dapat mengetahui jika ada gangguan yang terjadi pada sistem kelistrikan pada suatu gedung.

Penelitian dalam disertasi ini menyajikan tentang Sistem Pemantauan dan Pengendalian Pengoperasian Perangkat Kelistrikan Berbasis Internet of Things (IoT)

1.2. Rumusan Masalah

Adapun rumusan masalah dari penelitian ini adalah:

- 1.2.1. Bagaimana rancangan sistem pemantauan perangkat kelistrikan pada instalasi pemanfaatan gedung?
- 1.2.2. Bagaimana rancangan sistem pengendalian pemakaian peralatan dan pembebanan kelistrikan gedung berbasis *Internet of things* (IoT)?
- 1.2.3. Bagaimana rancangan sistem pemantauan dan pengendalian kerusakan peralatan kelistrikan?

- 1.2.4. Bagaimana rancangan sistem pemantauan, pendeteksi dan pengendalian kerusakan secara otomatis?

1.3. Tujuan Penelitian

- 1.3.1. Menghasilkan sistem pemantauan perangkat kelistrikan pada instalasi pemanfaatan gedung.
- 1.3.2. Menghasilkan sistem pengendalian pemakaian peralatan dan pembebanan kelistrikan gedung berbasis *Internet of things* (IoT).
- 1.3.3. Menghasilkan sistem pemantauan dan pengendalian kerusakan peralatan kelistrikan
- 1.3.4. Menghasilkan sistem pemantauan dan pengendalian kerusakan perangkat kelistrikan secara otomatis

1.4. Kegunaan Penelitian

Hasil penelitian ini diharapkan dapat bermanfaat untuk hal-hal sebagai berikut:

- 1.4.1. Pemantauan terhadap Jaringan Distribusi Tenaga Listrik menjadi lebih baik, dalam keadaan normal, gangguan dan emergency dapat dilakukan secara cepat atau *real-time*.
- 1.4.2. Dengan adanya sistem *fault detection* dan diagnosis yang diterapkan pada peralatan kelistrikan, tercapainya pemulihan gangguan yang lebih baik, dengan pengawasan dan penanganan gangguan yang dilakukan dengan cepat dan tanpa ke lapangan.

BAB 2

PEMANTAUAN KONSUMSI ENERGI UNTUK SMART HOME

2.1. Abstract

MULIADI, *Pemantauan Konsumsi Energi Smart Home* (dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Penggunaan teknologi IoT memungkinkan pemantauan berbasis web untuk mengumpulkan informasi statistik terkait pengukuran parameter energi listrik, seperti pengukuran arus (I), tegangan (V), dan daya (P) secara real time. Penelitian ini bertujuan untuk merancang sebuah sistem monitoring konsumsi energi listrik secara real time berbasis web pada suatu rumah tertentu. Node periferan pemantauan dibangun menggunakan chipset pengontrol ESP8266 untuk mengarahkan relai dan sensor arus PZEM-004T. Pengontrol diprogram menggunakan aplikasi Arduino IDE dan memanfaatkan kinerja bahasa C yang kuat untuk membangun kode. Antarmuka aplikasi berbasis web juga dibuat menggunakan editor teks 3-teks luhur menggunakan HTML; Bahasa pemrograman CSS sebagai frontend; dan perangkat lunak PHP dan MySQL sebagai backend. Pengguna dapat mengontrol fungsi perangkat elektronik tertentu yang dipasang menggunakan tombol ON / OFF secara langsung melalui situs web yang dirancang. Melalui cara ini, pengguna juga dapat mengamati nilai arus dan daya yang dikonsumsi oleh setiap perangkat yang dioperasikan. Profil konsumsi daya kumulatif untuk interval waktu tertentu untuk setiap peralatan elektronik juga divisualisasikan secara grafis pada menu situs web. Hasil pengujian praktikum didapatkan persentase error pembacaan arus dari sensor PZEM-004T terhadap nilai arus teoritis. Dari pendataan beban listrik, persentase error beban lampu 0,5%, beban rice cooker 0,7% dan beban dispenser 0,1%.

Kata kunci: Smart Home, IoT Network, Green Technology, Graphical Visualization, and Web-Server Application

2.2. Pendahuluan

Sistem pemantauan adalah proses rutin pengumpulan data dan pengukuran kemajuan objek yang dipantau yang berfokus pada proses dan keluaran. Pemantauan akan memberikan informasi tentang status dan kecenderungan pengukuran dan evaluasi asalkan terjadi terus menerus yang umumnya dilakukan untuk tujuan tertentu, seperti untuk memeriksa proses suatu objek atau untuk mengevaluasi kondisi tertentu [1] [2]. Sistem monitoring energi diharapkan dapat menghemat energi sehingga terjadi penghematan biaya, peningkatan daya saing dan perbaikan kualitas lingkungan. Pemantauan dan audit energi merupakan pilihan alternatif dalam efisiensi energi [3][4].

Saat ini, Internet of Things (IoT) menjadi teknologi pendukung yang populer untuk sistem pemantauan karena semua perangkat terhubung ke internet sehingga pemantauan dapat dilakukan tanpa batasan waktu dan tempat. Teknologi ini memungkinkan pengembangan perangkat pintar, seperti laboratorium pintar, rumah pintar, dan kota pintar. Salah satu komponen IoT adalah sensor yang terhubung dengan pengontrol atau prosesor sehingga parameter yang dipantau dan dianalisis dapat dilihat secara real-time [6]. Penggunaan IoT yang mampu berkomunikasi dengan perangkat sistem dan menggunakan data perangkat untuk kontrol dan pemantauan [5]

Pemantauan hemat energi adalah sistem pemantauan yang menggunakan IoT atau IoT-enabled yang dapat memantau konsumsi listrik dengan sistem untuk mengatur penggunaan peralatan listrik yang digunakan [11]. Pemantauan penggunaan peralatan listrik rumah tangga untuk memberikan informasi kepada konsumen tentang konsumsi listrik sehingga dapat dikelola dengan baik [12]. Sistem monitoring online yang terintegrasi dengan Power Quality (PQ) dan menggunakan instrumen virtual serta dapat diakses secara luas dan memanfaatkan IoT [13].

Penelitian tentang monitoring berbasis IoT telah dilakukan oleh beberapa peneliti. Pada tahun 2017, A. R. Al-Ali dkk menggunakan IoT untuk

Sistem Manajemen Energi Rumah Pintar. Hasil penelitian menunjukkan bahwa pengguna dapat memantau dan mengontrol perangkat dari jarak jauh dan melakukan tagihan secara online menggunakan perangkat seluler [4]. Pada tahun 2018, Dikpride Despa dkk menghasilkan prototipe KWh meter digital yang digunakan untuk mengetahui besarnya konsumsi energi di laboratorium berbasis teknologi Internet of Things (IoT) [9]. Pada tahun yang sama, Komkrit Chooruang dan Kraison Meekul mengembangkan sistem pemantauan tegangan, arus, daya aktif, dan konsumsi daya secara keseluruhan. [10].

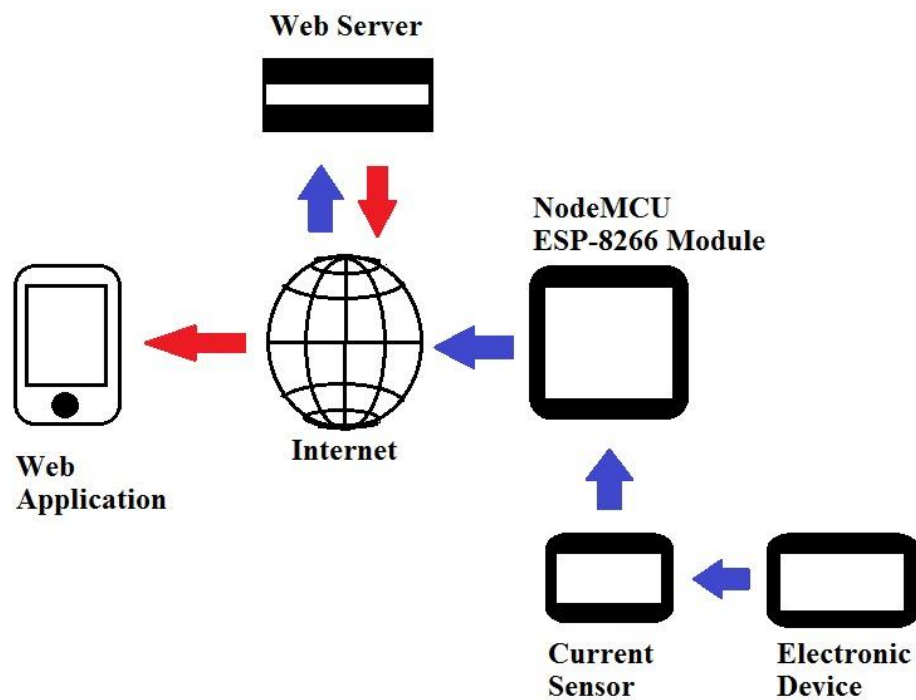
Ali A. S. Ramschie dkk pada tahun 2019 merancang sistem monitoring untuk monitoring konsumsi listrik untuk peralatan Air Conditioning (AC) dimana sistem tersebut dapat memonitor dan menginformasikan besarnya konsumsi energi listrik dan harga yang dibayarkan untuk pengoperasian peralatan AC melalui smartphone atau melalui web pelayan[1]. Sementara itu, A. D. Pangestu dkk pada tahun 2019 membangun sistem monitoring yang berfokus pada pembacaan arus dan daya terhadap beban induktif dan beban resistif dengan Arduino Nodemcu Esp8266. Tingkat akurasi yang diperoleh berada pada kisaran 96% hingga 99% [2].

Pada tahun 2020, penelitian tentang sistem pemantauan konsumsi daya listrik dilakukan oleh Sanjeev Joshi dan Dr. Kiran V berfokus pada pemantauan parameter daya seperti tegangan, arus, frekuensi, faktor daya, daya aktif, daya reaktif, dan daya semu dan dapat juga memprediksi kekuatan alat dan mengidentifikasi jenis perangkat yang terhubung [6].

Pada penelitian ini, sistem monitoring difokuskan pada konsumsi peralatan rumah tangga berbasis IoT saat ini dengan peralatan yang diamati yaitu bola lampu, rice cooker dan dispenser. Proses monitoring dilakukan dengan web dan smartphone. Sensor yang digunakan adalah sensor arus PZEM-004T karena memiliki akurasi pembacaan sensor yang tinggi. Selain itu juga digunakan NodeMCU yang merupakan platform IoT open source dengan Arduino IDE. Kit pengembangan ini didasarkan pada modul ESP8266 yang mengintegrasikan GPIO dan PWM (Pulse Width Modulation). NodeMCU ESP8266 digunakan sebagai pengolah data pada sistem

monitoring ini. Dengan sistem pemantauan ini, pengguna dapat memantau konsumsi daya listrik peralatan rumah tangga dengan lebih mudah.

2.3. Metode



Gambar 2.1. Schematic proposed system

Skema rancangan teknologi yang diusulkan ditunjukkan pada Gambar 2.1. dimana dalam sistem menggunakan beberapa perangkat yaitu mikrokontroler nodemcu, sensor PZEM-004T, aplikasi web berupa personal computer atau smartphone, dan perangkat elektronik berupa beban listrik yang akan dipantau konsumsi daya. sensor membaca daya listrik yang digunakan oleh peralatan perangkat (beban), nilai yang terbaca pada sensor dikirim ke mikrokontroler. Setelah data dikirim ke mikrokontroler, diproses dan dikirim ke webserver menggunakan koneksi internet. Setelah data diterima oleh webserver, disimpan dalam database yang dapat diakses atau dipantau melalui perangkat aplikasi web.

Setiap komponen menggunakan tegangan suplai 5V dari adaptor yang terhubung ke port micro usb pada modul ESP8266. Rangkaian suplai tegangan ditunjukkan pada Gambar 2.2. terdiri dari transformator, dioda, kondensator elektrolit dan IC7805. Hook up dari sistem monitoring konsumsi arus ditunjukkan pada Gambar 2.3. pada sistem ini digunakan nodemcu sebagai mikrokontroler yang dilengkapi dengan wifi untuk koneksi dengan internet, yang nantinya akan dihubungkan juga dengan web server. Sedangkan sensor yang digunakan untuk memantau konsumsi arus adalah sensor PZEM-004T yang memiliki beberapa parameter pengukuran yaitu arus, tegangan, daya, faktor daya, dan frekuensi. Sensor PZEM-004T dilengkapi dengan trafo arus yang berfungsi untuk mengubah efek medan magnet yang terdeteksi pada sistem beban listrik (beban), menjadi sinyal listrik yang akan diproses oleh sensor PZEM-004 T menjadi arus dan daya digunakan dalam beban listrik. Keluaran dari sensor PZEM-004T merupakan sinyal yang diumpankan ke mikrokontroler melalui komunikasi RX-TX antara sensor dengan mikrokontroler nodemcu.

Karakteristik Modul PZEM-004T adalah

- a. Mengukur konsumsi listrik.
- b. Antarmuka serial UART dengan kecepatan 9600 bps.
- c. Tegangan suplai 5V.
- d. Kemungkinan menghubungkan layar LCD atau LED.

Spesifikasi Parameter Modul PZEM-004T menurut datasheet adalah sebagai berikut.

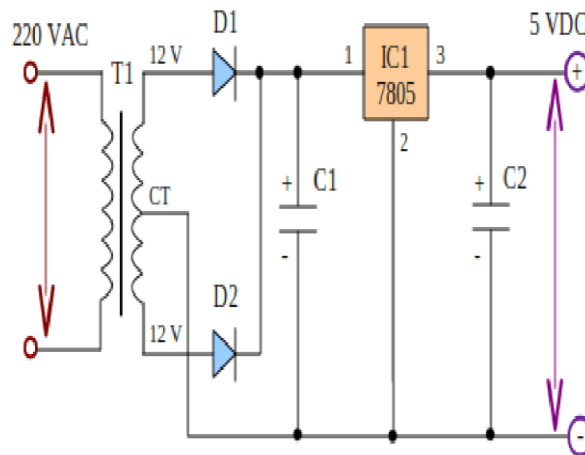
- a. Tegangan kerja: 80-260VAC
- b. Tegangan uji: 80-260VAC
- c. Nilai daya: 100A / 22000W
- d. Frekuensi operasi: 45 - 65Hz
- e. Akurasi pengukuran: 1.0

Perangkat lunak terdiri dari modul ESP 8266 NodeMCU menggunakan aplikasi Arduino IDE dan perangkat lunak web server

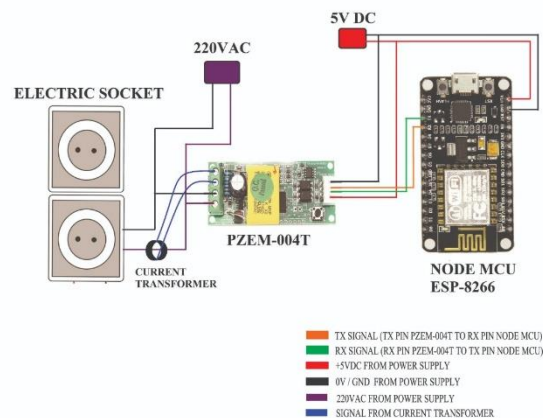
menggunakan bahasa pemrograman PHP dan aplikasi sublime sebagai editor. Tahapan sistem pemantauan untuk pemantauan konsumsi listrik dengan beban peralatan rumah tangga ditunjukkan pada Gambar 2.4.

Alur kerja gambar 2.4. adalah pengaturan sistem yang disebut "pengaturan sistem", dalam sistem pengaturan ini terdapat pengaturan koneksi yaitu koneksi mikrokontroler dengan wifi dan koneksi mikrokontroler dengan alamat web server yang digunakan. Selain itu juga terdapat deklarasi variabel dan inisialisasi port mikrokontroler. Deklarasi variabel meliputi deklarasi variabel tegangan, arus, dan daya yang merupakan hasil pembacaan sensor PZEM-004T. Sedangkan inisialisasi port mikrokontroler meliputi penamaan/inisialisasi pin mikrokontroler sebagai RX dan TX yang merupakan jalur komunikasi antara mikrokontroler Node MCU dengan sensor PZEM-004T. Setelah itu mikrokontroler memindai sensor PZEM-004T, jika koneksi ke sensor berhasil maka data dari sensor akan disimpan di memori mikrokontroler. Setelah itu, mikrokontroler akan terhubung ke wifi dan web server yang telah diatur pada tahap setup system. Jika Anda belum terhubung ke web server, mikrokontroler akan terus memindai hingga akhirnya terhubung ke web server. Setelah terhubung ke web server, mikrokontroler kemudian mengirimkan data ke webserver, dan webserver menerima data dan menyimpannya dalam database. Esp terputus dari server web.

Dalam perancangan program WebServer digunakan beberapa aplikasi pendukung antara lain aplikasi Sublime Text sebagai Text Editor dan aplikasi Xampp untuk koneksi ke Database Server. Sublime Text adalah aplikasi Text Editor yang digunakan untuk desain web dan template pada tampilan website.



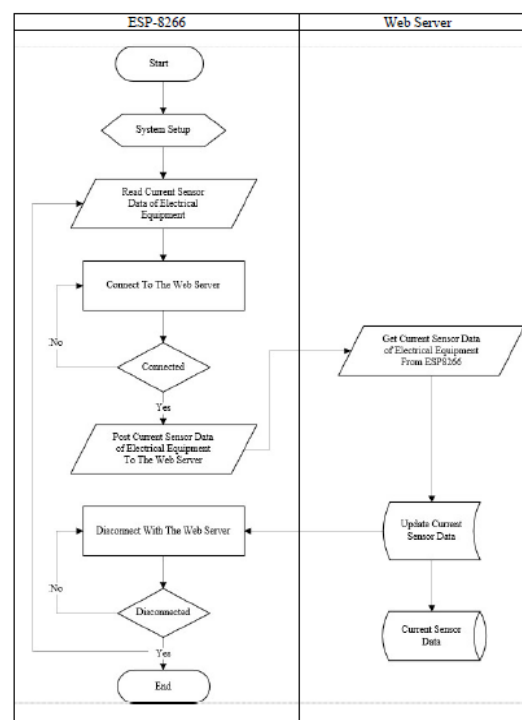
Gambar 2.2. Power supply circuit



Gambar 2.3. Hook up of current consumption monitoring system

Software desain Sublime Text meliputi: connection.php, index.php, reset.php, ACS712.cpp, dan get10lastdata.php. Layer connection.php ini berfungsi untuk menghubungkan WebServer dengan hardware NodeMCU ESP8266. Setiap data yang dibaca pada sensor arus PZEM-004T akan ditampilkan dalam bentuk grafik di situs web. Pada index.php, layer ini berfungsi untuk mendesain tampilan back-end dan front-end website. Reset.php berfungsi sebagai tombol reset untuk mengkalibrasi ulang data yang ditampilkan pada website. Sedangkan get10lastdata.php adalah fungsi untuk menampilkan 10 data terakhir dari pembacaan sensor arus yang ditampilkan dalam bentuk grafik pada website.

Program aplikasi XAMPP berfungsi sebagai server lokal untuk menangani berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP dapat digunakan untuk menguji kinerja fitur atau menampilkan konten pada website tanpa harus terkoneksi dengan internet, atau istilahnya website offline. XAMPP bekerja secara offline seperti web hosting.



Gambar 2.4. Flowchart system

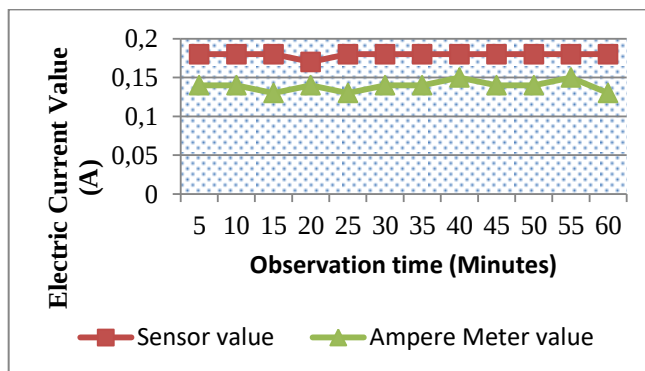
Pada penelitian ini pengujian sistem dilakukan dalam tiga skenario yaitu pengambilan data arus dan daya dengan beban lampu, ricecooker dan dispenser. Langkah-langkah sistem pengujian dijelaskan sebagai berikut:

- Menempatkan setiap komponen pada posisinya masing-masing.,
- Menghubungkan komponen pengontrol (ESP 8266) ke sumber DC 5 Volt.
- Menghubungkan beban dengan sumber tegangan
- Menghubungkan Laptop dan Modul ESP 8266 ke jaringan internet
- Mengakses browser web
- Menekan tombol untuk menampilkan grafik di web

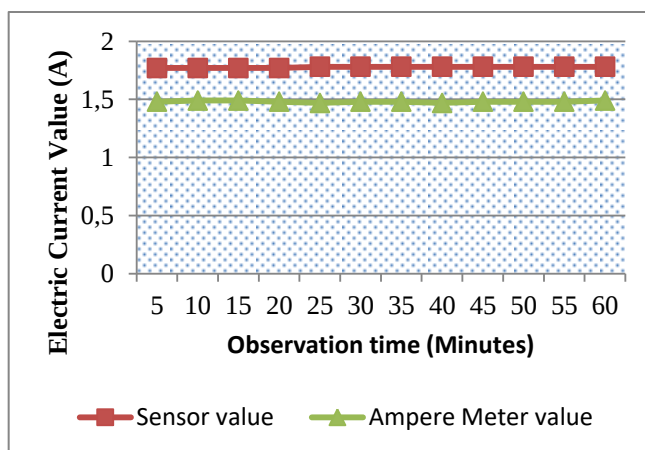
2.3. Hasil dan Pembahasan

Hasil dari sistem monitoring konsumsi arus listrik yang diusulkan ditunjukkan dari Gambar 2.5. sampai Gambar 2.7. Beban bola lampu, rice cooker, dan dispenser yang digunakan masing-masing adalah 40 watt, 350 watt, dan 745 watt dengan tegangan referensi sebesar 220 volt. Konsumsi daya dari ketiga beban ditunjukkan pada Gambar 2.8.

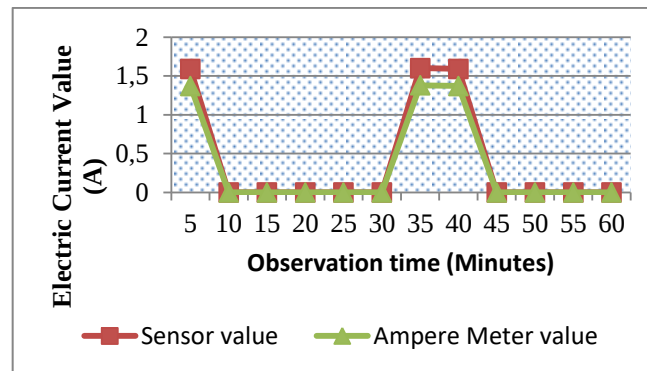
Gambar 2.8. menunjukkan hasil konsumsi daya lampu peralatan listrik, penanak nasi dan dispenser. Hasilnya adalah jumlah konsumsi daya dari semua beban yang terpasang pada saat itu.



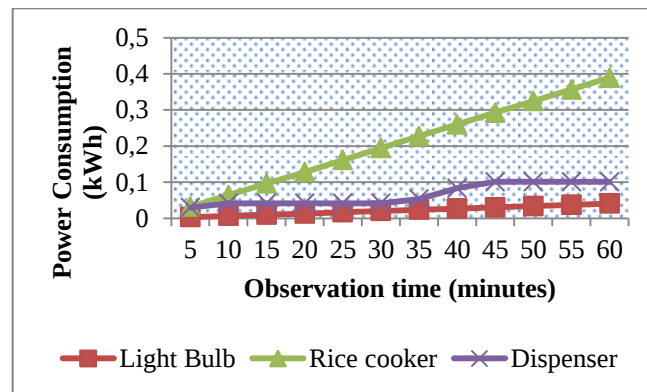
Gambar 2.5. The result of light bulb load measurement



Gambar 2.6. The result of rice cooker load measurement



Gambar 2.7. The result of dispenser load measurement



Gambar 2.8. The result of power consumption

Berdasarkan Gambar 2.5., rata-rata persentase error yang diperoleh dari pembacaan sensor PZEM-004T pada beban bola lampu adalah 0,5%. Dalam mengukur beban Rice Cooker dilakukan dengan cara memanaskan air yang dimasukkan ke dalam wadah yang ada di dalam Rice Cooker. Kesalahan pengukuran arus pada beban rice cooker adalah 0,7%. Sedangkan pada Gambar 2.7. dapat dilihat bahwa dari menit ke 10 sampai menit ke 30 nilai arusnya terbaca 0 A dan pada menit ke 45 sampai 60 nilai arusnya juga terbaca 0 A. Hal ini dikarenakan pada menit tersebut dispenser berhenti memanaskan, sehingga tidak ada beban listrik yang digunakan. Persentase kesalahan dalam pengukuran ini adalah 0,1%.

2.4. Kesimpulan

Sistem monitoring konsumsi arus listrik berbasis IoT sebagai bentuk pemanfaatan green technology telah dilakukan pada penelitian ini. Ada 3 alat listrik yang digunakan sebagai beban yaitu lampu bohlam, rice cooker dan dispenser. Pengamatan dilakukan selama 60 menit dengan pengiriman data setiap 5 menit. Hasil yang diperoleh menunjukkan bahwa sistem berhasil memonitor arus dan konsumsi daya secara real-time melalui web server menggunakan komputer atau smartphone. Untuk pengembangan ke depan, akan dipelajari optimasi sistem monitoring.

2.5. Daftar Pustaka

- [1] A. Ramschie, J. Makal, V. Ponggawa, and K. Kunci, "IoT Based Electric Power Consumption Monitoring System for Air Conditioner Equipment," pp. 14–19. [in Indonesian]
- [2] A. D. Pangestu et al., "ELECTRICAL LOAD MONITORING SYSTEM BASED ON ARDUINO NODEMCU ESP8266," vol. 4, no. 1, pp. 187–197, 2019. [in Indonesian]
- [3] U. Mudjiono et al., "Energy Monitoring and Audit for Efficient Electrical Energy Consumption at UPT. PP Politeknik Perkapalan Negeri," pp. 187–194, 2019. [in Indonesian]
- [4] I. A. Zuolkernan, M. Rashid, R. Gupta, and M. Alikarar, "A Smart Home Energy Management System Using IoT and Big Data Analytics Approach," vol. 63, no. 4, pp. 426–434, 2017.
- [5] V. D. S. Chacko, "IoT based Online Power Consumption Monitoring of a Distribution transformer feeding Domestic/Commercial Consumer loads," in *2019 4th International Conference on Information Systems and Computer Networks, ISCON 2019*, 2019.
- [6] S. Joshi and V. Kiran, "Design and Development of Power Monitoring System using IoT Technology," *International Journal of Advanced Science and Technology* Vol. 29, No. 7, (2020), pp. 8128-8138
- [7] A. Ramschie, "Modeling of Energy-Saving Mode Systems on Air Conditioning Equipment," *2018 Int. Conf. Appl. Sci. Technol.*, pp. 492–497, 2018.
- [8] F. Integrated, H. E. Linear, and C. Sensor, "ACS712," pp. 1–12.
- [9] D. Despa, "Monitoring and Management of Electrical Energy in Laboratory Building Based on Internet of Things (IoT)," pp. 2–6. [in Indonesian]

- [10] K. Chooruang, "Design of an IoT Energy Monitoring System," 2018 16th International Conference on ICT and Knowledge Engineering (ICT&KE) 16th Int. Conf. ICT Knowl. Eng., pp. 1–4, 2018.
- [11] H. K. H. B. A. A. S. Kiran, "Towards Efficient Energy Monitoring Using IoT," in *Proceedings of the 21st International Multi Topic Conference, INMIC 2018*, 2018
- [12] D. F. F. A. L. M. P. S. S. Houdi, "Design of an electricity consumption measurement system for Non Intrusive Load Monitoring," *2019 10th International Renewable Energy Congress, IREC 2019*, pp. 1-6, 2019.
- [13] N. N. A. D. A. I. Isdawimah, " Real-Time Monitoring of Power Quality for Web Based Electrical Power Panel Using LabVIEW," in *ICECOS 2019 - 3rd International Conference on Electrical Engineering and Computer Science, Proceeding*, 2019.

BAB 3

PENDETEKSIAN GANGGUAN DAN DIAGNOSIS SISTEM IoT KELISTRIKAN

3.1. Abstrak

MULIADI, *Peneteksian Gangguan Dan Diagnosis Sistem IoT Kelistrikan*
(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Bangunan modern terdiri dari berbagai peralatan seperti pemanas, ventilasi, pendingin udara (HVAC) dan penerangan. Setiap peralatan terdiri dari beberapa komponen seperti pompa, kipas, saluran, sensor, lampu, kabel dll. Semua peralatan dapat dipantau dan dikelola oleh sistem manajemen gedung. Semua komponen ini dapat mengalami kerusakan yang dapat disebabkan aus karena pemakaian dalam waktu yang lama, kesalahan konfigurasi, dan masalah komunikasi. Kerusakan peralatan mempengaruhi biaya perawatan dan khususnya efisiensi energi. Penelitian ini bertujuan untuk menghasilkan sistem monitoring konsumsi arus penerangan (lampu) dengan deteksi cahaya penerangan, konsumsi arus penyejuk udara (AC) dengan deteksi suhu ruangan dan konsumsi arus stok kontak dengan deteksi panas kabel. Pendeteksian arus, cahaya lampu, suhu ruangan dan panas kabel dilakukan untuk mengetahui kesalahan yang terjadi pada peralatan listrik. Monitoring konsumsi arus penerangan dengan deteksi cahaya lampu, konsumsi arus penyejuk udara dengan deteksi suhu ruangan dan konsumsi arus stok kontak dengan deteksi panas kabel dilakukan melalui web server firebase menggunakan komputer atau smartphone. Hasil penelitian menunjukkan bahwa sistem yang dibuat berhasil memonitor konsumsi arus, mendeteksi cahaya lampu, mendeteksi suhu ruangan dan panas kabel secara realtime. Hasil pembacaan sensor memberikan informasi mengenai kondisi peralatan listrik yang digunakan. Sistem ini dapat digunakan untuk mengetahui peralatan listrik yang mengalami gangguan sehingga memudahkan untuk memperbaiki. Penelitian ini dapat dikembangkan dengan system berbasis android.

Kata kunci: Internet of Things, Web Service, Condition Monitoring

3.2. Pendahuluan

Konsep ketahanan energi telah menjadi sebuah tantangan besar bagi pembangunan ekonomi berkelanjutan seluruh dunia saat ini [1]. Menurut Badan Energi Internasional, permintaan listrik global diperkirakan akan meningkat lebih banyak dari dua pertiga pada tahun 2035 [2]. Di Indonesia, kebutuhan listrik terus meningkat setiap tahunnya sejalan dengan peningkatan pertumbuhan ekonomi nasional. Listrik merupakan kebutuhan utama bagi manusia. Area yang telah dialiri listrik dapat meningkatkan perekonomian dan kemakmuran di daerah tersebut. Rasio Elektrifikasi Indonesia dalam tahun 2018 sudah mencapai 98,05% [3] [4].

Jaringan daya cerdas terdiri dari teknologi keamanan, perangkat pintar, komunikasi dan pembangkit distribusi dan konsumsi dari pengiriman energi yang tangguh, andal, aman dan berkelanjutan dan terintegrasi [5]. Sistem yang menyediakan informasi secara realtime dan memberikan gambaran secara garis besar tentang konsumsi energi dan mengidentifikasi pemborosan energi [6]. Menurut beberapa peneliti IoT adalah sistem perangkat komputasi yang saling terkait, mesin mekanik dan digital, objek, hewan atau orang-orang yang dilengkapi dengan pengidentifikasi unik dan kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi antara manusia dengan manusia, manusia dengan komputer [7]. Banyak peneliti telah mempelajari konsep, aplikasi dan keamanan IoT [8]. Kemampuan real time dari IoT merupakan fitur utama untuk memantau dan mengontrol aplikasi system daya [9] [10]. Aplikasi web juga telah dikembangkan untuk memantau data secara online [11]. Beban listrik di rumah dapat dikontrol dari jarak jauh melalui komputer / ponsel pintar dalam kasus kelebihan beban atau saat tidak ada orang di rumah [12].

Ada beberapa manfaat penerapan IoT dalam jaringan tenaga listrik cerdas antara lain: 1) Peningkatan keandalan, ketahanan, kemampuan beradaptasi dan efisiensi energi [5], 2) peningkatan kemampuan penginderaan [5]. Teknologi IoT dan big data dapat dimanfaatkan untuk mengelola konsumsi energi dengan lebih baik di sector perumahan,

komersial dan industry [13]. Sistem pemantauan konsumsi energi rumah dan peringatan pada ubidot cloud service dengan teknologi IoT yang menggunakan modul sensor ACS712 yang dihubungkan dengan mikrokontroller dapat digunakan untuk mengukur konsumsi daya di rumah [14].

Perangkat dan teknologi IoT seperti sensor IoT adalah pendorong utama yang berkontribusi pada pertumbuhan mikrogrid/nanogrid dengan beban yang berkemampuan IoT, transportasi listrik, dan sistem penyimpanan energi terdistribusi. Hal ini menyebabkan peran pelanggan berkembang yang dapat menghasilkan tenaga listrik secara lokal dari sumber terbarukan untuk memenuhi kebutuhan dan memungkinkan untuk pertukaran daya dengan jaringan tenaga listrik [15].

Sensor IoT menawarkan beberapa manfaat saat digunakan di rumah pintar (smarthome) diantaranya adalah membantu meminimalkan pemborosan energi, mengurangi biaya, secara aktif memantau lingkungan rumah. Pemilik rumah dapat menciptakan lingkungan hidup yang nyaman di dalam rumah dengan sensor lingkungan rumah pintar dengan IoT [16]. Peralatan listrik seperti pendingin udara dan lampu penerangan dapat dikontrol dan dipantau menggunakan teknologi IoT [17]. Sensor cahaya memantau tingkat pencahayaan di dalam rumah sehingga dapat menyesuaikan dengan pencahayaan yang alami yang bersumber dari cahaya matahari [16]. Sensor suhu dan kelembaban berfungsi memantau kondisi pemanasan dan pendinginan di dalam rumah sehingga dapat juga mengurangi pemborosan energi karena dapat mematikan pendingin atau pemanas yang ada [16]. Allegro® ACS712 memberikan hasil yang ekonomis dan presisi solusi untuk pembacaan arus AC atau DC di industri, otomotif, komersial, dan sistem komunikasi. Perangkat paket memungkinkan implementasi yang mudah seperti aplikasi kontrol motor, deteksi beban dan manajemen catu daya mode-sakelar, dan perlindungan arus lebih [18].

Keandalan Sensor IoT dan Kebutuhannya dapat ditingkatkan dengan cara menurunkan kepadatan arus dan mengoperasikan sensor IoT pada suhu yang lebih dingin. Keandalan sensor IoT dapat ditingkatkan dengan

tambahan biaya. Siklus hidup produk berbanding lurus keandalan sensor IoT, ada yang sangat singkat sekitar 1-3 tahun dengan sensor IoT yang lebih murah dan kurang dapat diandalkan, siklus hidup yang lebih lama sekitar 3-5 tahun dengan sensor IoT yang sedikit mahal dan dapat diandalkan, siklus hidup yang lebih lama sekitar 10-20 tahun dengan sensor IoT yang sangat mahal dan paling andal [5].

Bangunan modern terdiri dari berbagai peralatan seperti pemanas, ventilasi, dan pendingin udara (HVAC) dan penerangan. Setiap peralatan terdiri dari beberapa komponen seperti pompa, kipas, saluran, sensor, lampu, kabel dll. Semua peralatan dapat dipantau dan dikelola oleh sistem manajemen gedung. Semua komponen ini dapat mengalami kerusakan yang dapat disebabkan aus karena pemakaian dalam waktu yang lama, kesalahan konfigurasi, dan masalah komunikasi. Kerusakan peralatan mempengaruhi biaya perawatan dan khususnya efisiensi energi [19] [20]. Pengoperasian sistem HVAC yang tidak normal dapat mengakibatkan peningkatan penggunaan energi serta kualitas udara dalam ruangan yang buruk, ketidaknyamanan termal, dan produktivitas yang rendah [21]

Beban peralatan seperti *Heating, Ventilation, and Air-Conditioning* (HVAC) bervariasi tergantung pada jenis dan lokasi bangunan, tetapi kerusakan pada salah satu peralatan dapat membuat hingga 50% dari total konsumsi energi sehingga dapat menyebabkan kehilangan energi yang besar [19]. IoT dapat digunakan untuk meningkatkan efisiensi energi, meningkatkan pangsa energi terbarukan, dan mengurangi dampak lingkungan dari penggunaan energi [22]. IoT memanfaatkan kombinasi aplikasi perangkat lunak cerdas bersama dengan perangkat elektronik untuk membangun jaringan pertukaran data yang efektif. Untuk menerapkan ini, tertanam biaya rendah dan konsumsi daya rendah Modul Wi-Fi ESP8266 digunakan untuk mengoperasikan saluran relai, yang bertindak sebagai sakelar untuk mengontrol peralatan rumah tangga [22]. Sebagai sumber perolehan informasi, keakuratan dan keandalan output array sensor sangat penting untuk kinerja sistem pengukuran dan kontrol. Node MCU digunakan untuk mengumpulkan data sensor secara langsung yang menghemat chip

kontrol utama yang terpisah sehingga biaya yang dibutuhkan lebih murah [23].

Teknik *Fault detection and diagnostics* (FDD) dapat digunakan untuk memantau sistem kelistrikan bangunan serta mendeteksi dan mendiagnosa kerusakan peralatan kelistrikan [2]. Internet of Things (IoT) berfungsi untuk menggabungkan banyak sensor nirkabel ke dalam objek yang sama untuk memenuhi aplikasi utama IoT (smart building, smart city dan industri 4.0), Multimeasurement Wireless Sensor (MWS) mampu secara bersamaan mengukur beberapa data lingkungan yang berbeda (suhu, tekanan, getaran, dll.) [24]. Analisis (IoT) adalah sarana penting untuk memperoleh pengetahuan dan mendukung aplikasi untuk smart home. Peralatan dan perangkat yang terhubung di dalam smart home menghasilkan sejumlah besar data tentang konsumen dengan aktivitas sehari-hari. Analisis IoT dapat membantu mempersonalisasi aplikasi yang menguntungkan pemilik rumah dan industri yang terus berkembang yang perlu memanfaatkan profil konsumen [25].

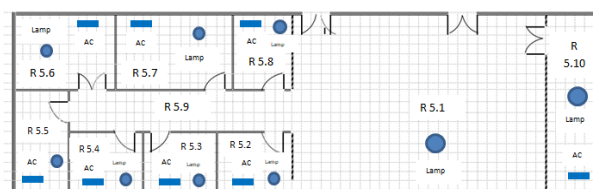
Semua komponen ini dapat mengalami kerusakan yang dapat disebabkan aus karena pemakaian dalam waktu yang lama, kesalahan konfigurasi, dan masalah komunikasi. Kerusakan peralatan mempengaruhi biaya perawatan dan khususnya efisiensi energi [19]. Diagnosis kawat menjadi sangat penting untuk memastikan keselamatan, keamanan, integritas dan kinerja yang optimal. Jaringan sensor yang terhubung dengan obyek merupakan solusi untuk mendiagnosis kesalahan yang terjadi [26]. IoT dapat menggabungkan banyak sensor nirkabel ke dalam obyek yang sama untuk aplikasi seperti smart building, smart city dan industry 4.0 [27].

Pada suatu sistem tenaga listrik dibutuhkan suatu sistem pengamanan yang handal yang mampu mendeteksi gangguan dan kesalahan yang terjadi pada sistem tersebut. Hal tersebut diperlukan agar suplai energi listrik dapat dipertahankan dengan baik. Agar sistem yang dimiliki mampu menjaga pelayanan suplai tenaga listrik dengan baik, maka sistem pengamanan harus memenuhi syarat *reliable*, selektif, sensitif, memiliki waktu operasi yang cepat, ekonomis dan sederhana.

3.3. Metode

Sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan. Adapun tahapan perancangan sistem monitoring dan deteksi sistem kelistrikan adalah perancangan *prototype* laboratorium radio dan telekomunikasi, perancangan hardware dan perancangan software. Selanjutnya sistem diuji coba pada satu ruangan dengan peralatan penerangan (lampu) dan penyejuk udara (AC). Adapun tahapan ujicoba adalah memonitoring konsumsi arus pada lampu dengan mendeteksi cahaya lampu dan memonitoring konsumsi arus pada penyejuk ruangan dengan deteksi suhu ruangan.

Perancangan *prototype* laboratorium telekomunikasi radio dan gelombang pendek yang terdiri dari sembilan ruangan yaitu ruang laboratorium telekomunikasi radio dan gelombang pendek (R 5.1), ruang AP 2 (R 5.2), ruang AP 1 (R 5.3), ruang professor (R 5.4), ruang asisten (R 5.5), ruang pertemuan (R 5.6), ruang mahasiswa R 5.7), ruang laboran (R 5.8), ruang komputer dan PABX (R 5.10) dan koridor (R 5.9). Adapun denah laboratorium telekonikasi radio dan gelombang pendek. Setiap ruangan dilengkapi peralatan penerangan, penyejuk ruangan dan komputer kecuali ruangan laboratorium telekomunikasi radio dan gelombang pendek yang hanya dilengkapi dengan penerangan dan peralatan praktikum seperti pada Gambar 3.1.

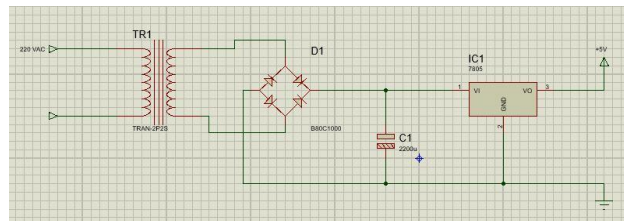


Gambar 3.1. Denah laboratorium telekomunikasi radio dan gelombang pendek

Perancangan perangkat keras (hardware) terdiri dari perancangan power supply, pemasangan sensor arus ACS 712, pemasangan sensor

cahaya LDR (Photosensitive light sensor), dan sensor suhu DHT22 pada modul Wemos D1 R1 dan modul ESP8266

Perancangan power supply dengan output +5V. Power supply ini berfungsi sebagai sumber tegangan dari modul Wemos D1 R1 dan modul ESP8266. Adapun gambar rangkaiannya seperti pada Gambar 3.2.



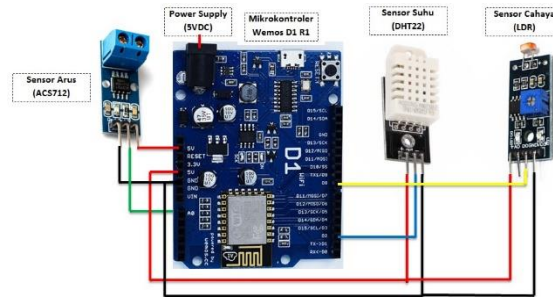
Gambar 3.2. Rangkaian power supply output +5V

Pemasangan sensor arus ACS712 yang terdiri dari 3 pin yaitu VCC, GND dan OUT pada modul Wemos D1 board ESP8266. VCC dan GND dihubungkan dengan tegangan +5 Volt dan Ground power supply, OUT sensor arus dihubungkan dengan pin A0 pada modul Wemos D1 board ESP8266.

Pemasangan Photosensitive light sensor yaitu optical cahaya LDR 4 pin yang terdiri dari Vcc, Gnd, D0 dan A0 pada modul Wemos D1 board ESP8266 . Vcc dan Gnd dihubungkan dengan power supply +5V, Ao dihubungkan dengan pin D8 modul Wemos D1 board ESP8266

Pemasangan sensor suhu DHT22 yang terdiri dari 3 pin yaitu +5Vcc, Gnd, dan Data pada modul Wemos D1 board ESP8266. Pin +5Vcc dan Gnd dihubungkan dengan power supply dan Data dihubungkan dengan pin D2 modul Wemos D1 board ESP8266

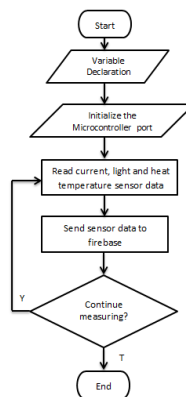
Pemasangan sensor suhu ACS712, sensor cahaya LDR, dan sensor suhu DHT22 dapat ditunjukkan seperti pada Gambar 3.3.



Gambar 3.3. Pemasangan sensor arus ACS712, sensor cahaya, dan sensor suhu DHT22 pada modul Wemos D1 board ESP8266

Perancangan perangkat lunak terdiri dari beberapa tahap yaitu pembuatan flowchart sistem monitoring arus, cahaya, suhi, program pembacaan arus, program pembacaan sensor cahaya dan program pembacaan sensor suhu, program penampil pada papan tampilan dan perancangan web server untuk komunikasi antara wemos R1 board ESP8266 dengan perangkat komputer atau smartphone menggunakan jaringan internet

Flowchart sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan seperti pada Gambar 3.4.



Gambar 3.4. Flowchart sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan

Program pembacaan arus, program pembacaan sensor cahaya dan program pembacaan sensor suhu, program penampil pada papan tampilan, dan perancangan web server untuk komunikasi antara wemos D1 board

ESP8266 dengan perangkat komputer atau smartphone menggunakan sistem internet adalah:

```
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
#include <SPI.h>
#include <Wire.h>
#include <SimpleDHT.h>
#include <ACS712.h>

// Set to run firebase and wifi
#define FIREBASE_HOST "monitoring-and-control-ab.firebaseio.com"
#define FIREBASE_AUTH "0qYhJouHUFGRIYxKTvktMJqluleBr7DiAwkX1uOe"
#define WIFI_SSID "Muliandira"
#define WIFI_PASSWORD "R4h4s14@"

//Temperature sensor DHT22
int pinDHT22 = D8;
SimpleDHT22 dht22(pinDHT22);
float temperature = 0;
float humidity = 0;
int err = SimpleDHTErrSuccess;

//Light sensor LDR, Temp sensor LM35
byte analogPin = A0;
int adc = 0;
float tempLM35 = 0;

//Current sensor ACS712
ACS712 ACS(A0, 5, 1023, 185); // max. 5A = 185
float mA ;
float current ;

void setup() {
//Switch Multiplexer
pinMode (D4,OUTPUT);
```

```

pinMode (D5,OUTPUT);
pinMode (D6,OUTPUT);
Serial.begin(9600);
//Firebase controlling
pinMode (D2, OUTPUT); //panel
pinMode (D3, OUTPUT); //penyejuk ruangan
pinMode (D9, OUTPUT); //penerangan
pinMode (D10, OUTPUT); //stok kontak
//Buzzer bip
pinMode (D11, OUTPUT);
// Connect to wifi.
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
while (WiFi.status() != WL_CONNECTED) {
  Serial.println("Connected");
  delay(500);}
//Connect to firebase
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.set("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Panel Switch",1);
Firebase.set("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Air Conditioning Switch",1);
Firebase.set("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Lamp Switch",1);
Firebase.set("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Socket Switch",1);
}
int swpanel;
int swpenyejukruangan;
int swpenerangan;
int swstokkontak;
void loop() {
digitalWrite (D11,HIGH); delay(100); digitalWrite (D11,LOW); delay(100);

```

```

panelACS712();          //Current sensor panel processing
airconditionerACS712(); //Current sensor air conditioner processing
lampACS712();           //Current sensor lamp processing
stokcontactACS712();    //Current sensor stok contact processing
tempDHT22sensor();      //Temp sensor DHT22 processing
lightsensor();          //Light sensor LDR processing
tempLM35sensor();       //Temp sensor LM35 processing

//Controller

swpanel = Firebase.getInt("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Panel Switch");
if (swpanel==1) {Serial.println("Panel ON"); digitalWrite(D2,LOW);}
if (swpanel==0) {Serial.println("Panel OFF"); digitalWrite(D2,HIGH); }

swpenyejukruangan = Firebase.getInt("Telecommunication Radio &
Microwave Laboratory /Computer & PABX Room /Air Conditioning Switch");
if (swpenyejukruangan==1) {Serial.println("Penyejuk Ruangan ON");
digitalWrite(D3,LOW);}
if (swpenyejukruangan==0) {Serial.println("Penyejuk Ruangan OFF");
digitalWrite(D3,HIGH); }

swpenerangan = Firebase.getInt("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Lamp Switch");
if (swpenerangan==1) {Serial.println("Penerangan ON");
digitalWrite(D9,LOW); }
if (swpenerangan==0) {Serial.println("Penerangan OFF");
digitalWrite(D9,HIGH);}

swstokkontak = Firebase.getInt("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Socket Switch");
if (swstokkontak==1) {Serial.println("Stok Kontak ON");
digitalWrite(D10,LOW); }
if (swstokkontak==0) {Serial.println("Stok Kontak OFF");
digitalWrite(D10,HIGH); }

//delay(60000);
}

```

```

//Collection void for system.
void panelACS712(){
  digitalWrite(D4,LOW);
  digitalWrite(D5,LOW);
  digitalWrite(D6,LOW);
  mA = ACS.mA_AC();
  current = (mA / 1000)-0.05;
  if (current>=0){String panel = String (current,2)+ String (" A");
    Firebase.setString ("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Panel Switch /Panel Current", panel); delay(10);}
  else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Panel Switch /Panel Current", "0 A");
delay(10);}}
void airconditionerACS712(){
  digitalWrite(D4,HIGH);
  digitalWrite(D5,LOW);
  digitalWrite(D6,LOW);
  mA = ACS.mA_AC();
  current = (mA / 1000)-0.05;
  if (current>=0){String airconditionercurrent = String (current,2)+ String (" A");
    Firebase.setString ("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Air Conditioning Switch /Air Conditioning Current",
airconditionercurrent); delay(10);}
  else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Air Conditioning Switch /Air
Conditioning Current", "0 A"); delay(10);}}
void lampACS712(){
  digitalWrite(D4,LOW);
  digitalWrite(D5,HIGH);
  digitalWrite(D6,LOW);
  mA = ACS.mA_AC();
  current = (mA / 1000)-0.05;

```

```

    if (current>=0){String lamp = String (current,2)+ String (" A");
        Firebase.setString ("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Lamp Switch /Lamp Current", lamp); delay(10);}
    else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Lamp Switch /Lamp Current", "0 A");
delay(10);}}
void stokcontactACS712(){
    digitalWrite(D4,HIGH);
    digitalWrite(D5,HIGH);
    digitalWrite(D6,LOW);
    mA = ACS.mA_AC();
    current = (mA / 1000)-0.05;
    if (current>=0){String stokcontact = String (current,2)+ String (" A");
        Firebase.setString ("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Socket Switch /Socket Current", stokcontact);
delay(10);}
    else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Socket Switch /Socket Current", "0
A"); delay(10);}}
void tempDHT22sensor(){
    //MuxTempDHT22();
    if ((err = dht22.read2(&temperature, &humidity, NULL)) !=
SimpleDHTErrSuccess){return;}
    String DHT22Temp = String (temperature) + String (" °C");
    Firebase.setString("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Air Conditioning Switch /Temperature Sensor
Value", DHT22Temp); delay(10);}
void lightsensor(){
    digitalWrite(D4,LOW);
    digitalWrite(D5,LOW);
    digitalWrite(D6,HIGH);
    adc = analogRead(analogPin);

```

```

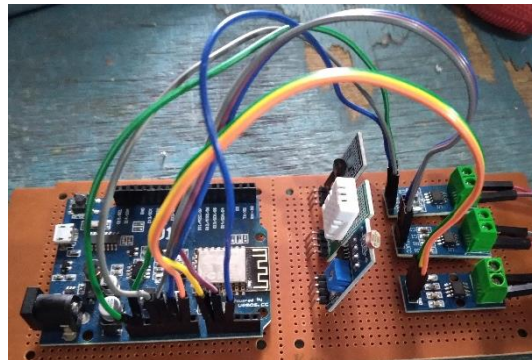
if (adc <500){Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Lamp Switch /Light Sensor Condition",
"Light" );delay(10);}
else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Lamp Switch /Light Sensor Condition",
"No Light");delay(10);}}
void tempLM35sensor(){
    digitalWrite(D4,HIGH);
    digitalWrite(D5,LOW);
    digitalWrite(D6,HIGH);
    adc = analogRead(analogPin);
    tempLM35 = (adc/3.222);
    String tempLM35A= String (tempLM35) + String (" °C");
    Firebase.setString ("Telecommunication Radio & Microwave Laboratory
/Computer & PABX Room /Socket Switch /Thermal Sensor
Value/Temperature",tempLM35A );delay(10);
    if (tempLM35 <=70){Firebase.setString ("Telecommunication Radio &
Microwave Laboratory /Computer & PABX Room /Socket Switch /Socket
Temperature/Condition", "Secure" );delay(10);}
    else {Firebase.setString ("Telecommunication Radio & Microwave
Laboratory /Computer & PABX Room /Socket Switch /Socket
Temperature/Condition", "Warning");delay(10);}}

```

Pengujian sistem monitoring dengan menggunakan web service firebase, langkah pertama adalah login dengan akun google yang sudah didaftar pada web service firebase, buka firebase.google.com, klik Go to console, pilih monitoring lab-radio-telkom, pilih develop sehingga tampil menu develop, pilih database sehingga tampil monitoring Lab-Radio-Telkom.

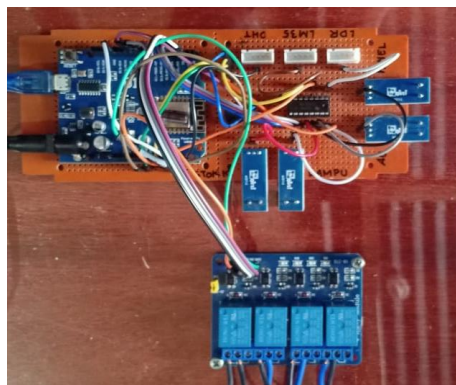
3.4. Hasil dan Pembahasan

Hasil pemasangan sensor arus ACS712, sensor cahaya (*light sensor*), sensor suhu DHT22, dan sensor panas (*Thermal sensor*) LM35 pada modul Wemos R1 board ESP8266 seperti pada Gambar 3.5.



Gambar 3.5. Hasil pemasangan sensor arus, sensor cahaya sensor suhu dan sensor panas

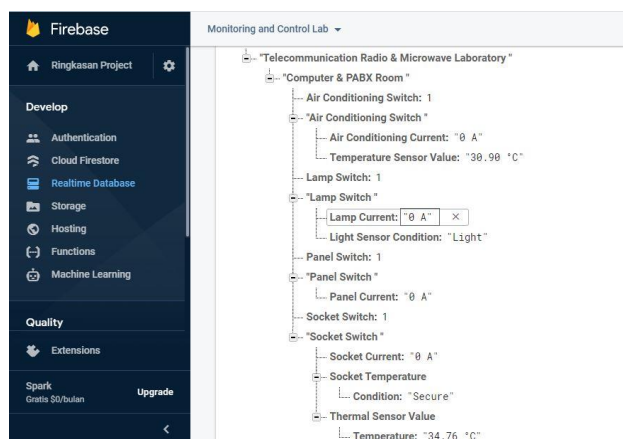
Hasil pemasangan sistem pengontrolan untuk mengaktifkan penyejuk udara (AC), penerangan (Lampu), dan stok kontak menggunakan relay yang dipasang pada masing-masing saklar peralatan yang digunakan seperti pada Gambar 3.6.



Gambar 3.6. Pemasangan sistem pengontrolan untuk mengaktifkan penyejuk udara (AC), penerangan (Lampu), dan stok kontak menggunakan relay pada modul Wemos D1 board ESP8266

Hasil perancangan web service firebase pembacaan arus penerangan dengan deteksi cahaya, hasil pembacaan arus penyejuk udara dengan deteksi suhu ruangan dan hasil pembacaan arus stok kontak

dengan deteksi panas menggunakan komputer atau smartphone seperti pada Gambar 3.7.



Gambar 3.7. Hasil perancangan web service firebase monitoring arus lampu, penyejuk ruangan, stok kontak dan deteksi cahaya lampu, suhu ruangan dan panas stok kontak

Hasil monitoring konsumsi arus penerangan (Lampu) jenis LED dengan daya 55W, konsumsi arus penyejuk ruangan (AC) dengan daya 330 Watt dengan arus 1,6 A, konsumsi arus stok kontak yang terhubung dengan personal computer (PC) dengan daya power supply 430 watt dan LCD monitor dengan arus 1,5 A selama 60 menit dengan menggunakan sensor suhu ACS 712 ditunjukkan pada Tabel 3.1.

Tabel 3.1. Hasil monitoring konsumsi arus penerangan (lampu), penyejuk ruangan (ac), arus stok kontak, dan arus panel menggunakan sensor arus ACS712

No.	Switchs condition				Time (Minute)	Current values of sensor			
	SW 1	SW 2	SW 3	SW 4		Panel current (A)	Lamp current (A)	AC current (A)	Socket current (A)
1	OFF	OFF	OFF	OFF	-	0	0	0	0
2	ON	ON	ON	ON	6 th	2,50	0,28	1,47	0,69
3	ON	ON	ON	ON	12 th	2,76	0,29	1,53	0,71
4	ON	ON	ON	ON	18 th	2,58	0,24	1,50	0,68
5	ON	ON	ON	ON	24 th	2,49	0,22	1,48	0,67
6	ON	ON	ON	ON	30 th	2,53	0,25	1,52	0,65
7	ON	ON	ON	ON	36 th	2,56	0,29	1,49	0,60
8	ON	ON	ON	ON	42 th	0,90	0,22	0,13	0,59
9	ON	ON	ON	ON	48 th	2,51	0,25	1,50	0,62
10	ON	ON	ON	ON	54 th	2,65	0,29	1,52	0,69
11	ON	ON	ON	ON	60 th	2,53	0,29	1,53	0,66

Hasil pengukuran konsumsi arus penerangan (Lampu) jenis LED dengan daya 55W, konsumsi arus penyejuk ruangan (AC) dengan daya 330 Watt dengan arus 1,6 A, konsumsi arus stok kontak yang terhubung dengan personal computer (PC) dengan daya power supply 430 watt dan LCD monitor dengan arus 1,5 A selama 60 menit dengan menggunakan *clamp meter* ditunjukkan pada Tabel 3.2.

Tabel 3.2. Hasil Pengukuran Konsumsi Arus Penerangan (Lampu), Penyejuk Ruangan (AC), Arus Stok Kontak, Dan Arus Panel Menggunakan Clamp Meter

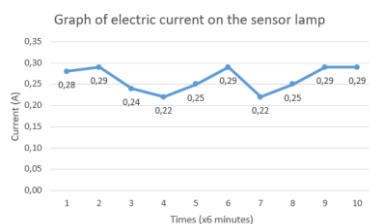
No.	Switchs condition				Time (Minute)	Current values using clamp meter			
	SW 1	SW 2	SW 3	SW 4		Panel current (A)	Lamp current (A)	AC current (A)	Socket current (A)
1	OFF	OFF	OFF	OFF	-	0	0	0	0
2	ON	ON	ON	ON	6 th	2,38	0,27	1,40	0,66
3	ON	ON	ON	ON	12 th	2,62	0,28	1,45	0,67
4	ON	ON	ON	ON	18 th	2,45	0,23	1,43	0,65
5	ON	ON	ON	ON	24 th	2,37	0,21	1,41	0,64
6	ON	ON	ON	ON	30 th	2,40	0,24	1,44	0,62
7	ON	ON	ON	ON	36 th	2,43	0,28	1,42	0,57
8	ON	ON	ON	ON	42 th	0,86	0,21	0,12	0,56
9	ON	ON	ON	ON	48 th	2,38	0,24	1,43	0,59
10	ON	ON	ON	ON	54 th	2,52	0,28	1,44	0,66
11	ON	ON	ON	ON	60 th	2,40	0,28	1,45	0,63

Hasil monitoring suhu ruangan menggunakan sensor DHT22, monitoring suhu thermal atau panas kabel menggunakan sensor LM35 dan monitoring kondisi cahaya penerangan dalam ruangan menggunakan sensor cahaya selama 60 menit ditunjukkan pada Tabel 3.3.

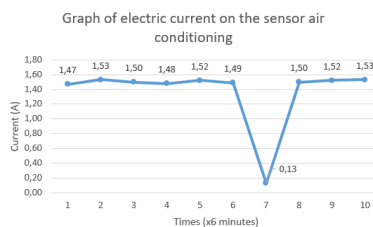
Tabel 3.3. Hasil Monitoring Kondisi Cahaya Penerangan, Suhu Ruangan, dan Suhu Thermal Kabel Menggunakan Sensor Cahaya dht22 dan lm35

No.	Switchs condition				Time (Minute)	Light sensor condition	Temper ature sensor value ("C)	Thermal sensor value ("C)
	SW 1	SW 2	SW 3	SW 4				
1	OFF	OFF	OFF	OFF	-	NO LIGHT	28,30	29,50
2	ON	ON	ON	ON	6 th	LIGHT	27,20	28,70
3	ON	ON	ON	ON	12 th	LIGHT	24,70	26,40
4	ON	ON	ON	ON	18 th	LIGHT	25,40	26,60
5	ON	ON	ON	ON	24 th	LIGHT	22,90	24,40
6	ON	ON	ON	ON	30 th	LIGHT	18,65	20,35
7	ON	ON	ON	ON	36 th	LIGHT	18,10	19,30
8	ON	ON	ON	ON	42 th	LIGHT	16,55	18,05
9	ON	ON	ON	ON	48 th	LIGHT	21,70	23,40
10	ON	ON	ON	ON	54 th	LIGHT	20,35	21,55
11	ON	ON	ON	ON	60 th	LIGHT	19,10	20,60

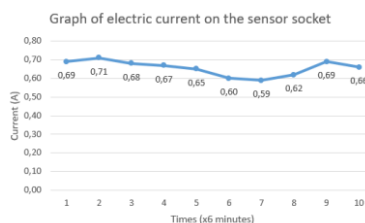
Grafik konsumsi arus pada lampu, konsumsi arus pada *Air Conditioning* (AC) dan konsumsi arus pada stok kontak, menggunakan sensor arus ACS712 ditunjukkan seperti pada Gambar 3.8.



(a)



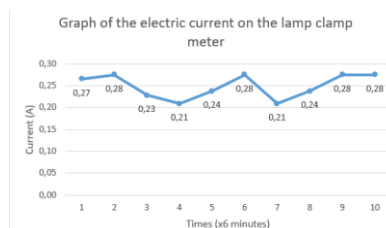
(b)



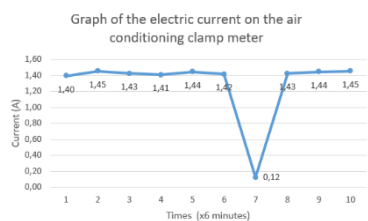
(c)

Gambar 3.8,. Grafik konsumsi arus menggunakan sensor arus ACS712 (a) Lamp (b) Air Conditioner (AC) (c) Stok Kontak

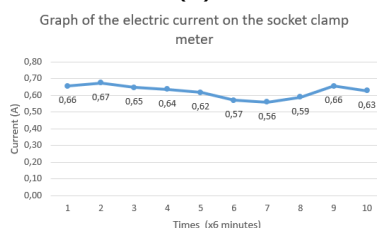
Grafik konsumsi arus pada lampu, konsumsi arus pada *Air Conditioning* (AC) dan konsumsi arus pada stok kontak, menggunakan *clamp meter* ditunjukkan seperti pada Gambar 3.9.



(a)



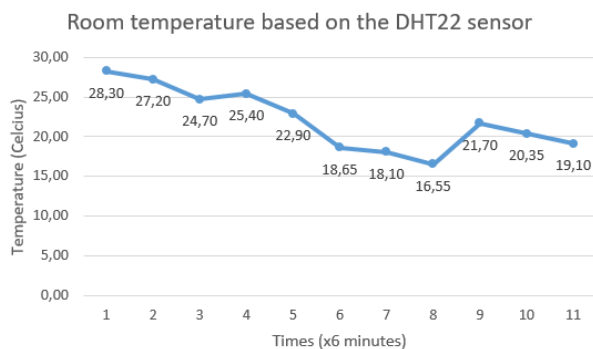
(b)



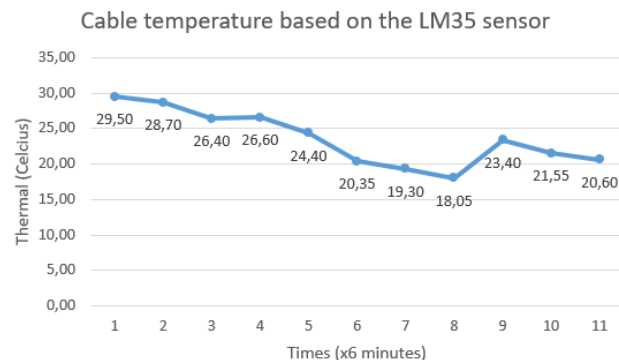
(c)

Gambar 3.9. Grafik konsumsi arus menggunakan *clamp meter* (a) Lamp (b) Air Conditioner (AC) (c) Stok Kontak (*socket*)

Grafik kondisi suhu ruangan dan kondisi suhu thermal (panas) kabel ditunjukkan pada Gambar 3.10. – 3.11.



Gambar 3.10. Grafik kondisi suhu ruangan menggunakan sensor DHT22



Gambar 3.11. Grafik kondisi suhu thermal (panas) Kabel menggunakan sensor LM35

Berdasarkan hasil penelitian menunjukkan bahwa *prototype* sistem monitoring energi panel listrik, penerangan (Lampu) dengan deteksi cahaya, penyejuk udara (*Air Conditioning*) dengan deteksi suhu ruangan, monitoring energi pada stok kontak dengan deteksi suhu thermal (panas) kabel berhasil dirancang dan berhasil diakses melalui komputer atau smartphone menggunakan aplikasi *web service firebase*.

Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada alat penerangan (lampu) yang dayanya 50 watt dengan deteksi cahaya lampu seperti pada tabel 1 menunjukkan bahwa konsumsi arus rata-rata 0,26 ampere selama 60 menit dan hasil deteksi cahaya lampu menunjukkan bahwa ada cahaya lampu selama 60 menit seperti yang ditunjukkan pada tabel 1. Hal ini menunjukkan bahwa selama ada arus yang mengalir pada alat penerangan (lampu) maka ada cahaya lampu yang terdeteksi, berarti lampu di ruangan tempat uji coba tetap menyala selama dimonitoring.

Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada alat penyejuk ruangan (*Air Conditioning*) yang dayanya 330 watt dengan arus 1.6 A dan deteksi suhu ruangan seperti pada tabel 1 dan gambar 20 menunjukkan bahwa konsumsi arus rata-rata 1,37 ampere selama 60 menit dan hasil deteksi suhu ruangan menunjukkan rata-rata 24,3 °C selama 60 menit. Hal ini menunjukkan bahwa selama 60 menit monitoring

arus dan deteksi suhu ruangan pada alat penyejuk ruangan (AC) maka ada arus yang mengalir dan terdeteksi suhu ruangan yang menurun secara signifikan dari suhu awal 28,30°C setelah 60 menit menjadi 19,10°C yang berarti bahwa alat penyejuk ruangan (AC) di ruangan tempat uji coba masih berfungsi dengan baik selama dimonitoring.

Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada stok kontak yang terhubung dengan personal computer (PC) dengan daya power supply 430 watt dan LCD monitor dengan arus 1,5A dan deteksi suhu thermal (panas) kabel seperti pada tabel 1 dan gambar 21 menunjukkan bahwa konsumsi arus rata-rata 0,66 ampere selama 60 menit dan hasil deteksi suhu thermal (panas) menunjukkan rata-rata 25,9 °C selama 60 menit. Hal ini menunjukkan bahwa selama 60 menit monitoring arus dan deteksi suhu thermal (panas) kabel, arus mengalir dan suhu thermal (panas) kabel masih dalam kondisi aman yaitu rata-rata 25,9 °C.

Hasil pengukuran arus menggunakan *clamp meter* pada alat penerangan (lampu) yang dayanya 50 watt seperti pada tabel 2 dan gambar 17 menunjukkan bahwa konsumsi arus rata-rata 0,25 ampere selama 60 menit. hasil pengukuran arus menggunakan *clamp meter* pada *Air Conditioning* (AC) yang dayanya 330 watt dengan arus 1.6 A seperti pada tabel 2 dan gambar 18 menunjukkan bahwa konsumsi arus rata-rata 1,30 ampere selama 60 menit. Hasil pengukuran arus menggunakan *clamp meter* pada stok kontak yang terhubung dengan personal computer (PC) dengan daya power supply 430 watt dan LCD monitor dengan arus 1,5A seperti pada tabel 2 dan gambar 19 menunjukkan bahwa konsumsi arus rata-rata 0,62 ampere selama 60 menit.

Berdasarkan hasil pengukuran arus menggunakan sensor arus ACS712 dan pengukuran arus menggunakan *clamp meter* pada peralatan penerangan lampu, *Air Conditioning* dan stok kontak yang terhubung dengan *personal computer* dan *LCD monitor* menunjukkan perbedaan rata-rata 0,04 A atau sekitar 5,26 %.

3.5. Kesimpulan

Sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan dapat diakses melalui komputer atau smartphone menggunakan aplikasi *web service firebase*. Pendeteksian cahaya lampu menggunakan sensor cahaya menunjukkan bahwa penerangan (lampu) pada ruangan yang dimonitoring dalam kondisi baik. Pendeteksian suhu ruangan yang mengalami penurunan selama monitoring menunjukkan bahwa kondisi penyejuk udara (AC) pada ruangan yang dimonitoring dalam kondisi berfungsi dengan baik. Pendeteksian suhu thermal (panas) kabel masih menunjukkan nilai pada kondisi aman yaitu sekitar 25,6 °C.

Sistem ini dapat digunakan untuk mengetahui system kelistrikan yang mengalami gangguan sehingga memudahkan untuk memperbaiki.

Pengembangan penelitian dapat dilakukan dengan optimalisasi sensor dalam mendeteksi kesalahan yang sering terjadi pada peralatan listrik atau sistem kelistrikan dengan notifikasi melalui aplikasi telegram dan system pengontrolan berbasis android.

3.6. Daftar Pustaka

- [1] H. Fatima, F. Farouk and H. B. Pascal, "Multi-objective optimization methodology for net zero energy buildings," *Journal of Building Engineering*, vol. 16, pp. 57-71, 2018.
- [2] M. A. Hannan, M. Faisal, P. J. Ker, L. H. Mun, K. Parvin, T. M. I. Mahlia and F. Blaabjerg, "A Review of Internet of Energy Based," *Transaction IEEE*, vol. 3536, 2018.
- [3] A. I. Malakani, D. H. Arhanto, B. G. D. Wicaksono and A. Purwadi, "Study and Design of Off-Grid PV Power System in Pirien, Asmat Regency, Papua Province using MATLAB/SIMULINK," in *International Conference on High Voltage Engineering and Power Systems (ICHVEPS)*, Bali-Indonesia, 2019.
- [4] K. M. Komkrit Chooruang, "Design of an IoT Energy Monitoring System, 978-1-5386-7159-7/18/\$31.00 ©2018 IEEE," in *International Conference on ICT and Knowledge Engineering*, 2018.
- [5] B. Guneet, K. Ganesh, S. V. Rajendra and B. Richard, "Review of Internet of Things (IoT) in Electric Power and Energy System," *IEEE*

INTERNET OF THINGS JOURNAL, vol. 4662, 2018.

- [6] D. Santos and J. C. Ferreira, "IoT Power Monitoring System for Smart Environments," *Sustainability*, vol. 5355, 2019.
- [7] A. Gillis, L. Rosencrance, S. Shea and I. Wigmore, Internet of Things (IoT), 2020.
- [8] T. D. Mcallister, S. El-tawab and a. M. H. Heydari, "Localization of Health Center Assets Through an IoT Environment (LoCATE)," pp. 132-137, 2017.
- [9] L. Zhao, I. Matsuo, Y. Zhou and W.-J. Lee, "Design of an Industrial IoT-Based Monitoring System for Power Substation," *IEEE Transaction on Industry Applications*, 2019.
- [10] M. D. Mudaliar and N. Sivakumar, "IoT based real time energy monitoring system using Raspberry Pi," *Internet of Things (IoT)*, vol. 2542, 2020.
- [11] A. Hamied, A. Mellit and M. Z. a. R. Birouk, "IoT-based experimental prototype for monitoring of photovoltaic arrays," *International Conference on Applied Smart Systems (ICASS'2018)*, pp. 24-25, 2018.
- [12] S. Balamurugan and D. Saravanakamalam, "Energy Monitoring and Management using Internet of Things," *International Conference on Power and Embedded Drive Control (ICPEDC)*, 2017.
- [13] I. A. Zulkernan, M. Rasyid, R. Gupta and a. M. Alikarar, "A Smart Home Energy Management System Using IoT and Big Data Analytics Approach," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 426-434, 2017.
- [14] Z. H. C. Soh, I. H. Hamzah, S. A. C. Abdullah, M. A. Shafie, S. N. Sulaiman and K. Daud, "Energy Consumption Monitoring and Alert System via IoT," in *International Conference on Future Internet of Things and Cloud (FiCloud)*, 2019.
- [15] B. D. Lounsbury, "Weighing the Advantages of Distributed and Centralized Energy Storage".
- [16] P. Type, "Unlock 20% Off Unlock".
- [17] W. ". Hartman, E. V. Alexander Hansen, S. El-Tawab and K. Altaï, "Energy Monitoring and Control Using Internet of Things (IoT) System," *IEEE*, 2018.
- [18] I. Allegro MicroSystems, Fully Integrated, Hall Effect-Based Linear Current Sensor with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor ACS712.
- [19] J. Kim, J. Cai and a. J. E. Braun, Common Faults and Their Prioritization in Small Commercial Buildings, National Renewable Energy Laboratory, 2018.
- [20] C. Haithem, A. S. Bayraktar and C. Çeken, "Energy Management in Smart Buildings by Using M2M Communication," *IEEE*, 2019.
- [21] M. s. Mirnaghi and F. Haghighat, "Fault Detection and Diagnosis of large-scale HVAC system in Building using data-driven Methods,"

Energy and Building, vol. 229, 2020.

- [22] R. K. K. a. S. Yerroju, "Energy Efficient Home Automation Using IoT," *International Conference on Communication, Computing and Internet of Things (IC3IoT)*, 2018.
- [23] T. Qiang, G. Guangling, L. hai and C. L. a. W. Han, Nodemcu-based Low-cost Smart Home Node Design, 2018.
- [24] M. J. a. H. R. S. Claudio Giovanni Mattera, "Online Energy Simulator for building fault detection and diagnostics using dynamic energy performance model," *International Journal of Low-Carbon Technologies*, pp. 1-9, 2018.
- [25] A. Yassine, S. Singh and M. S. Hossain, "IoT big data analytics for smart homes with fog and cloud computing," *Future Generation Computer Systems*, 2018.
- [26] E. Cabanillas, C. Layer, M. Kafal and a. A. Dupret, "Enhancing the Spatial Resolution for Wire Fault Detection Systems using Multi-Carrier Signals," *IEEE SENSORS JOURNAL*, 2018.
- [27] M. Alex, H. Mrad and A. Parsad, "Implementation of a Realibility Test Protocol for a Multimeasurement Sensor dedicated to Industrial Applications of the Internet of Things," in *Journal Pre-proofs*, 2019.

BAB 4

KENDALI PERANGKAT KELISTRIKAN PINTAR

4.1. Abstrak

MULIADI, *Kendali Perangkat Kelistrikan Pintar*

(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Konsumsi energi listrik setiap tahun selalu mengalami peningkatan yang salah satu penyebabnya adalah gaya hidup. Gaya hidup yang buruk seperti lupa mematikan perangkat elektronik yang tidak terpakai ketika sudah berada di luar rumah menjadi salah satu faktor meningkatnya konsumsi energi listrik yang berakibat pada pemborosan. Hal ini dapat terjadi dikarenakan sistem pengontrolan perangkat masih dilakukan secara manual. Salah satu solusi untuk menyelesaikan permasalahan ini adalah dengan menerapkan sistem pengontrolan perangkat jarak jauh. Beberapa penelitian telah mengembangkan sistem kontrol jarak jauh namun tidak disertai fitur yang dapat mengetahui kondisi perangkat dalam keadaan normal atau rusak sehingga sulit mengidentifikasi keberhasilan pengontrolan. Pada paper ini telah didesain dan dibangun sistem pengontrolan perangkat elektronik berbasis Internet of Things yang dilengkapi dengan fitur monitoring kondisi perangkat. Sensor cahaya, temperature, thermal dan arus listrik disematkan pada sistem untuk mengetahui kondisi perangkat. Pengujian kontrol pada sistem berhasil dengan nilai rata-rata response time 3.32 detik dan kebenaran akurasi pembacaan sensor diatas 89%. Dengan diterapkannya sistem ini pengguna dapat dengan mudah melakukan pengawasan dan kontrol perangkat elektronik dimanapun dan kapanpun sehingga peluang penggunaan energi listrik yang efisien dan efektif dapat lebih ditingkatkan.

Kata kunci: Smart Control, Electrical Device Control, Internet of Things, Monitoring, Control

4.2. Pendahuluan

Listrik merupakan salah satu sumber energi kehidupan yang sangat dibutuhkan masyarakat untuk menjalani kehidupannya. Seiring berjalannya waktu, konsumsi energi listrik setiap tahun mengalami peningkatan, seperti yang terjadi di Indonesia selama tiga tahun berturut-turut dari tahun 2017 sampai dengan 2019 jumlah konsumsi energi listrik meningkat sebesar 60 Wh per kapita [1]. Banyak hal yang dapat menyebabkan meningkatnya konsumsi energi listrik, salah satunya gaya hidup [2]. Gaya hidup yang buruk dalam penggunaan energi listrik seperti lupa menonaktifkan perangkat elektronik ketika tidak terpakai dapat disebabkan oleh sistem pengontrolan perangkat yang masih manual yaitu dengan cara menonaktifkan perangkat secara fisik langsung dari perangkatnya.

Penerapan Internet of Things sebagai solusi permasalahan dalam berbagai aspek kehidupan telah diterapkan [3]–[10] dan hal ini dapat menjadi salah satu solusi dari permasalahan pemanfaatan energi listrik yang berlebihan dengan membangun sistem kontrol perangkat elektronik jarak jauh. Sistem tersebut memungkinkan pengguna dapat mengontrol perangkat listrik walaupun sudah berada jauh dari perangkat secara realtime.

Beberapa penelitian telah dilakukan dalam membangun sistem kontrol perangkat elektronik jarak jauh, seperti yang telah dilakukan oleh ALRikabi et al [11] membangun sistem kontrol perangkat elektronik menggunakan protocol komunikasi GSM. Perintah pengontrolan dikirimkan melalui layanan SMS untuk mengaktifkan dan menonaktifkan perangkat elektronik seperti kulkas, air conditioner, ceiling fan, lighting, dan water cooler. Indra, D et al [12] membangun sistem pengontrolan perangkat listrik menggunakan Raspberry Pi 2 melalui komunikasi Wi-Fi. Didapatkan hasil bahwa jarak maksimal untuk mengontrol sistem menggunakan Laptop adalah 30 meter dan menggunakan smartphone adalah 20 meter. Eslam Al-Hassan [13] membangun smart power socket yang dikontrol untuk menghubungkan dan memutus arus listrik pada stock contact melalui komunikasi Zigbee. Nasution, T H et al [14] membangun sistem pengontrolan

perangkat elektronik melalui relay menggunakan SIM 900 sebagai komunikasi GSM via SMS dan Arduino sebagai microcontroller. Sistem yang dibangun dapat mengaktifkan dan menonaktifkan relay secara paralel.

Sistem kontrol perangkat elektronik dapat dilengkapi dengan fitur pemantauan kondisi perangkat dengan tujuan untuk memastikan bahwa perangkat elektronik yang dikontrol dalam keadaan normal dan tidak ada kendala dalam pengoperasiannya, sehingga pengontrolan dapat berjalan dengan baik. Penelitian-penelitian yang telah dijabarkan sebelumnya hanya berfokus mendesain dan membangun sistem pengontrolannya saja. Pada paper ini telah dirancang dan dibangun sistem pengontrolan perangkat elektronik berbasis Internet of Things. Pemantauan dan pengontrolan dapat dilakukan dengan jarak jauh dan realtime menggunakan aplikasi yang terinstall pada smartphone. Sistem yang dibangun dilengkapi dengan fitur pemantauan kondisi perangkat dengan menempatkan sensor cahaya, temperature, thermal dan arus. Diharapkan setelah diterapkannya sistem ini, pengguna dapat memantau dan mengontrol perangkat elektronik dengan mudah serta dapat mengetahui kondisi perangkat dari jarak jauh secara realtime sehingga peluang pemanfaatan energi listrik dapat lebih efektif dan efisien.

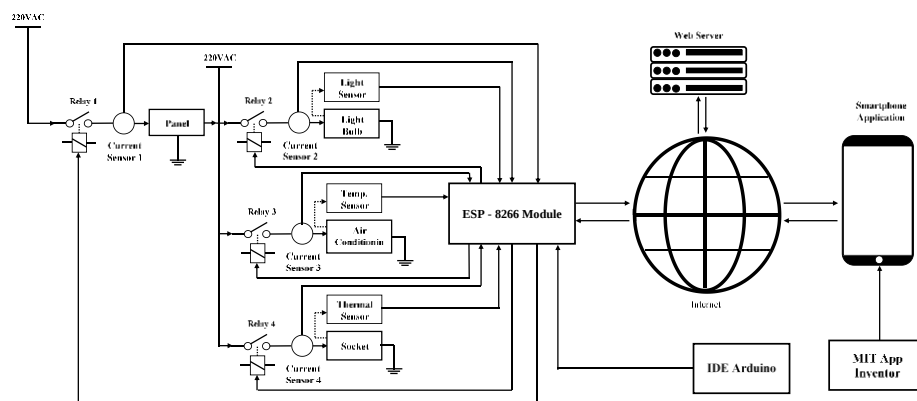
Pada bagian satu paper ini menjelaskan latar belakang serta penelitian yang terkait, bagian kedua membahas metode penelitian, bagian ketiga membahas hasil penelitian dan dilakukan analisis dari hasil yang telah didapatkan dan bagian keempat menyimpulkan hasil yang telah didapatkan dari penelitian ini.

4.3. Metode

Pada bagian ini dijelaskan desain arsitektur sistem, rangkaian hardware dan software yang digunakan, serta cara kerja sistem yang dibangun. Bagian ini juga menjelaskan metode pengujian yang dilakukan pada penelitian ini.

4.3.1. Architecture System

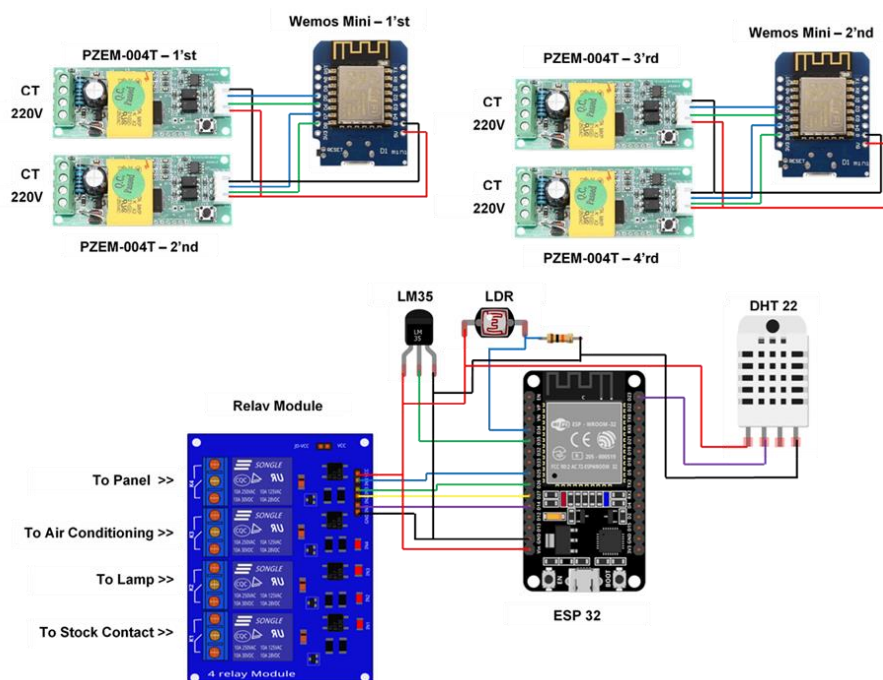
Rancangan sistem menggunakan empat buah sensor yang berfungsi untuk memantau kondisi kerja perangkat elektronik yang digunakan yaitu Light sensor digunakan untuk mendeteksi cahaya pada ruangan untuk menentukan kondisi lampu ruangan dalam keadaan normal atau tidak. Temperature sensor digunakan untuk mengetahui nilai suhu ruangan pada saat Air Conditioner (AC) sudah aktif dan menentukan kondisi AC berjalan normal atau tidak. Selanjutnya Thermal sensor diletakkan pada socket listrik untuk mengetahui suhu socket dalam keadaan normal atau dalam keadaan panas berlebihan dan yang terakhir sensor arus diletakkan pada kabel untuk mengetahui nilai arus yang tersalurkan pada perangkat elektronik. Keempat sensor ini terhubung ke microcontroller sebagai masukan data yang selanjutnya data tersebut diteruskan ke web server melalui Internet untuk diolah dan diteruskan kepada pengguna melalui aplikasi yang terinstall pada smartphone. Pengguna dapat mengontrol status aktif dan non aktif perangkat elektronik yang terhubung dengan microcontroller melalui Internet. Ketika perintah kontrol diterima maka microcontroller akan mengirimkan instruksi ke Relay untuk menghubungkan atau memutus arus sesuai dengan instruksi yang diterima dari microcontroller. Pengguna dapat memantau secara langsung informasi dari sensor dan status control yang diberikan serta mengetahui anomaly perangkat elektronik yang terdaftar lebih dini contohnya ketika status control lampu sudah aktif sedangkan informasi light sensor memberikan informasi bahwa tidak ada cahaya yang terdeteksi sehingga pengguna dapat menyimpulkan ada kerusakan pada lampu yang digunakan. Desain arsitektur system yang dibangun dapat dilihat pada Gambar 4.1..



Gambar 4.1. Architecture System

4.3.2. Hardware Schematic

Rangkaian hardware yang dibangun dibagi menjadi 3 bagian, yang dimana masing-masing bagian terdiri dari satu microcontroller. Untuk lebih jelasnya dapat dilihat pada Gambar 4.2.



Gambar 4.2. Hardware Schematic

Desain sistem menggunakan dua buah Wemos D1 Mini [15] sebagai main controller yang terhubung dengan masing-masing PZEM-004T module [16] sebagai sensor pengukur arus. Satu buah ESP32 [17] sebagai main

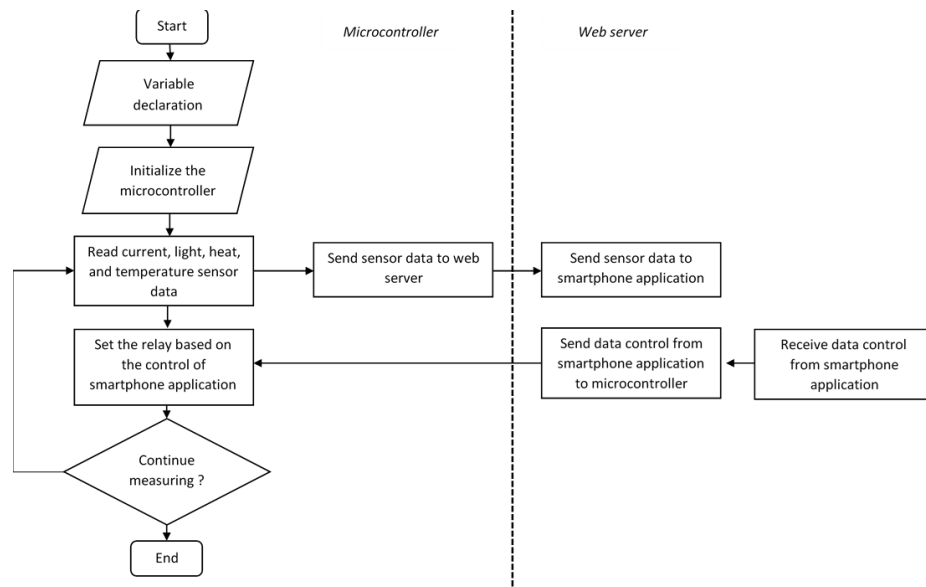
controller untuk mengontrol tiga buah sensor yaitu LDR [18] sebagai light sensor untuk mengukur tingkat cahaya pada ruangan, LM35 [19] sebagai sensor pengukur panas yang diletakkan pada electrical socket dan DHT22 [20] sebagai sensor yang mengukur suhu ruangan. Relay 4 channel module [21] terhubung dengan ESP32 yang digunakan untuk menghubungkan dan memutus arus pada panel listrik, AC, lampu dan socket listrik yang tidak terhubung dengan perangkat elektronik.

4.3.3. Software

Listring program disusun menggunakan Arduino Integrated Development Environment (IDE) [22] dan selanjutnya diupload ke ESP32 dan Wemos Mini D1. Di sisi web server digunakan Firebase [23] untuk mengolah data yang telah diterima dari controller untuk diteruskan kepada user dan juga sebaliknya. Pengembangan aplikasi mobile pada system ini menggunakan MIT App Inventor [24]. Untuk memudahkan desain rangkaian pada controller system digunakan Diptrace [25] sebagai aplikasi desain circuit yang mudah digunakan.

4.3.4. Flowchart System

Microcontroller akan selalu standby untuk membaca nilai dari sensor dan mengirmkannya ke web server untuk diolah. Setelah data diolah, web server mengirimkan data tersebut ke aplikasi smartphone pengguna untuk mengetahui status perangkat elektronik yang terdaftar. Pengguna mengirimkan instruksi melalui aplikasi yang akan diteruskan ke controller system yang selanjutnya controller system akan memberikan instruksi ke relay untuk menghubungkan atau memutuskan arus sesuai dengan data perintah dari aplikasi pengguna. Cara kerja system ini dilakukan secara berulang-ulang hingga system dinonaktifkan. Untuk lebih jelasnya terkait work flow dari system yang telah dibangun dapat dilihat pada Gambar 4.3.



Gambar 4.3. Flowchart System

4.3.5. Testing Method

Sistem diterapkan dan diuji pada satu unit rumah yang tersedia layanan Internet Indihome sebagai salah satu provider Internet terbesar di Indonesia [26] dengan kecepatan transfer data sebesar 72 Mbps. Pengujian terdiri dari tiga bagian, yaitu bagian pertama menguji fungsi control, bagian kedua menguji akurasi kebenaran pembacaan sensor dan bagian ketiga menguji nilai response time pada saat dilakukan pengontrolan. Penjelasan model pengujian dijelaskan pada poin dibawah ini:

4.3.5.1.Control: Pengujian control dilakukan dengan cara pengguna menjalankan aplikasi yang terinstall pada smartphone dan pengguna berada jauh dari rumah untuk memberikan instruksi mengaktifkan dan menonaktifkan masing-masing sebanyak 10 kali pengujian dengan jeda setiap pengujiannya selama satu menit.

4.3.5.2.Sensor: Pengujian akurasi pembacaan sensor temperature, sensor cahaya, sensor thermal dan sensor arus dilakukan sebanyak 20 kali pembacaan data dengan jeda waktu selama satu menit setiap pembacaan. Hasil pembacaan akan dibandingkan dengan perangkat pengukuran yang telah dikomersilkan untuk menentukan seberapa besar akurasi sensor yang digunakan dengan rumus perhitungan akurasi sebagai berikut:

$$Accuracy = 100\% - Error Rate \quad (1)$$

dengan rumus error rate yang digunakan adalah

$$Error Rate = \frac{Sensor Value - Commercial Measuring Device}{Commercial Measuring Device} \times 100\% \quad (2)$$

4.3.5.3. Response Time: Pengujian response time untuk melihat besaran waktu yang dibutuhkan mulai dari memberikan perintah kontrol kepada perangkat hingga perangkat tersebut melakukan perintah yang telah dikirimkan sebanyak 20 kali pengiriman dengan jeda setiap pengiriman selama satu menit. Rumus response time yang digunakan adalah

$$Response Time = T2 - T1 \quad (3)$$

T1 = Time when give command to the devices

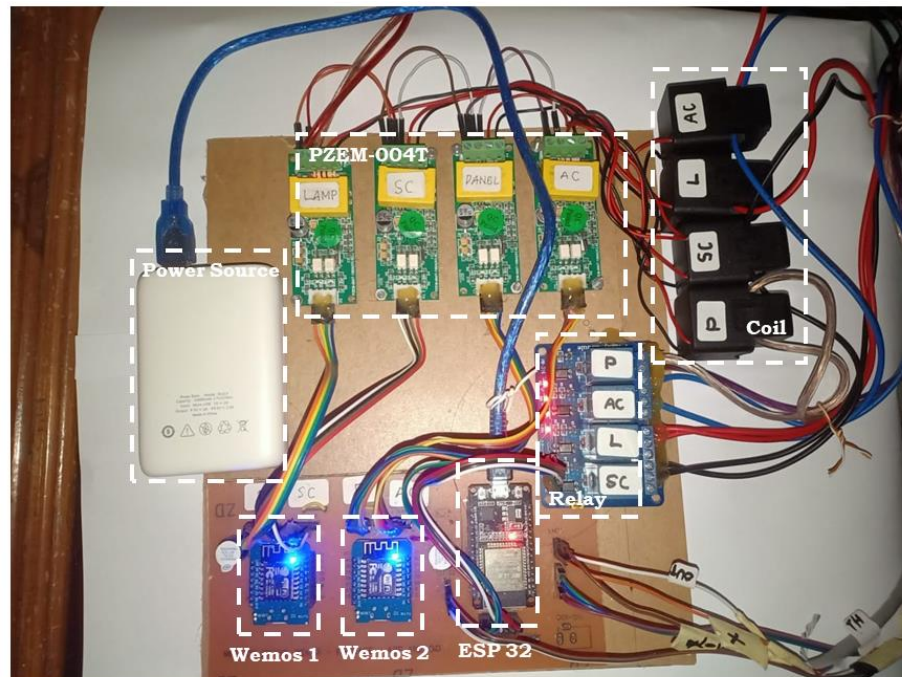
T2 = Time when devices respond to the command

4.4. Hasil dan Pembahasan

Setelah menjelaskan metode penelitian yang digunakan, pada bagian ini dijelaskan hasil dan pembahasan dari sistem yang telah dibangun menjadi dua bagian pembahasan, yaitu (1) hasil prototype system dari sisi hardware dan software dan (2) hasil pengujian sistem yang telah dilakukan.

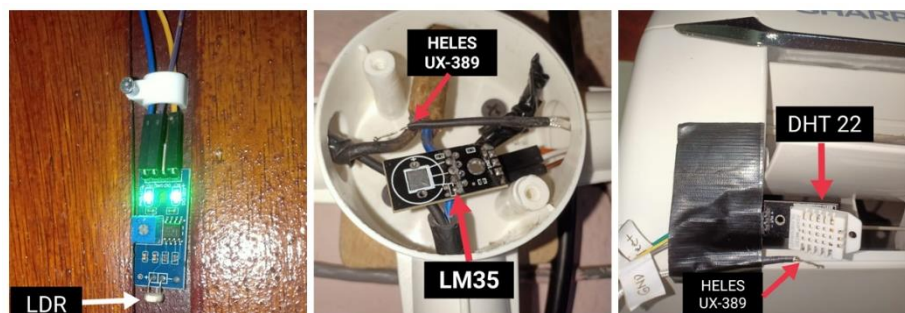
4.4.1. System Prototype

System prototype dibangun menjadi dua bagian, yaitu bagian pertama dapat dilihat pada Gambar 4.4. yang menempatkan Wemos Mini D1 dan ESP32 pada satu tempat. Wemos Mini D1 terhubung dengan empat buah sensor PZEM-004T dan ESP32 terhubung dengan Relay 4 Channel dengan Sensor yang ditempatkan terpisah pada bagian kedua.



Gambar 4.4. System of Prototype

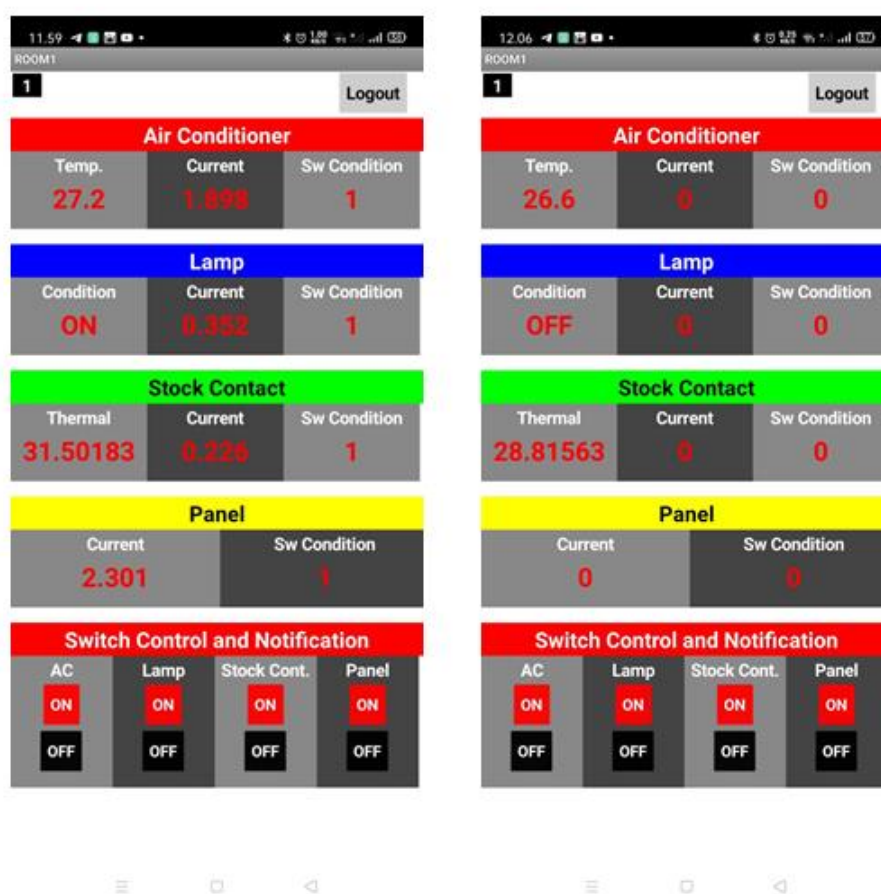
Gambar 4.5. menggambarkan bagian kedua dari system yang terdiri dari Sensor LDR yang ditempatkan di dinding ruangan, Sensor LM35 ditempatkan pada percabangan kabel yang juga ditempatkan bersamaan dengan Sensor Helex UX-389 untuk membandingkan hasil pengukuran terhadap suhu kabel. Sensor DHT 22 ditempatkan dibawah AC juga dengan Sensor Helex UX-389 untuk mengetahui perbandingan pengukuran suhu ruangan terhadap kedua Sensor yang digunakan.



Gambar 4.5. Sensor Placements

Tampilan Aplikasi pada gambar 4.6. telah dibangun menjadi dua bagian yaitu bagian atas tampilan dan bagian bawah tampilan. Bagian atas menampilkan status perangkat elektronik rumah tangga yang terhubung.

Pada AC terdiri dari nilai temperature, nilai arus dan kondisi switch. Untuk lampu terdapat parameter status kondisi lampu, nilai arus dan juga kondisi switch dan parameter yang terdapat pada stock contact terdapat nilai dari thermal, arus dan kondisi switch. System juga melakukan pemantauan pada panel listrik yang terdiri dari dua nilai parameter yaitu nilai arus dan kondisi switch. Kondisi switch yang ada pada tampilan aplikasi diberi inisial SW Condition. Jika SW Condition bernilai satu maka switch dalam keadaan on dan jika bernilai nol maka switch dalam keadaan off.



Gambar 4.6. Application Interface

Pada bagian bawah menampilkan tombol untuk mengontrol perangkat elektronik yang terhubung dengan dua kondisi yaitu on dan off.

4.4.2. Testing Results

Berdasarkan metode pengujian yang telah dijelaskan pada bagian dua, hasil pengujian dibagi menjadi lima bagian yang pertama adalah

aplikasi berhasil mengendalikan lampu dan AC melalui aplikasi yang telah terinstall di smartphone sebanyak 10 kali masing-masing dalam status off ke on dan status on ke off. Hasil pengujian bagian pertama dapat dilihat pada Table 4.1.

Table 4.1. Testing Results of Electrical Devices Control

Control testing in 20 times		
Status	Light	Air Conditioner
On	Success	Success
Off	Success	Success

Bagian kedua pada Table 4.2. didapatkan hasil pengujian bahwa pembacaan sensor LDR berhasil mendeteksi kondisi cahaya pada ruangan, sensor DHT22 berhasil membaca nilai temperature pada ruangan, sensor LM35 berhasil membaca nilai thermal dari kabel dan sensor PZEM-004T berhasil membaca nilai arus dari perangkat elektronik yang terhubung dengan jumlah pengujian masing-masing sensor sebanyak 20 kali.

Table 4.2. Testing Results of Sensor Reading

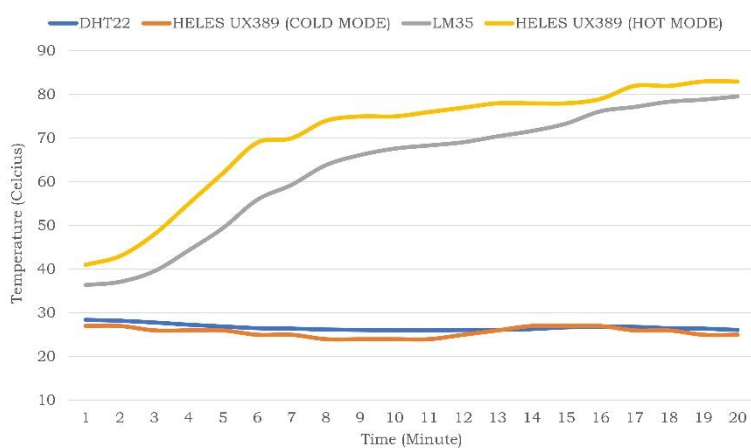
Sensor Read in 20 times			
LDR	DHT22	LM35	PZEM-004T
Success	Success	Success	Success

Bagian ketiga hasil akurasi pembacaan sensor DHT22 yang diuji selama 20 menit dengan membandingkan pembacaan dari HELES UX389 [27] didapatkan hasil bahwa DHT22 mendapatkan nilai akurasi sebesar 95.78% dan dengan model pengujian yang sama dilakukan pada sensor LM35 dengan nilai akurasi sebesar 89.70%. Graph perbandingan pembacaan sensor DHT22 dan sensor LM35 dengan HELES UX389 dapat dilihat pada gambar 4.7.

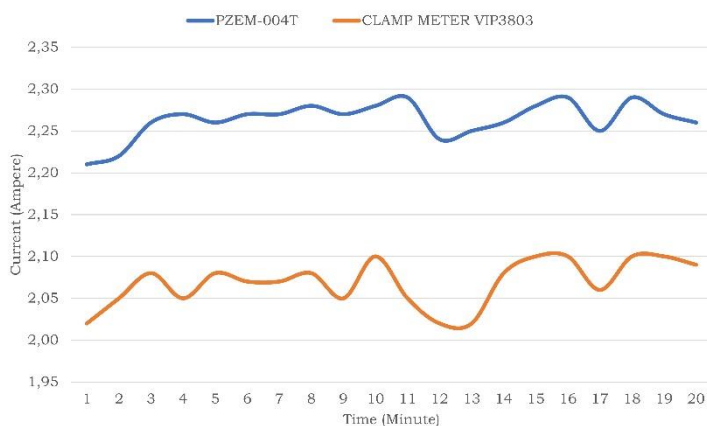
Bagian keempat pada gambar 4.8. menggambarkan hasil pengujian akurasi pembacaan sensor PZEM-004T dalam mendeteksi arus yang dibandingkan dengan alat pengukur arus komersial tipe VIP3803 [28] selama

20 menit didapatkan hasil bahwa nilai akurasi dari sensor PZEM-004T sebesar 91.39%.

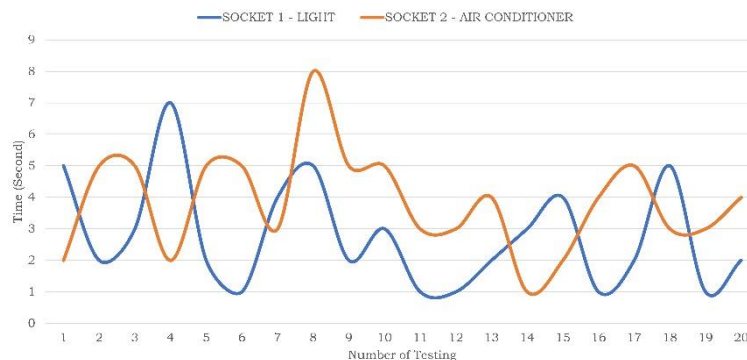
Selain melakukan pengujian terhadap akurasi sensor, bagian lima sistem juga diuji dari sisi response time. Pengujian dilakukan sebanyak 20 kali pengontrolan dengan dua perangkat elektronik yang berbeda yaitu pada socket satu untuk menghubungkan dan memutus arus pada lampu dengan hasil nilai rata-rata response time 2.8 detik dan untuk socket kedua pada AC dengan hasil nilai rata response time 3.85 detik. Pada pengujian kedelapan di socket dua didapatkan nilai respons time terlama selama 8 detik dan response time tercepat didapatkan selama 1 detik untuk kedua socket. Graph hasil pengujian response time dapat dilihat pada gambar 4.9.



Gambar 4.7. Graph of DHT 22 and LM35 Sensor Accuracy



Gambar 4.8. Graph of PZEM-004T Sensor Accuracy



Gambar 4.9. Graph of Response Time

4.5. Kesimpulan

Pada paper ini sistem pengontrolan perangkat elektronik berbasis Internet of Things yang dilengkapi dengan fitur monitoring kondisi perangkat telah berhasil dirancang dan dibangun. Pengujian kontrol dan pembacaan sensor sebanyak 20 kali semuanya berhasil. Ketepatan akurasi pembacaan semua sensor yang digunakan diatas 89% dan nilai rata-rata response time selama 3.32 detik. Sistem dilengkapi dengan aplikasi yang dapat diinstall pada smartphone dengan operating system Android untuk melakukan pengontrolan dan memonitoring kondisi perangkat elektronik dan memberikan notifikasi kepada pengguna jika perangkat dalam rusak sehingga dapat memudahkan pengguna dalam mengelola konsumsi energi listrik dan melakukan maintenance perangkat elektronik secara efektif dan efisien. Kedepannya sistem kontrol dapat dikembangkan menjadi sistem otomatisasi pengontrolan perangkat elektronik dengan tujuan kelalaian pengguna dalam pengontrolan dapat lebih diminimalisir.

4.6. Ucapan Terima kasih

Ucapan terima kasih disampaikan kepada Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi dan Fakultas Teknik Universitas Hasanuddin yang telah mendanai penelitian ini dan juga kepada tim

akademik yang telah membantu sehingga penelitian ini dapat diselesaikan dengan baik.

4.7. Daftar Pustaka

- [1] BPS, "Badan Pusat Statistik." <https://www.bps.go.id/indicator/7/1156/1/konsumsi-listrik-per-kapita.html> (accessed Aug. 24, 2021).
- [2] Sask Power, "Factors Affecting Power Use." <https://www.saskpower.com/Efficiency-Programs-and-Tips/Saving-Power-at-Home/Saving-Tips-and-Programs/Factors-Affecting-Power-Use> (accessed Aug. 24, 2021).
- [3] H. T. S. Alrikabi and N. Ali Jasim, "Design and Implementation of Smart City Applications Based on the Internet of Things," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 13, p. 4, Jul. 2021, doi: 10.3991/ijim.v15i13.22331.
- [4] N. Ali Hussien, A. A. Daleh Al-Magsoosi, H. T. S. AlRikabi, and F. Theyab Abed, "Monitoring the Consumption of Electrical Energy Based on the Internet of Things Applications," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 07, p. 17, Apr. 2021, doi: 10.3991/ijim.v15i07.20183.
- [5] N. A. Hussien, S. A. alradha Alsaidi, I. K. Ajlan, M. F. Mohamed Firdhous, and H. T. S. Alrikabi, "Smart Shopping System with RFID Technology Based on Internet of Things," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 04, p. 17, Mar. 2020, doi: 10.3991/ijim.v14i04.13511.
- [6] Y. Irawan, R. Wahyuni, M. -, H. Fonda, M. Luthfi Hamzah, and R. Muzawi, "Real Time System Monitoring and Analysis-Based Internet of Things (IoT) Technology in Measuring Outdoor Air Quality," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 10, p. 224, May 2021, doi: 10.3991/ijim.v15i10.20707.
- [7] O. Yahya, H. T. S. Alrikabi, R. M. Al_ airaji, and M. Faezipour, "Using Internet of Things Application for Disposing of Solid Waste," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 13, p. 4, Aug. 2020, doi: 10.3991/ijim.v14i13.13859.
- [8] H. El Mrabet and A. Ait Moussa, "IoT-School Attendance System Using RFID Technology," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 14, p. 95, Aug. 2020, doi: 10.3991/ijim.v14i14.14625.
- [9] E. Husni, "Driving and Fuel Consumption Monitoring with Internet of Things," *Int. J. Interact. Mob. Technol.*, vol. 11, no. 3, p. 78, Apr. 2017, doi: 10.3991/ijim.v11i3.6473.
- [10] N. Yuniarti, D. Hariyanto, S. Yatmono, and M. Abdillah, "Design and Development of IoT Based Water Flow Monitoring for Pico Hydro Power Plant," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 07, p. 69, Apr. 2021, doi: 10.3991/ijim.v15i07.18425.
- [11] H. T. S. ALRikabi, A. H. Alaidi, and K. Nasser, "The Application of

- Wireless Communication in IOT for Saving Electrical Energy.,” *Int. J. Interact. Mob. Technol.* academia.edu, 2020, [Online]. Available: <https://www.academia.edu/download/62866292/Iot20200407-45035-1vq8qea.pdf>.
- [12] D. Indra, “Design Web-Based Electrical Control System Using Raspberry Pi,” *J. Inf. Technol. Its Util. ...*, 2019, [Online]. Available: <http://repository.umi.ac.id/id/eprint/270>.
- [13] E. Al-Hassan, H. Shareef, M. M. Islam, A. Wahyudie, and ..., “Improved smart power socket for monitoring and controlling electrical home appliances,” *IEEE ...*, 2018, [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/8454720/>.
- [14] T. H. Nasution, M. A. Muchtar, I. Siregar, and ..., “Electrical appliances control prototype by using GSM module and Arduino,” *2017 4th ...*, 2017, [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/7939237/>.
- [15] “LOLIN D1 mini — WEMOS documentation.” https://www.wemos.cc/en/latest/d1/d1_mini.html (accessed Aug. 28, 2021).
- [16] “PZEM-004T V3 Module | Arduino & NodeMCU Code, Circuit, Pinout And Library.” <https://innovators-guru.com/pzem-004t-v3/> (accessed Aug. 28, 2021).
- [17] “ESP32 Wi-Fi & Bluetooth MCU | Espressif Systems.” <https://www.espressif.com/en/products/socs/esp32> (accessed Aug. 28, 2021).
- [18] “Light Dependent Resistors (LDR) - Working, Construction, Symbol, Applications.” <https://www.circuitstoday.com/ldr-light-dependent-resistors> (accessed Aug. 28, 2021).
- [19] “LM35 Temperature Sensor: Pinout, Diagrams, Equivalents & Datasheet.” <https://components101.com/sensors/lm35-temperature-sensor> (accessed Aug. 28, 2021).
- [20] “DHT22 Sensor Pinout, Specs, Equivalents, Circuit & Datasheet.” <https://components101.com/sensors/dht22-pinout-specs-datasheet> (accessed Aug. 28, 2021).
- [21] “5V Four-Channel Relay Module - Pin Diagram, Specifications, Applications, Working.” <https://components101.com/switches/5v-four-channel-relay-module-pinout-features-applications-working-datasheet> (accessed Aug. 28, 2021).
- [22] “Arduino - Environment.” <https://www.arduino.cc/en/guide/environment> (accessed Aug. 28, 2021).
- [23] “Documentation | Firebase.” <https://firebase.google.com/docs> (accessed Aug. 28, 2021).
- [24] “MIT App Inventor | Explore MIT App Inventor.” <https://appinventor.mit.edu/> (accessed Aug. 28, 2021).
- [25] “DipTrace - Schematic and PCB Design Software.” <https://diptrace.com/> (accessed Aug. 28, 2021).
- [26] “IndiHome | Internet Rumah dan Cepat Hingga 300Mbps.”

- <https://indihome.co.id/> (accessed Aug. 28, 2021).
- [27] I. Fatimah, B. Indarto, D. Anggoro, and B. Arfianto, "Thingview Free: Application of Smart Greenhouse for Chilli Plant on Android," *J. Phys. Conf. Ser.*, vol. 1951, no. 1, p. 012040, Jun. 2021, doi: 10.1088/1742-6596/1951/1/012040.
- [28] "Jual Clamp Meter Tang Ampere VIP HELES - 3803 N 3803N HELES di Lapak WOBBLE | Bukalapak." <https://www.bukalapak.com/p/industrial/tools/3o72a8o-jual-clamp-meter-tang-ampere-vip-heles-3803-n-3803n-heles?from=list-product&pos=0> (accessed Aug. 28, 2021).

BAB 5

MONITORING KONSUMSI DAN RUGI ENERGI BERBASIS IoT

5.1. Abstrak

MULIADI, *Monitoring Konsumsi dan Rugi Energi Berbasis IoT*
(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Pemanfaatan energi listrik meningkat rata-rata sebesar 5% pada setiap tahunnya namun hal ini belum tentu menandakan bahwa energi listrik yang digunakan sesuai dengan tujuan yang menyebabkan terjadinya pemborosan energi. Salah satu faktornya adalah pengguna sulit melakukan pemantauan konsumsi energi secara konvensional. Beberapa energy monitoring system telah dikembangkan namun belum dapat menghitung jumlah energy yang terbuang sementara informasi ini sangat penting untuk dianalisis jika terjadi jumlah lost energi yang terlalu besar. Pada paper ini dibangun energy monitoring system berbasis Internet of Things menggunakan PZEM-004T sensor module yang dapat memonitoring jumlah konsumsi dan lost energi secara realtime. Setelah melakukan pengujian, sistem yang dibangun mendapatkan hasil akurasi kebenaran sebesar 97.96% dari hasil perbandingan dengan dua unit electrical meter yang berbeda. Dengan dikembangkannya EMS ini maka pengguna dapat melakukan pemantauan konsumsi dan lost energy listrik dimanapun dan kapanpun melalui Internet secara fleksibel.

Kata kunci: Energy Monitoring System, Consumption Energy, Lost Energy, Monitoring, Internet of Things, IoT, PZEM-004T Sensor

5.2. Pendahuluan

Berdasarkan data dari U.S. Energy Information Administration (EIA), pemanfaatan energi listrik di seluruh dunia pada tahun 2015 sampai dengan tahun 2018 mengalami peningkatan rata-rata sebanyak 5% pada setiap tahunnya [1]. Pada umumnya pemanfaatan energi listrik digunakan pada pencahayaan bangunan, transportasi, industri dan perangkat elektronik rumah tangga [2].

Peningkatan konsumsi energi listrik belum tentu menunjang produktivitas sehari-hari yang pada akhirnya menyebabkan terjadinya

pemborosan energi. Salah satu sebab terjadinya pemborosan energi adalah pengguna mengalami kesulitan untuk menghitung jumlah energi yang telah dikonsumsi setiap waktunya [3] sehingga tidak dapat mengontrol dengan baik penggunaan energi listriknya. Selain mengalami kesulitan pengguna juga mengalami kejenuhan melakukan pengecekan konsumsi energi melalui alat pengukur energi listrik secara konvensional [4] [5] yang terpasang pada lokasi bangunan. Lost Energi [6] [7] [8] juga merupakan salah satu factor yang menyebabkan meningkatnya penggunaan konsumsi energi listrik terlebih lagi lost energi adalah energi listrik yang terbuang percuma sehingga sangat perlu diketahui oleh pengguna besaran lost energi dalam penggunaannya.

Energy Monitoring System (EMS) merupakan sebuah solusi dari permasalahan yang telah dijabarkan sebelumnya untuk dapat mengatasi kesulitan dan kejenuhan pengguna dalam melakukan monitoring konsumsi energi listrik secara konvensional. EMS menampilkan penggunaan energi secara digital sehingga memudahkan pengguna untuk mengetahui jumlah konsumsi energinya. Pemanfaatan sistem Internet of Things yang telah diterapkan pada beberapa penelitian [9] [10] [11] [12] [13] [14] [15] membuat sistem dapat diakses dimana saja dan kapan saja sehingga pengguna dapat lebih fleksibel untuk melakukan pengecekan konsumsi energi.

Beberapa penelitian telah dilakukan untuk membangun EMS, seperti yang telah dilakukan oleh Sergio H. M. S. Andrade, et al. [16] membangun sistem monitoring konsumsi energi listrik melalui aplikasi mobile secara realtime menggunakan Arduino Uno sebagai microcontroller dan LoRa EXP32 SX1278 sebagai komunikasi internal antara node yang lain. Central Unit mengirimkan data ke Internet menggunakan ESP8266. Eslam Al-Hassan, et al. [17] membangun Smart Power Socket untuk memonitoring konsumsi energi perangkat listrik rumah tangga menggunakan Arduino Nano dengan Xbee sebagai module komunikasi Zigbee. Shaojun Gan, et al. [18] membangun sistem monitoring konsumsi energi berbasis IoT menggunakan ABB B24 digital power meter sebagai pengukur konsumsi energi dan Raspberry Pi 2 sebagai controller serta LoRa sebagai komunikasi utama pada

sistem untuk lingkungan Industri. Lusi Susanti, et al. [19] mengembangkan sistem monitoring konsumsi energi listrik berbasis IoT menggunakan sensor arus SCT013 dan Wemos D1 sebagai central unit yang meneruskan data ke website application. Siriwat Wasoontarajoen, et al. [20] membangun perangkat IoT untuk memonitoring konsumsi energy listrik menggunakan sensor PZEM-004t yang mengukur tegangan dan arus, Arduino Uno sebagai controller dan ESP8266 sebagai perangkat pengirim data ke internet dan diterima melalui mobile application.

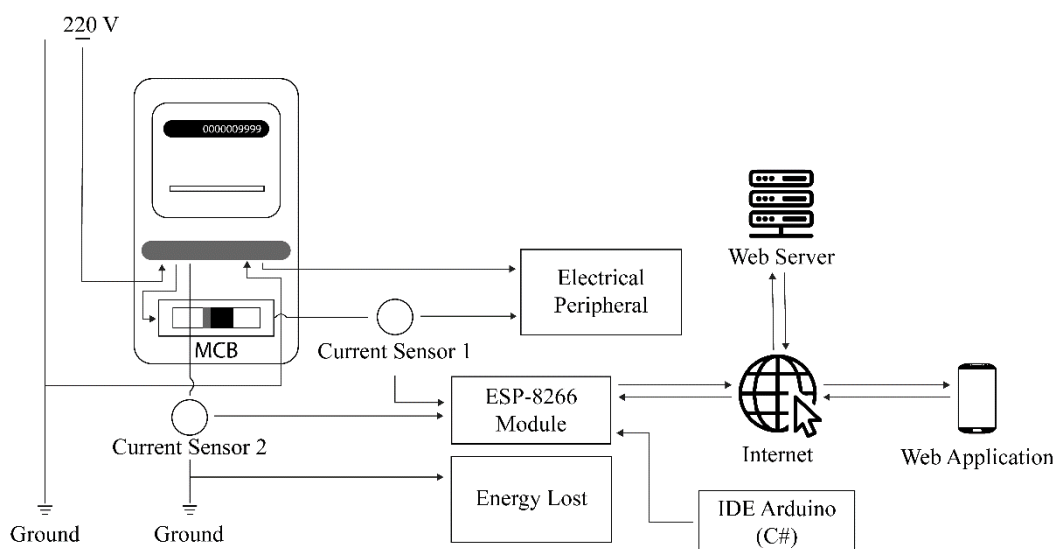
EMS yang telah dikembangkan hanya fokus pada monitoring konsumsi energi sedangkan lost energi juga merupakan salah satu factor yang perlu dilakukan monitoring untuk meningkatkan efektifitas penggunaan energi listrik. Maka dari itu, pada paper ini dikembangkan Sistem Monitoring Energi yang dapat memonitoring konsumsi dan lost energi listrik, setelah itu pengujian akurasi terhadap sensor power yang digunakan dengan membandingkan hasil pengukuran terhadap alat ukur listrik yang sudah tersedia di pasaran. Dengan diterapkannya Sistem Monitoring Energi yang telah dibangun, diharapkan pengguna dapat mengetahui jumlah konsumsi energi yang telah digunakan pada setiap waktunya dan mengetahui jumlah lost energi. Sistem ini dapat membantu pengguna untuk menganalisis hal yang menyebabkan lost energi diluar batasan normal yang dapat meningkatkan pembayaran listrik dan pemborosan energi yang berlebihan.

Pada bagian satu paper ini menjelaskan latar belakang dan penelitian terkait, bagian kedua membahas metode penelitian, bagian ketiga membahas hasil dan analisis dari penelitian yang telah dilakukan serta bagian keempat membahas kesimpulan penelitian ini.

5.3. Metode

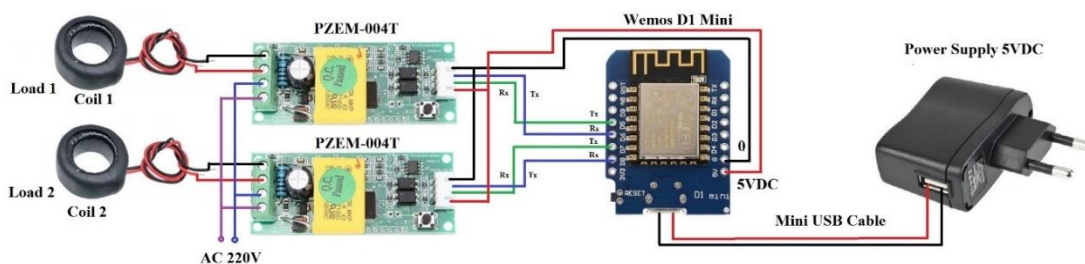
Arsitektur sistem yang dibangun dapat dilihat pada Gambar 5.1. Dua power sensor ditempatkan pada pengukur listrik yang satunya ditempatkan pada jalur MCB untuk menghitung konsumsi energi dan satunya lagi ditempatkan pada jalur ground untuk menghitung lost energy. Data dari

sensor akan diteruskan ke Central Unit untuk dilakukan proses pengolahan data yang selanjutnya dikirim ke Web Server sebagai pusat penyimpanan data sensor. Pengguna dapat mengakses data konsumsi dan lost energi melalui website application yang dapat diakses melalui Internet.



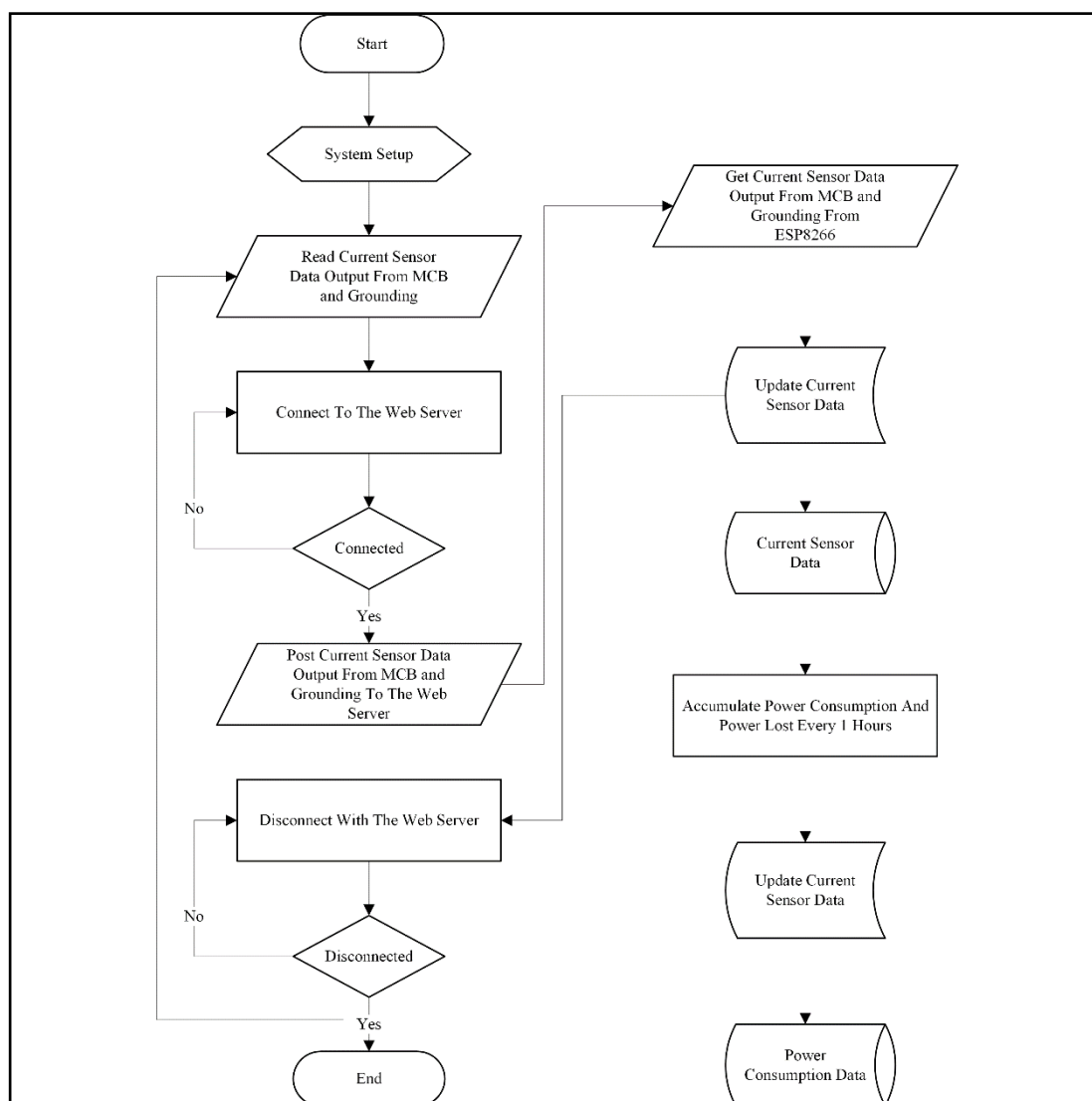
Gambar 5.1. Architecture System

Power sensor menggunakan PZEM-004T untuk mengukur tegangan, arus dan power [21], Wemos D1 Mini yang sudah termasuk dengan module ESP8266 [22] terpilih sebagai perangkat central unit yang menerima data dari sensor dan meneruskannya ke Web Server menggunakan komunikasi Wifi karena lebih murah dan compatible dibandingkan dengan ESP32 [23] Rangkaian komponen hardware yang digunakan dapat dilihat pada Gambar 5.2. dan cara kerja sistem yang dibangun dapat dilihat pada Gambar 5.3.



Gambar 5.2. Schematic Hardware

Energy monitoring system diimplementasikan pada rumah tinggal yang memiliki besaran tegangan sebesar 220V dengan arus sebesar 6A sehingga kapasitas daya yang dapat dicapai sebesar 1320W. Penelitian ini melakukan dua pengujian monitoring yaitu pengujian pada konsumsi energi dan lost energy.



Gambar 5.3. Flowchart of Design System

5.3.1. Pengujian energy consumption: PZEM-004T Sensor ditempatkan pada kabel yang berasal dari electrical meter untuk dapat mengetahui jumlah power, nilai tegangan dan nilai arus secara realtime dalam setiap waktunya sehingga dapat diketahui jumlah

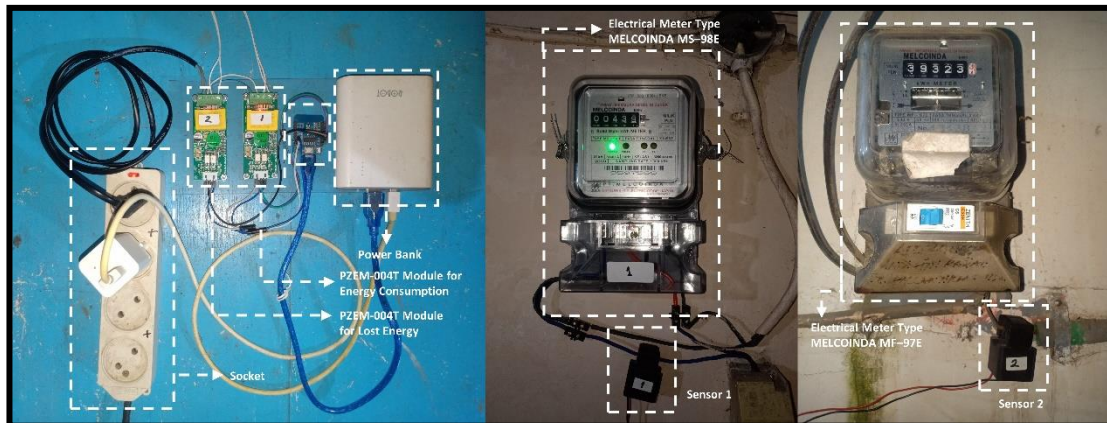
konsumsi energi yang digunakan. Validasi pengujian sensor juga dilakukan dengan membandingkan hasil pengukuran dari dua unit electrical meter dengan tipe MELCOINDA MS-98E (cite) dan MELCOINDA MF-97E (cite) menggunakan error rate equations untuk mengetahui akurasi PZEM-004T Sensor dalam menghitung jumlah power pada setiap jamnya (kWh).

$$Error Rate = \frac{Sensor Value - Electrical Meter Value}{Electrical Meter Value} \times 100\% \quad (1)$$

- 5.3.2. Pengujian lost energy: kami menempatkan PZEM-004 Sensor pada kabel yang menuju ground untuk menghitung jumlah arus yang terjadi secara realtime selanjutnya dikalikan dengan jumlah tegangan statis sebesar 220V sehingga didapatkan hasil jumlah power yang terbuang pada setiap jamnya (kWh).

5.4. Hasil dan Pembahasan

PZEM-004T module sensor dengan nomor 1 digunakan untuk menghitung jumlah konsumsi energi dan PZEM-004T module sensor dengan nomor 2 digunakan untuk menghitung jumlah energi yang terbuang. Kedua sensor terhubung dengan Wemos D1 sebagai Central Unit (CU) untuk mengolah data sensor yang akan dikirimkan ke Web Server menggunakan module ESP8266 yang tersemat pada CU. Sistem dilengkapi dengan energi cadangan menggunakan Power Bank untuk mengantisipasi terjadinya electrical down sehingga EMS dapat beroperasi sesuai dengan fungsinya. Prototype terhubung dengan dua unit electrical meter yang menjadi bahan validasi pengukuran sensor yang digunakan. Model prototype sistem yang telah dibangun dapat dilihat pada Gambar 5.4.

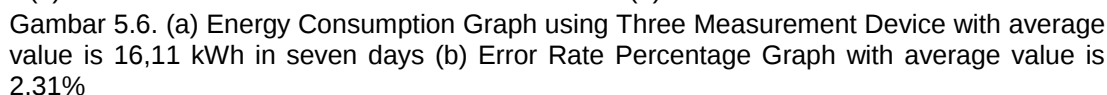
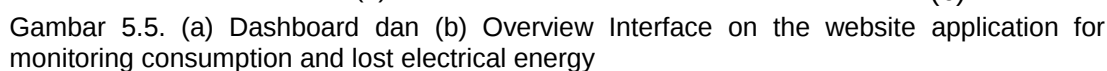


Gambar 5.4. Prototype System

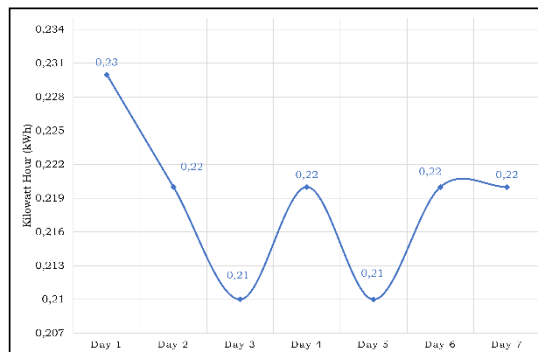
Pengguna dapat memantau jumlah konsumsi dan lost energy secara realtime dimanapun dan kapanpun melalui aplikasi website yang telah dibangun. Aplikasi ini dirancang dengan user interface yang sederhana dan data yang ditampilkan sesuai dengan kebutuhan sistem sehingga pengguna mudah untuk mengoperasikannya. Aplikasi monitoring memiliki dua menu yaitu menu Dashboard dan Overview.

- 5.4.1. Dashboard: tampilan ini menunjukkan nilai power, power factor, tegangan dan arus yang terpantau secara realtime. Nilai-nilai yang disebutkan dapat dilihat dari sisi jumlah konsumsi energi yang sedang digunakan dan lost energi yang telah terhitung. Dashboard juga dilengkapi dengan chart jumlah konsumsi dan lost energy disetiap waktunya.
- 5.4.2. Overview: nilai jumlah konsumsi dan lost energi secara detail mulai dari nilai secara realtime, nilai kemarin, nilai dari saat ini hingga bulan terakhir dan nilai penggunaan bulan terakhir ditampilkan pada menu Overview. Tampilan overview juga dilengkapi dengan chart yang menampilkan jumlah penggunaan pada setiap jam dan hari dari sisi konsumsi dan lost energi. Tampilan dashboard bagian (a) dan overview bagian (b) dapat dilihat pada Gambar 5.5.

Setelah dilakukan pengujian pada energy monitoring system selama tujuh hari, didapatkan hasil bahwa jumlah rata-rata konsumsi energi yang



Hasil pengujian dari pengukuran nilai energi listrik yang terbuang selama tujuh hari dengan menggunakan PZEM-004T module sensor, didapatkan nilai rata-rata sebesar 0.22 kWh. Gambar 5.7. menampilkan nilai lost energy pada setiap harinya.



Gambar 5.7. Lost Energy Consumption Graph using PZEM-004T module sensor

5.5. Kesimpulan

Penelitian ini telah berhasil mengembangkan sistem monitoring konsumsi energy dan lost energy menggunakan sensor PZEM-004T berbasis Internet of Things. Didapatkan hasil pengujian akurasi sensor dengan rata-rata error rate sebesar 2.31% dari hasil validasi pembacaan dua unit electrical meter tipe MELCOINDA MF-97E dan MELCOINDA MS-98E. Hal ini menandakan bahwa sistem monitoring ini memiliki akurasi kebenaran dalam mendeteksi konsumsi energi sebesar 97,69%. Untuk energi listrik yang terbuang, EMS mendeteksi nilai kWh rata-rata selama tujuh hari sebesar 0.22 kWh. Sistem ini sudah dapat diimplementasikan secara nyata sebagai alat monitoring konsumsi dan lost energi listrik yang fleksibel. Pada penelitian selanjutnya dapat dilakukan dengan menggunakan sensor power yang lain untuk meningkatkan akurasi pembacaan.

5.6. Daftar Pustaka

- [1] "International - U.S. Energy Information Administration (EIA)." <https://www.eia.gov/international/data/world> (accessed Jun. 02, 2021).
- [2] "Global electricity consumption continues to rise faster than population - Today in Energy - U.S. Energy Information Administration (EIA)." <https://www.eia.gov/todayinenergy/detail.php?id=44095> (accessed Jun. 02, 2021).
- [3] Syamsuddin B, "IoT Based Energy Monitoring System," 20210602.
- [4] "How to Read Your Electric Meter: An Energy Use Guide," *Chariot Energy*. <https://chariotenergy.com/chariot-university/read-your-electric-meter-energy-consumption-guide/> (accessed Jun. 02, 2021).
- [5] "What is an Electric Meter?," *Enertiv - Smart Building Technology & Submetering Solutions*. <https://www.enertiv.com/resources/faq/what-is-electric-meter> (accessed Jun. 02, 2021).
- [6] donnellymech, "5 Sources for Energy Loss in Commercial Buildings | Donnelly," *Donnelly Mechanical*, Jan. 24, 2017. <https://donnellymech.com/blog/5-common-sources-for-energy-loss-in-commercial-buildings/> (accessed Jun. 02, 2021).
- [7] J. Schonek, "How big are Power line losses?," *Schneider Electric Blog*, Mar. 25, 2013. <https://blog.se.com/energy-management-energy-efficiency/2013/03/25/how-big-are-power-line-losses/> (accessed Jun. 02, 2021).
- [8] "Lost In Transmission: How Much Electricity Disappears Between A Power Plant And Your Plug?," *Inside Energy*. <http://insideenergy.org/2015/11/06/lost-in-transmission-how-much-electricity-disappears-between-a-power-plant-and-your-plug/> (accessed Jun. 02, 2021).
- [9] E. Palantei *et al.*, "6 Monopole Elements Array Intelligent Antennas for IoT Based Environmental Surveillance Network," *EPI Int. J. Eng.*, vol. 3, no. 2, pp. 126–131, Jan. 2021, doi: 10.25042/epi-ije.082020.06.
- [10] "A real-time monitoring system based on ZigBee and 4G communications for photovoltaic generation," *CSEE J. Power Energy Syst.*, Mar. 2020, doi: 10.17775/CSEEJPES.2019.01610.
- [11] R. Morello, C. De Capua, G. Fulco, and S. C. Mukhopadhyay, "A Smart Power Meter to Monitor Energy Flow in Smart Grids: The Role of Advanced Sensing and IoT in the Electric Grid of the Future," *IEEE Sens. J.*, vol. 17, no. 23, pp. 7828–7837, Dec. 2017, doi: 10.1109/JSEN.2017.2760014.
- [12] L. Zhao, S. Qu, J. Zeng, and Q. Zhao, "Energy-Saving and Management of Telecom Operators' Remote Computer Rooms Using IoT Technology," *IEEE Access*, vol. 8, pp. 166197–166211, 2020, doi: 10.1109/ACCESS.2020.3022641.

- [13] H. Ko, S. Pack, and V. C. M. Leung, "Spatiotemporal Correlation-Based Environmental Monitoring System in Energy Harvesting Internet of Things (IoT)," *IEEE Trans. Ind. Inform.*, vol. 15, no. 5, pp. 2958–2968, May 2019, doi: 10.1109/TII.2018.2889778.
- [14] A. Hendra, E. Palantei, Syafaruddin, M. S. Hadis, N. Zulkarnaim, and M. F. Mansyur, "Wireless Sensor Network Implementation for IoT-Based Environmental Security Monitoring," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 875, p. 012093, Jul. 2020, doi: 10.1088/1757-899X/875/1/012093.
- [15] Muliadi, M. Y. Fahrezi, I. S. Areni, E. Palantei, and A. Achmad, "A Smart Home Energy Consumption Monitoring System Integrated with Internet Connection," in *2020 IEEE International Conference on Communication, Networks and Satellite (Comnetsat)*, Batam, Indonesia, Dec. 2020, pp. 75–80. doi: 10.1109/Comnetsat50391.2020.9328960.
- [16] S. H. M. S. Andrade, G. O. Contente, L. B. Rodrigues, L. X. Lima, N. L. Vijaykumar, and C. R. L. Frances, "A Smart Home Architecture for Smart Energy Consumption in a Residence With Multiple Users," *IEEE Access*, vol. 9, pp. 16807–16824, 2021, doi: 10.1109/ACCESS.2021.3051937.
- [17] E. Al-Hassan, H. Shareef, Md. M. Islam, A. Wahyudie, and A. A. Abdrabou, "Improved Smart Power Socket for Monitoring and Controlling Electrical Home Appliances," *IEEE Access*, vol. 6, pp. 49292–49305, 2018, doi: 10.1109/ACCESS.2018.2868788.
- [18] S. Gan, K. Li, Y. Wang, and C. Cameron, "IoT Based Energy Consumption Monitoring Platform for Industrial Processes," in *2018 UKACC 12th International Conference on Control (CONTROL)*, Sheffield, Sep. 2018, pp. 236–240. doi: 10.1109/CONTROL.2018.8516828.
- [19] L. Susanti, D. Fatrias, D. Ichwana, H. Kamil, and M. V. Putri, "A Configuration System for Real-Time Monitoring and Controlling Electricity Consumption Behavior," in *2018 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung - Padang, Indonesia, Oct. 2018, pp. 442–447. doi: 10.1109/ICITSI.2018.8696022.
- [20] S. Wasoontarajaroen, K. Pawasan, and V. Chamnanphrai, "Development of an IoT device for monitoring electrical energy consumption," in *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Phuket, Oct. 2017, pp. 1–4. doi: 10.1109/ICITEED.2017.8250475.
- [21] admin, "Get to know PZEM-004T Electronic Modules for Electrical Measurement Tools," *NN Digital | Learn Arduino, ESP8266 / NodeMCU, STM32, Raspberry Pi, Microcontroller and Other Information Technology*, Aug. 07, 2019. <https://www.nn-digital.com/en/blog/2019/08/07/get-to-know-pzem-004t-electronic-modules-for-electrical-measurement-tools/> (accessed Jun. 02, 2021).
- [22] "WeMos D1 Mini ESP8266 Development Board | Open ImpulseOpen Impulse." <https://www.openimpulse.com/blog/products-page/product->

- category/wemos-d1-mini-esp8266-development-board/ (accessed Jun. 02, 2021).
- [23] M. Rizal, M. S. Hadis, R. Angriawan, and A. Arifin, "Evaluasi Kinerja Bluetooth Pada Modul Esp32 Di Lingkungan Line Of Sight," *J. Embed. Syst. Secur. Intell. Syst.*, vol. 1, no. 1, pp. 42–47, 15052020.

BAB 6

SENSOR JAMAK IoT UNTUK SMART HOME

6.1. Abstrak

MULIADI, *Sensor Jamak IoT untuk Smart Home*

(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Pemanfaatan teknologi smarthome bertujuan untuk memberikan efisiensi dan efektifitas dalam penggunaannya namun pencapaian itu belum maksimal dikarenakan belum adanya sistem pendeteksi kesalahan dalam pengoperasiannya. Pada paper ini diusulkan sistem pendeteksi kesalahan pada pengoperasian teknologi smarthome yang diterapkan pada control sistem AC, Lampu dan electrical socket. Sistem akan secara otomatis memberikan notifikasi kepada pengguna terkait failure yang terjadi dan melakukan tindakan solusi awal terhadap failure yang ditemukan. Pengujian prinsip kerja sebanyak 10 kali pada setiap sistem dilakukan dan sistem dapat bekerja dengan baik sesuai dengan prinsip kerja juga pengujian response time dilakukan sebanyak 40 kali pengiriman data control kepada server dengan nilai rata-rata response time secara keseluruhan pada setiap sistem sebesar 635,1 ms.

Kata kunci: Failure Detection System, Smarthome, Internet of Things, Smart Automation, System

6.2. Pendahuluan

Listrik merupakan energi vital yang menunjang produktifitas kehidupan manusia sehari-hari. Semakin bertambahnya tahun, peningkatan pemanfaatan energi listrik juga ikut bertambah, hal ini dibuktikan dari data yang disajikan oleh U.S. Energy Information Administration pada tahun 2015 sampai dengan tahun 2018 mengalami peningkatan sekitar 10% di setiap tahunnya [1]. Di Indonesia, penggunaan energi listrik meningkat sebesar 4,42% pada periode Januari hingga September tahun 2021 dibandingkan pada tahun 2020 dengan periode bulan yang sama. Rumah tangga merupakan sektor pemanfaatan energi listrik terbesar di Indonesia dengan

nilai capaian konsumsi energi sebesar 85,43 TWh dengan presentase pangsa pasar sebesar 46% dari total konsumsi energi listrik [2]. Hal ini membuktikan bahwa pengelolaan konsumsi energi listrik di sektor rumah tangga khususnya di Indonesia adalah hal yang penting untuk dikaji pemanfaatannya agar lebih efektif dan efisien serta aman.

Pemanfaatan energi listrik yang tidak sesuai dengan prosedur keamanan dalam penggunaannya dapat menyebabkan bencana kepada penggunanya seperti kebakaran dan yang paling minimal adalah pemborosan energi yang menyebabkan tingginya tagihan listrik. Berdasarkan data yang disampaikan oleh The Electrical Safety Foundation International (ESFI), kebakaran rumah yang diakibatkan oleh penggunaan energi listrik sebanyak 51.000 kasus yang menyebabkan hampir lebih dari 500 kematian dan kerugian material sebesar 1,3 miliar USD Dollar setiap tahunnya [3]. Belum lagi pemborosan energi listrik di Indonesia disebabkan 80% dari kelalaian manusia [4]. Pemborosan energi listrik dapat disebabkan oleh sifat lupa mematikan lampu, barang elektronik yang selalu tertancap di stop kontak, charging ponsel semalaman, AC tetap aktif walaupun tidak ada orang atau juga AC tersebut tidak dapat dideteksi mengalami kerusakan atau tidak [5].

Beberapa solusi telah diterapkan untuk mengatasi masalah yang telah disampaikan, mulai dari aspek kesadaran manusia itu sendiri hingga pemanfaatan Internet of things sebagai alat yang mempermudah penyelesaian masalah pada kehidupan sehari-hari [6][7][8][9][10]. Pemanfaatan teknologi internet of things yang spesifik pada perumahan disebut sebagai smart home system [11][12][13]. Dengan smart home system, penggunaan energi akan dimonitoring oleh kecerdasan buatan sehingga dapat menentukan penggunaan energi yang efektif dan efisien untuk pengguna, seperti menyalakan lampu secara otomatis di sebuah ruangan jika ruangan tersebut ada penghuninya atau butuh cahaya dan menonaktifkan lampu jika penghuni sudah keluar ruangan atau ruangan tersebut sudah cukup cahaya. Penerapan teknologi ini bisa menghemat energi 30-40% [14]

Beberapa penelitian telah melakukan pengembangan smart home Hamdi W. Rotib, et al [15] membangun teknologi smart home pada kendali lampu jarak jauh melalui website application berbasis internet of things. Haris Isyanto, et al [16] mengembangkan teknologi smart home pada kendali lampu, tv, dan kipas angin berbasis perintah suara menggunakan Google Assistant yang ditujukan untuk memudahkan penyandang disabilitas dalam mengendalikan perangkat elektronik rumah tangganya. Mohamed S. Soliman, et al. [17] mengembangkan teknologi smart home pada kendali lampu, kipas dan CCTV secara otomatis berdasarkan nilai dari pembacaan sensor menggunakan microcontroller Arduino dan LabView sebagai platform aplikasi untuk menampilkan data monitoring sensor melalui PC. Chico Hermanu, et al [18] membangun sistem pengendalian lampu dengan dua mode, yaitu manual dan otomatis. Pada manual lampu di control menggunakan saklar fisik, dan saklar pada aplikasi tidak dapat mengontrol lampu sedangkan pada otomatis lampu di control melalui aplikasi dan saklar tidak berfungsi. Pengembangan smarthome dibangun menggunakan ESP8266. Teddy Surya Gunawan, et al [19] membangun prototipe smart home system dengan pemanfaatan teknologi Internet of Things. Sistem yang dibangun dapat memonitoring kadar gas dan nilai temperature ruangan serta status buka dan tutup pintu secara realtime. Sistem juga dapat mengontrol beberapa peralatan elektronik rumah tangga seperti lampu, penguncian pintu dan kipas melalui website application.

Penelitian yang telah dikerjakan belum memiliki sistem deteksi kesalahan sehingga jika sistem mengalami kegagalan dapat menyebabkan pemborosan energi bahkan sistem tidak dapat berfungsi sebagaimana mestinya.

Pada paper ini diusulkan sistem pendeteksi kesalahan pada teknologi smarthome sehingga pengguna dapat mengetahui jika penerapan smarthome yang digunakan mengalami kendala dan dapat segera diatasi. Misalkan AC dalam keadaan aktif namun suhu ruangan tidak mengalami perubahan, sistem dapat memberikan secara otomatis mematikan AC dan memberikan notifikasi kepada pengguna bahwa AC dalam keadaan rusak.

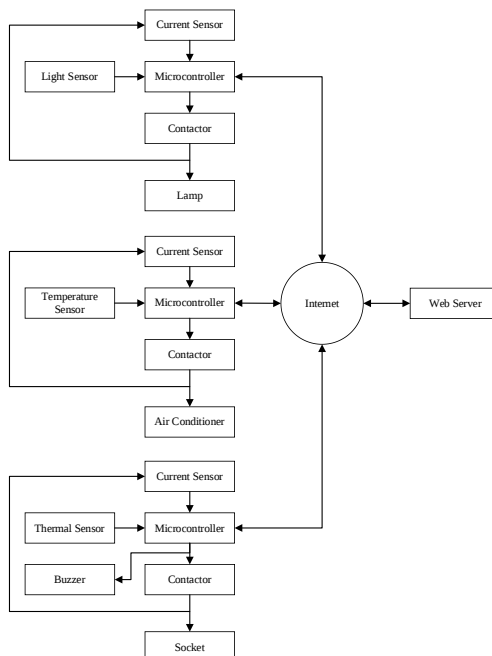
Dengan adanya sistem ini diharapkan pengguna dapat lebih mudah melakukan pemeliharaan hingga perawatan terhadap teknologi smarthome yang diimplementasikan agar tujuan dari penerapan teknologi tersebut dapat tercapai.

Pada bagian 1 paper ini menjelaskan latar belakang penelitian ini diusulkan, bagian kedua membahas metode penelitian yang dilakukan, bagian ketiga membahas hasil dan analisis dari penelitian yang telah dilakukan dan bagian keempat menyimpulkan hasil yang telah didapatkan dari penelitian ini.

6.3. Metode

6.3.1. Architecture System

Sistem dibangun menjadi tiga bagian yang terpisah yaitu bagian (1) sistem yang mengontrol dan memonitoring perangkat elektronik lampu, (2) untuk air conditioner (3) dan untuk electrical socket. Setiap sistem dilengkapi dengan failure alerts. Arsitektur sistem digambarkan pada Gambar 6.1.



Gambar 6.1. Architecture System

6.3.2. Hardware and Software

Setiap sistem menggunakan perangkat masing-masing satu EPS8266 module, satu Relay 250V/10A dilengkapi dengan socket, satu PZEM-004T sensor, power supply 5V2A dan satu electrical socket dengan 1 slot untuk sistem control lampu dan ac dan 2 slot untuk sistem control electrical socket. EPS8266 berfungsi sebagai control processing unit yang dimana modul ini sudah dilengkapi dengan antena WiFi untuk mengirim dan menerima data dari Internet. Relay 250V/10A berfungsi untuk menghubungkan dan memutus arus listrik, PZEM-004T sensor berfungsi untuk memonitoring nilai arus yang ada pada masing-masing sistem, power supply 5V2A untuk memberikan arus listrik DC kepada perangkat elektronik yang ada pada sistem dan electrical socket sebagai terminal power listrik AC. Sistem pengontrol lampu ruangan dilengkapi dengan Light Dependent Resistor (LDR) sensor untuk mengambil nilai cahaya disekitar, Sistem Air conditioner dilengkapi dengan temperature sensor dengan tipe DHT-22 untuk memonitoring nilai temperature, dan sistem control electrical outlet dilengkapi thermal sensor dengan tipe LM35 untuk mengetahui nilai temperature pada kabel dan buzzer sebagai alarm jika panas pada kabel sudah berada diatas ambang batas. Software yang digunakan untuk menyusun program dan memasukkannya ke EPS8266 adalah Arduino IDE dan Firebase sebagai pengolah data yang ada pada server.

6.3.3. Workflow

Untuk cara kerja kami membagi menjadi tiga bagian yang dijelaskan dibawah ini

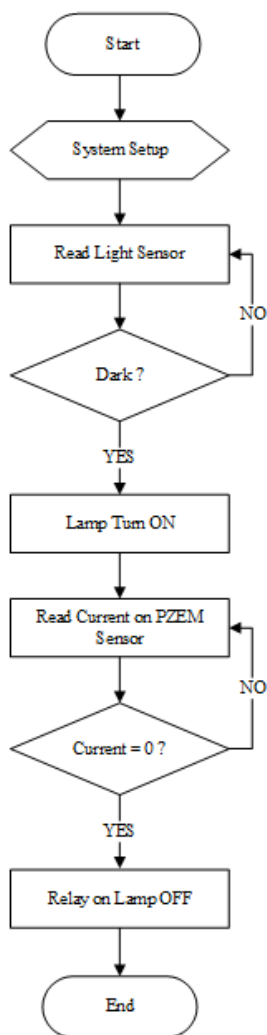
6.3.3.1. Sistem kontrol lampu

Sistem membaca nilai dari LDR sensor, jika nilai sensor menunjukkan daerah sekitar berada dalam kegelapan maka sistem akan melakukan pengontrolan secara otomatis untuk menghubungkan arus listrik melalui relay yang terhubung dengan lampu sehingga lampu dalam keadaan menyala namun dalam jangka waktu satu menit PZEM-004T sensor membaca nilai arus listrik pada lampu masih tetap bernilai 0 maka sistem akan memutuskan arus listrik pada relay yang menandakan bahwa lampu

dalam keadaan rusak. Untuk lebih jelasnya alur cara kerja sistem kontrol lampu dapat dilihat pada Gambar 6.2.

6.3.3.2. Sistem kontrol AC

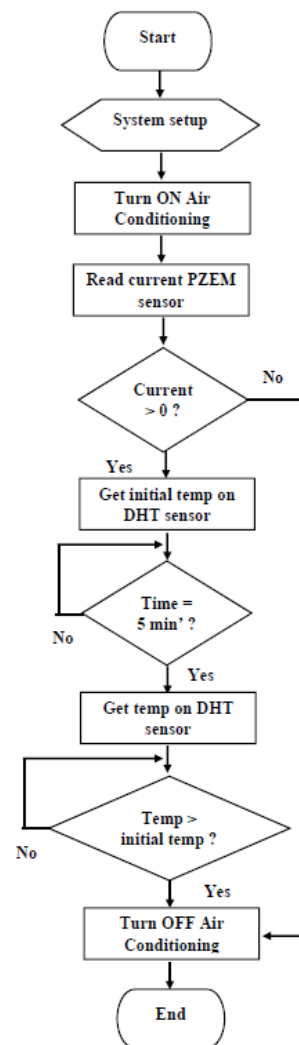
Pada saat pengguna mengaktifkan AC maka PZEM-004T sensor akan membaca nilai arus yang terhubung pada relay ke AC, jika nilai arus tidak sama dengan 0 maka sistem akan lanjut membaca nilai DHT-22 selanjutnya nilai tersebut akan disimpan sebagai data temperatur awal, namun jika arus bernilai 0 maka sistem secara otomatis akan menon-aktifkan AC dan ini digambarkan sebagai kondisi pertama AC dalam keadaan rusak. Selanjutnya setelah 5 menit DHT-22 sensor akan membaca lagi nilai suhu disekitar, jika nilai suhu awal lebih besar dibandingkan nilai suhu setelah 5 menit maka sistem tetap mengaktifkan AC karena AC dalam keadaan normal namun jika suhu awal lebih kecil dibandingkan dengan nilai suhu sekarang maka sistem akan menon-aktifkan AC yang berarti AC dalam keadaan rusak. Skema alur cara kerja sistem kontrol AC dapat dilihat pada Gambar 6.3.



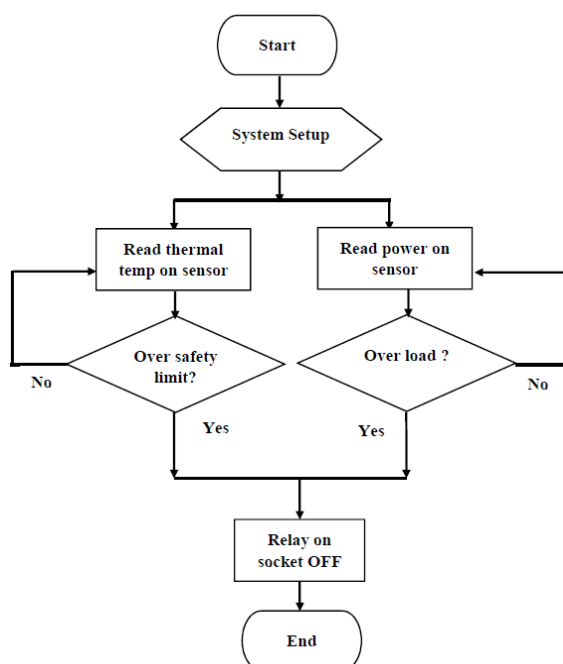
Gambar 6.2. Algorithm Flow of Smart Lamp Control

6.3.3.3. Sistem kontrol electrical socket

Sistem membaca LM35 sensor untuk mengetahui suhu kabel dalam keadaan aman atau tidak jika aman maka sistem akan tetap mengalirkan arus listrik pada socket tetapi jika suhu telah mencapai ambang batas aman maka sistem akan secara otomatis akan memutus arus listrik pada socket. Sistem pengontrolan juga membaca nilai arus pada socket menggunakan PZEM-004T jika nilai arus sudah mencapai ambang batas arus maksimal yang aman sistem akan memberikan notifikasi kepada pengguna untuk segera mengurangi beban arus dalam waktu satu menit namun jika beban arus tidak berkurang selama waktu yang telah ditentukan maka sistem secara otomatis akan memutus arus pada electrical socket. Gambar 6.4. menggambarkan alur kerja dari sistem control electrical socket.



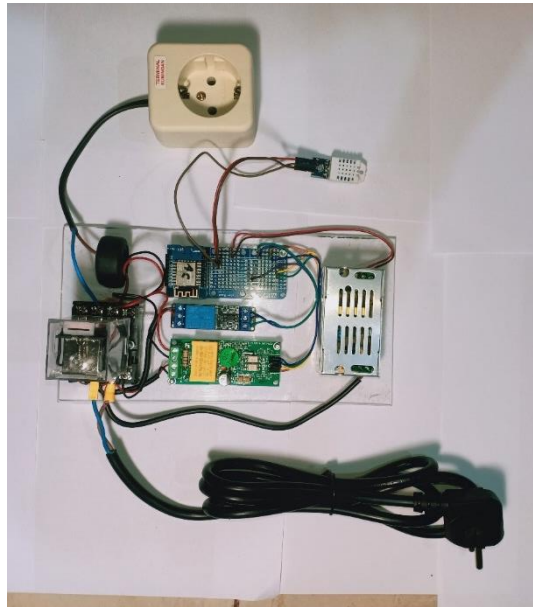
Gambar 6.3. Algorithm Flow of Smart AC Control



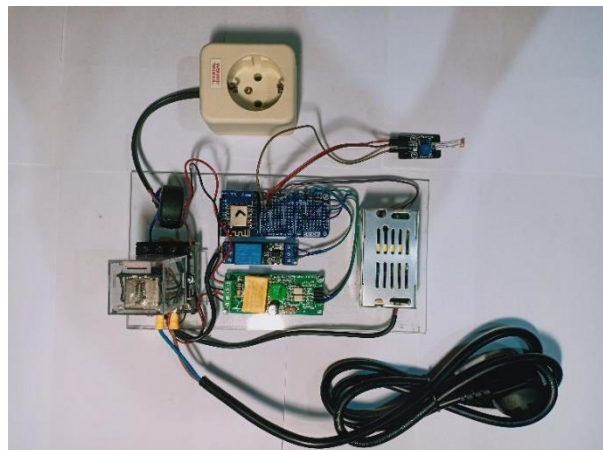
Gambar 6.4. Algorithm Flow of Smart Electrical Socket Control

6.4. Hasil dan Pembahasan

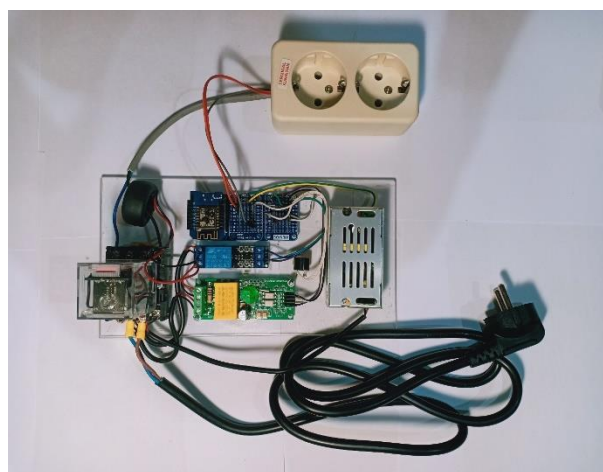
Prototype dari Smart Air Conditioner Control, Smart Light control, dan Smart electrical socket memiliki kesamaan model rangkaian dan komponen yang digunakan hanya saja yang membedakan adalah satu unit sensor dan jumlah electrical socket yang berbeda. Masing-masing rangkaian menggunakan satu unit microcontroller EPS8266 yang telah dilengkapi dengan module WiFi, satu unit sensor arus menggunakan PZEM-004T, satu unit power supply yang memiliki tegangan 5 Volt dengan nilai arus sebesar 2A, dan satu unit contractor. Untuk perbedaannya pada rangkaian sistem control AC ditambah satu unit sensor DHT22, sistem control lampu ditambah satu unit sensor LDR dan masing-masing sistem memiliki satu slot electrical socket, sedangkan untuk sistem control electrical socket ditambah dengan satu unit sensor LM35 dengan dua slot electrical socket. Gambar 6.5. adalah hasil rangkaian prototype untuk sistem control AC, Gambar 6.6. untuk sistem control Lampu dan Gambar 6.7. untuk sistem control electrical socket.



Gambar 6.5. Prototype of Air Conditioner Controller



Gambar 6.6. Prototype of Light Controller



Gambar 6.7. Prototype of Electrical Socket Controller

Prinsip kerja dari ketiga sistem telah diuji sebanyak 10 kali yang dimana hasil pengujian dijabarkan pada Table 6.1.

TABLE 6.1. TESTING OF PRINCIPLE WORK

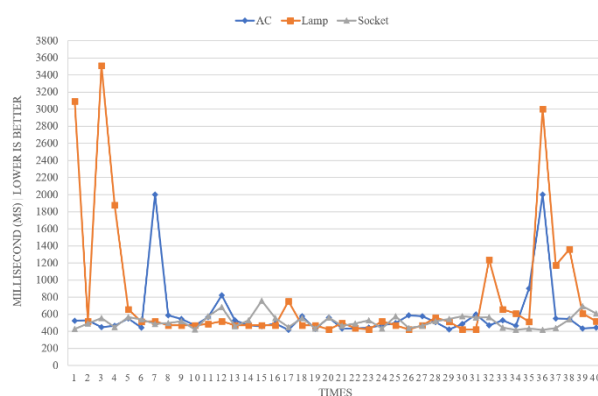
System	Testing Description	Result
AC	Nilai arus = 0 maka AC dinon-aktifkan secara otomatis	Success
	Nilai arus > 0 maka AC tetap diizinkan aktif oleh sistem	Success
	Nilai suhu awal $\leq$ Nilai suhu setelah 5 menit AC aktif maka AC dinon-aktifkan secara otomatis	Success
	Nilai suhu awal > Nilai suhu setelah 5 menit AC aktif maka AC tetap diizinkan aktif oleh sistem	Success
Light	Selama satu menit lampu diaktifkan namun nilai arus = 0 maka Contractor memutuskan arus listrik	Success
	Selama satu menit lampu diaktifkan namun nilai arus > 0 maka Contractor tetap	Success

	menghubungkan arus listrik	
	Nilai LDR Sensor >200 artinya terang	Success
	Nilai LDR Sensor <200 artinya gelap	Success
Socket	Nilai suhu kabel >70° maka Contractor memutuskan arus pada socket	Success
	Nilai suhu kabel <70° maka Contractor tetap menghubungkan arus pada socket	Success
	Nilai arus > 2A maka sistem memberikan notifikasi kepada pengguna selama satu menit	Success
	Setelah satu menit notifikasi namun nilai arus > 2A maka contractor memutus arus pada socket	Success
	Setelah satu menit notifikasi dan nilai arus < 2A maka contractor tetap menghubungkan arus pada	Success

	socket	
	Nilai arus < 2A maka contractor menghubungkan arus pada socket	Success

Ketiga sistem juga telah diuji sebanyak 40 kali pengiriman instruksi control secara manual dan mendapatkan hasil packet loss adalah 0% yang artinya tidak ada data yang tidak berhasil dikirim dan untuk hasil pengujian response time dijabarkan pada Gambar 6.8

Pengujian response time telah dilakukan dan diperoleh hasil response time tercepat pada control socket adalah 415 millisecond (ms), control AC 417 ms, dan control lampu 422 ms dan rata-rata response time yang didapatkan untuk control socket adalah 516.4 ms, control AC 591.4 ms dan control lampu 797.5 ms. Response time sangat dipengaruhi oleh traffic data dan kekuatan sinyal seluler yang digunakan pada saat pengujian, response time paling lambat didapatkan pada control lampu dengan nilai 3509 ms.



Gambar 6.8. Result of Response Time

6.5. Kesimpulan

Sistem pendeteksi kesalahan pada teknologi smarthome yang terdiri dari sistem monitoring dan pengontrolan pada AC, Lampu dan Socket

berhasil dibangun dan hasil pengujian prinsip kerja sebanyak 10 kali berhasil beroperasi sesuai dengan prinsip kerja yang telah dirancang sehingga pengguna dapat mengetahui kesalahan yang terjadi pada teknologi smarthome yang digunakan dan bagian apa yang perlu ditangani. Uji Quality of Service dari sisi response time telah dilakukan sebanyak 40 kali di masing-masing sistem dan mendapatkan rata-rata response time sebesar 516.4 ms pada Socket system, 591.4 ms pada sistem AC dan pada sistem lampu sebesar 797.5 ms dengan nilai rata-rata response time secara keseluruhan sebesar 635.1 ms.

6.6. Daftar Pustaka

- [1] "International - U.S. Energy Information Administration (EIA)."
<https://www.eia.gov/international/data/world/electricity/> (accessed Nov. 16, 2021).
- [2] "Konsumsi Listrik Capai 187,78 TWh per September 2021."
<https://www.cnnindonesia.com/ekonomi/20211017103219-85-708832/konsumsi-listrik-capai-18778-twh-per-september-2021>
(accessed Nov. 16, 2021).
- [3] "Home Electrical Fires - Electrical Safety Foundation International."
<https://www.esfi.org/home-electrical-fires/> (accessed Nov. 16, 2021).
- [4] O. A. Yola, "Studi Perilaku Konsumsi Energi Listrik Di Universitas Andalas," Jan. 2020.
- [5] "7 Pemborosan Energi Terbesar yang Biasa Dilakukan di Rumah."
<https://www.idntimes.com/life/inspiration/stella/pemborosan-energi-terbesar-yang-dilakukan-orang/7> (accessed Nov. 16, 2021).
- [6] Pasnur, A. Hendra, M. S. Hadis, and Ardiansa, "Sistem Kontrol Suhu Ideal Kandang Ayam Broiler Berbasis Internet of Things," in *Prosiding Seminar Nasional Komunikasi dan Informatika #3*, 2019, pp. 79–82, [Online]. Available: <https://jurnal-dev.kominfo.go.id/index.php/snki/article/view/2553>.
- [7] A. Hendra, E. Palantei, Syafaruddin, M. S. Hadis, N. Zulkarnaim, and M. F. Mansyur, "Wireless Sensor Network Implementation for IoT-Based Environmental Security Monitoring," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 875, no. 1, p. 012093, Jun. 2020, doi: 10.1088/1757-899X/875/1/012093.
- [8] E. Palantei et al., "6 Monopole Elements Array Intelligent Antennas for IoT Based Environmental Surveillance Network," *EPI Int. J. Eng.*, vol. 3, no. 2, pp. 126–131, Jan. 2021, doi: 10.25042/epi-ije.082020.06.
- [9] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of Things applications: A systematic review," *Comput. Networks*, vol. 148, pp.

- 241–261, Jan. 2019, doi: 10.1016/j.comnet.2018.12.008.
- [10] B. Petersen, H. Bindner, B. Poulsen, and S. You, “Smart Grid Communication Infrastructure Comparison—For Distributed Control of Distributed Energy Resources Using Internet of Things Devices,” *Int. J. Electr. Electron. Eng. Telecommun.*, pp. 7–14, 2018, doi: 10.18178/ijeetc.7.1.7-14.
 - [11] D. D. Furszyfer Del Rio, B. K. Sovacool, N. Bergman, and K. E. Makuch, “Critically reviewing smart home technology applications and business models in Europe,” *Energy Policy*, vol. 144, p. 111631, Sep. 2020, doi: 10.1016/j.enpol.2020.111631.
 - [12] R. Ford, M. Pritoni, A. Sanguinetti, and B. Karlin, “Categories and functionality of smart home technology for energy management,” *Build. Environ.*, vol. 123, pp. 543–554, Oct. 2017, doi: 10.1016/j.buildenv.2017.07.020.
 - [13] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, “A review of smart home applications based on Internet of Things,” *J. Netw. Comput. Appl.*, vol. 97, pp. 48–65, Nov. 2017, doi: 10.1016/j.jnca.2017.08.017.
 - [14] “How smart homes save energy.” <https://www.smart-energy.com/features-analysis/how-smart-homes-save-energy/> (accessed Nov. 16, 2021).
 - [15] H. W. Rotib, M. B. Nappu, Z. Tahir, A. Arief, and M. Y. A. Shiddiq, “Electric Load Forecasting for Internet of Things Smart Home Using Hybrid PCA and ARIMA Algorithm,” *Int. J. Electr. Electron. Eng. Telecommun.*, pp. 425–430, 2021, doi: 10.18178/ijeetc.10.6.425-430.
 - [16] H. Isyanto, A. S. Arifin, and M. Suryanegara, “Design and Implementation of IoT-Based Smart Home Voice Commands for disabled people using Google Assistant,” in *2020 International Conference on Smart Technology and Applications (ICoSTA)*, Feb. 2020, pp. 1–6, doi: 10.1109/ICoSTA48221.2020.1570613925.
 - [17] M. S. Soliman, A. A. Alahmadi, A. A. Maash, and M. O. Elhabib, “Design and Implementation of a Real-Time Smart Home Automation System Based on Arduino Microcontroller Kit and LabVIEW Platform,” *Int. J. Appl. Eng. Res.*, vol. 12, no. 18, pp. 7259–7264, 2017, [Online]. Available: http://www.ripublication.com/ijaer17/ijaerv12n18_27.pdf.
 - [18] C. Hermanu, H. Magfiroh, H. P. Santoso, Z. Arifin, and C. Harsito, “Dual Mode System of Smart Home Based on Internet of Things,” *J. Robot. Control*, vol. 3, no. 1, pp. 26–31, 2022, doi: 10.18196/jrc.31147/.
 - [19] T. S. Gunawan *et al.*, “Prototype Design of Smart Home System using Internet of Things,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 7, no. 1, p. 107, Jul. 2017, doi: 10.11591/ijeecs.v7.i1.pp107-115.

BAB 7

Sistem Pemantauan dan Pengendalian Terintegrasi untuk Pengoperasian Perangkat Kelistrikan IoT

7.1. Abstrak

MULIADI, *Sistem Pemantauan dan Pengendalian Terintegrasi untuk Pengoperasian Perangkat Kelistrikan IoT*

(dibimbing oleh: Andani Achmad, Intan Sari Areni dan Elyas Palantei)

Penelitian tentang Teknik Fault detection and diagnostics (FDD) dapat digunakan untuk memantau sistem kelistrikan bangunan serta mendeteksi dan mendiagnosa kerusakan peralatan kelistrikan. (Hannan et al., 2018). FDD adalah proses secara otomatis mengetahui dan mengisolasi kesalahan dalam bangunan untuk perlindungan sistem dari kerusakan. Beberapa aplikasi FDD dari bangunan dikembangkan dan diteliti berdasarkan hubungan antara suhu, tekanan, dan termodinamika untuk deteksi dan diagnosis kesalahan. Penelitian dalam disertasi ini menyajikan tentang Sistem Pemantauan dan Pengendalian Pengoperasian Perangkat Kelistrikan Berbasis Internet of Things (IoT) secara terpusat. Beberapa Perangkat yang dirancang dan dikembangkan dengan pemodelan sedemikian rupa agar memudahkan dalam pengoperasian, bersifat reconfigurable dan dimanis. Pemantauan, pengontrolan, pendeteksian kerusakan pada perangkat kelistrikan dapat diakses pada sebuah sistem yang terpusat dan terintegrasi. Pengujian prinsip kerja sistem sebanyak 10 kali pada setiap sistem dilakukan dan sistem dapat bekerja dengan baik sesuai dengan prinsip kerja. Pengujian respon time dilakukan sebanyak 25 kali pengiriman data control ke server dengan nilai rata-rata respon time secara keseluruhan sebesar 15 detik. Setelah melakukan pengujian Aplikasi android MACSys menggunakan Firebase Test Lab menggunakan device virtual dengan spesifikasi pixel 5e dan API level 30, orientasi portrait dan dinyatakan lulus/berhasil uji dan dapat berjalan dengan baik pada perangkat Samsung, vivo dan redmi.

Kata kunci: internet of things, Failure Detection system, fire base test lab, automation system, system

7.2. Pendahuluan

Pembangunan ekonomi berkelanjutan sangat dipengaruhi oleh konsep ketahanan energi (Harkouss et al., 2018). Badan Energi Internasional memperkirakan permintaan listrik global akan mengalami peningkatan secara signifikan pada tahun 2035 (Hannan et al., 2018).

Peningkatan efisiensi energi dapat mengurangi permintaan kelistrikan Negara, sehingga memberikan penghematan moneter, gas rumah kaca dan pengurangan polutan lainnya, dan peningkatan keamanan energi. Penggunaan energi secara efisien dapat menurunkan permintaan listrik dan mengurangi resiko kekurangan listrik yang berdampak pada pertumbuhan ekonomi. Permintaan energi puncak di Indonesia diperkirakan mencapai tiga kali lipat antara tahun 2020 dan 2030, untuk konsumsi pengkondisian udara (Air Conditioner) (McAllister et al., 2017)

PT. PLN (Persero) merupakan satu-satunya Badan Usaha Milik Negara yang bergerak di bidang penyediaan sampai penyaluran jasa tenaga listrik. Seiring berkembangnya teknologi, industri dan meningkatnya kebutuhan akan energi listrik, maka dibutuhkan pasokan energi listrik dan penyaluran yang andal. Keandalan akan pasokan energi listrik adalah merupakan kepuasan pelanggan. PT. PLN (Persero) menjaga keandalan sistem penyaluran tenaga listrik menggunakan sistem pengoperasian yang mempunyai tingkat keandalan tinggi, dikarenakan PT. PLN (Persero) mempunyai visi diakui sebagai perusahaan kelas dunia yang dinilai dari SAIDI (System Average Interruption Duration Index) dan SAIFI (System Average Interruption Frequency Index). Untuk itu diperlukan sistem operasi yang mempunyai keandalan baik, sehingga dapat meminimalisir pemadaman untuk menjaga kepuasan pelanggan (Soleh, 2014). Selain perbaikan dari pihak PLN, perlu pula adanya perbaikan dan penghematan disisi pengguna atau pelanggan PLN dalam menggunakan listrik sebaik dan seefisien mungkin.

Penggunaan listrik secara umum masih didominasi oleh sektor industri dan gedung perkantoran secara global (Jain et al., 2016). Penggunaan peralatan kelistrikan pada gedung umumnya belum memiliki

sistem manajemen energi yang pemakaiannya dapat dipantau, begitupun pengendalian peralatan listrik yang masih bersifat konvensional. Kekurangan sistem yang ada saat ini adalah terkadang peralatan-peralatan listrik lupa dinonaktifkan padahal sudah tidak digunakan. Manajemen energi dalam gedung telah menjadi tujuan internasional teknologi modern (Li et al., 2013). Penelitian menunjukkan bahwa sekitar 40% dari total energi di dunia dikonsumsi oleh gedung, yang merupakan sepertiga dari emisi Greenhouse Gas (GHG) (Huang et al., 2017) (Mytafides et al., 2017). Penelitian lain menunjukkan bahwa 49% dari total energi dikonsumsi oleh gedung pada tahun 2014 (Wu et al., 2020) dan 60% dari konsumsi energi gedung karena tujuan pemanasan dan pendinginan (Lizana et al., 2017). Karena itu, konsumsi energi dan pengaruhnya terhadap perubahan iklim paling banyak menjadi masalah yang perlu diselesaikan di sektor pemanfaatan energi pada gedung (Santamouris, 2016) (Sun et al., 2018) (Cao et al., 2016).

Bangunan modern terdiri dari berbagai peralatan seperti pemanas, ventilasi dan pendingin udara (HAVAC) dan penerangan. Setiap peralatan terdiri dari beberapa komponen seperti pompa, kipas, saluran, sensor, lampu, kabel dan lain-lain. Semua peralatan dapat dipantau dan dikelola oleh sistem manajemen gedung (BMS). Semua komponen ini dapat mengalami kerusakan yang dapat disebabkan karena aus karena pemakaian dalam waktu yang lama, kerusakan konfigurasi, dan masalah komunikasi. Kerusakan peralatan mempengaruhi biaya perawatan dan khususnya efisiensi energi (Mills, 2011)(Kim et al., 2018). Building Energy Management System (BEMS) dengan menggunakan Internet of Energy (IoE) dapat memaksimalkan efisiensi energi dengan meminimalkan kerugian dan dampak lingkungan (Hannan et al., 2018).

Beban peralatan seperti Heating, Ventilation, and Air-Conditioning (HVAC) bervariasi tergantung pada jenis dan lokasi bangunan, tetapi kerusakan pada salah satu peralatan dapat membuat hingga 50% dari total konsumsi energi sehingga dapat menyebabkan kehilangan energi yang besar (Kim et al., 2018)(Pe, 2008). Penelitian menunjukkan bahwa penghematan energi antara 20% hingga 30% dapat dicapai dengan cara re-

commissioning malfungsi sistem HVAC (Agency, 2005). Sistem HVAC sering disesuaikan untuk bangunan spesifik sehingga kurangnya integrasi sistem yang berkualitas (Yu, Woradechjumroen and Yu, 2014).

Telah banyak penelitian yang dilakukan untuk mencapai tujuan efisiensi energi. Termasuk pada penelitian tentang sensor array dengan keuntungan secara simultan mendeteksi beberapa parameter yang berbeda digunakan secara luas dalam banyak aplikasi teknik seperti teknik lingkungan, teknik kimia, teknik biologi, dan teknik aerospace (Chen et al., 2016) Sebagai sumber perolehan informasi, keakuratan dan keandalan output array sensor sangat penting untuk kinerja sistem pengukuran dan kontrol.

Pada dasarnya sistem yang dibangun dapat memonitoring energi yang digunakan oleh peralatan kelistrikan seperti AC, peralatan pencahayaan, dan peralatan kelistrikan yang terkoneksi dengan stok kontak yang ada seperti komputer PC atau adaptor Laptop. Pengontrolan juga dengan mudah dapat dilakukan karena system terkoneksi dengan jaringan internet atau terintegrasi dengan Internet of Things (IoT). Pengontrolan dilakukan menggunakan smartphone dengan aplikasi android. Sistem ini juga dilengkapi dengan peringatan dini terhadap gangguan yang terjadi sehingga teknisi dengan cepat dapat mengetahui jika ada gangguan yang terjadi pada sistem kelistrikan pada suatu gedung

Selanjutnya penelitian tentang Teknik Fault detection and diagnostics (FDD) dapat digunakan untuk memantau sistem kelistrikan bangunan serta mendeteksi dan mendiagnosa kerusakan peralatan kelistrikan. (Hannan et al., 2018). FDD adalah proses secara otomatis mengetahui dan mengisolasi kesalahan dalam bangunan untuk perlindungan sistem dari kerusakan. Beberapa aplikasi FDD dari bangunan dikembangkan dan diteliti berdasarkan hubungan antara suhu, tekanan, dan termodinamika untuk deteksi dan diagnosis kesalahan

Selain penghematan dari penggunaan energi listrik, Hal yang perlu dan dibutuhkan suatu system kelistrikan yaitu pengamanan yang handal dan mampu mendeteksi gangguan dan kesalahan yang terjadi pada sistem

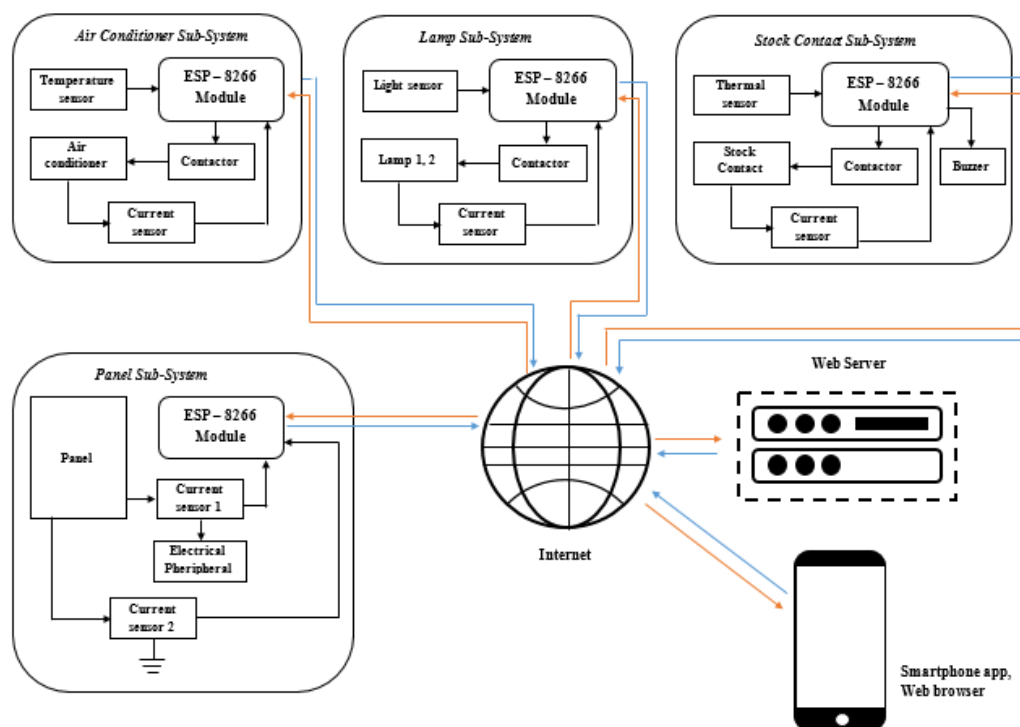
tersebut. Hal tersebut diperlukan agar suplai energi listrik dapat dipertahankan dengan baik. Agar sistem yang dimiliki mampu menjaga pelayanan suplai tenaga listrik dengan baik, maka sistem pengaman harus memenuhi syarat reliable, selektif, sensitif, memiliki waktu operasi yang cepat, ekonomis dan sederhana.

Penelitian dalam disertasi ini menyajikan tentang Sistem Pemantauan dan Pengendalian Pengoperasian Perangkat Kelistrikan Berbasis Internet of Things (IoT) secara terpusat. Beberapa Perangkat jenis yang dirancang dan dikembangkan dengan pemodelan sedemikain rupa agar memudahkan dalam pengoperasian, bersifat reconfigurable dan dimanis. Pemantauan, pengontrolan, pendeteksian kerusakan pada perangkat kelistrikan dapat di akses pada sebuah sistem yang terpusat dan terintegrasi.

7.3. Metode

7.3.1. Architecture System

Sistem dibangun menjadi empat bagian yang terpisah yaitu bagian (1) sistem yang memonitoring konsumsi energi pada panel listrik dinamakan Panel Sub-system, (2) sistem yang memonitoring dan mengontrol pengoperasian Air Conditioner (AC) dinamakan Air Conditioner Sub-System dilengkapi dengan pendeteki kerusakan dan proteksi, (3) sistem monitoring dan pengontrolan lampu secara manual dan otomatis. Sistem ini dinamakan Lamp Sub-System, (4) sistem monitoring dan pengontrolan stok kontak yang dilengkapi dengan pendeteksi kelebihan suhu dan kelebihan beban dan proteksi. Sistem ini dinamakan Stok Kontak Sub-System. Arsitektur sistem digambarkan pada gambar 7.1



Gambar 7.1. Arsitektur sistem (MACSys)

7.3.2. Hardware dan Software

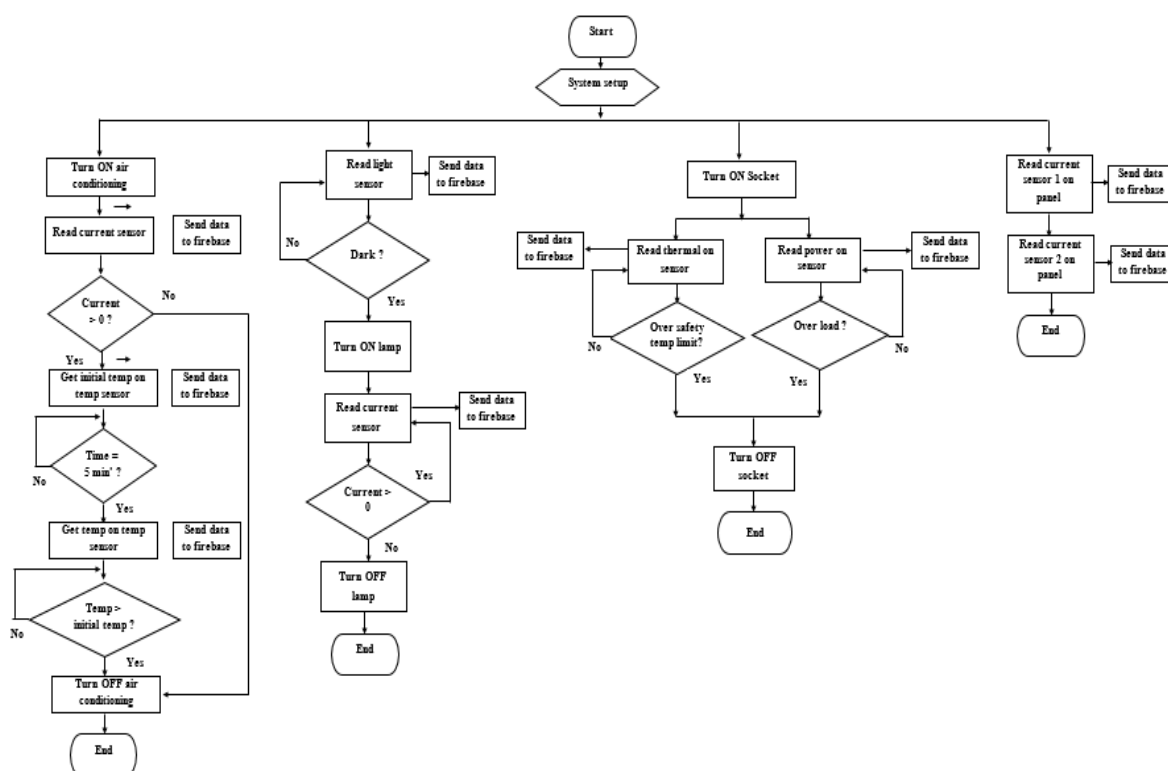
Air Conditioner Sub-System menggunakan perangkat yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, sensor suhu DHT-22, power supply 5V/2A dilengkapi dengan kabel AC, Relay 1 channel opto 5V, switch on/off dan stok kontak. *Lamp Sub-Sys* menggunakan perangkat yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, sensor cahaya LDR, power supply 5V/2A dilengkapi dengan kabel AC, Relay 1 channel opto 5V dua buah, switch on/off dan stok kontak dua buah. *Electrichal Socket Sub-System* menggunakan perangkat yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, sensor panas DS18B20, power supply 5V/2A dilengkapi dengan kabel AC, Relay 1 channel opto 5V, switch on/off dan stok kontak. *Panel Sub-System* menggunakan perangkat yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, power supply 5V/2A dan satu stok kontak.

Software yang digunakan untuk menyusun program dan memasukkannya ke mikrokontroler adalah arduino IDE, firebase sebagai

pengolah data yang ada pada server, web server untuk menyimpan data hasil monitoring dan kodular digunakan untuk membuat aplikasi android.

7.3.3. Workflow

Cara kerja sistem monitoring dan pengontrolan (MACSys) sesuai dengan flowchart sistem seperti pada gambar 7.2.



Gambar 7.2. Flowchart monitoring and Controlling system (MACSys)

Berdasarkan flowchart tersebut cara kerja sistem dibagi menjadi 4 bagian yang dijelaskan sebagai berikut:

7.3.3.1. Air Conditioner Sub-System (Modul AC)

Pada saat pengguna mengaktifkan AC maka sensor PZEM-004T akan membaca nilai arus yang terhubung pada relay ke AC dan memberikan informasi pada bagian *warning*. Jika arus bernilai nol maka sistem secara otomatis akan menon-aktifkan AC. jika nilai arus tidak sama dengan nol maka sistem akan lanjut membaca nilai DHT-22, selanjutnya nilai tersebut

disimpan sebagai data temperatur awal. Selama AC aktif maka temperature ruangan akan dimonitoring secara realtime, setelah waktu mencapai 5 menit maka sistem akan membandingkan nilai suhu ruangan realtime dengan suhu awal, jika suhu ruangan realtime sama atau lebih besar dari suhu awal maka sistem akan menon-aktifkan AC. Air Conditioner (AC) akan terus diaktifkan jika sampai waktu 5 menit suhu ruangan realtime lebih rendah dibandingkan dengan suhu awal.

7.3.3.2. *Lamp sub-sys (Modul Lampu)*

Modul lampu terdiri dari 2 bagian yaitu lampu yang bekerja secara otomatis dan lampu yang bekerja secara manual. Prinsip kerja sistem lampu otomatis adalah apabila sensor cahaya LDR mendapatkan cahaya berada pada kategori terang maka sistem akan mengaktifkan relay untuk mematikan lampu dan jika sensor cahaya LDR tidak mendapatkan cahaya atau dalam kondisi gelap maka sistem akan mengaktifkan relay untuk menyalakan lampu. Sistem lampu otomatis dapat juga memonitoring tegangan dan arus listrik yang mengalir pada balon lampu dan dapat mendeteksi balon lampu dalam kondisi berfungsi atau tidak berfungsi. Prinsip kerja sistem lampu yang bekerja secara manual adalah balon lampu yang terpasang pada sistem lampu manual dapat dikendalikan menggunakan aplikasi.

7.3.3.3. *Electrical Socket Sub-System (Modul Stok Kontak)*

Modul Stok Kontak terdiri dari sistem yang berfungsi untuk over heat dan sistem yang berfungsi untuk overload. Jika suhu kabel yang ada di stok kontak meningkat hingga melebihi 70 derajat Celsius maka sistem akan memutus arus listrik ke stok kontak dan ada notifikasi di aplikasi bahwa kabel kelebihan panas (overheat). Jika power listrik atau daya listrik melebihi yang ditetapkan yaitu 40 watt maka sistem akan memutus arus listrik ke stok kontak dan ada notifikasi di aplikasi bahwa stok kontak kelebihan beban (overload).

7.3.3.4. *Panel Sub-System* (Modul panel listrik)

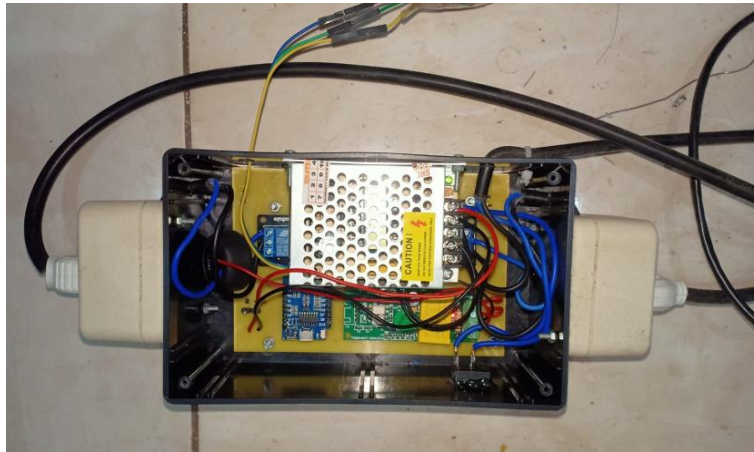
Modul panel listrik berfungsi untuk memonitoring penggunaan energi listrik secara realtime yang dapat tersimpan pada web server, Arus listrik dan tegangan listrik.

7.4. Hasil dan Pembahasan

Modul *Air Conditioner Sub-System*, *Lamp Sub-Sys*, *Electrichal Socket Sub-System*, dan *Panel Sub-System* menggunakan komponen yang sama yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, power supply 5V/2A dilengkapi dengan kabel AC, switch on/off dan stok kontak. Komponen tambahan untuk modul *Air Conditioner Sub-System* adalah sensor DHT-22 dan Relay 1 channel opto 5V *Lamp Sub-Sys* adalah sensor cahaya LDR dan Relay 1 channel opto 5V, *Electrichal Socket Sub-System* adalah sensor panas DS18B20, Relay 1 channel opto 5V, dan stok kontak 2 slot. Gambar 7.3 adalah modul *Air Conditioner Sub-System*, gambar 7.4 adalah moul *Lamp Sub-Sys*, gambar 7.5 adalah modul *Electrichal Socket Sub-System*, gambar 7.6 adalah gambar modul *Panel Sub-System*.



Gambar 7.3 Modul *Air Conditioner Sub-System*



Gambar 7.4 Modul *Lamp Sub-System*



Gambar 7.5 Modul *Electrichal Socket Sub-System*



Gambar 7.6. modul *Panel Sub-System*

Prinsip kerja MACSys yang terdiri 4 modul yaitu *Air Conditioner Sub-System*, *Lamp Sub-System*, *Electrichal Socket Sub-System* dan *Panel Sub-System* yang telah diuji sebanyak 20 kali yang dimana hasil pengujian dijabarkan pada Tabel 7.1. Pengujian dilakukan dengan mengetes fungsi dari hardware yang digunakan dengan teknik pengujian blackbox. Pengujian blackbox juga dilakukan untuk aplikasi MACSys dengan menguji coba aplikasi sebanyak 20 kali pengujian. Hasil pengujian blackbox untuk aplikasi MACSys dijabarkan pada tabel 7.2.

Tabel 7.1. Hasil pengujian blackbox untuk hardware

No.	Perangkat sistem	Input tes / kondisi	Hasil yang diharapkan	Output tes	Hasil tes
1	Sistem AC	Sistem AC di on kan	Sistem standby	Sistem standby	Sukses
		Saklar on AC pada aplikasi android ditekan	Sistem menyalakan AC	AC menyala	Sukses
		Suhu berubah menjadi lebih rendah dari suhu awal setelah 5 menit	Perangkat AC terus dinyalakan	AC menyala secara terus menerus	Sukses
		Suhu tidak berubah menjadi lebih rendah dari suhu awal setelah 5 menit	Perangkat dipadamkan, notifikasi dikirim ke aplikasi	AC dipadamkan dan notifikasi diterima di aplikasi	Sukses
		Saklar off AC pada aplikasi android ditekan	Sistem memadamkan AC	AC padam	Sukses
2	Sistem lampu	Sistem lampu di on kan	Sistem standby	Sistem standby	Sukses
		Saklar on auto lampu pada aplikasi ditekan	Sistem lampu otomatis dinyalakan	Sistem lampu otomatis menyala	Sukses
		Teras gedung gelap	Lampu teras dinyalakan	Lampu teras menyala	Sukses
		Teras gedung terang	Lampu teras dipadamkan	Lampu teras padam	Sukses
		Saklar off auto lampu pada	Sistem lampu otomatis dipadamkan,	Sistem lampu otomatis	Sukses

		aplikasi ditekan	sensor cahaya tidak merespon	padam, dan sensor cahaya tidak merespon	
		Tombol on lampu manual pada aplikasi ditekan	Lampu manual / ruangan dinyalakan	Lampu manual / ruangan menyala	Sukses
		Tombol off lampu manual pada aplikasi ditekan	Lampu manual / ruangan dipadamkan	Lampu manual / ruangan padam	Sukses
3	Sistem soket	Sistem soket di on kan	Sistem soket standby	Sistem soket standby	Sukses
		Saklar on soket pada aplikasi ditekan	Sistem soket dinyalakan	Sistem soket menyala	Sukses
		Beban lebih rendah dari 40 watt	Soket dinyalakan terus	Soket menyala terus	Sukses
		Beban sebesar 40 watt keatas	Soket dipadamkan, notifikasi dikirim ke aplikasi	Soket padam, notifikasi diterima di aplikasi	Sukses
		Suhu kabel di bawah 70 derajat celcius	Soket dinyalakan terus	Soket menyala terus	Sukses
		Suhu kabel sebesar 70 derajat ke atas	Soket dipadamkan, notifikasi dikirim ke aplikasi	Soket padam, notifikasi dikirim ke aplikasi	Sukses
		Saklar off soket pada aplikasi ditekan	Sistem soket dipadamkan	Sistem soket padam	Sukses
4	Panel	Sistem panel di on kan	Sistem panel dinyalakan	Sistem panel menyala	Sukses

		Sistem on	Selama sistem di on kan, data arus konsumsi dan grounding dikirim ke aplikasi	Data arus konsumsi dan grounding dikirim ke aplikasi secara terus menerus	Sukses
--	--	-----------	---	---	--------

Tabel 7.2. Hasil pengujian aplikasi android MACSys

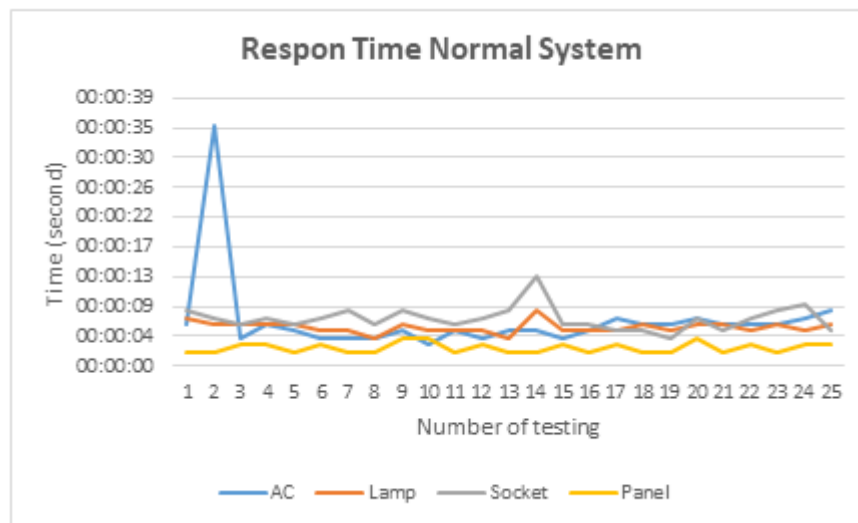
No.	Input tes	Hasil yang diharapkan	Output tes	Hasil tes
1	Icon aplikasi MACsys	Aplikasi terbuka	Aplikasi terbuka	Sukses
2	Tombol Sign In ditekan	Screen Home terbuka	Screen home ditampilkan	Sukses
3	Icon AC ditekan	Monitoring dan Kontrol sistem AC terbuka, dan memunculkan data dari mikrokontroler	Monitoring dan Kontrol sistem AC terbuka, dan memunculkan data dari mikrokontroller	Sukses
4	Saklar off AC ditekan	Saklar off berubah menjadi on, dan mengaktifkan sistem AC	Saklar off berubah menjadi on, dan mengaktifkan sistem AC	Sukses
5	Saklar on AC ditekan	Saklar on berubah menjadi off, dan menonaktifkan sistem AC	Saklar on berubah menjadi off, dan menonaktifkan sistem AC	Sukses
6	Icon home pada screen AC ditekan	Screen dialihkan ke menu home	Screen dialihkan ke menu home	Sukses
7	Icon lampu ditekan	Monitoring dan kontrol sistem lampu terbuka, dan memunculkan data dari	Monitoring dan kontrol sistem lampu terbuka, dan memunculkan data dari	Sukses

		mikrokontroler	mikrokontroler	
8	Saklar off lampu auto ditekan	Saklar off berubah menjadi on, dan mengaktifkan sistem lampu auto	Saklar off berubah menjadi on, dan mengaktifkan sistem lampu auto	Sukses
9	Saklar on lampu auto ditekan	Saklar on berubah menjadi off, dan menonaktifkan sistem lampu auto	Saklar on berubah menjadi off, dan menonaktifkan sistem lampu auto	Sukses
10	Saklar off lampu manual ditekan	Saklar off berubah menjadi on, dan mengaktifkan sistem lampu manual	Saklar off berubah menjadi on, dan mengaktifkan sistem lampu manual	Sukses
11	Saklar on lampu auto ditekan	Saklar on berubah menjadi off, dan menonaktifkan sistem lampu manual	Saklar on berubah menjadi off, dan menonaktifkan sistem lampu manual	Sukses
12	Icon home pada screen lampu ditekan	Screen dialihkan ke menu home	Screen dialihkan ke menu home	Sukses
13	Icon soket ditekan	Monitoring dan Kontrol sistem soket terbuka, dan memunculkan data dari mikrokontroler	Monitoring dan Kontrol sistem soket terbuka, dan memunculkan data dari mikrokontroler	Sukses
14	Saklar off soket ditekan	Saklar off berubah menjadi on, dan mengaktifkan sistem soket	Saklar off berubah menjadi on, dan mengaktifkan sistem soket	Sukses

15	Saklar on AC ditekan	Saklar on berubah menjadi off, dan menonaktifkan sistem soket	Saklar on berubah menjadi off, dan menonaktifkan sistem soket	Sukses
16	Icon home pada screen soket ditekan	Screen dialihkan ke menu home	Screen dialihkan ke menu home	Sukses
17	Icon panel ditekan	Monitoring sistem panel terbuka, dan memunculkan data dari mikrokontroler	Monitoring sistem panel terbuka, dan memunculkan data dari mikrokontroler	Sukses
18	Icon home pada screen panel ditekan	Screen dialihkan ke menu home	Screen dialihkan ke menu home	Sukses

MACSys juga telah diuji sebanyak 25 kali pengiriman instruksi control secara manual dan mendapatkan hasil packet loss adalah 0% yang berarti semua data berhasil dikirim dan untuk hasil pengujian respon time sistem dalam kondisi normal dijabarkan pada gambar 7.7.

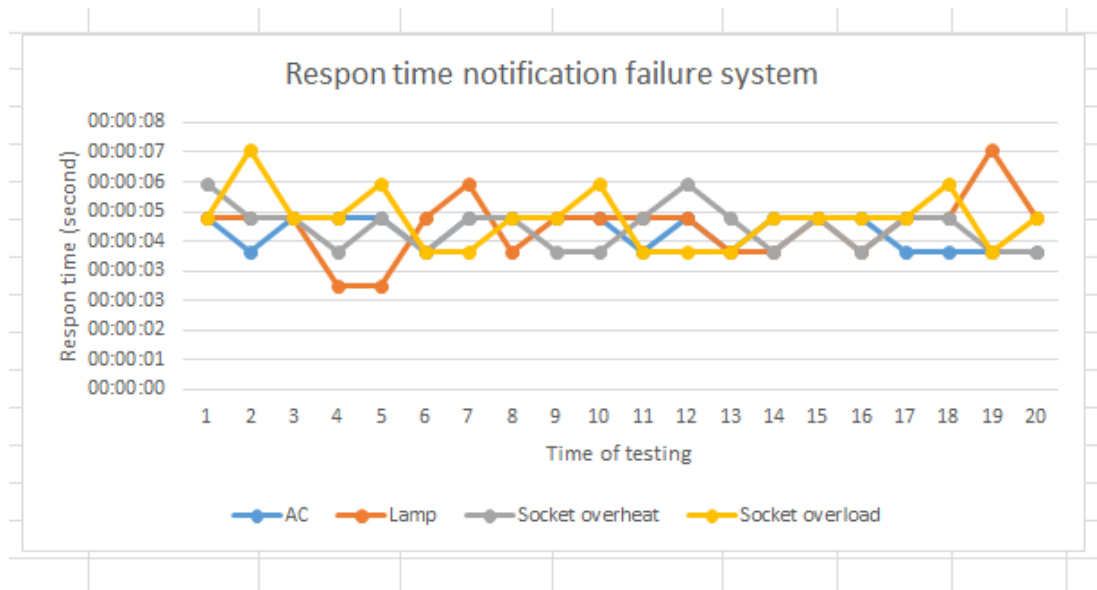
Pengujian respon time telah dilakukan dan diperoleh hasil respon time tercepat pada modul panel sub-system adalah 2 detik, modul air conditioner sub-system 3 detik, modul lamp sub-system dan modul electric socket sub-system adalah 4 detik. Respon time sangat dipengaruhi oleh traffic data dan kekuatan sinyal seluler yang digunakan pada saat pengujian, respon time yang paling lambat didapatkan pada modul air conditioner sub-system pada awal pengujian yaitu 35 detik.



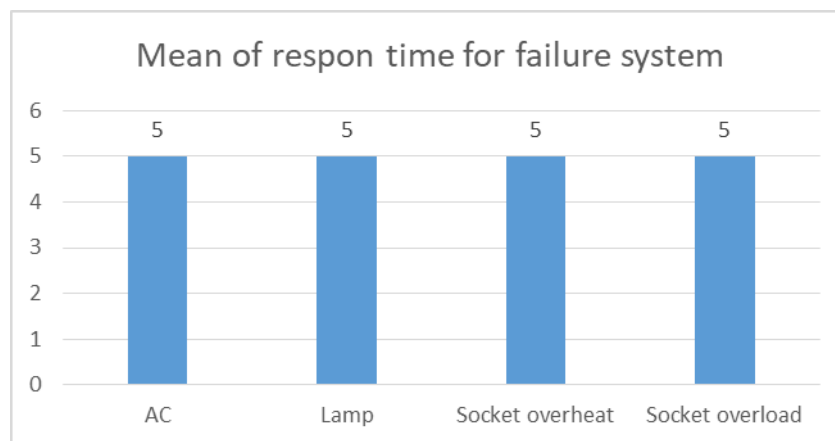
Gambar 7.7. Respon time MACSys dalam kondisi normal

MACSys dapat bekerja secara otomatis pada modul lamp sub-system dengan prinsip kerja jika kondisi lingkungan gelap maka lampu secara otomatis menyala dan jika kondisi lingkungan terang maka lampu akan padam secara otomatis. Pendeteksi kesalahan secara otomatis juga dapat bekerja pada modul air conditioner sub-system, lamp sub-system dan electric socket. Pemberitahuan hasil pendeteksian kesalahan dapat dilihat pada aplikasi android MACSys.

Respon time pada saat mendeteksi kesalahan yang dilakukan sebanyak 20 kali dijabarkan pada gambar 7.8. Respon time tercepat pada modul air conditioner sub-system adalah 4 detik. Respon time pemberitahuan panas lebih dan beban lebih pada modul socket adalah 5 detik. Rata-rata respon time pada saat mendeteksi kesalahan adalah 5 detik seperti pada gambar 7.9



Gambar 7.8. Respon time MACSys dalam kondisi deteksi kesalahan



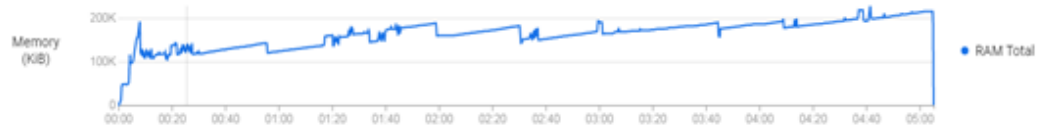
Gambar 7.9. Rata-rata Respon time MACSys dalam kondisi deteksi kesalahan

Selain pengujian respon time, pengujian menggunakan Firebase test lab dilakukan pada aplikasi Android MACSys. Pengujian dilakukan dengan cara: Aplikasi MACSys diuji melalui Test Lab firebase menggunakan device virtual dengan spesifikasi **pixel 5e** dan **API level 30**, orientasi **portrait** dan dinyatakan **lulus/berhasil** uji. Hasil test menunjukkan bahwa time to initial display adalah 614ms, Graphics stats dengan Missed Vsync 11%, high input latency 8%, slow UI thread 21%, slow draw commands 4%, slow bitmap uploads 0%.

Pengujian untuk CPU (percent), memory dan network dijabarkan pada gambar 7.10, gambar 7.11, dan gambar 7.12.



Gambar 7.10. Hasil pengujian CPU

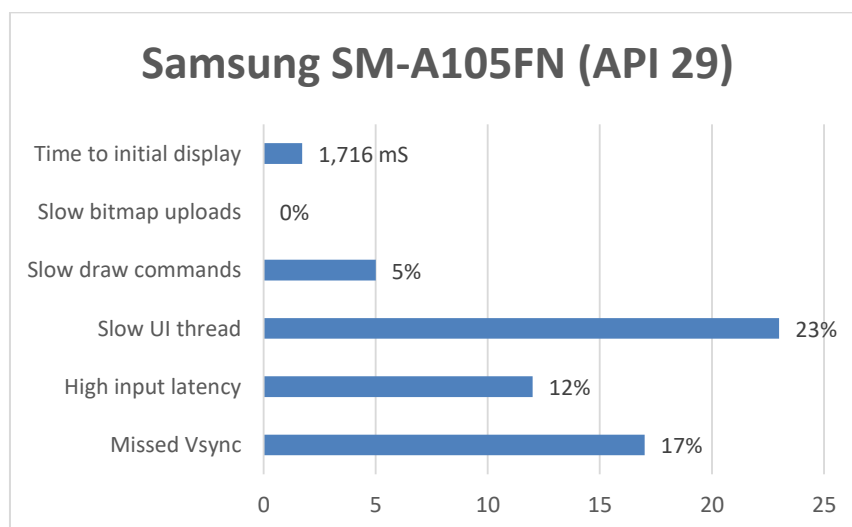


Gambar 7.11. Hasil pengujian Memory

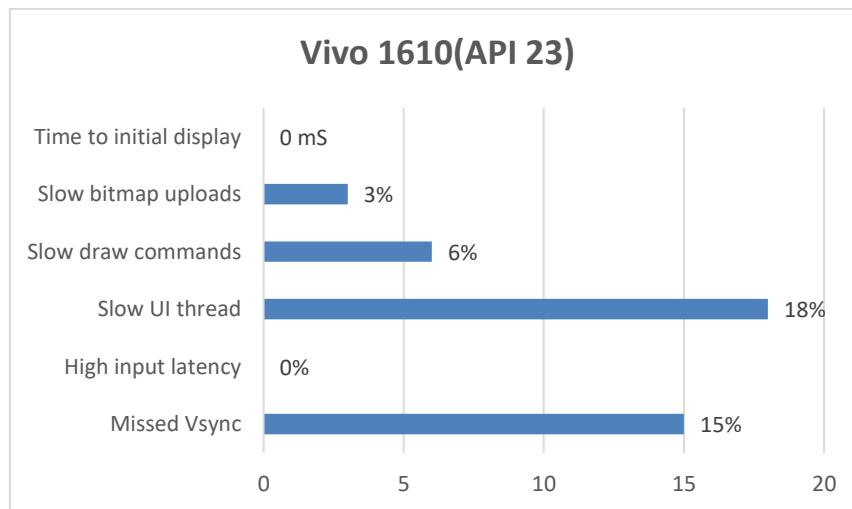


Gambar 7.12. Hasil pengujian network

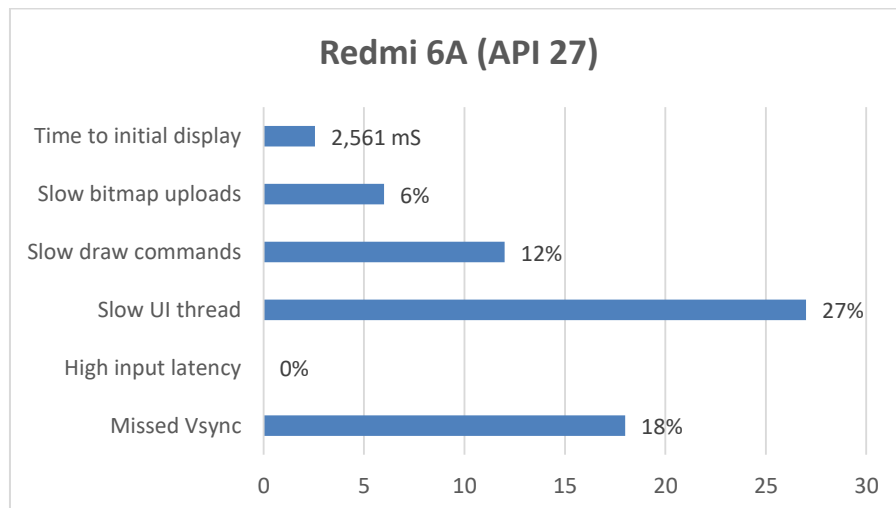
Pengujian performance aplikasi android MACSys untuk Samsung SM-A105FN (API 29) dijabarkan pada gambar 7.13, Vivo 1610 (API 23) dijabarkan pada gambar 7.14, dan Redmi 6A (API27) dijabarkan pada gambar 7.15. Pengujian performance aplikasi terkait dengan Missed Vsync, high input latency, slow UI thread, slow draw commands, slow bitmap uploads dan time to initial display.



Gambar 7.13 Hasil pengujian performance Samsung SM-A105FN (API 29)



Gambar 7.14 Hasil pengujian performance Vivo 1610 (API 23)



Gambar 7.15 Hasil pengujian performance Redmi 6A (API 27)

7.5. Kesimpulan

MACSys memiliki sistem pendeteksi kesalahan yang dapat terjadi pada perangkat kelistrikan yang terpasang seperti *air conditioner*, lampu dan electrical socket. Pendeteksi kesalahan pada MACSys berhasil dibangun dan hasil pengujian prinsip kerja sebanyak 10 kali dan berhasil beroperasi sesuai dengan prinsip kerja yang telah dirancang sehingga pengguna dapat mengetahui kesalahan yang terjadi pada perangkat kelistrikan dan bagian apa yang perlu ditangani. Uji Quality of Service dari sisi respon time telah dilakukan sebanyak 25 kali di masing-masing sistem dan mendapatkan rata-

rata respon time terbesar pada modul air conditioner sub-system 35 detik, modul electric socket sub-system 13 detik, modul modul lamp sub-system 8 detik dan modul panel sub-system 4 detik dengan rata-rata respon time keseluruhan adalah 15 detik. Aplikasi MACSys diuji melalui Test Lab firebase menggunakan device virtual dengan spesifikasi **pixel 5e** dan **API level 30**, orientasi **portrait** dan dinyatakan **lulus/berhasil** uji.

BAB 8

PEMBAHASAN UMUM

Penelitian Model Surveilans Otomatis Terintegrasi Dinamis untuk Operasional Perangkat-Perangkat IoT Kelistrikan merupakan pengembangan dari beberapa penelitian yang telah lalu. Beberapa penelitian terkait dapat dilihat pada gambar 8.1:

2021	Smart home Energy Consumption Monitoring System Integrated with Internet Connection Smarthome dengan monitoring konsumsi arus peralatan rumah tangga serta melihat tingkat akurasi dari sensor
2022	IoT Based Consumption and Lost Energy Monitoring System using PZEM-004T Sensor Pengukuran konsumsi energi dan energi terbuang pada penggunaan peralatan smarthome
2022	IoT Network of Sensor Array for Intrusion Detection and Diagnosis of Electrical System Berfokus pada pendeteksi gangguan dan diagnosis kerusakan pada peralatan kelistrikan
2022	Smart Electrical Device Control With Intrusion Detection Alert Berfokus pada pengontrolan perangkat jika terdeteksi gangguan untuk menghindari kerusakan
2022	Multi IoT Sensors For Failure Detection In Smart Home Berfokus pada deteksi dan diagnosis kerusakan dan melakukan aksi secara otomatis dalam mencegah kerusakan
2022	Sistem Pemantauan dan Pengendalian Terintegrasi untuk Pengoperasian Perangkat Kelistrikan IoT Berfokus pada pengembangan sistem terintegrasi dalam manajemen perangkat IoT kelistrikan

Gambar 8.1 Penelitian Terkait

Penelitian diawali dengan perancangan sistem pemanfaatan perangkat kelistrikan pada instalasi pemanfaatan gedung. Sistem pemantauan konsumsi energi pada *smart home* menggunakan koneksi internet berhasil dibangun. Pada penelitian ini sistem pemantauan yang dilakukan berfokus pada konsumsi arus peralatan listrik rumah tangga

seperti rice cooker, lampu penerangan, dan dispenser. Proses monitoring dapat dilakukan dengan web dan smartphone. Sensor yang digunakan adalah sensor PZEM-004T yang memiliki ketepatan pembacaan sensor yang tinggi, selain itu menggunakan node MCU ESP8266. Validasi hasil pembacaan sensor dengan nilai teori, membandingkan hasil pengukuran clamp meter dengan teori. Berdasarkan hasil perbandingan tersebut menunjukkan bahwa tingkat akurasi pembacaan sensor jauh lebih akurat. (bab2)

Penelitian selanjutnya **adalah IoT Based Consumption and Lost Energy Monitoring System using PZEM-004T Sensor**. Pada penelitian ini dikembangkan sistem pemantauan energi yang dapat memantau jumlah konsumsi dan lost energi secara realtime dimanapun dan kapanpun melalui aplikasi website yang telah dibangun. Aplikasi monitoring memiliki dua menu yaitu dashboard dan overview. Menu dashboar menampilkan nilai power, power factor, tegangan dan arus yang terpantau secara realtime. Dashboard juga dilengkapi dengan chart jumlah konsumsi dan lost energi di setiap waktunya. Menu overview menampilkan konsumsi dan lost energi secara detail mulai dari nilai secara realtime, nilai kemarin, nilai saat ini hingga bulan terakhir dan nilai penggunaan bulan terakhir ditampilkan di overview. Overview juga dilengkapi dengan chart yang enampilkan jumlah penggunaan pada setiap jam dan hari dari sisi konsumsi dan lost energi. Pengujian dilakukan selama tujuh hari dengan membandingkan hasil pengukuran PZEM-004T module sensor dengan electrical meter tipe melcoinda MS-98E (kWh meter komersil) dan electrical meter tipe melcoinda MS-97E (milik PLN). Hasil pengukuran sensor sebesar 16,39 kWh, electrical meter tipe melcoinda MS-98E sebesar 15,86 kWh, dan electrical meter tipe melcoinda MS-97E sebesar 16,12 kWh. Nilai rata-rata akurasi berdasarkan persentasi error rate sensor dengan electrical meter tipe melcoinda MS-97E sebagai acuan vlidasi mendapatkan nilai rata-rata error rate lebih kecil yaitu sebesar 1,48%. (bab 5)

Berdasarkan hasil penelitian pada **Jaringan Iot Sensor Array Untuk Deteksi Gangguan Dan Diagnosis Sistem Kelistrikan** menunjukkan

bahwa *prototype* sistem monitoring energi panel listrik, penerangan (Lampu) dengan deteksi cahaya, penyejuk udara (*Air Conditioning*) dengan deteksi suhu ruangan, monitoring energi pada stok kontak dengan deteksi suhu thermal (panas) kabel berhasil dirancang dan berhasil diakses melalui komputer atau smartphone menggunakan aplikasi *web service firebase*. Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada alat penerangan (lampu) yang dayanya 50 watt dengan deteksi cahaya lampu menunjukkan bahwa konsumsi arus rata-rata 0,26 ampere selama 60 menit dan hasil deteksi cahaya lampu menunjukkan bahwa ada cahaya lampu selama 60 menit, Hal ini menunjukkan bahwa selama ada arus yang mengalir pada alat penerangan (lampu) dan ada cahaya lampu yang terdeteksi, berarti lampu di ruangan tempat uji coba tetap menyala selama dimonitoring. Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada alat penyejuk ruangan (*Air Conditioning*) yang dayanya 330 watt dengan arus 1.6 A dan deteksi suhu ruangan menunjukkan bahwa konsumsi arus rata-rata 1,37 ampere selama 60 menit dan hasil deteksi suhu ruangan menunjukkan rata-rata 24,3 °C selama 60 menit. Hal ini menunjukkan bahwa selama 60 menit monitoring arus dan deteksi suhu ruangan pada alat penyejuk ruangan (AC) maka ada arus yang mengalir dan terdeteksi suhu ruangan yang menurun secara signifikan dari suhu awal 28,30°C setelah 60 menit menjadi 19,10°C yang berarti bahwa alat penyejuk ruangan (AC) di ruangan tempat uji coba masih berfungsi dengan baik selama dimonitoring. Berdasarkan hasil monitoring arus menggunakan sensor arus ACS712 pada stok kontak yang terhubung dengan personal computer (PC) dengan daya power supply 430 watt dan LCD monitor dengan arus 1,5A dan deteksi suhu thermal (panas) kabel menunjukkan bahwa konsumsi arus rata-rata 0,66 ampere selama 60 menit dan hasil deteksi suhu thermal (panas) menunjukkan rata-rata 25,9 °C selama 60 menit. Hal ini menunjukkan bahwa selama 60 menit monitoring arus dan deteksi suhu thermal (panas) kabel, arus mengalir dan suhu thermal (panas) kabel masih dalam kondisi aman yaitu rata-rata 25,9 °C. Berdasarkan hasil pengukuran arus menggunakan sensor arus ACS712 dan pengukuran arus

menggunakan *clamp meter* pada peralatan penerangan lampu, *Air Cinditioning* dan stok kontak yang terhubung dengan *personal computer* dan *LCD monitor* menunjukkan perbedaan rata-rata 0,04 A atau sekitar 5,26 %. (bab 3)

Pengujian aplikasi berhasil mengendalikan lampu dan AC melalui aplikasi yang telah terinstall di smartphone sebanyak 10 kali masing-masing dalam status off ke on dan status on ke off pada penelitian **Smart Electrical Device Control With Intrusion Detection Alert**. pembacaan sensor LDR berhasil mendeteksi kondisi cahaya pada ruangan, sensor DHT22 berhasil membaca nilai temperature pada ruangan, sensor LM35 berhasil membaca nilai thermal dari kabel dan sensor PZEM-004T berhasil membaca nilai arus dari perangkat elektronik yang terhubung dengan jumlah pengujian masing-masing sensor sebanyak 20 kali. Hasil akurasi pembacaan sensor DHT22 yang diuji selama 20 menit dengan membandingkan pembacaan dari HELES UX389 [27] didapatkan hasil bahwa DHT22 mendapatkan nilai akurasi sebesar 95.78% dan dengan model pengujian yang sama dilakukan pada sensor LM35 dengan nilai akurasi sebesar 89.70%. Hasil pengujian akurasi pembacaan sensor PZEM-004T dalam mendeteksi arus yang dibandingkan dengan alat pengukur arus komersial tipe VIP3803 [28] selama 20 menit didapatkan hasil bahwa nilai akurasi dari sensor PZEM-004T sebesar 91.39%. Selain melakukan pengujian terhadap akurasi sensor, sistem juga diuji dari sisi response time. Pengujian dilakukan sebanyak 20 kali pengontrolan dengan dua perangkat elektronik yang berbeda yaitu pada socket satu untuk menghubungkan dan memutus arus pada lampu dengan hasil nilai rata-rata response time 2.8 detik dan untuk socket kedua pada AC dengan hasil nilai rata response time 3.85 detik. Pada pengujian di socket dua didapatkan nilai respons time terlama selama 8 detik dan response time tercepat didapatkan selama 1 detik untuk kedua socket. (bab 4)

Prototype dari Smart Air Conditioner Control, Smart Light control, dan Smart electrical socket pada penelitian **Multi Iot Sensors For Failure Detection In Smart Home** berhasil dibangun. Ketiga sistem telah diuji sebanyak 40 kali pengiriman instruksi control secara manual dan

mendapatkan hasil packet loss adalah 0% yang artinya tidak ada data yang tidak berhasil dikirim dan pengujian response time telah dilakukan dan diperoleh hasil response time tercepat pada control socket adalah 415 millisecond (ms), control AC 417 ms, dan control lampu 422 ms dan rata-rata response time yang didapatkan untuk control socket adalah 516.4 ms, control AC 591.4 ms dan control lampu 797.5 ms. Response time sangat dipengaruhi oleh traffic data dan kekuatan sinyal seluler yang digunakan pada saat pengujian, response time paling lambat didapatkan pada control lampu dengan nilai 3509 ms. (bab 6)

Modul *Air Conditioner Sub-System*, *Lamp Sub-Sys*, *Electrichal Socket Sub-System*, dan *Panel Sub-System* yang dintegrasikan menggunakan aplikasi android dengan nama MACSys pada penelitian **Model Sistem Surveiln Otomatis Terintegrasi dan Dinamis Untuk Operasional Perangkat-Perangkat Iot Kelistrikan** berhasil dibangun dan diimplentasikan pada instalasi listrik rumah tinggal dan institusi. Semua modul menggunakan komponen yang sama yaitu: mikrokontroler Wemos mini D1 modul, sensor PZEM-004T, power supply 5V/2A dilengkapi dengan kabel AC, switch on/off dan stok kontak. Komponen tambahan untuk modul *Air Conditioner Sub-System* adalah sensor DHT-22 dan Relay 1 channel opto 5V *Lamp Sub-Sys* adalah sensor cahaya LDR dan Relay 1 channel opto 5V, *Electrichal Socket Sub-System* adalah sensor panas DS18B20, Relay 1 channel opto 5V, dan stok kontak 2 slot. MACSys juga telah diuji sebanyak 25 kali pengiriman instruksi control secara manual dan mendapatkan hasil packet loss adalah 0% yang berarti semua data berhasil dikirim dan untuk hasil pengujian respon time sistem dalam kondisi normal Pengujian respon time telah dilakukan dan diperoleh hasil respon time tercepat pada modul panel sub-system adalah 2 detik, modul air conditioner sub-system 3 detik, modul lamp sub-system dan modul electrichal socket sub-system adalah 4 detik. Respon time sangat dipengaruhi oleh traffic data dan kekuatan sinyal seluler yang digunakan pada saat pengujian, respon time yang paling lambat didapatkan pada modul air conditioner sub-system pada awal pengujian yaitu 35 detik. MACSys dapat bekerja secara otomatis

pada modul lamp sub-system dengan prinsip kerja jika kondisi lingkungan gelap maka lampu secara otomatis menyala dan jika kondisi lingkungan terang maka lampu akan padam secara otomatis. Pendeteksi kesalahan secara otomatis juga dapat bekerja pada modul air conditioner sub-system, lamp sub-system dan electrichal socket. Pemberitahuan hasil pendeteksian kesalahan dapat dilihat pada aplikasi android MACSys. Respon time pada saat mendeteksi kesalahan yang dilakukan sebanyak 10 kali dijabarkan pada 6.8. Respon time tercepat pada modul air conditioner sub-system adalah 4 detik. Respon time pemberitahuan panas lebih dan beban lebih pada modul socket adalah 5 detik. Selain pengujian respon time, pengujian menggunakan Firebase test lab dilakukan pada aplikasi Android MACSys. Pengujian dilakukan dengan cara: Aplikasi MACSys diuji melalui Test Lab firebase menggunakan device virtual dengan spesifikasi **pixel 5e** dan **API level 30**, orientasi **portrait** dan dinyatakan **lulus/berhasil** uji. Hasil test menunjukkan bahwa time to initial display adalah 614ms, Graphics stats dengan Missed Vsync 11%, high input latency 8%, slow UI thread 21%, slow draw commands 4%, slow bitmap uploads 0%. Pengujian performance aplikasi android MACSys untuk Samsung SM-A105FN (API 29), Vivo 1610 (API 23) dan Redmi 6A (API27) terkait dengan Missed Vsync, high input latency, slow UI thread, slow draw commands, slow bitmap uploads dan time to initial display. (Bab 7)

Secara keseluruhan penelitian ini mengembangkan system yang terintegrasi dalam manajemen perangkat IoT kelistrikan. Monitoring, Kontrol, deteksi dan diagnosis kerusakan dengan beberapa jenis peralatan kelistrikan diintegrasikan dengan sebuah sistem untuk mengelola dan mengatur secara keseluruhan. Beberapa topic penelitian yang telah dilakukan menghasilkan suatu sistem yang sudah diuji semua perangkat yang digunakan mengenai prinsip kerja, akurasi dan respon time pada saat dioperasikan. Sistem yang telah dibangun merupakan pengembangan sistem yang telah dibuat pada penelitian sebelumnya dan diintegrasikan sehingga menjadi satu kesatuan sistem yang terintegrasi, dinamis dan konfigurabel. Sistem yang dibangun juga memenuhi aspek praktis, mudah, efektif, efisien

dan ekonomis. Produk yang dihasilkan adalah sistem terintegrasi, dinamis dan reconfigurable serta diimplementasikan dalam skala laboratorium yaitu pada Laboratorium Radio, Telekomunikasi dan Gelombang Mikro.

Aplikasi android MACSys dapat memonitoring dan mengontrol modul-modul yang dibangun dan diimplementasikan dimana saja posisinya dan kapanpun akan digunakan dengan syarat terkoneksi dengan jaringan internet. **Aplikasi android MACSys** sangat tepat digunakan untuk memonitoring dan mengontrol peralatan kelistrikan yang berada pada lokasi yang berbeda-beda dan jumlah yang banyak. Sistem ini juga sifatnya stand alone sehingga memungkinkan modul yang diaplikasikan disesuaikan dengan kebutuhan, modul yang satu tidak terpengaruh dengan modul yang lain. **Aplikasi android MACSys** ini juga dapat digunakan untuk memonitoring konsumsi energi secara realtime dan hasilnya tersimpan pada database yang ada di web server sehingga dapat menjadi salah satu pilihan untuk mengetahui penggunaan energi pada gedung atau kampus dalam memudahkan untuk proses audit energi setiap tahun.

Beberapa pengujian telah dilakukan untuk memastikan bahwa sistem dapat berjalan dengan baik di beberapa perangkat dan diberbagai kondisi. Pengujian respon time untuk memastikan aplikasi dapat memberi informasi sedini mungkin tentang informasi real pada peralatan kelistrikan. Dari sisi kemampuan komputasi perangkat yang terpasang aplikasi MACSys dilakukan pengujian stress test dengan beberapa tipe smartphone dengan spesifikasi yang berbeda. Hasil pengujian menunjukkan dengan spesifikasi rendah, aplikasi MACSys masih dapat berjalan dengan baik.

Model surveillance kelistrikan berbasis IoT bekerja secara otomatis dan terkonfigurasi secara mudah dalam jaringan dan terintegrasi telah memperlihatkan prospek pengaplikasian untuk skala menengah hingga skala industry (skala besar). Karena realibiliti, efisiensi, akurasi dalam monitoring konsumsi energy dan los energy. Respon time (waktu yang digunakan untuk pengiriman data ke server) pada saat sistem bekerja dengan normal dan pada saat terjadi gangguan pada sistem.

Penggunaan TKDN dan TRL (Technology Readiness Level) tingkat kesiaapterapan teknologi saat ini masih berada pada level 7 hingga 8. Sedangkan TKDN dari produk sistem surveillance perangkat-perangkat kelistrikan IoT sekitar 70%.

BAB 9

KESIMPULAN UMUM

Sistem monitoring konsumsi arus listrik berbasis IoT sebagai bentuk pemanfaatan green technology telah dilakukan pada penelitian ini. Ada 3 alat listrik yang digunakan sebagai beban yaitu lampu bohlam, rice cooker dan dispenser. Pengamatan dilakukan selama 60 menit dengan pengiriman data setiap 5 menit. Hasil yang diperoleh menunjukkan bahwa sistem berhasil memonitor arus dan konsumsi daya secara real-time melalui web server menggunakan komputer atau smartphone

Sistem monitoring konsumsi energy dan lost energy menggunakan sensor PZEM-004T berbasis Internet of Things berhasil dibangun. Didapatkan hasil pengujian akurasi sensor dengan rata-rata error rate sebesar 2.31% dari hasil validasi pembacaan dua unit electrical meter tipe MELCOINDA MF-97E dan MELCOINDA MS-98E. Hal ini menandakan bahwa sistem monitoring ini memiliki akurasi kebenaran dalam mendeteksi konsumsi energi sebesar 97,69%. Untuk energi listrik yang terbuang, EMS mendeteksi nilai kWh rata-rata selama tujuh hari sebesar 0.22 kWh. Sistem ini sudah dapat diimplementasikan secara nyata sebagai alat monitoring konsumsi dan lost energi listrik yang fleksibel

Sistem monitoring energi penerangan (Lampu) dengan deteksi cahaya dan penyejuk udara (AC) dengan deteksi suhu ruangan dapat diakses melalui komputer atau smartphone menggunakan aplikasi *web service firebase* berhasil dirancang dengan Pendeteksian cahaya lampu menggunakan sensor cahaya menunjukkan bahwa penerangan (lampu) pada ruangan yang dimonitoring dalam kondisi baik. Pendeteksian suhu ruangan yang mengalami penurunan selama monitoring menunjukkan bahwa kondisi penyejuk udara (AC) pada ruangan yang dimonitoring dalam kondisi berfungsi dengan baik. Pendeteksian suhu thermal (panas) kabel masih menunjukkan nilai pada kondisi aman yaitu sekitar 25,6 °C. Sistem ini dapat digunakan untuk mengetahui system kelistrikan yang mengalami gangguan sehingga memudahkan untuk memperbaiki.

Sistem pengontrolan perangkat elektronik berbasis Internet of Things yang dilengkapi dengan fitur monitoring kondisi perangkat telah berhasil dirancang dan dibangun. Pengujian kontrol dan pembacaan sensor sebanyak 20 kali semuanya berhasil. Ketepatan akurasi pembacaan semua sensor yang digunakan diatas 89% dan nilai rata-rata response time selama 3.32 detik. Sistem dilengkapi dengan aplikasi yang dapat diinstall pada smartphone dengan operating system Android untuk melakukan pengontrolan dan memonitoring kondisi perangkat elektronik dan memberikan notifikasi kepada pengguna jika perangkat dalam rusak sehingga dapat memudahkan pengguna dalam mengelola konsumsi energi listrik dan melakukan maintenance perangkat elektronik secara efektif dan efisien. Kedepannya sistem kontrol dapat dikembangkan menjadi sistem otomatisasi pengontrolan perangkat elektronik dengan tujuan kelalaian pengguna dalam pengontrolan dapat lebih diminimalisir

Sistem pendeteksi kesalahan pada teknologi smarthome yang terdiri dari sistem monitoring dan pengontrolan pada AC, Lampu dan Socket berhasil dibangun dan hasil pengujian prinsip kerja sebanyak 10 kali berhasil beroperasi sesuai dengan prinsip kerja yang telah dirancang sehingga pengguna dapat mengetahui kesalahan yang terjadi pada teknologi smarthome yang digunakan dan bagian apa yang perlu ditangani. Uji Quality of Service dari sisi response time telah dilakukan sebanyak 40 kali di masing-masing sistem dan mendapatkan rata-rata response time sebesar 516.4 ms pada Socket system, 591.4 ms pada sistem AC dan pada sistem lampu sebesar 797.5 ms dengan nilai rata-rata response time secara keseluruhan sebesar 635.1 ms.

MACSys memiliki sistem pendeteksi kesalahan yang dapat terjadi pada perangkat kelistrikan yang terpasang seperti *air conditioner*, lampu dan electric socket. Pendeteksi kesalahan pada MACSys berhasil dibangun dan hasil pengujian prinsip kerja sebanyak 10 kali dan berhasil beroperasi sesuai dengan prinsip kerja yang telah dirancang sehingga pengguna dapat mengetahui kesalahan yang terjadi pada perangkat kelistrikan dan bagian apa yang perlu ditangani. Uji Quality of Service dari sisi respon time telah

dilakukan sebanyak 25 kali di masing-masing sistem dan mendapatkan rata-rata respon time terbesar pada modul air conditioner sub-system 35 detik, modul electrichal socket sub-system 13 detik, modul modul lamp sub-system 8 detik dan modul panel sub-system 4 detik dengan rata-rata respon time keseluruhan adalah 15 detik. Aplikasi MACSys diuji melalui Test Lab firebase menggunakan device virtual dengan spesifikasi **pixel 5e** dan **API level 30**, orientasi **portrait** dan dinyatakan **lulus/berhasil** uji.

SARAN

Model surveillance kelistrikan berbasis IoT bekerja secara otomatis dan terkonfigurasi secara mudah dalam jaringan dan terintegrasi telah memperlihatkan prospek pengaplikasian untuk skala menengah hingga skala industri (skala besar). Karena realibiliti, efisiensi, akurasi dalam monitoring konsumsi energy dan los energy. Respon time (waktu yang digunakan untuk pengiriman data ke server) pada saat sistem bekerja dengan normal dan pada saat terjadi gangguana pada sistem. Kondisi animo dan kebutuhan masyarakat dan industry akan sangat besar sehingga keberlanjutan kegiatan R & D pada kedua aspek

Perangkat keras:

Pemutakhiran kinerja operasi 4 modul IoT terkait dengan komponen mikrokontroler dan sensor yang digunakan

Perangkat lunak

Aplikasi android yang telah didesain dan diimplementasikan memiliki kekurangan diantaranya hanya sebagian simbol yang bersifat aktif dan symbol yang lain hanya bersifat informasi.

Peningkatan penggunaan TKDN (dan TRL (Technology Readiness Level) tingkat kesiapterapan teknologi saat ini masih berada pada level 7 hingga 8. Sedangkan TKDN dari produk sistem surveillance perangkat-perangkat kelistrikan IoT sekitar 70%. Oleh karena itu upaya riset berkelanjutan harus diimplementasikan untuk meningkatkan TKDL dan TRL.

DAFTAR PUSTAKA

- Abid, A. J. (2017). *Internet of Energy: A Design to Manage Energy Consumption for Limited Resources Building*. June, 19–25.
- Agdas, D., Srinivasan, R. S., Frost, K., & Masters, F. J. (2015). Energy use assessment of educational buildings : Toward a campus-wide sustainable energy policy. *Sustainable Cities and Society*, 17, 15–21. <https://doi.org/10.1016/j.scs.2015.03.001>
- Akbar, Y. M. (2014). *Sistem Pemantau Jaringan Wireless dan Estimasi Arah Wireless Access Point berbasis Raspberry-Pi*. 16(1), 1–9.
- Alanne, K. (2015). Ac ce pt cr t. *Energy & Buildings*. <https://doi.org/10.1016/j.enbuild.2015.07.060>
- Ascione, F., Bianco, N., Francesca, R., Masi, D., Maria, G., & Peter, G. (2017). Energy retrofit of educational buildings : Transient energy simulations , model calibration and multi-objective optimization towards nearly zero-energy performance. *Energy & Buildings*, 144, 303–319. <https://doi.org/10.1016/j.enbuild.2017.03.056>
- Attia, S., Eleftheriou, P., Xeni, F., Morlot, R., Ménézo, C., Kostopoulos, V., Betsi, M., Kalaitzoglou, I., Pagliano, L., Cellura, M., Almeida, M., Ferreira, M., Baracu, T., Badescu, V., Crutescu, R., & Hidalgo-betanzos, J. M. (2019). Overview and future challenges of nearly zero energy buildings (nZEB) design in Southern Europe. *Energy & Buildings*, 155(2017), 439–458. <https://doi.org/10.1016/j.enbuild.2017.09.043>
- Azmi, R. (2013). *Analisis Migrasi Radio Trunking Analog ke Radio Trunking Digital di Indonesia*. 247–264.
- Cao, X., Dai, X., & Liu, J. (2016). Building energy-consumption status worldwide and the state-of-the-art technologies for zero-energy buildings during the past decade. *Energy & Buildings*, 128, 198–213. <https://doi.org/10.1016/j.enbuild.2016.06.089>
- Darmawanputra, S. F., Rachmat, H., & Sjafrizal, T. (2015). Perancangan Sistem Otomasi Terintegrasi dan Supervisory Control And Data Acquisition (SCADA) Pada Stasiun Kerja Ex-Turning, Drilling-Chamfering dan Treading di PT. ABC Menggunakan Jaringan Komunikasi Wireless. *E-Proceeding of Engineering*, 4502–4528.
- Ferrari, S. (2017). Ac ce pt cr t. *Sustainable Cities and Society*. <https://doi.org/10.1016/j.scs.2017.03.010>
- Ferreira, M., Almeida, M., & Rodrigues, A. (2017). Impact of co-benefits on the assessment of energy related building renovation with a nearly-zero

- energy target. *Energy & Buildings*.
<https://doi.org/10.1016/j.enbuild.2017.07.066>
- Firman, B., & Firmansyah, E. (2012). Implementasi Komunikasi Data Berbasis Zigbee Pada SCADA (Supervisory Control and Data Acquisition) PLTMH. *Jurnal Teknologi*, 5, 149–155.
- Guilherme Mussi., Tochi., L. B. Campos., C. E. Cugnasca. (2017). Home Automation Networks. A Survey.
- Hannan, M. A., Faisal, M., Ker, P. J., Mun, L. H., Parvin, K., Mahlia, T. M. I., & Blaabjerg, F. (2018). *A Review of Internet of Energy Based Building Energy Management Systems : Issues and Recommendations*. 3536(c).
<https://doi.org/10.1109/ACCESS.2018.2852811>
- Harkouss, F., Fardoun, F., & Biwole, P. H. (2018). Multi-objective optimization methodology for net zero energy buildings. *Journal of Building Engineering*, 16(December 2017), 57–71.
<https://doi.org/10.1016/j.jobbe.2017.12.003>
- Huang, Z., Lu, Y., Wei, M., & Liu, J. (2017). Performance Analysis of Optimal Designed Hybrid Energy Systems for Grid-connected Nearly/Net Zero Energy Buildings. *Energy*. <https://doi.org/10.1016/j.energy.2017.11.093>
- Jain, M., Hoppe, T., & Bressers, H. (2016). Analyzing sectoral niche formation : The case of net-zero energy buildings in India. *Environmental Innovation and Societal Transitions*, 1–17.
<https://doi.org/10.1016/j.eist.2016.11.004>
- Khajenasiri, I., Virgone, J., & Gielen, G. (2015). *A Presence-Based Control Strategy Solution for HVAC Systems*. 620–622.
- Kim, J., Cai, J., Braun, J. E., Technical, N., Stephen, M., Kim, J., Cai, J., & Braun, J. E. (2018). *Common Faults and Their Prioritization in Small Commercial Buildings Common Faults and Their Prioritization in Small Commercial Buildings*. February 2017.
- Li, D. H. W., Yang, L., & Lam, J. C. (2013). Zero energy buildings and sustainable development implications e A review. *Energy*.
<https://doi.org/10.1016/j.energy.2013.01.070>
- Linji Kong & Biao Ma (2019). Intelligent Manufacturing Model of Construction Indusry based on Internet of Things Technology
- Lizana, J., Chacartegui, R., Barrios-padura, A., & Manuel, J. (2017). *Advances in thermal energy storage materials and their applications towards zero energy buildings : A critical review*. 203, 219–239.
<https://doi.org/10.1016/j.apenergy.2017.06.008>
- Lou, S., Tsang, E. K. W., Li, D. H. W., Lee, E. W. M., & Lam, J. C. (2017).

- Towards zero energy school building designs in Hong Kong. *Energy Procedia*, 105, 182–187. <https://doi.org/10.1016/j.egypro.2017.03.299>
- McAllister, T. D., El-Tawab, S., & Heydari, M. H. (2017). Localization of Health Center Assets Through an IoT Environment (LoCATE). *2017 Systems and Information Engineering Design Symposium, SIEDS 2017*, 132–137. <https://doi.org/10.1109/SIEDS.2017.7937703>
- Mills, E. (2011). Building commissioning: A golden opportunity for reducing energy costs and greenhouse gas emissions in the United States. In *Energy Efficiency* (Vol. 4, Issue 2, pp. 145–173). <https://doi.org/10.1007/s12053-011-9116-8>
- Molina-huelva, M., Chacartegui, R., & Molina-huelva, M. (2017). *Accepted Manuscript*. <https://doi.org/10.1016/j.energy.2017.05.188>
- Morales, P., Letelier, V., & Mora, D. (2017). Implications of Life Cycle Energy Assessment of a new school building, regarding the nearly Zero Energy Buildings targets in EU: A case of Study. *Sustainable Cities and Society*. <https://doi.org/10.1016/j.scs.2017.03.016>
- Mytafides, A. C. K., Dimoudia, A., Mytafides, C. K., Dimoudi, A., & Zoras, S. (2017). Title : Transformation of a university building into a zero energy building in Mediterranean climate Transformation of a University Building into a Zero Energy Building in Mediterranean Climate. *Energy & Buildings*. <https://doi.org/10.1016/j.enbuild.2017.07.083>
- Rahman, M. A., Jakaria, A. H. M., & Al-Shaer, E. (2016). Formal Analysis for Dependable Supervisory Control and Data Acquisition in Smart Grids. *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 263–274. <https://doi.org/10.1109/DSN.2016.32>
- Richman, E. E. (n.d.). *“Measurement and Verification of Energy Savings and Performance from Advanced Lighting Controls,.”*
- Santamouris, M. (2016). ScienceDirect Innovating to zero the building sector in Europe : Minimising the energy consumption , eradication of the energy poverty and mitigating the local climate change. *SOLAR ENERGY*. <https://doi.org/10.1016/j.solener.2016.01.021>
- Soleh, M. (2014). Desain Sistem SCADA Untuk Peningkatan Pelayanan Pelanggan Dan Efisiensi Operasional Sistem Tenaga Listrik di APJ Cirebon. *Jurnal Telekomunikasi Dan Komputer*, 5(1), 31–32.
- Stouffer, K., Joe, F., & Keut, K. (2006). Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control System Security. In *Nist Special Publication*. [https://scadahacker.com/library/Documents/ICS_Basics/SCADA Basics -](https://scadahacker.com/library/Documents/ICS_Basics/SCADA_Basics_-)

NCS TIB 04-1.pdf

- Sun, Y., Huang, G., Xu, X., & Lai, A. C. (2018). Building-group-level performance evaluations of net zero energy buildings with non-collaborative controls. *Applied Energy*, 212(July 2017), 565–576. <https://doi.org/10.1016/j.apenergy.2017.11.076>
- Susanto, H., & Lysbetti, N. (2015). *ANALISA PENERAPAN SISTEM SCADA PADA PENGENDALIAN JARINGAN TEGANGAN MENENGAH 20 kV PT . PLN AREA*. 1–9.
- Won, J., Kwae, Y., & Woo, I. (2012). *Energy Procedia Automatic sensor arrangement system for building energy and environmental management*. <https://doi.org/10.1016/j.egypro.2011.12.887>
- Wu, W., Guo, J., Li, J., Hou, H., Meng, Q., & Wang, W. (2020). A multi-objective optimization design method in zero energy building study : A case study concerning small mass buildings in cold district of China. *Energy & Buildings*, 158(2018), 1613–1624. <https://doi.org/10.1016/j.enbuild.2017.10.102>

Lampiran

1. Listing dan Penjelasan Program

Program mikrokontroler dibuat menggunakan software Arduino IDE. Berikut listing beserta penjelasan program mikrokontroler yang telah dibuat.

a. Air Conditioner

Berikut listing untuk *air conditioner* dijelaskan secara bertahap.

1) Library (pustaka)

Library merupakan kumpulan kode program arduino dasar yang dikemas untuk memberikan perintah terhadap suatu komponen agar bekerja sesuai dengan fungsinya. Sebelum memasuki program inti, di awal penulisan program pada Arduino IDE harus dimasukkan atau diinkludkan beberapa library yaitu;

- a) `#include "FirebaseESP8266.h"`, yaitu library firebase untuk ESP8266.
- b) `#include <ESP8266Wifi.h>`, yaitu library wifi ESP8266.
- c) `#include <PZEM004Tv30.h>`, yaitu library sensor arus PZEM004T.
- d) `#include "DHT.h"`, yaitu library sensor temperatur DHT22.
- e) `#include <ESP8266HTTPClient.h>`, yaitu library untuk ESP8266 sebagai client yang melakukan request pada server.

```
#include "FirebaseESP8266.h"
#include <ESP8266Wifi.h>
#include <PZEM004Tv30.h>
#include "DHT.h"
#include <ESP8266HTTPClient.h>
```

2) Define (defenisi)

Pendefinisian dalam hal ini yang dimaksud adalah penentuan perangkat keras ataupun perangkat lunak yang akan digunakan sebagai pendukung bekerjanya mikrokontroler.

Beberapa pendefinisian yang harus dituliskan yaitu;

- a) `#define FIREBASE_HOST`, yang diisi dengan nama hosting firebase yang telah dibuat.
- b) `#define FIREBASE_AUTH`, yang diisi dengan firebase autentikasi firebase yang akan digunakan.
- c) `#define WIFI_SSID`, yang diisi dengan nama wifi direct yang akan digunakan untuk koneksi ke internet.
- d) `#define WIFI_PASSWORD`, yang diisi dengan password wifi yang akan digunakan.

```
//Wifi-FireBase
#define FIREBASE_HOST "fds-android-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "KywiAMupDG4wOoL3E3kVeULboWQMACAdLfy9V7sv"
#define WIFI_SSID "Muliadi"
#define WIFI_PASSWORD "qwerty1234"
```

Selanjutnya pendefinisian sensor-sensor input dan aktuator output yang akan digunakan seperti pada listing program berikut.

```
//PZEMAC
PZEM004Tv30 pzem(D6, D5);
//Tempsensor-RelayAC
#define DHTPIN D4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
#define relayAC D1
```

- a) PZEM004Tv30 pzem (D6,D5), yaitu penentuan pin D6 sebagai RX dan D5 sebagai TX untuk sensor arus PZEM004T.
- b) #define DHTPIN D4, yaitu penentuan pin D4 sebagai input dari sensor DHT.
- c) #define DHTTYPE DHT22, penentuan tipe sensor DHT yang digunakan yaitu DHT22.
- d) #define relayAC D1, yaitu penentuan D1 sebagai output relay untuk air conditioner.

3) *Variable* (variabel)

Variabel merupakan tempat penyimpanan data yang akan diproses, pembuatan variabel diawali dengan penulisan tipe data yang akan digunakan dan diikuti dengan nama variabel seperti pada listing program berikut.

```

FirebaseData firebaseData;
unsigned long timing = 0;
unsigned long timing2 = 0;
unsigned long timing1 = millis();
float temp;
float roomTemp;
float currentAC;
float voltageAC;
boolean statusPZEM_AC = true;
int counter;

```

Ada beberapa tipe data yang digunakan yaitu;

- firebaseData, untuk mengolah data dari firebase.
- unsigned long timing, untuk mengolah nilai millis.
- float temp, untuk mengolah data temperatur dalam satuan desimal (floating).
- float currentAC, untuk mengolah data arus pada air conditioner dalam satuan desimal (floating).
- float voltageAC, untuk mengolah data tegangan pada air conditioner dalam satuan desimal (floating).
- boolean statusPZEM_AC untuk mengolah status dari sensor arus PZEM004T.
- int counter, untuk mengolah data counter (penghitung) yang dibuat manual menggunakan data integer.

Selanjutnya variabel yang digunakan untuk pengirim data pada web server sebagai berikut :

```

HTTPClient http;
int httpCode;
String payload;
String host_post= "http://project3.monlab.site/index.php/post/ac";
String status = "Normal";
String toggle = "OFF";

```

Ada beberapa tipe data yang digunakan yaitu;

- HTTPClient, untuk protokol pengiriman data dari HTTP request.
- String payload, untuk mengolah data feedback dari request
- String host_post, untuk menginisialisasi domain host server
- String status, sebagai data yang dikirim mewakili kondisi sistem
- String toggle, sebagai data yang dikirim mewakili kondisi relay.

4) Void Setup

Void setup merupakan bagian program yang pertama dan hanya sekali diproses oleh mikrokontroler. Pada bagian ini mikrokontroler melakukan beberapa proses seperti yang terlihat pada listing program berikut ini.

```
void setup() {
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    counter++;
    if (counter >= 40) {break;} delay(500);}

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  dht.begin();
  pinMode(relayAC, OUTPUT);
  digitalWrite (relayAC, HIGH);
  roomTemp = dht.readTemperature();
  delay(8000);
  Firebase.setFloat(firebaseData, "/AC/initTemp", roomTemp);
}
```

- a) `Wifi.begin(WIFI_SSID, WIFI_PASSWORD)`, berfungsi untuk mengaktifkan wifi dengan menyesuaikan nama SSID dengan password.
- b) `counter++` , berfungsi untuk menghitung timer sampai 40 detik, jika telah melewati 40 detik namun tidak dapat terhubung dengan wifi tersebut maka mikrokontroler beralih pada mode offline.
- c) `Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH)`, berfungsi untuk menghubungkan mikrokontroler dengan firebase dengan menyesuaikan firebase host dengan firebase auth. Proses ini akan berhasil jika mikrokontroler dalam kondisi online. Namun jika dalam kondisi offline maka bagian ini akan diabaikan atau dilewatkan, dengan demikian mikrokontroler tidak akan mengirim data ke firebase.
- d) `dht.begin()` , berfungsi untuk mengaktifkan sensor DHT (sensor temperatur).
- e) `pinMode (relayAC,OUTPUT)`, untuk menyatakan bahwa relayAC merupakan OUTPUT.
- f) `digitalWrite(relayAC, HIGH)`, memberikan logika awal berupa HIGH pada relay.
- g) `roomTemp=dht.readTemperature()` , berfungsi untuk membaca suhu awal ruangan.
- h) `Firebase.setFloat(firebaseData,"/AC/initTemp",roomTemp)`, berfungsi untuk mengirim data ke firebase yang berisikan nilai suhu awal ruangan.

5) Void Loop

Void loop merupakan bagian program yang akan diproses oleh mikrokontroler secara terus menerus selama mikrokontroler bekerja. Pada bagian ini mikrokontroler melakukan beberapa proses seperti yang terlihat pada listing program berikut ini.

```
void loop() {
  check_mode:
  if (Firebase.getString(firebaseData, "/AC/switchAc")) {
    if (firebaseData.to<String>() == "1") { goto mode_on;}
    if (firebaseData.to<String>() == "0") {goto mode_off;}
    goto check_mode;
  }
```

Pada awal dinyalakan, mikrokontroler melakukan proses pengecekan mode (check mode) yaitu mengambil data perintah dari firebase, jika perintah dari firebase (switch AC) berlogika 1 (ON), maka mikrokontroler mengaktifkan air conditioner dan menjalankan sistem. Mode on ditunjukkan pada listing berikut ini.

```
mode_on:
  if (Firebase.getString(firebaseData, "/AC/switchAc")) {
    if (firebaseData.to<String>() == "0") { goto mode_off;}}
  timing2 = millis()-timing1;
  timing = timing2 / 60000;
  Firebase.setInt(firebaseData, "/AC/timeOnAc", timing);
  digitalWrite (relayAC, HIGH);
  temp = dht.readTemperature();
  if (isnan(temp)) { Firebase.setString(firebaseData, "/AC/temperatureAc", "-");
    Firebase.setString(firebaseData, "/AC/warningAc", "Check_temperature_sensor");}
  else { Firebase.setFloat(firebaseData, "/AC/temperatureAc", temp);
    Firebase.setString(firebaseData, "/AC/warningAc", "Temp_Sens_Normal");}
```

Mode on;

- Firebase.setInt(firebaseData, "/AC/timeOnAc", timing), yaitu mikrokontroler mengirim data waktu ON air conditioner (timing) ke firebase.
- digitalWrite (relayAC, HIGH), yaitu mikrokontroler menyalakan atau mengaktifkan air conditioner.
- temp = dht.readTemperature(), yaitu mikrokontroler membaca nilai temperatur ruangan.
- Firebase.setFloat(firebaseData, "/AC/temperatureAc", temp), yaitu mikrokontroler mengirim data nilai temperatur ruangan ke firebase.

Selanjutnya pembacaan sensor arus PZEM004T, diperlihatkan pada listing berikut.

```

statusPZEM_AC = true;
currentAC = pzem.current();
voltageAC = pzem.voltage();
if (!isnan(currentAC)) { Firebase.setFloat(firebaseData, "/AC/currentAc", currentAC);
                        Firebase.setFloat(firebaseData, "/AC/voltageAc", voltageAC);
                        Firebase.setString(firebaseData, "/AC/warningAc", "Current_Sens_Normal");}

else { Firebase.setString(firebaseData, "/AC/currentAc", "-");
      Firebase.setString(firebaseData, "/AC/conditionAc", "Not detected");
      Firebase.setString(firebaseData, "/AC/warningAc", "Check_current_sensor");
statusPZEM_AC = false;}

```

- a) statusPZEM_AC = true, yaitu perintah mengaktifkan sensor arus air conditioner.
- b) currentAC = pzem.current(), yaitu perintah membaca arus air conditioner.
- c) voltageAC = pzem.voltage(), yaitu perintah membaca tegangan air conditioner.
- d) Firebase.setFloat(firebaseData, "/AC/currentAc", currentAC), yaitu mengirim nilai arus air conditioner ke firebase.
- e) Firebase.setFloat(firebaseData, "/AC/voltageAc", voltageAC), yaitu mengirim nilai tegangan air conditioner ke firebase.

Selanjutnya program bersyarat pada sistem ditunjukkan pada listing berikut.

```

if (timing >= 1 && currentAC > 0.0)
{ Firebase.setFloat(firebaseData, "/AC/currentAc", currentAC);
  Firebase.setFloat(firebaseData, "/AC/voltageAc", voltageAC);
  Firebase.setString(firebaseData, "/AC/conditionAc", "TURN ON");
  Firebase.setString(firebaseData, "/AC/warningAc", "Normal");}

```

Penjelasan:

Jika waktu menyala (ON) air conditioner telah melebihi 1 menit dan arus lebih besar dari 0A, maka mikrokontroler mengirim data ke firebase berupa nilai arus, nilai tegangan, kondisi air conditioner, dan catatan peringatan "Normal".

```

if (timing >= 1 && currentAC == 0.0)
{ Firebase.setFloat(firebaseData, "/AC/currentAc", currentAC);
  Firebase.setFloat(firebaseData, "/AC/voltageAc", voltageAC);
  Firebase.setString(firebaseData, "/AC/conditionAc", "Damaged");
  Firebase.setString(firebaseData, "/AC/warningAc", "Fault_detected");}

```

Penjelasan:

Jika waktu menyala (ON) air conditioner telah melebihi 1 menit dan arus 0A, maka mikrokontroler mengirim data ke firebase berupa nilai arus, nilai tegangan, kondisi air conditioner, dan catatan peringatan “Fault detected”.

```
while (timing >= 5 && temp >= roomTemp)
{ digitalWrite (relayAC, LOW);
  Firebase.setFloat(firebaseData, "/AC/currentAc", currentAC);
  Firebase.setFloat(firebaseData, "/AC/voltageAc", voltageAC);
  Firebase.setString(firebaseData, "/AC/conditionAc", "TURN OFF");
  Firebase.setString(firebaseData, "/AC/warningAc", "Fault_detected");}
goto mode_on;
```

Penjelasan:

Jika air conditioner telah ON selama 5 menit ke atas dan temperatur ruangan masih sama atau lebih besar dari temperatur awal ruangan, maka relay air conditioner di OFF kan (air conditioner dinonaktifkan). Namun jika temperatur ruangan menjadi lebih rendah dari temperatur awal ruangan, maka air conditioner akan terus dinyalakan.

Namun jika perintah dari firebase (switch AC) berlogika 0 (OFF), maka mikrokontroler menonaktifkan air conditioner sistem dalam kondisi standby. Seperti yang diperlihatkan pada listing berikut.

```
mode_off:
digitalWrite (relayAC, LOW);
if (Firebase.getString(firebaseData, "/AC/switchAc")) {
  if (firebaseData.to<String>() == "1") {
    timing1 = millis();
    timing = timing1 -millis();
    Firebase.setInt(firebaseData, "/AC/timeOnAc", timing);
    goto mode_on;
  }
  Firebase.setString(firebaseData, "/AC/timeOnAc", "-");
  Firebase.setString(firebaseData, "/AC/currentAc", "-");
  Firebase.setString(firebaseData, "/AC/conditionAc", "-");
  Firebase.setString(firebaseData, "/AC/warningAc", "OFF");
  Firebase.setString(firebaseData, "/AC/voltageAc", "-");}
goto mode_off;}
```

- a) digitalWrite (relayAC, LOW), yaitu mikrokontroler akan memberikan logika LOW pada relay dan air conditioner dan sistem akan dinonaktifkan.
- b) Mikrokontroler mengirim data ke firebase berupa nilai arus, nilai tegangan, kondisi air conditioner, dan catatan peringatan “OFF”.

Selanjutnya pengiriman seluruh data ke server, diperlihatkan pada listing berikut.

```
String data = "?current=" + String(currentAC) + "&voltage=" + String(voltageAC)
host_post += data;
http.begin(host_post); //Specify request destination
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
httpCode=http.GET();
payload=http.getString(); // get data from webhost continously
Serial.println(payload);
http.end(); //Close connection
```

Penjelasan:

Pada baris pertama data digabung dalam sebuah baris String data kemudian digabung/dimasukkan bersama dengan data dari host_post yang selanjutnya memulai melakukan request pada host_post dengan menggunakan header dan request type GET. Hasil dari request dimasukkan dalam variabel payload dan diakhiri dengan pemutusan koneksi pada server.

b. Lampu

Berikut listing untuk lampu dijelaskan secara bertahap.

1) Library (pustaka)

Sebelum memasuki program inti, di awal penulisan program pada Arduino IDE harus dimasukkan atau diinkludkan beberapa library yaitu;

- a) #include "FirebaseESP8266.h", yaitu library firebase untuk ESP8266.
- b) #include <ESP8266Wifi.h>, yaitu library wifi ESP8266.
- c) # <PZEM004Tv30.h, yaitu library sensor arus PZEM004T.
- d) #include "ESP8266HTTPClient, yaitu library untuk ESP8266 sebagai client yang melakukan request pada server.

```
#include <FirebaseESP8266.h>
#include <ESP8266WiFi.h>
#include <PZEM004Tv30.h>
#include <ESP8266HTTPClient.h>
```

2) Define (defenisi)

Beberapa pendefinisian yang harus dituliskan yaitu;

- a) #define FIREBASE_HOST, yang diisi dengan nama hosting firebase yang telah dibuat.
- b) #define FIREBASE_AUTH, yang diisi dengan firebase autentikasi firebasae yang akan digunakan.
- c) #define WIFI_SSID, yang diisi dengan nama wifi direct yang akan digunakan untuk koneksi ke internet.
- d) #define WIFI_PASSWORD, yang diisi dengan password wifi yang akan digunakan.

```
//Wifi-FireBase
#define FIREBASE_HOST "fds-android-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "KywiAMupDG4wOoL3E3kVeULboWQMACAdLf9V7sv"
#define WIFI_SSID "Muliadi"
#define WIFI_PASSWORD "qwerty1234"
```

Selanjutnya pendefenisian sensor-sensor input dan aktuator output yang akan digunakan seperti pada listing program berikut.

```
//PZEMLamp
PZEM004Tv30 pzem(D6, D5);
//Lightsensor-RelayLamp
#define lightLDR D7
#define relayLamp D1
#define relayLamp2 D2
```

- a) PZEM004Tv30 pzem (D6,D5), yaitu penentuan pin D6 sebagai RX dan D5 sebagai TX untuk sensor arus PZEM004T.
- b) #define lightLDR D7, yaitu penentuan pin D7 sebagai input dari sensor LDR.
- c) #define relayLamp D1, penentuan pin D1 sebagai output relay untuk lampu 1.
- d) #define relayLamp D2, penentuan pin D2 sebagai output relay untuk lampu 2.

3) Variable (variabel)

Berikut variabel yang digunakan.

```

FirebaseData firebaseData;
unsigned long timing = 0;
float currentLamp;
float voltageLamp;
boolean statusPZEM_Lamp = true;
int counter;

```

Ada beberapa tipe data yang digunakan yaitu;

- a) firebaseData, untuk mengolah data dari firebase.
- b) unsigned long timing, untuk mengolah nilai millis.
- c) float currentLamp, untuk mengolah data arus pada lampu dalam satuan desimal (floating).
- d) float voltageLamp, untuk mengolah data tegangan pada lampu dalam satuan desimal (floating).
- e) boolean statusPZEM_AC untuk mengolah status dari sensor arus PZEM004T.
- f) int counter, untuk mengolah data counter (penghitung) yang dibuat manual menggunakan data integer.

Selanjutnya variabel yang digunakan untuk pengirim data pada web server sebagai berikut :

```

HTTPClient http;
int httpCode;
String payload;
String host_post= "http://project3.monlab.site/index.php/post/lamp";
String status = "Normal";
String toggle = "OFF";
String light = "Null";

```

Ada beberapa tipe data yang digunakan yaitu;

- f) HTTPClient, untuk protokol pengiriman data dari HTTP request.
- g) String payload, untuk mengolah data feedback dari request
- h) String host_post, untuk menginisialisasi domain host server
- i) String status, sebagai data yang dikirim mewakili kondisi sistem
- j) Sting light, sebagai data yang dikirim mewakili pembacaan sensor
- k) String toggle, sebagai data yang dikirim mewakili kondisi relay.

4) Void Setup

Berikut program yang pertama dan hanya sekali dibaca oleh mikrokontroler.

```
void setup() {
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    counter++;
    if (counter >= 40) {break;}
    delay(500);}
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  Firebase.setString(firebaseData, "/Lamp/cond", "Standby");
  pinMode(relayLamp, OUTPUT);
  pinMode(relayLamp2, OUTPUT);
  digitalWrite (relayLamp, HIGH);
  digitalWrite (relayLamp2, HIGH);}
}
```

- a) `Wifi.begin(WIFI_SSID, WIFI_PASSWORD)`, berfungsi untuk mengaktifkan wifi dengan menyesuaikan nama SSID dengan password.
- b) `counter++` , berfungsi untuk menghitung timer sampai 40 detik, jika telah melewati 40 detik namun tidak dapat terhubung dengan wifi tersebut maka mikrokontroler beralih pada mode offline.
- c) `Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH)`, berfungsi untuk menghubungkan mikrokontroler dengan firebase dengan menyesuaikan firebase host dengan firebase auth. Proses ini akan berhasil jika mikrokontroler dalam kondisi online. Namun jika dalam kondisi offline maka bagian ini akan diabaikan atau dilewatkan, dengan demikian mikrokontroler tidak akan mengirim data ke firebase.
- d) `Firebase.setString(firebaseData, "/Lamp/cond", "Standby")`, berfungsi untuk mengirim data ke firebase yang berisikan tulisan "Standby".
- e) `pinMode (relayLamp, OUTPUT)`, untuk menyatakan bahwa relayLamp merupakan OUTPUT.
- f) `pinMode (relayLamp2, OUTPUT)`, untuk menyatakan bahwa relayLamp2 merupakan OUTPUT.
- g) `digitalWrite(relayLamp, HIGH)`, memberikan logika awal berupa HIGH pada relay 1.
- h) `digitalWrite(relayLamp2, HIGH)`, memberikan logika awal berupa HIGH pada relay 2.

5) *Void Loop*

Pada bagian void loop mikrokontroler melakukan beberapa proses seperti yang terlihat pada listing program berikut ini.

```

void loop() {
  check_mode:
  if (Firebase.getString(firebaseData, "/Lamp/switchL")) {
    if (firebaseData.to<String>() == "1") {goto mode_on;}
    if (firebaseData.to<String>() == "0") {goto mode_off;}}
  goto check_mode;

```

Pada awal dinyalakan, mikrokontroler melakukan proses pengecekan mode (check mode) yaitu mengambil data perintah dari firebase, jika perintah dari firebase (switch Lamp) berlogika 1 (ON), maka mikrokontroler mengaktifkan lampu dan menjalankan sistem. Mode on ditunjukkan pada listing berikut ini.

```

mode_on:
  if (Firebase.getString(firebaseData, "/Lamp/switchL")) {
    if (firebaseData.to<String>() == "0") {goto mode_off;}}
  if (Firebase.getString(firebaseData, "/Lamp/switchL2")) {
    if (firebaseData.to<String>() == "1") {
      digitalWrite (relayLamp2, LOW) ;}
    if (firebaseData.to<String>() == "0") {
      digitalWrite (relayLamp2, HIGH) ;}}
  timing2 = millis() - timing1;
  timing = timing2 / 60000;
  Firebase.setInt(firebaseData, "/Lamp/timeOnL", timing);
  int value_ldr = digitalRead(lightLDR);
  if (value_ldr == HIGH) {
    digitalWrite (relayLamp, LOW) ;
    Firebase.setString(firebaseData, "/Lamp/sensCond", "DARK");
    Firebase.setString(firebaseData, "/Lamp/lampCond", "TURN ON");
    Firebase.setString(firebaseData, "/Lamp/warningL", "Normal");}
  else {
    digitalWrite (relayLamp, HIGH) ;
    Firebase.setString(firebaseData, "/Lamp/sensCond", "LIGHT");
    Firebase.setString(firebaseData, "/Lamp/lampCond", "TURN OFF");
    Firebase.setString(firebaseData, "/Lamp/warningL", "Normal");}

```

Mode on;

- Firebase.setInt(firebaseData, "/Lamp/timeOnL", timing), yaitu mikrokontroler mengirim data waktu ON lampu (timing) ke firebase.
- int value_ldr(digitalRead(lightLDR), yaitu mikrokontroler membaca data digital dari sensor LDR, dan tipe datanya adalah integer.
- if(value_ldr==HIGH){digitalWrite(relayLamp,LOW); else{digitalWrite(relayLamp,HIGH)}}, yaitu jika LDR berlogika HIGH (gelap), maka relay lampu diberikan logika LOW (lampu dinyalakan), dan sebaliknya.

Selanjutnya pembacaan sensor arus PZEM004T, diperlihatkan pada listing berikut.

```
statusPZEM_Lamp = true;
currentLamp = pzem.current();
voltageLamp = pzem.voltage();
if (!isnan(currentLamp)) {
  Firebase.setFloat(firebaseData, "/Lamp/currentL", currentLamp);
  Firebase.setFloat(firebaseData, "/Lamp/voltageL", voltageLamp);}
else {
  Firebase.setString(firebaseData, "/Lamp/currentL", "Not detected");
  Firebase.setString(firebaseData, "/Lamp/voltageL", "Not detected");
  statusPZEM_Lamp = false;}
```

- a) statusPZEM_Lamp = true, yaitu perintah mengaktifkan sensor arus lampu.
- b) currentLamp = pzem.current(), yaitu perintah membaca arus lampu.
- c) voltageLamp = pzem.voltage(), yaitu perintah membaca tegangan lampu.
- d) Firebase.setFloat(firebaseData, "/Lamp/currentL", currentL), yaitu mengirim nilai arus lampu ke firebase.
- e) Firebase.setFloat(firebaseData, "/Lamp/voltageL", voltageL), yaitu mengirim nilai tegangan lampu ke firebase.

Selanjutnya program bersyarat pada sistem ditunjukkan pada listing berikut.

```
while ( value_1dr == HIGH && currentLamp == 0) {
  Firebase.setString(firebaseData, "/Lamp/sensCond", "DARK");
  Firebase.setString(firebaseData, "/Lamp/lampCond", "Damaged");
  Firebase.setString(firebaseData, "/Lamp/warningL", "Fault detected");
  statusPZEM_Lamp = true; currentLamp = pzem.current();}
goto mode_on;}
```

Penjelasan:

Jika waktu menyala (ON) arus lampu tetap 0A (tidak ada arus), maka mikrokontroler mengirim data ke firebase berupa informasi “Fault detected”.

Namun jika perintah dari firebase (switch Lamp) berlogika 0 (OFF), maka mikrokontroler menonaktifkan lampu dan sistem dalam kondisi standby. Seperti yang diperlihatkan pada listing berikut.

```

mode_off:
digitalWrite (relayLamp, HIGH);
if (Firebase.getString(firebaseData, "/Lamp/switchL")) {
  if (firebaseData.to<String>() == "1") { timing1 = millis();
    timing = timing1 - millis();
    Firebase.setInt(firebaseData, "/Lamp/timeOnL", timing);
    goto mode_on;}
  Firebase.setString(firebaseData, "/Lamp/timeOnL", "-");
  Firebase.setString(firebaseData, "/Lamp/sensCond", "-");
  Firebase.setString(firebaseData, "/Lamp/lampCond", "-");
  Firebase.setString(firebaseData, "/Lamp/warningL", "OFF");
  Firebase.setString(firebaseData, "/Lamp/currentL", "-");
  Firebase.setString(firebaseData, "/Lamp/voltageL", "-");}
if (Firebase.getString(firebaseData, "/Lamp/switchL2")) {
  if (firebaseData.to<String>() == "1") {digitalWrite (relayLamp2, LOW) ;}
  if (firebaseData.to<String>() == "0") {digitalWrite (relayLamp2, HIGH) ;}
  goto mode_off;
}

```

Penjelasan :

- a) digitalWrite (relayLamp, HIGH);
 if (Firebase.getString(firebaseData, "/Lamp/switchL"))
 {if (firebaseData.to<String>() == "1") { timing1 = millis();timing =
 timing1 - millis(); Firebase.setInt(firebaseData, "/Lamp/timeOnL",
 timing); goto mode_on;}}

Pada mode off, relayLamp akan diberikan logika HIGH (lampu dimatikan) dan waktu ON akan direset. Setelah itu sistem akan kembali standby, jika data switchL dari firebase berubah menjadi ON (logika 1), maka sistem akan dialihkan ke mode ON.

- b) if (Firebase.getString(firebaseData, "/Lamp/switchL2")) {
 if (firebaseData.to<String>() == "1") {digitalWrite (relayLamp2,
 LOW) ;}
 if (firebaseData.to<String>() == "0") {digitalWrite (relayLamp2,
 HIGH) ;}

Pada mode ON, jika data switchL2 dari firebase berlogika 1, maka relay lampu diberikan kondisi LOW (lampu dinyalakan). Dan jika data switchL2 dari firebase berlogika 0, maka relay lampu diberikan kondisi HIGH (lampu dimatikan). Lampu 2 tidak dipengaruhi oleh sensor LDR, namun dapat dikendalikan secara manual.

Selanjutnya pengiriman seluruh data ke server, diperlihatkan pada listing berikut.

```
String data = "?current=" + String(currentAC) + "&voltage=" + String(voltageAC)
host_post += data;
http.begin(host_post); //Specify request destination
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
httpCode=http.GET();
payload=http.getString(); // get data from webhost continously
Serial.println(payload);
http.end(); //Close connection
```

Penjelasan:

Pada baris pertama data digabung dalam sebuah baris String data kemudian digabung/dimasukkan bersama dengan data dari host_post yang selanjutnya memulai melakukan request pada host_post dengan menggunakan header dan request type GET. Hasil dari request dimasukkan dalam variabel payload dan diakhiri dengan pemutusan koneksi pada server.

c. Stok Kontak

Berikut listing untuk lampu dijelaskan secara bertahap.

1) Library (pustaka)

Sebelum memasuki program inti, di awal penulisan program pada Arduino IDE harus dimasukkan atau diinkludkan beberapa library yaitu;

- a) #include "FirebaseESP8266.h", yaitu library firebase untuk ESP8266.
- b) #include <ESP8266Wifi.h>, yaitu library wifi ESP8266.
- c) #include <PZEM004Tv30.h>, yaitu library sensor arus PZEM004T.
- d) #include <OneWire.h>, yaitu library untuk sensor suhu DS18B20
- e) #include <DallasTemperature.h>, yaitu library untuk sensor suhu DS18B20.
- f) #include "ESP8266HTTPClient", yaitu library untuk ESP8266 sebagai client yang melakukan request pada server.

```
#include <FirebaseESP8266.h>
#include <ESP8266WiFi.h>
#include <PZEM004Tv30.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <ESP8266HTTPClient.h>
```

2) Define (defenisi)

Beberapa pendefinisian yang harus dituliskan yaitu;

- a) #define FIREBASE_HOST, yang diisi dengan nama hosting firebase yang telah dibuat.

- b) `#define FIREBASE_AUTH`, yang diisi dengan firebase autentikasi firebase yang akan digunakan.
- c) `#define WIFI_SSID`, yang diisi dengan nama wifi direct yang akan digunakan untuk koneksi ke internet.
- d) `#define WIFI_PASSWORD`, yang diisi dengan password wifi yang akan digunakan.

```
//Wifi-FireBase
#define FIREBASE_HOST "fds-android-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "KywiAMupDG4wOoL3E3kVeULboWQMACAdLf9V7sv"
#define WIFI_SSID "Muliadi"
#define WIFI_PASSWORD "qwerty1234"
```

Selanjutnya pendefinisian sensor-sensor input dan aktuator output yang akan digunakan seperti pada listing program berikut.

```
//PZEMSC
PZEM004Tv30 pzem(D6, D5);
//RelaySC
#define buzzer D8
#define relaySC D1
const int oneWireBus = D2;
```

- a) PZEM004Tv30 pzem (D6,D5), yaitu penentuan pin D6 sebagai RX dan D5 sebagai TX untuk sensor arus PZEM004T.
- b) `#define buzzer D8`, yaitu penentuan pin D8 sebagai output ke buzzer.
- c) `#define relaySC D1`, penentuan pin D1 sebagai output relay untuk stok kontak.
- d) `const int oneWire = D2` , penentuan D2 sebagai input dari sensor suhu DS18B20.

3) *Variable* (variabel)

Berikut variabel yang digunakan.

```

FirebaseData firebaseData;
unsigned long timing = 0;
float currentSC;
float powerSC;
float voltageSC;
boolean statusPZEM_SC = true;
unsigned long startmillis, currentmillis;
int counter;
int jeda = 0;
OneWire oneWire(oneWireBus);
DallasTemperature sensors(&oneWire);

```

Ada beberapa tipe data yang digunakan yaitu;

- a) firebaseData, untuk mengolah data dari firebase.
- b) unsigned long timing, untuk mengolah nilai millis.
- c) unsigned long startmillis, currentmillis, untuk mengolah nilai millis.
- d) float currentSC, untuk mengolah data arus pada stok kontak dalam satuan desimal (floating).
- e) float voltageSC, untuk mengolah data tegangan pada stok kontak dalam satuan desimal (floating).
- f) float powerSC, untuk mengolah data daya pada stok kontak dalam satuan desimal (floating).
- g) boolean statusPZEM_AC untuk mengolah status dari sensor arus PZEM004T.
- h) int counter, untuk mengolah data counter (penghitung) yang dibuat manual menggunakan data integer.
- i) int jeda, untuk mengolah data jeda counter.
- j) OneWire oneWire(oneWireBus), untuk mengolah data sensor suhu DS18B20.
- k) DallasTemperature sensor (&oneWire), untuk mengolah data sensor DS18B20.

Selanjutnya variabel yang digunakan untuk pengirim data pada web server sebagai berikut :

```

HTTPClient http;
int httpCode;
String payload;
String host_post= "http://project3.monlab.site/index.php/post/shock";
String status = "Normal";
String toggle = "OFF";

```

Ada beberapa tipe data yang digunakan yaitu;

- l) HTTPClient, untuk protokol pengiriman data dari HTTP request.
- m) String payload, untuk mengolah data feedback dari request
- n) String host_post, untuk menginisialisasi domain host server

- o) String status, sebagai data yang dikirim mewakili kondisi sistem
- p) String toggle, sebagai data yang dikirim mewakili kondisi relay.

4) Void Setup

Berikut program yang pertama dan hanya sekali dibaca oleh mikrokontroler.

```
void setup() {
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    counter++;
    if (counter >= 40) {break;}
    delay(500);}
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
  //Firebase.reconnectWiFi(true);
  Firebase.setString(firebaseData, "/SC/cond", "Standby");
  sensors.begin();
  pinMode(relaySC, OUTPUT);
  pinMode (buzzer, OUTPUT);
  digitalWrite (relaySC, LOW);}
```

- a) Wifi.begin(WIFI_SSID, WIFI_PASSWORD), berfungsi untuk mengaktifkan wifi dengan menyesuaikan nama SSID dengan password.
- b) counter++ , berfungsi untuk menghitung timer sampai 40 detik, jika telah melewati 40 detik namun tidak dapat terhubung dengan wifi tersebut maka mikrokontroler beralih pada mode offline.
- c) Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH), berfungsi untuk menghubungkan mikrokontroler dengan firebase dengan menyesuaikan firebase host dengan firebase auth. Proses ini akan berhasil jika mikrokontroler dalam kondisi online. Namun jika dalam kondisi offline maka bagian ini akan diabaikan atau dilewatkan, dengan demikian mikrokontroler tidak akan mengirim data ke firebase.
- d) Firebase.setString(firebaseData, "/SC/cond", "Standby"), berfungsi untuk mengirim data ke firebase yang berisikan tulisan "Standby".
- e) sensors.begin(), untuk memulai menjalankan sensor DS18B20.
- f) pinMode (relaySC, OUTPUT), untuk menyatakan bahwa relaySC merupakan OUTPUT.
- g) pinMode (buzzer, OUTPUT), untuk menyatakan bahwa buzzer merupakan OUTPUT.
- h) digitalWrite(relaySC, LOW), memberikan logika awal berupa LOW (ON) pada relay stok kontak.

5) *Void Loop*

Pada bagian void loop mikrokontroler melakukan beberapa proses seperti yang terlihat pada listing program berikut ini.

```
void loop() {
  check_mode:
  if (Firebase.getString(firebaseData, "/SC/switchSC")) {
    if (firebaseData.to<String>() == "1") {goto mode_on;}
    if (firebaseData.to<String>() == "0") {goto mode_off;}}
  goto check_mode;
```

Pada awal dinyalakan, mikrokontroler melakukan proses pengecekan mode (check mode) yaitu mengambil data perintah dari firebase, jika perintah dari firebase (switch SC) berlogika 1 (ON), maka mikrokontroler mengaktifkan lampu dan menjalankan sistem. Mode on ditunjukkan pada listing berikut ini.

```
mode_on:
  if (Firebase.getString(firebaseData, "/SC/switchSC")) {
    if (firebaseData.to<String>() == "0") {goto mode_off;}}
  timing2 = millis() - timing1;
  timing = timing2 / 60000;
  Firebase.setInt(firebaseData, "/SC/timeOnSC", timing);
  digitalWrite (relaySC, HIGH);
  sensors.requestTemperatures();
  float temperatureC = sensors.getTempCByIndex(0);
  Firebase.setFloat (firebaseData, "/SC/thermalSC", temperatureC);
  Firebase.setString(firebaseData, "/SC/warningSC", "-");
```

Mode on;

- d) `Firebase.setInt(firebaseData, "/SC/timeOnSC", timing)`, yaitu mikrokontroler mengirim data waktu ON stok kontak (timing) ke firebase.
- e) `digitalWrite(relaySC, HIGH)`, yaitu mikrokontroler memberikan kondisi HIGH (ON) pada relay stok kontak.
- f) `sensors.requestTemperatures()`, yaitu mikrokontroler meminta data temperature sensor DS18B20.
- g) `float temperatureC = sensors.getTempCByIndex(0)`, yaitu mikrokontroler mengambil data temperature sensor DS18B20 kemudian dikonversi menjadi celsius dengan bentuk desimal (floating).
- h) `Firebase.setFloat(firebaseData, "/SC/thermalSC", temperatureC)`, yaitu firebase mengirim data suhu sensor DS18B20 ke firebase.

Selanjutnya pembacaan sensor arus PZEM004T, diperlihatkan pada listing berikut.

```
//Read Current Sensor
statusPZEM_SC = true;
currentSC = pzem.current();
voltageSC = pzem.voltage();
if (!isnan(currentSC)) {
  Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC);
  Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC);}
else {
  Firebase.setString(firebaseData, "/SC/currentSC", "-");
  Firebase.setString(firebaseData, "/SC/voltageSC", "-");
  Firebase.setString(firebaseData, "/SC/conditionSC", "Not detected");
  Firebase.setString(firebaseData, "/SC/warningSC", "Check current sensor");
  statusPZEM_SC = false;}
```

- a) statusPZEM_SC = true, yaitu perintah mengaktifkan sensor arus stok kontak.
- b) currentSC = pzem.current(), yaitu perintah membaca arus stok kontak.
- c) voltageSC = pzem.voltage(), yaitu perintah membaca tegangan stok kontak.
- d) Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC), yaitu mengirim nilai arus stok kontak ke firebase.
- e) Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC), yaitu mengirim nilai tegangan stok kontak ke firebase.

Selanjutnya program bersyarat pada sistem ditunjukkan pada listing berikut.

```
if (temperatureC >= 71) {
  while (1) {
    digitalWrite (relaySC, LOW);
    digitalWrite (buzzer, HIGH); delay (100);
    digitalWrite (buzzer, LOW); delay (100);
    digitalWrite (buzzer, HIGH); delay (100);
    digitalWrite (buzzer, LOW); delay (100);
    Firebase.setFloat (firebaseData, "/SC/thermalSC", temperatureC);
    Firebase.setString(firebaseData, "/SC/warningSC", "Overheat");
    Firebase.setString(firebaseData, "/SC/conditionSC", "OFF");
    if (temperatureC < 71) {
      digitalWrite (relaySC, HIGH);break;}}}
```

Penjelasan:

Jika suhu stok kontak (sensor DS18B20) mencapai 71 derajat celcius ke atas, maka relay stok kontak akan di OFF kan dan buzzer akan mengeluarkan bunyi beep. Sedangkan jika suhu stok kontak dibawah 71 derajat celcius maka relay stok kontak akan di ON kan kembali dan buzzer berhenti mengeluarkan bunyi beep.

```

powerSC = pzem.power();
if ( !isnan(powerSC) ) {
  Firebase.setFloat(firebaseData, "/SC/powerSC", powerSC);}
else {Firebase.setString(firebaseData, "/SC/powerSC", "-");}
if (powerSC == 0.0)
{ Firebase.setString(firebaseData, "/SC/conditionSC", "No device installed");
  Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC);
  Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC);
  Firebase.setString(firebaseData, "/SC/warningSC", "-");
  Firebase.setFloat(firebaseData, "/SC/powerSC", powerSC );
  jeda = 0;}

```

Penjelasan:

Mikrokontroler mengambil data daya (power) dari sensor PZEM004T, lalu mengirim ke firebase. Jika nilai daya adalah 0 Watt, maka mikrokontroler akan mengirim data kondisi stok kontak yaitu “Not device installed”.

```

if (powerSC >= 1.0 && powerSC <= 400)
{ Firebase.setString(firebaseData, "/SC/conditionSC", "ON");
  Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC);
  Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC);
  Firebase.setString(firebaseData, "/SC/warningSC", "-");
  Firebase.setFloat(firebaseData, "/SC/powerSC", powerSC );
  jeda = 0;}

```

Penjelasan :

Jika daya pada stok kontak berkisar antara 1 – 400 Watt, maka mikrokontroler akan mengirim nilai daya yang terukur ke firebase serta tidak memberikan peringatan dalam hal ini stok kontak dalam kondisi aman (tidak kelebihan beban).

```

if (powerSC > 400 && powerSC <= 420)
{ Firebase.setString(firebaseData, "/SC/conditionSC", "ON");
  Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC);
  Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC);
  Firebase.setString(firebaseData, "/SC/warningSC", "Almost overload");
  Firebase.setFloat(firebaseData, "/SC/powerSC", powerSC );
  jeda = 0;}

```

Penjelasan :

Jika daya pada stok kontak berkisar antara 401 – 420 Watt, maka mikrokontroler akan mengirim nilai daya yang terukur ke firebase serta

memberikan peringatan “Almost overload”, dalam hal ini stok kontak dalam kondisi kurang aman (menghampiri kelebihan beban).

```

if (powerSC > 420) {
  digitalWrite (buzzer, HIGH); delay(10);
  digitalWrite (buzzer, LOW); delay(10);
  digitalWrite (buzzer, HIGH); delay(10);
  digitalWrite (buzzer, LOW); delay(10);
  currentmillis = millis ();
  if (currentmillis - startmillis >= 1000) { startmillis = currentmillis; jeda++;
    if (jeda >= 30) {
      while (1) {
        digitalWrite (relaySC, LOW);
        statusPZEM_SC = false;
        currentSC = pzem.current();
        powerSC = pzem.power();
        jeda = 0; } } }
  Firebase.setString(firebaseData, "/SC/conditionSC", "ON");
  Firebase.setFloat(firebaseData, "/SC/currentSC", currentSC);
  Firebase.setFloat(firebaseData, "/SC/voltageSC", voltageSC );
  Firebase.setString(firebaseData, "/SC/warningSC", "Unplug any load and press reset button");
  Firebase.setFloat(firebaseData, "/SC/powerSC", powerSC ); } }

```

Penjelasan :

Jika daya pada stok kontak melebihi 420 Watt, maka buzzer akan mengeluarkan bunyi beep pendek secara terus menerus hingga 30 detik. Jika beban dilepas dan daya kembali berada di bawah 420 Watt maka bunyi beep akan berhenti, namun jika bunyi beep pendek diabaikan hingga 31 detik, maka relay stok kontak akan di OFF kan dan mikrokontroller akan mengirim data ke firebase berupa informasi “Unplug and press reset button”. Untuk menyalakan kembali stok kontak, maka beberapa beban harus dilepas terlebih dahulu kemudian menekan tombol reset supaya sistem set up ulang.

Namun jika perintah dari firebase (switch SC) berlogika 0 (OFF), maka mikrokontroler menonaktifkan lampu dan sistem dalam kondisi standby. Seperti yang diperlihatkan pada listing berikut.

```

mode_off:
digitalWrite (relaySC, LOW);
if (Firebase.getString(firebaseData, "/SC/switchSC")) {
  if (firebaseData.to<String>() == "1") {timing1 = millis();
    timing = timing1 - millis();
    Firebase.setInt(firebaseData, "/SC/timeOnSC", timing);
    goto mode_on;}
  //timing = 0;
  Firebase.setString(firebaseData, "/SC/timeOnSC", "-");
  Firebase.setString(firebaseData, "/SC/currentSC", "-");
  Firebase.setString(firebaseData, "/SC/voltageSC", "-");
  Firebase.setString(firebaseData, "/SC/conditionSC", "-");
  Firebase.setString(firebaseData, "/SC/warningSC", "OFF");
  Firebase.setString(firebaseData, "/SC/powerSC", "-");}
goto mode_off;

```

Penjelasan:

Pada mode OFF relay stok kontak akan di OFF kan, mikrokontroler akan mengirim beberapa data ke firebase dan sistem berada dalam kondisi standby.

Selanjutnya pengiriman seluruh data ke server, diperlihatkan pada listing berikut.

```

String data = "?current=" + String(currentAC) + "&voltage=" + String(voltageAC)
host_post += data;
http.begin(host_post); //Specify request destination
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
httpCode=http.GET();
payload=http.getString(); // get data from webhost continously
Serial.println(payload);
http.end(); //Close connection

```

Penjelasan:

Pada baris pertama data digabung dalam sebuah baris String data kemudian digabung/dimasukkan bersama dengan data dari host_post yang selanjutnya memulai melakukan melakukan request pada host_post dengan menggunakan header dan request type GET. Hasil dari request dimasukkan dalam variabel payload dan diakhiri dengan pemutusan koneksi pada server.

d. Panel (Power Consumption and Lost)

Berikut listing untuk lampu dijelaskan secara bertahap.

1) Library (pustaka)

Sebelum memasuki program inti, di awal penulisan program pada Arduino IDE harus dimasukkan atau diinkludkan beberapa library yaitu;

- a) `#include "FirebaseESP8266.h"`, yaitu library firebase untuk ESP8266.
- b) `#include <ESP8266Wifi.h>`, yaitu library wifi ESP8266.
- c) `# <PZEM004Tv30.h>`, yaitu library sensor arus PZEM004T.
- d) `#include "ESP8266HTTPClient"`, yaitu library untuk ESP8266 sebagai client yang melakukan request pada server.

```
#include <FirebaseESP8266.h>
#include <ESP8266Wifi.h>
#include <PZEM004Tv30.h>
#include <ESP8266HTTPClient.h>
```

2) Define (defenisi)

Beberapa pendefinisian yang harus dituliskan yaitu;

- a) `#define FIREBASE_HOST`, yang diisi dengan nama hosting firebase yang telah dibuat.
- b) `#define FIREBASE_AUTH`, yang diisi dengan firebase autentikasi firebasae yang akan digunakan.
- c) `#define WIFI_SSID`, yang diisi dengan nama wifi direct yang akan digunakan untuk koneksi ke internet.
- d) `#define WIFI_PASSWORD`, yang diisi dengan password wifi yang akan digunakan.

```
//Wifi-FireBase
#define FIREBASE_HOST "fds-android-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "KywiAMupDG4wOoL3E3kVeULboWQMACAdLf9V7sv"
#define WIFI_SSID "Muliadi"
#define WIFI_PASSWORD "qwerty1234"
```

Selanjutnya pendefenisian sensor-sensor input dan aktuator output yang akan digunakan seperti pada listing program berikut.

```
//PZEMSC
PZEM004Tv30 pzem(D6, D5);
```

- a) PZEM004Tv30 pzem (D6,D5), yaitu penentuan pin D6 sebagai RX dan D5

- 6) *Variable* (variabel)
Berikut variabel yang digunakan.

```

FirebaseData firebaseData;
float currentPanelCons;
float voltagePanelCons;
boolean statusPZEM_Panel = true;
int counter;

```

Ada beberapa tipe data yang digunakan yaitu;

- g) firebaseData, untuk mengolah data dari firebase.
- h) float currentPanel, untuk mengolah data arus pada panel dalam satuan desimal (floating).
- i) float voltagePanel, untuk mengolah data tegangan pada panel dalam satuan desimal (floating).
- j) boolean statusPZEM_Panel untuk mengolah status dari sensor arus PZEM004T.
- k) int counter, untuk mengolah data counter (penghitung) yang dibuat manual menggunakan data integer.

Selanjutnya variabel yang digunakan untuk pengirim data pada web server sebagai berikut :

```

HTTPClient http;
int httpCode;
String payload;
String host_post= "http://project3.monlab.site/index.php/post/panel_cons";
String status;

```

Ada beberapa tipe data yang digunakan yaitu;

- a) HTTPClient, untuk protokol pengiriman data dari HTTP request.
- b) String payload, untuk mengolah data feedback dari request
- c) String host_post, untuk menginisialisasi domain host server
- d) String status, sebagai data yang dikirim mewakili kondisi sistem

- 3) *Void Setup*

Berikut program yang pertama dan hanya sekali dibaca oleh mikrokontroler.

```
void setup() {
  Serial.begin(115200);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    counter++;
    if (counter >= 40) {
      break;
    }
    delay(500);
  }
  Serial.println("WiFi connected");
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
```

- i) `Wifi.begin(WIFI_SSID, WIFI_PASSWORD)`, berfungsi untuk mengaktifkan wifi dengan menyesuaikan nama SSID dengan password.
- j) `counter++`, berfungsi untuk menghitung timer sampai 40 detik, jika telah melewati 40 detik namun tidak dapat terhubung dengan wifi tersebut maka mikrokontroler beralih pada mode offline.
- k) `Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH)`, berfungsi untuk menghubungkan mikrokontroler dengan firebase dengan menyesuaikan firebase host dengan firebase auth. Proses ini akan berhasil jika mikrokontroler dalam kondisi online. Namun jika dalam kondisi offline maka bagian ini akan diabaikan atau dilewatkan, dengan demikian mikrokontroler tidak akan mengirim data ke firebase.

4) *Void Loop*

Pada bagian void loop mikrokontroler melakukan beberapa proses seperti yang terlihat pada listing program berikut ini.

```

statusPZEM_Panel = true;
currentPanelCons = pzem.current();
voltagePanelCons = pzem.voltage();
if (!isnan(currentPanelCons)) {
    Firebase.setFloat(firebaseData, "/Panel/currentCons", currentPanelCons);
    Firebase.setFloat(firebaseData, "/Panel/voltageCons", voltagePanelCons);
    Firebase.setString(firebaseData, "/Panel/warningP", "Normal");
    status = "Normal";
}
else {
    Firebase.setString(firebaseData, "/Panel/currentCons", "-");
    Firebase.setString(firebaseData, "/Panel/voltageCons", "-");
    Firebase.setString(firebaseData, "/Panel/warningP", "Check_current_sensor_cons");
    statusPZEM_Panel = false;
    status = "Failure";
}

```

Pada awal dinyalakan, mikrokontroler melakukan proses pembacaan sensor, kemudian jika pembacaan berhasil mikrokontroler akan melakukan pengiriman data ke firebase antara lain :

- a) `Firebase.setString(firebaseData, "/Panel/CurrentCons", currentPanelCons)`, yaitu mikrokontroler mengirim data arus panel (`currentPanelCons`) ke firebase.
- b) `Firebase.setString(firebaseData, "/Panel/voltageCons", voltagePanelCons)`, yaitu mikrokontroler mengirim data tegangan panel (`voltagePanelCons`) ke firebase.
- c) `Firebase.setString(firebaseData, "/Panel/warningP", "Normal")`, yaitu mikrokontroler mengirim data string Normal ke firebase.

Selanjutnya pengiriman seluruh data ke server, diperlihatkan pada listing berikut.

```

String data = "?current=" + String(currentAC) + "&voltage=" + String(voltageAC)
host_post += data;
http.begin(host_post); //Specify request destination
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
httpCode=http.GET();
payload=http.getString(); // get data from webhost continously
Serial.println(payload);
http.end(); //Close connection

```

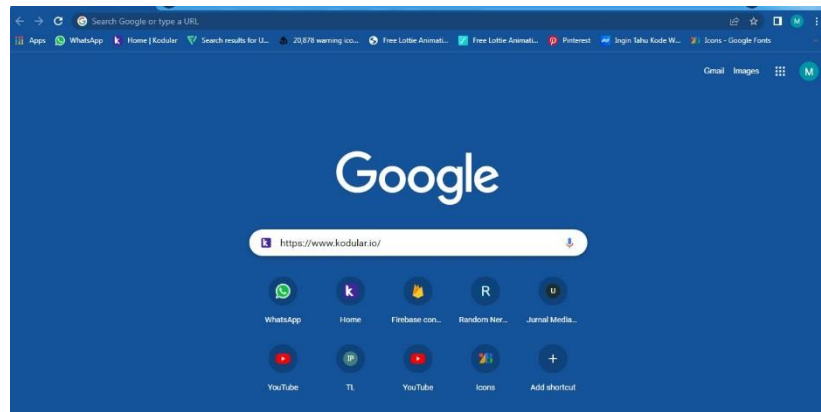
Penjelasan:

Pada baris pertama data digabung dalam sebuah baris `String data` kemudian digabung/dimasukkan bersama dengan data dari `host_post` yang selanjutnya memulai melakukan request pada `host_post` dengan menggunakan header dan request type GET. Hasil dari request dimasukkan dalam variabel `payload` dan diakhiri dengan pemutusan koneksi pada server.

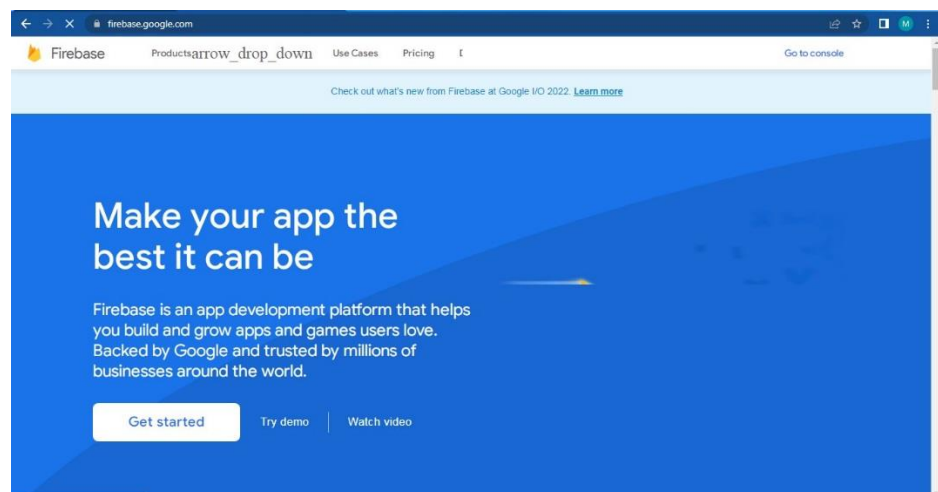
2. Pembuatan Firebase

Firebase merupakan tools database dari Google yang dapat kita gunakan untuk mengembangkan aplikasi. Langkah-langkah pembuatan database pada firebase yaitu sebagai berikut;

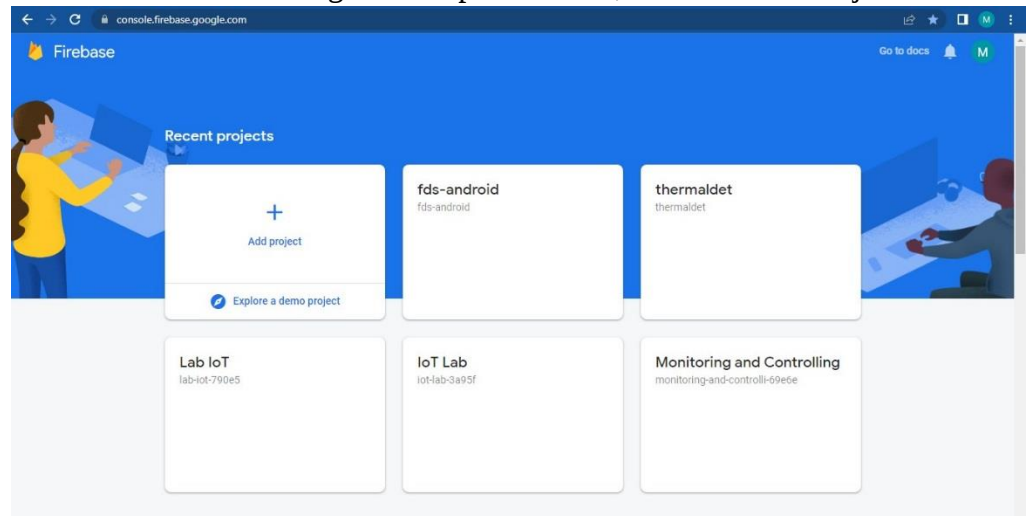
- a. Buka aplikasi google chrome (atau sejenisnya), pastikan komputer terkoneksi dengan internet (online).



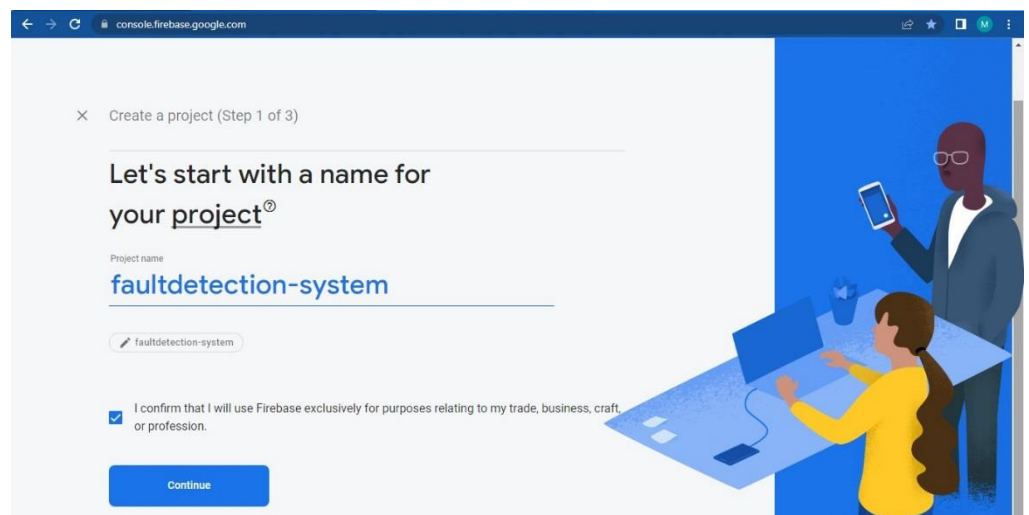
- a. Buka laman <https://firebase.google.com/> , seperti gambar berikut ini. Lalu klik Go to Console.



- b. Setelah itu akan muncul gambar seperti berikut, lalu klik Add Project.

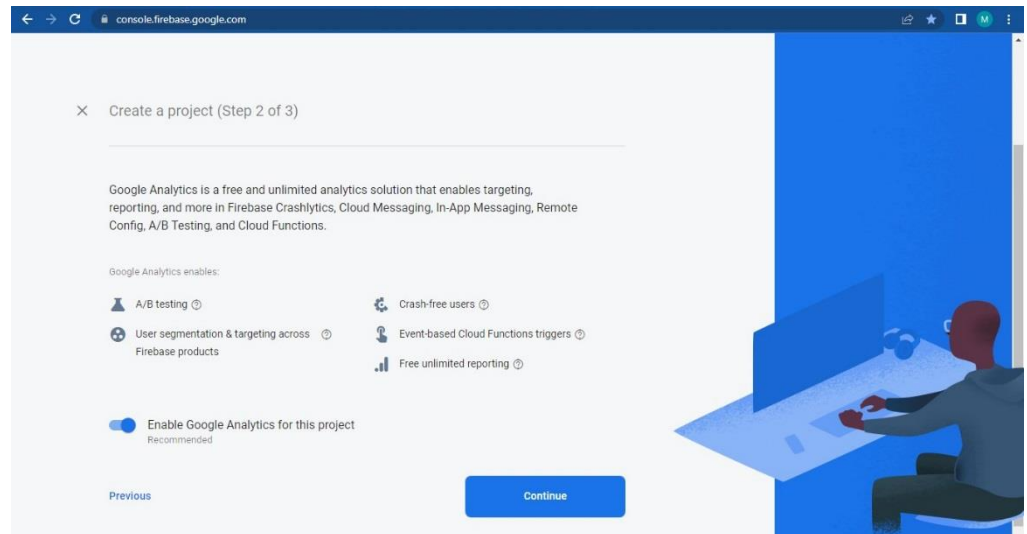


- c. Setelah mengklik Add Project, maka akan muncul gambar seperti di bawah ini.



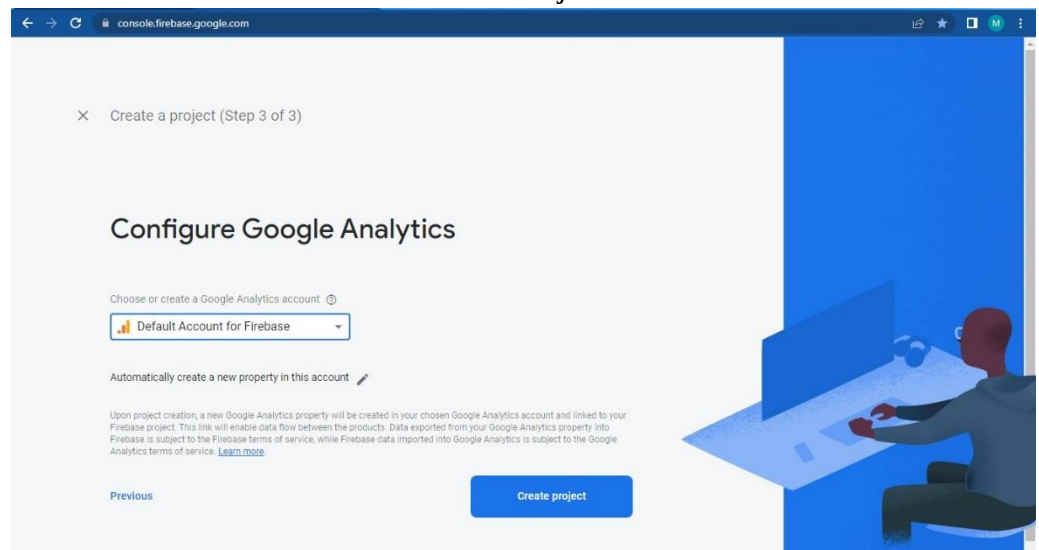
Pada bagian ini, masukkan nama project yang akan dibuat pada bagian Project Name.

- d. Setelah mengisi nama project, klik Continue maka akan muncul tampilan sebagai berikut.

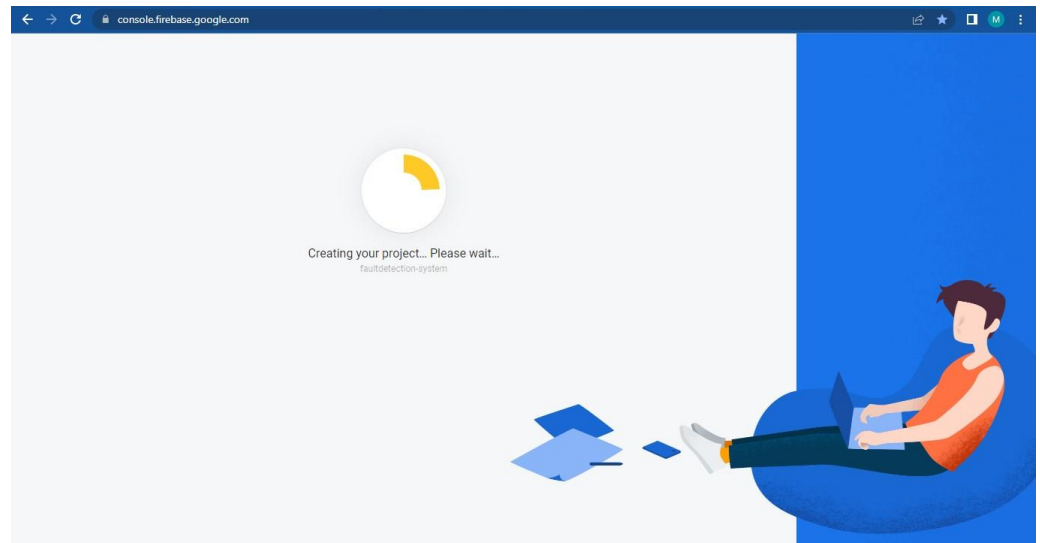


Klik Continue.

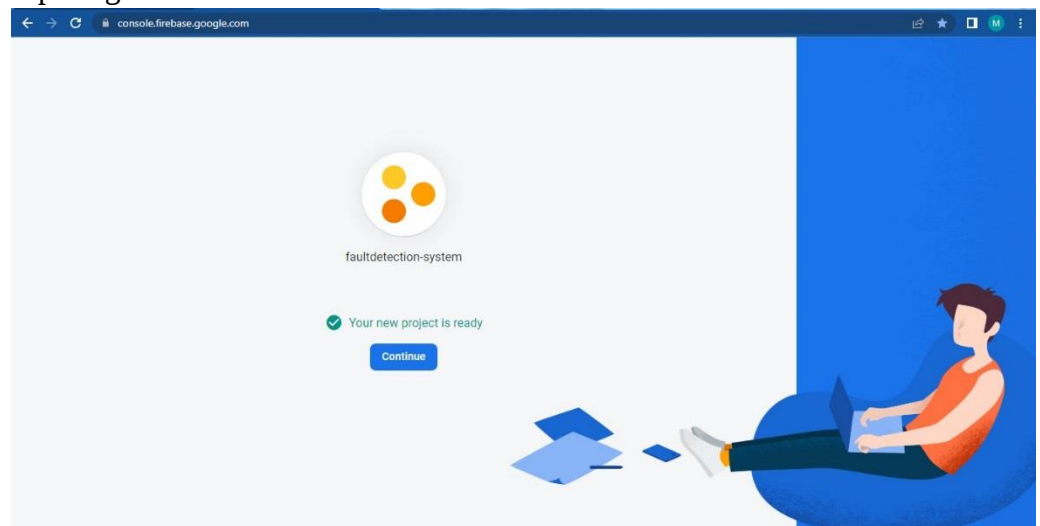
- e. Setelah itu akan muncul tampilan seperti berikut ini. Set bagian Choose and Create a Google Analytics Account menjadi Default Account for Firebase. Lalu klik Create Project.



- f. Setelah itu akan muncul tampilan seperti di bawah ini.

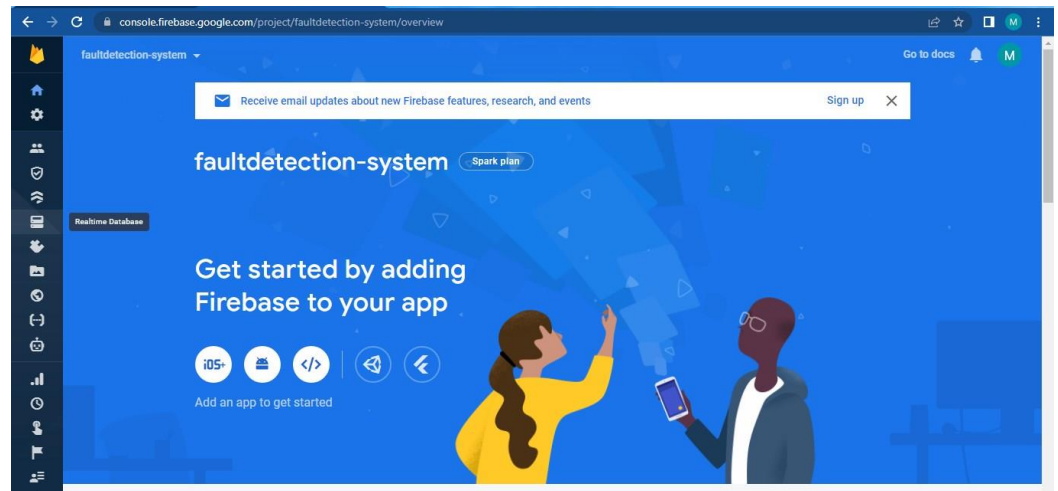


- g. Silahkan tunggu hingga sukses 100%. Lalu akan muncul tampilan seperti gambar berikut.

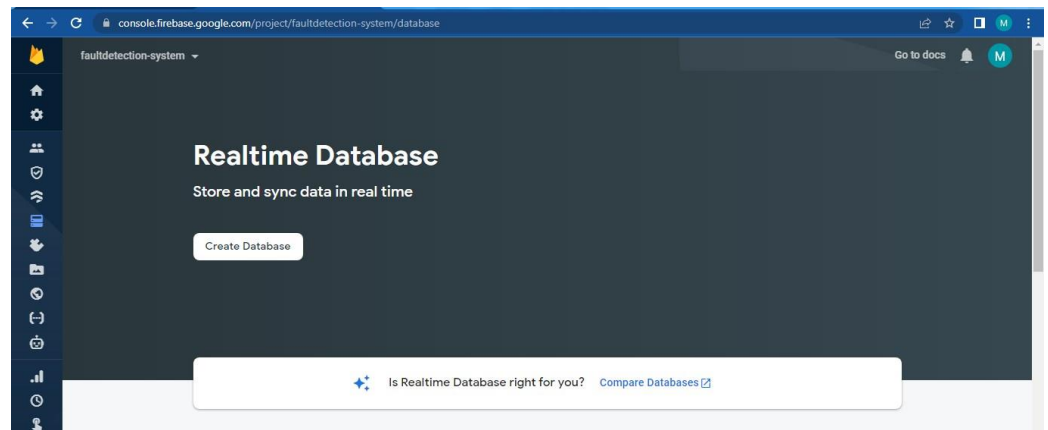


Bagian tersebut menandakan project telah berhasil dibuat. Silahkan klik Continue.

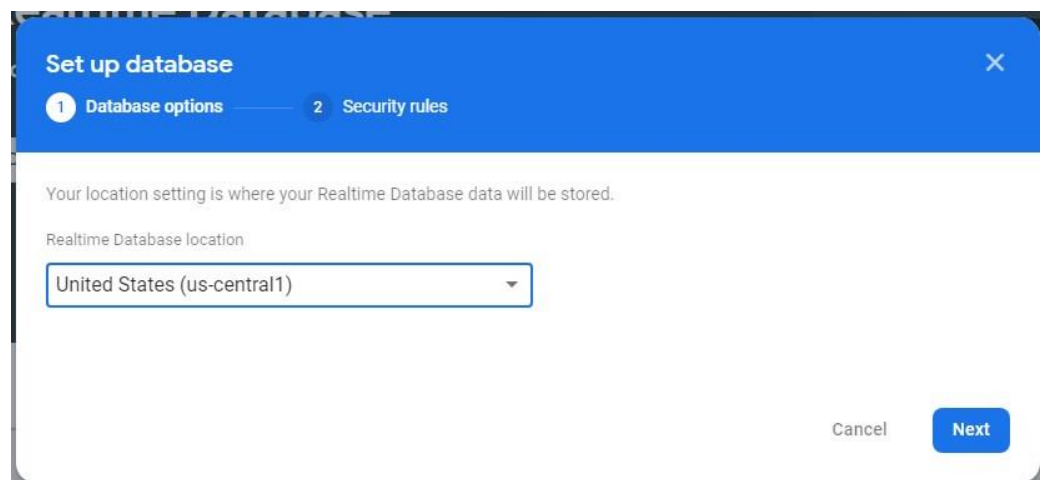
- h. Setelah mengklik Continue maka akan muncul gambar seperti berikut ini.



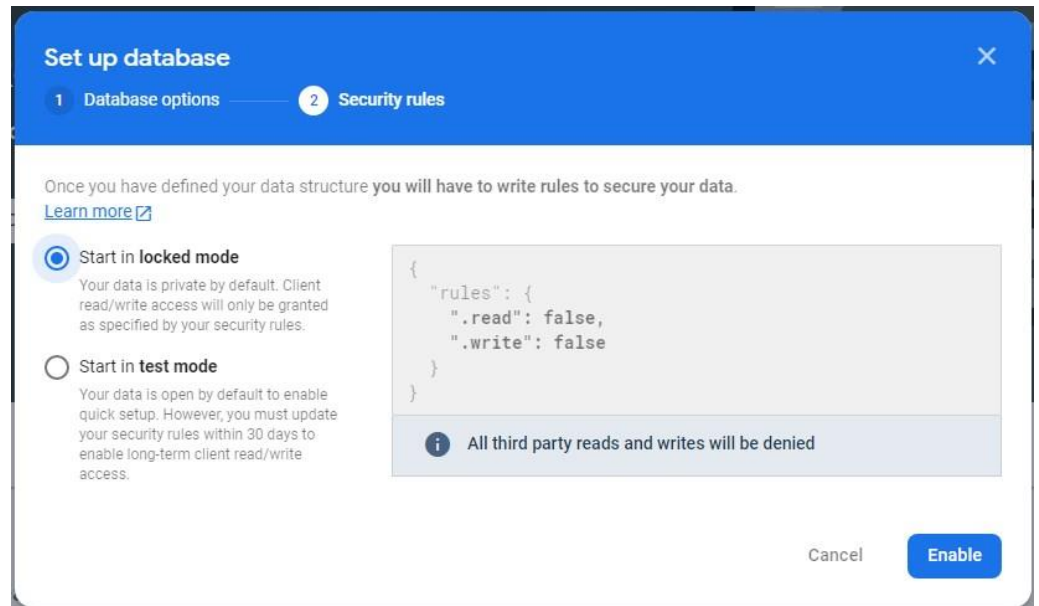
Pada bagian ini silahkan memilih dan klik icon Realtime Database. Setelah itu akan muncul gambar seperti gambar berikut.



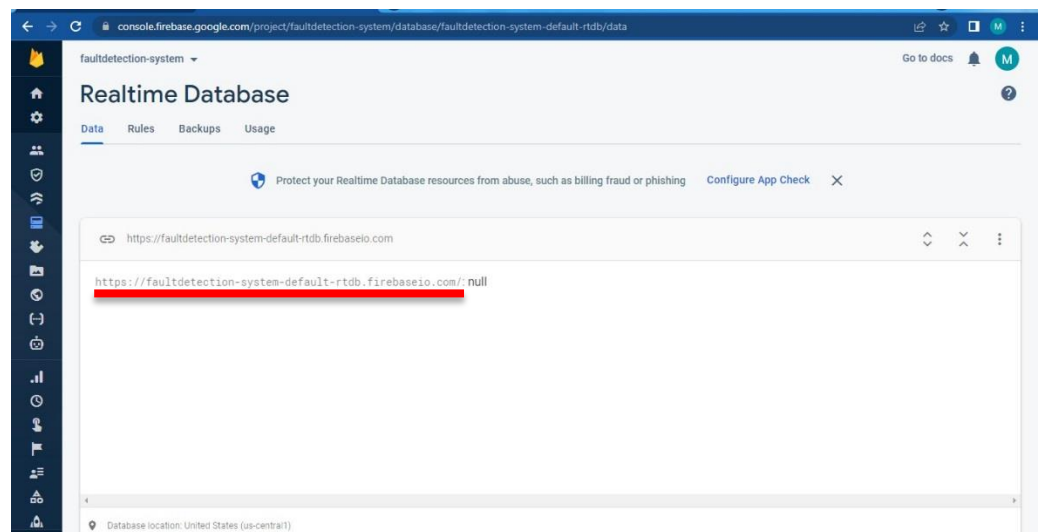
Setelah itu klik Create Database, lalu akan muncul lagi seperti berikut.



Pilih dan klik United States (us-central1), lalu klik Next. Setelah itu akan dimunculkan Step Database.

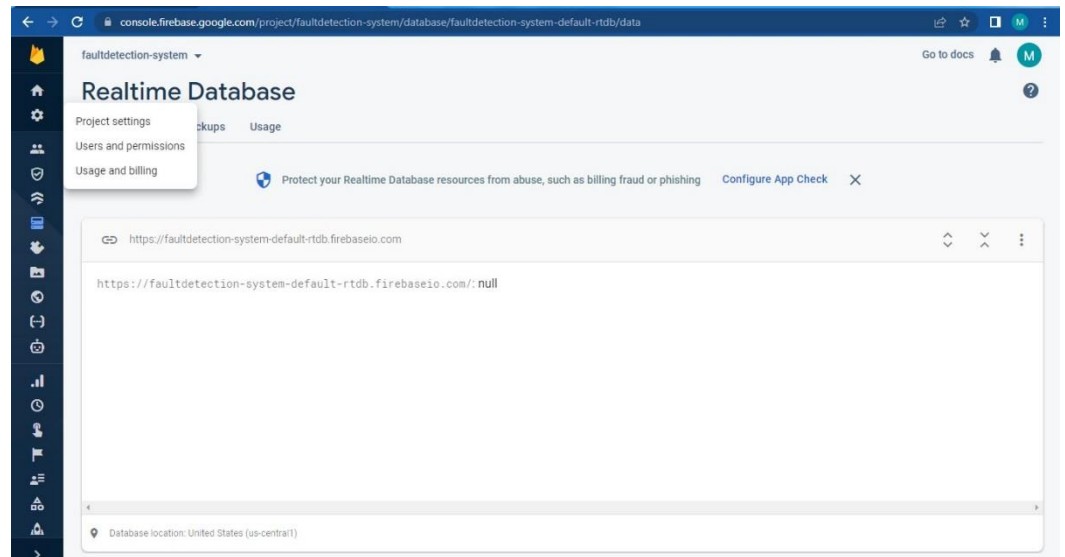


Pada bagian pilih Start in locked mode lalu klik Enable. Setelah itu akan muncul database baru yang masih kosong dan siap untuk digunakan. Terlihat pada gambar berikut ini.

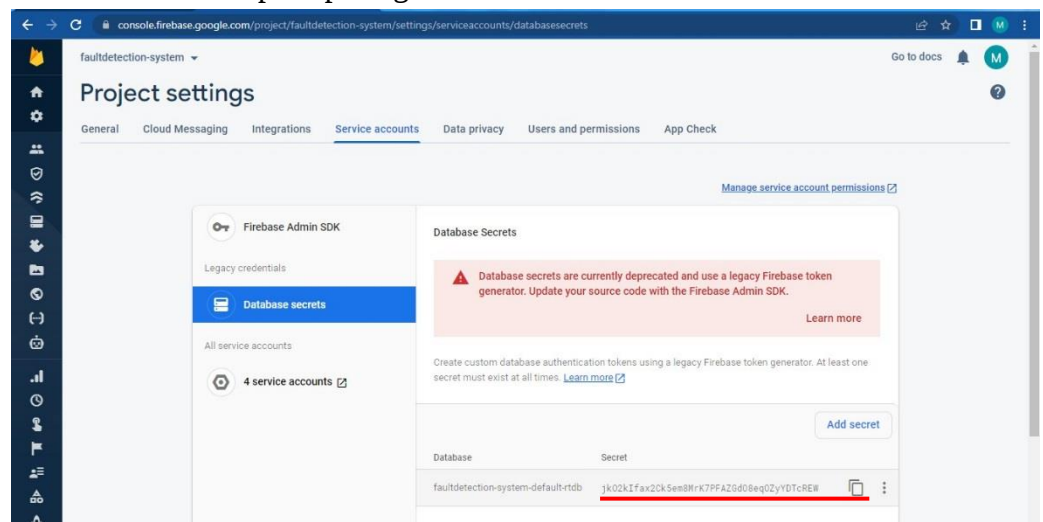


Bagian yang ditandai warna merah merupakan Firebase Host yang akan diinputkan pada program arduino untuk proses posting data pada firebase.

- i. Sedangkan untuk mendapatkan nomor unik atau disebut Firebase Authentication, dapat dilakukan melalui cara berikut ini.

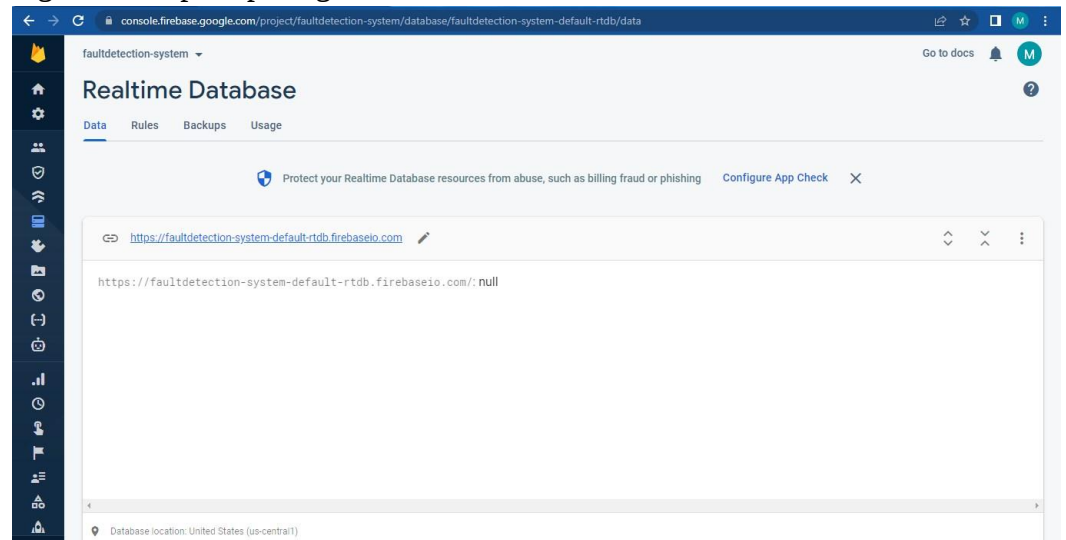


Pada bagian Project klik Project Setting, lalu pilih dan klik menu Service Account seperti pada gambar berikut ini.



Pada bagian Service Account klik Database Secret maka kode Auth akan dimunculkan seperti yang telah ditandai warna merah. Kode tersebut akan diinputkan pada bagian Firebase Auth di aplikasi mikrokontroler.

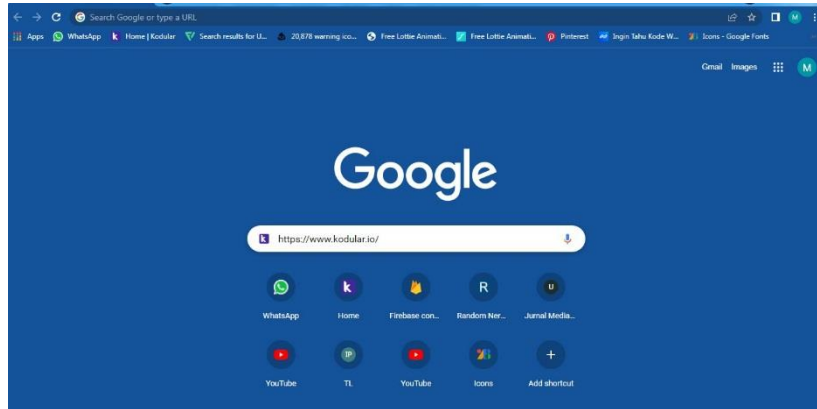
- j. Setelah melakukan langkah-langkah tersebut, maka akan kembali pada layar utama database yang baru saja dibuat dan siap untuk digunakan seperti pada gambar berikut ini.



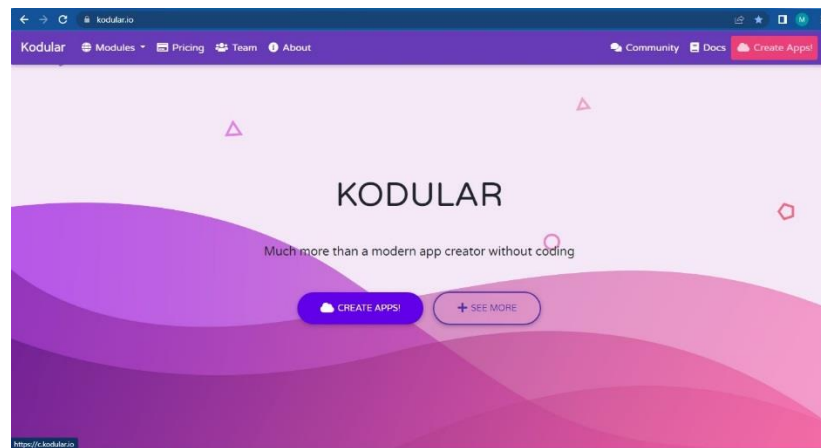
3. Pembuatan Aplikasi Android

Aplikasi yang digunakan pada penelitian ini dibuat menggunakan salah satu website online yang disebut Kodular. Kodular merupakan situs web yang menyediakan tools yang menyerupai MIT App Inventor untuk membuat aplikasi android dengan menggunakan block programming. Adapun software yang digunakan dalam membuat aplikasi yaitu google chrome, website kodular di “<https://www.kodular.io/>“, dan kodular companion untuk testing aplikasi di handphone android. Berikut langkah-langkahnya.

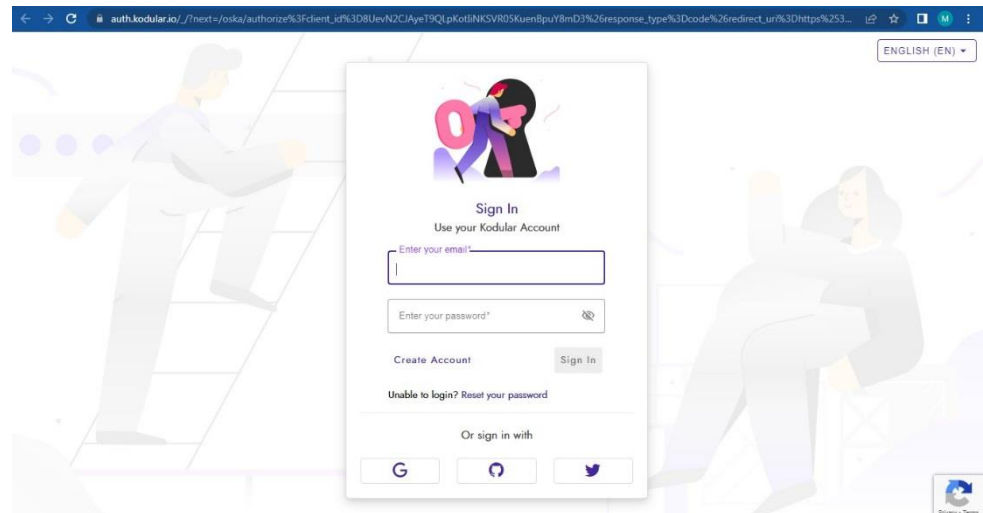
- k. Buka aplikasi google chrome (atau sejenisnya), pastikan komputer terkoneksi dengan internet (online).



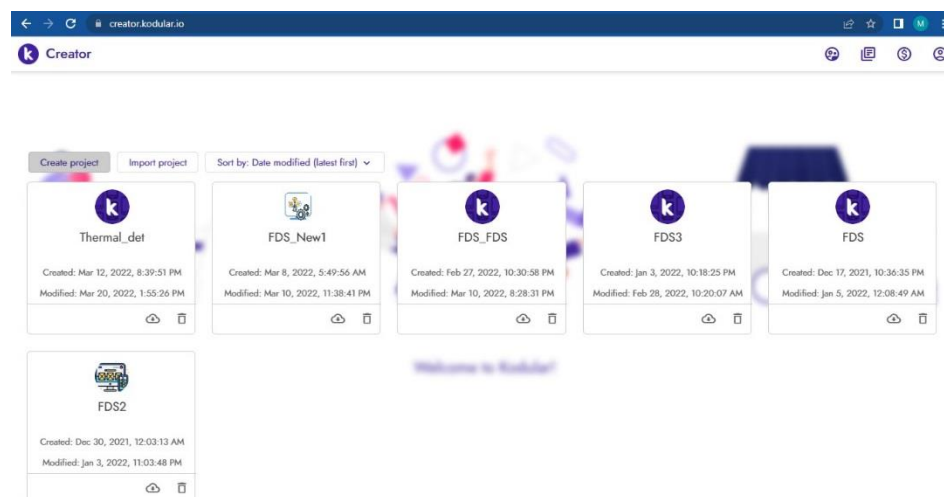
- l. Buka laman <https://www.kodular.io/>, seperti gambar berikut ini. Lalu klik Create Apps.



- m. Setelah mengklik Create Apps, maka akan muncul tampilan seperti berikut ini. Pada bagian ini terlebih dahulu harus membuat akun dengan mengklik Create Account lalu Sign in menggunakan akun yang telah dibuat.

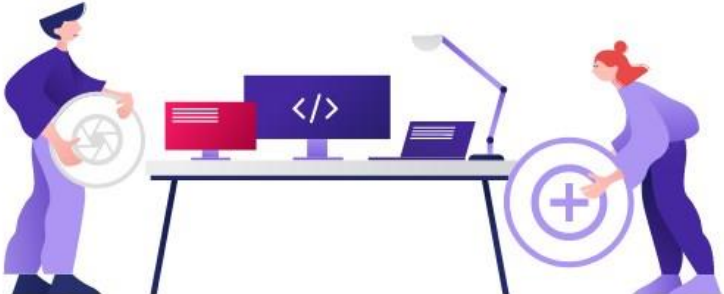


n. Jika berhasil Sign in maka akan muncul tampilan seperti berikut ini.



Untuk membuat project baru, klik “Create project” maka akan muncul tampilan seperti berikut ini.

Create new project



Give your new project a name

Project name

Project names must start with a letter and can contain only letters, numbers, and underscores

Cancel Next

Ketikkan nama project, kemudian klik “Next”. Setelah itu akan muncul lagi tampilan seperti gambar berikut ini.

Configure your project

These settings can be changed later



App Name

Theme
 Default

Minimum SDK Level
 Android 5.0 - 5.0.2 (API 21)

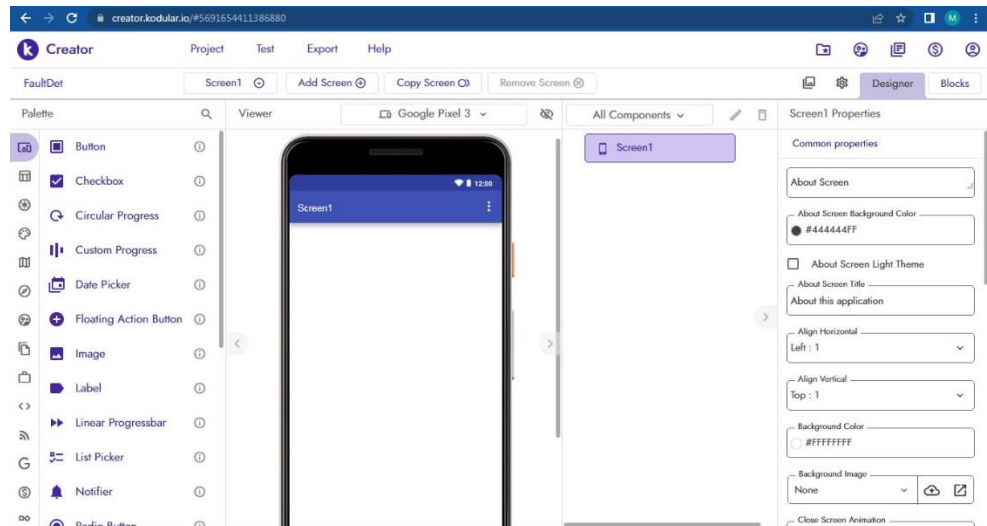
Package Name

Primary Color
 #3F51B5FF

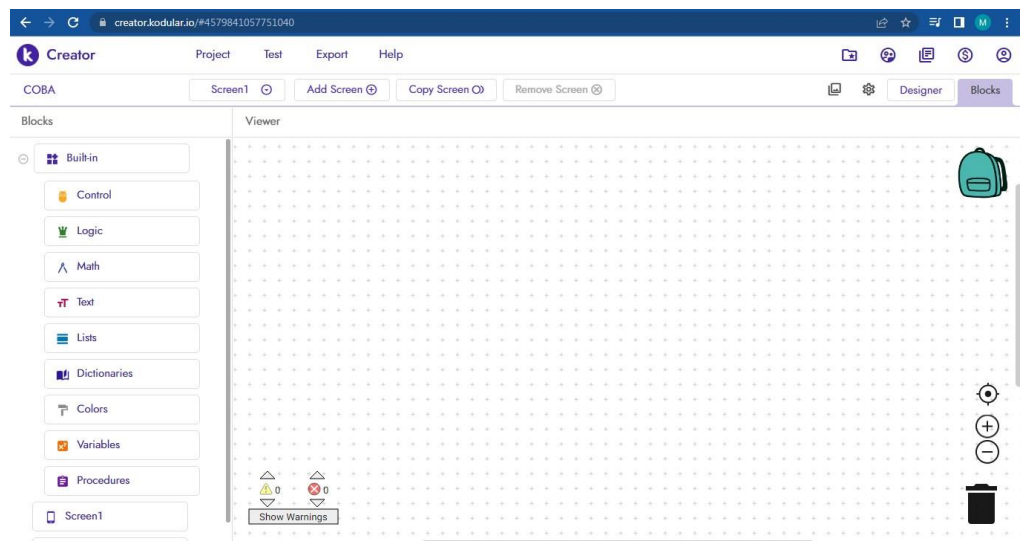
Back Cancel Finish

Pada bagian SDK level, silahkan tentukan minimal versi android yang dapat menggunakan aplikasi tersebut nantinya. Setelah itu klik “Finish”.

Setelah mengklik “Finish” maka akan terbuka lembar kerja seperti gambar berikut ini.

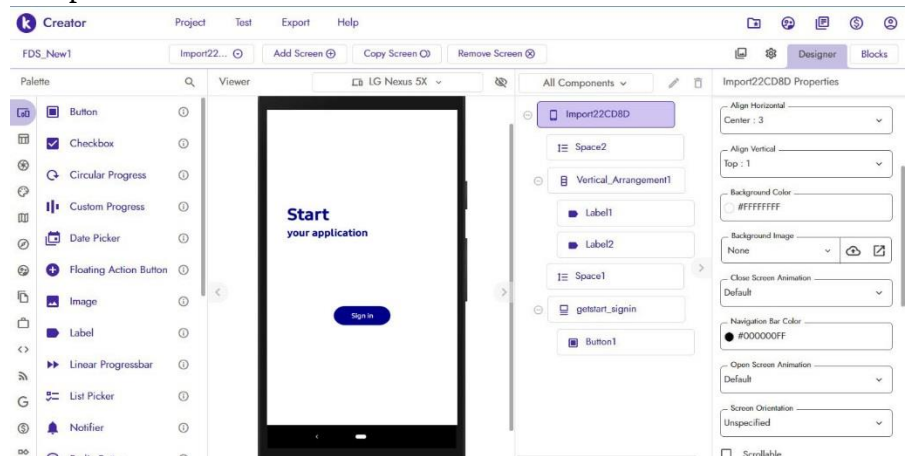


Pada lembar kerja project tersebut akan didesain tampilan aplikasi maupun coding. Gambar di atas menunjukkan lembar kerja bagian “Designer”, selain itu juga terdapat lembar kerja untuk codingan program atau “Blocks” seperti yang terlihat pada gambar berikut ini.

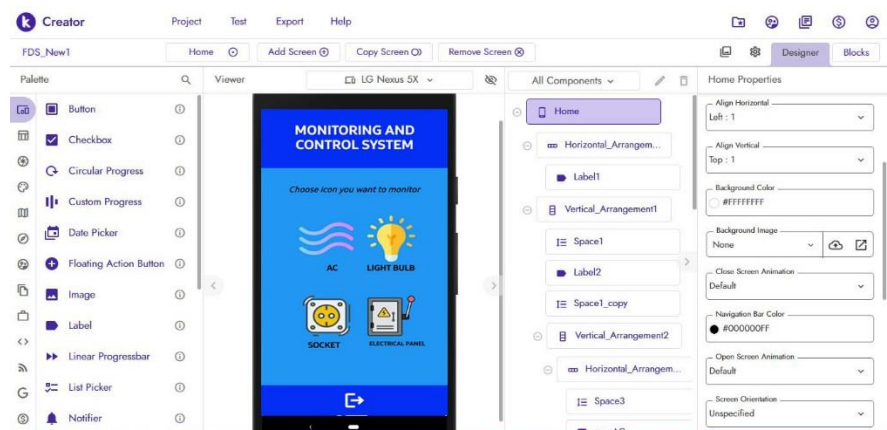


Berikut ini akan diperlihatkan dan dijelaskan mengenai desain dan codingan blok dari aplikasi android yang telah dibuat.

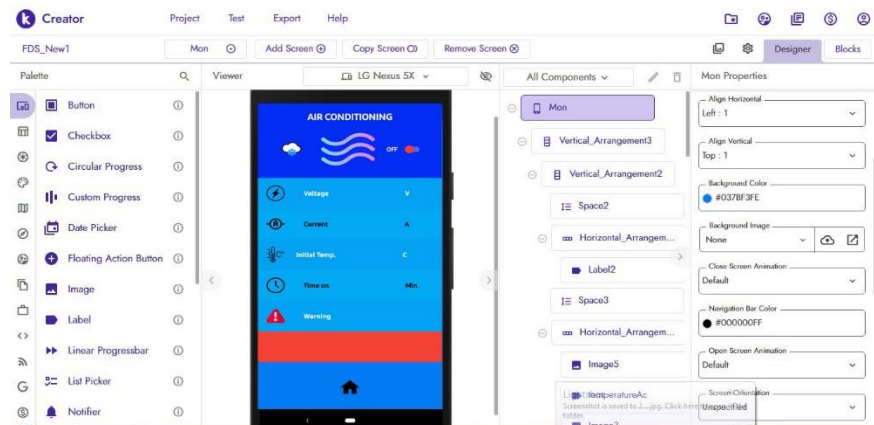
1) Tampilan awal.



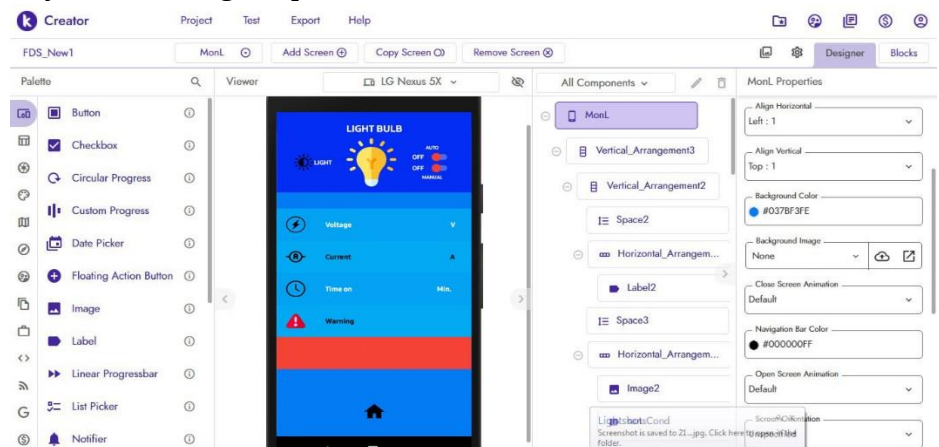
2) Layar utama / Home



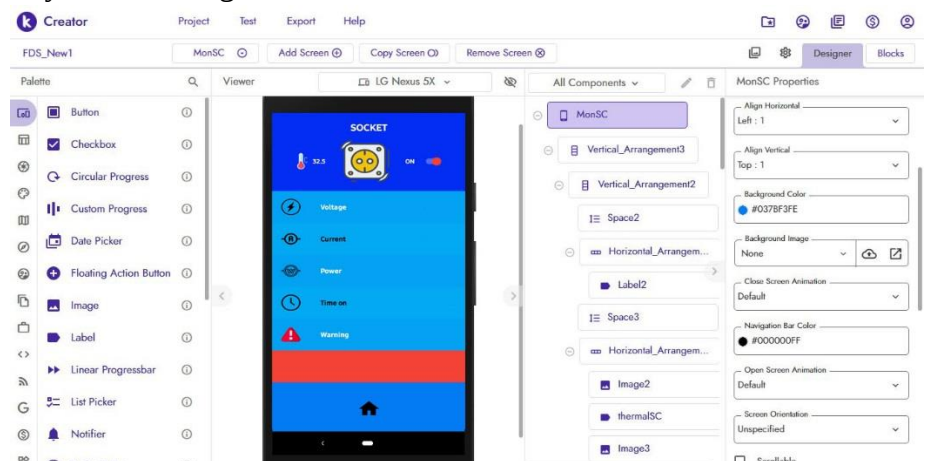
3) Layar monitoring AC / penyejuk ruangan.



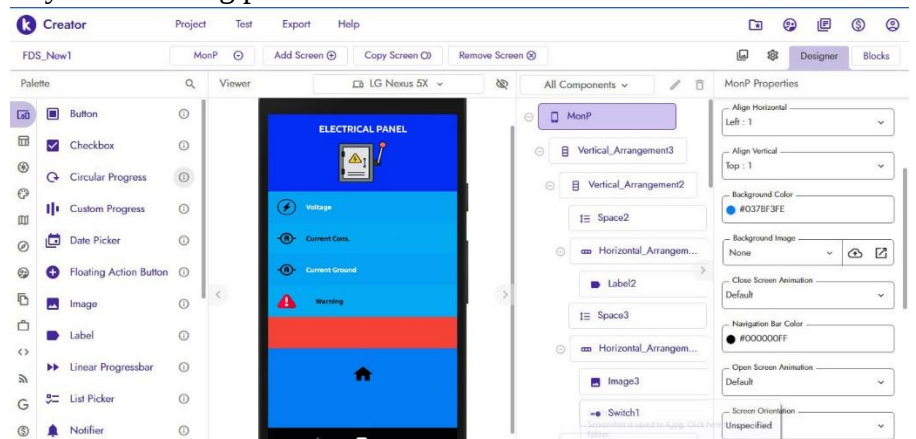
4) Layar monitoring lampu



5) Layar monitoring stok kontak.

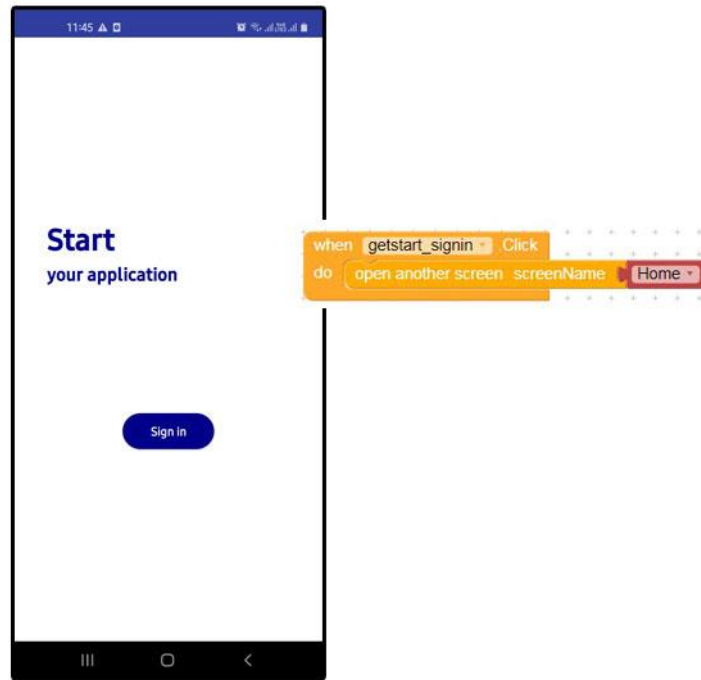


6) Layar monitoring panel.



Berikut ini akan diperlihatkan codingan blok dari aplikasi android yang telah dibuat.

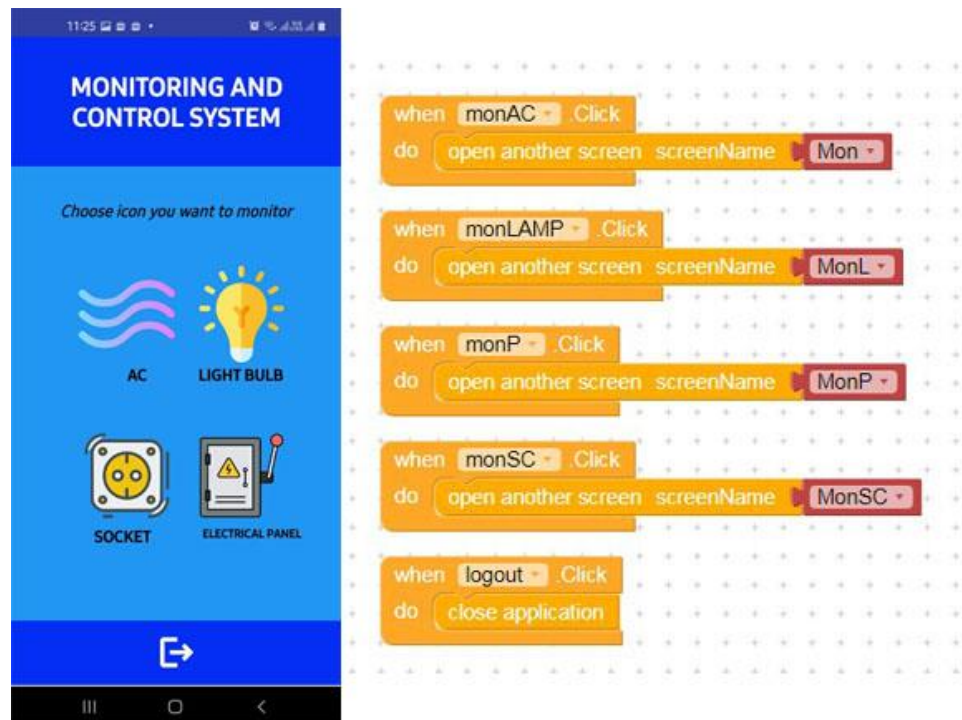
1) Tampilan awal



Penjelasan:

Saat tombol “Sign in” ditekan maka akan diarahkan ke screen “Home”.

2) Layar utama / home.



Penjelasan:

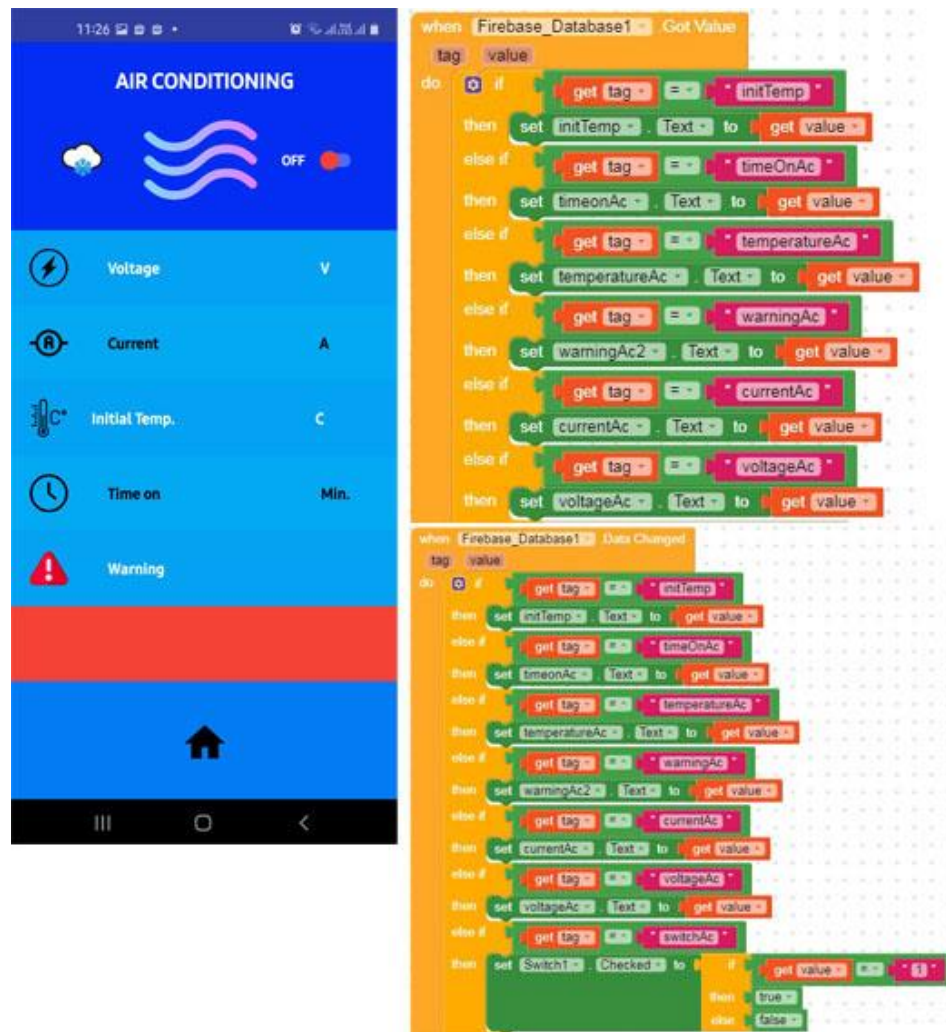
- Jika menekan icon AC maka akan diarahkan ke screen AC
- Jika menekan icon Ligh bulb maka akan diarahkan ke screen Ligh Bulb
- Jika menekan icon Socket maka akan diarahkan ke screen Socket
- Jika menekan icon Electrical Panel maka akan diarahkan ke screen Electrical Panel
- Jika menekan icon Close maka aplikasi akan ditutup.

3) Layar monitoring AC



Penjelasan:

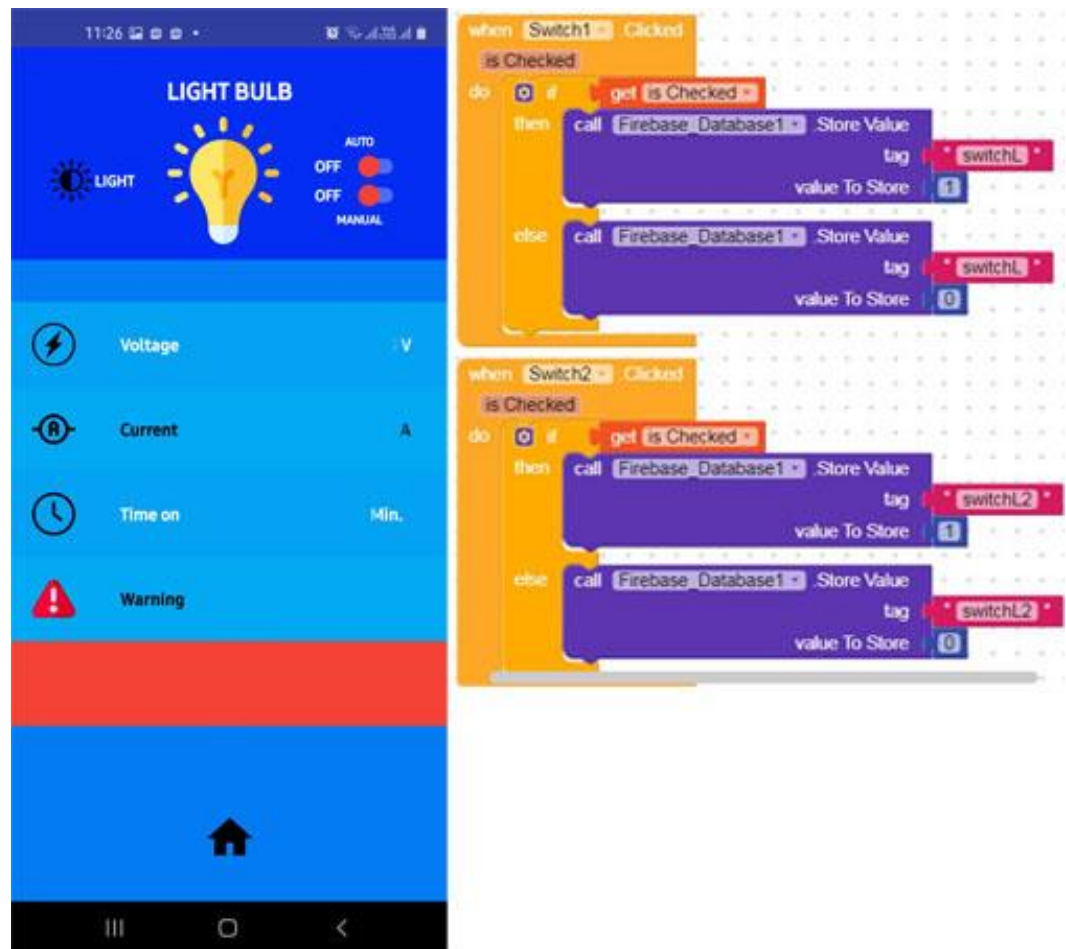
- Jika switch AC ditekan ke ON, maka sistem AC pada hardware akan diaktifkan.
- Jika switch AC ditekan ke OFF, maka sistem AC pada hardware akan dinonaktifkan.
- Jika icon Home ditekan, maka tampilan akan diarahkan ke layar utama / home.
- Pada saat inisialisasi screen, aplikasi akan mengambil data nilai (get value) dari firebase, yang berupa voltage, current, initial temp, time on, warning, dan data switch.



Penjelasan:

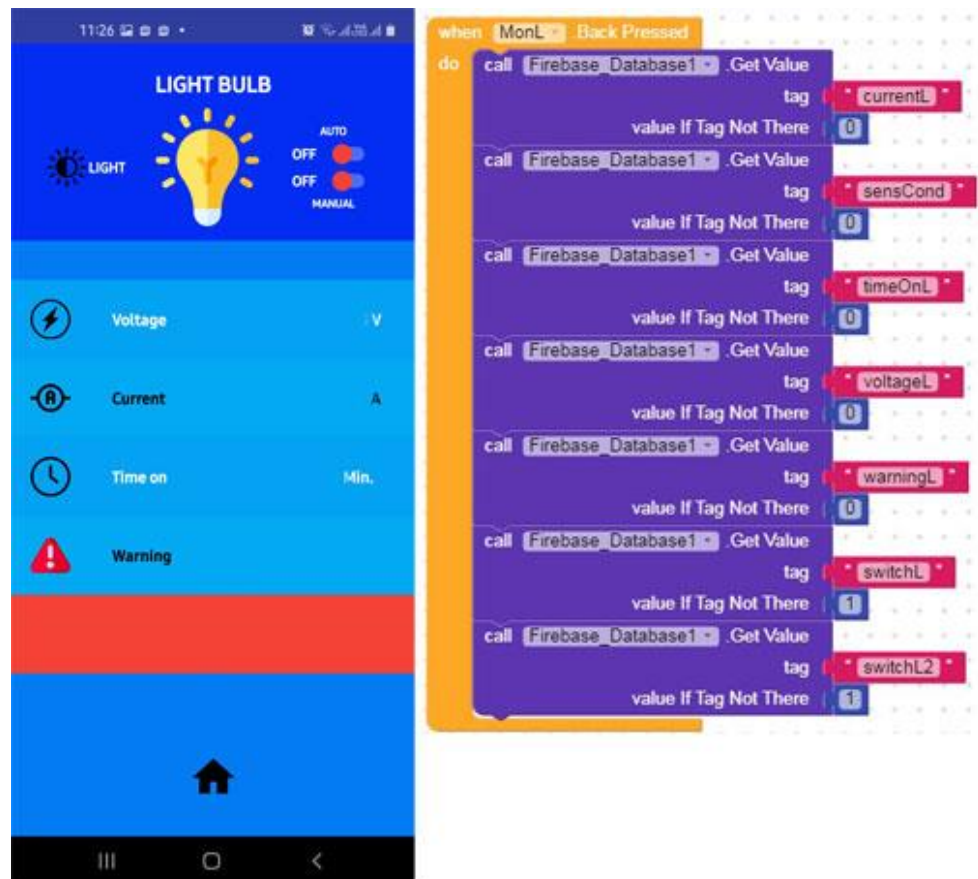
Setelah semua data telah didapat (got value) dari firebase maka aplikasi akan mengganti data (data change) pada tampilan monitoring AC.

4) Layar monitoring lampu.



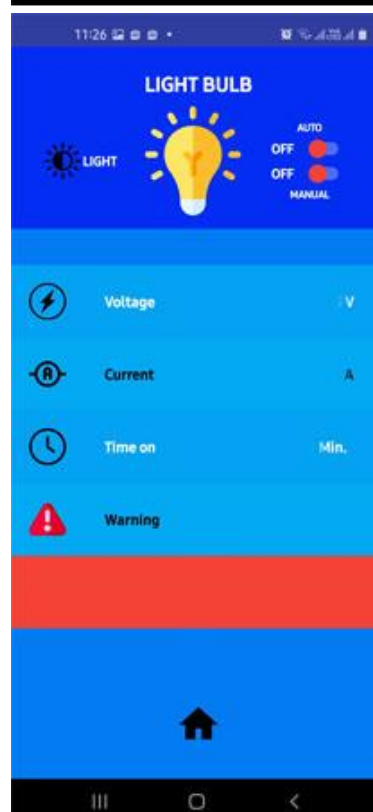
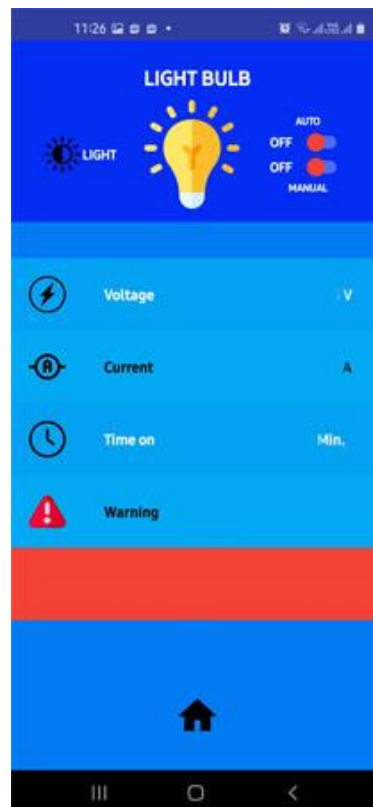
Penjelasan:

- Jika switch AUTO ditekan ke ON, maka sistem lampu pada hardware akan diaktifkan.
- Jika switch AUTO ditekan ke OFF, maka sistem lampu pada hardware akan dinonaktifkan.
- Jika switch MANUAL ditekan ke ON, maka lampu manual pada hardware akan diaktifkan.
- Jika switch MANUAL ditekan ke OFF, maka sistem lampu manual pada hardware akan dinonaktifkan.
- Jika icon Home ditekan, maka tampilan akan diarahkan ke layar utama / home.



Penjelasan:

Pada saat inisialisasi screen, aplikasi akan mengambil data nilai (get value) dari firebase, yang berupa voltage, current, time on, warning, dan data switch.



```

when Firebase_Database1 Got Value
  tag value
do
  if
    get tag == currentL
  then
    set currentL . Text to get value
  else if
    get tag == sensCond
  then
    set sensCond . Text to get value
  else if
    get tag == timeOnL
  then
    set timeOnL . Text to get value
  else if
    get tag == voltageL
  then
    set voltageL . Text to get value
  else if
    get tag == warningL
  then
    set warningL2 . Text to get value

```

```

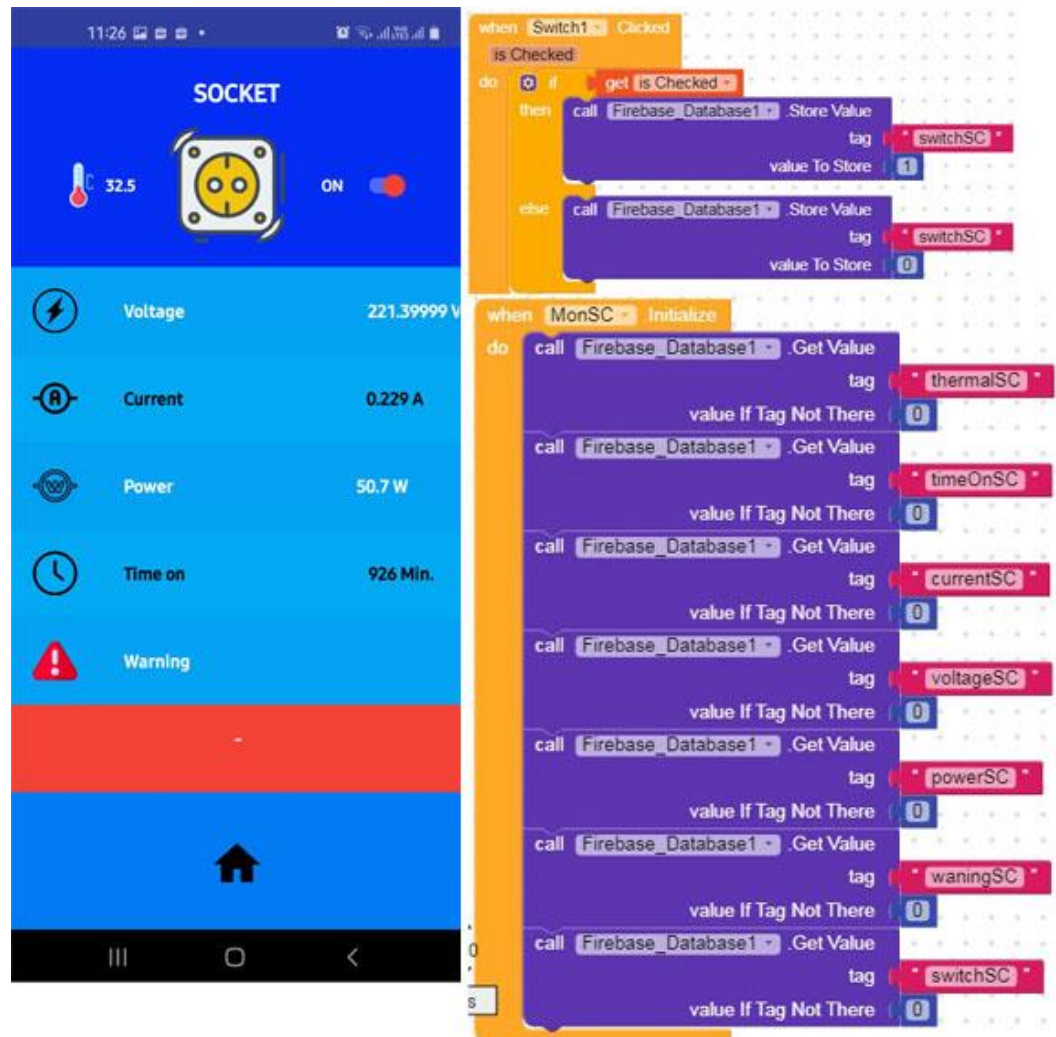
when Firebase_Database1 Data Changed
  tag value
do
  if
    get tag == currentL
  then
    set currentL . Text to get value
  else if
    get tag == sensCond
  then
    set sensCond . Text to get value
  else if
    get tag == timeOnL
  then
    set timeOnL . Text to get value
  else if
    get tag == voltageL
  then
    set voltageL . Text to get value
  else if
    get tag == warningL
  then
    set warningL2 . Text to get value
  else if
    get tag == switchL
  then
    set Switch1 . Checked to if
      get value == 1
    then true
    else false
  else if
    get tag == switchL2
  then
    set Switch2 . Checked to if
      get value == 1
    then true
    else false

```

Penjelasan:

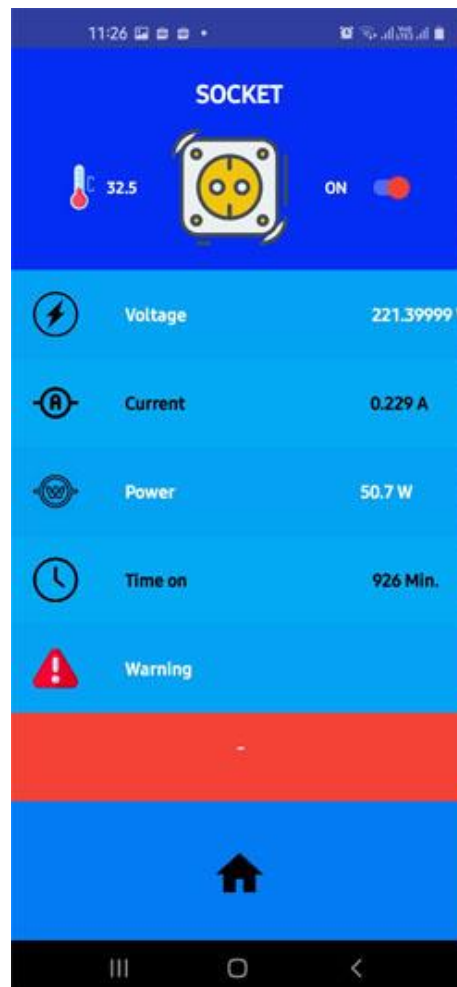
Setelah semua data telah didapat (got value) dari firebase maka aplikasi akan mengganti data (data change) pada tampilan monitoring lampu.

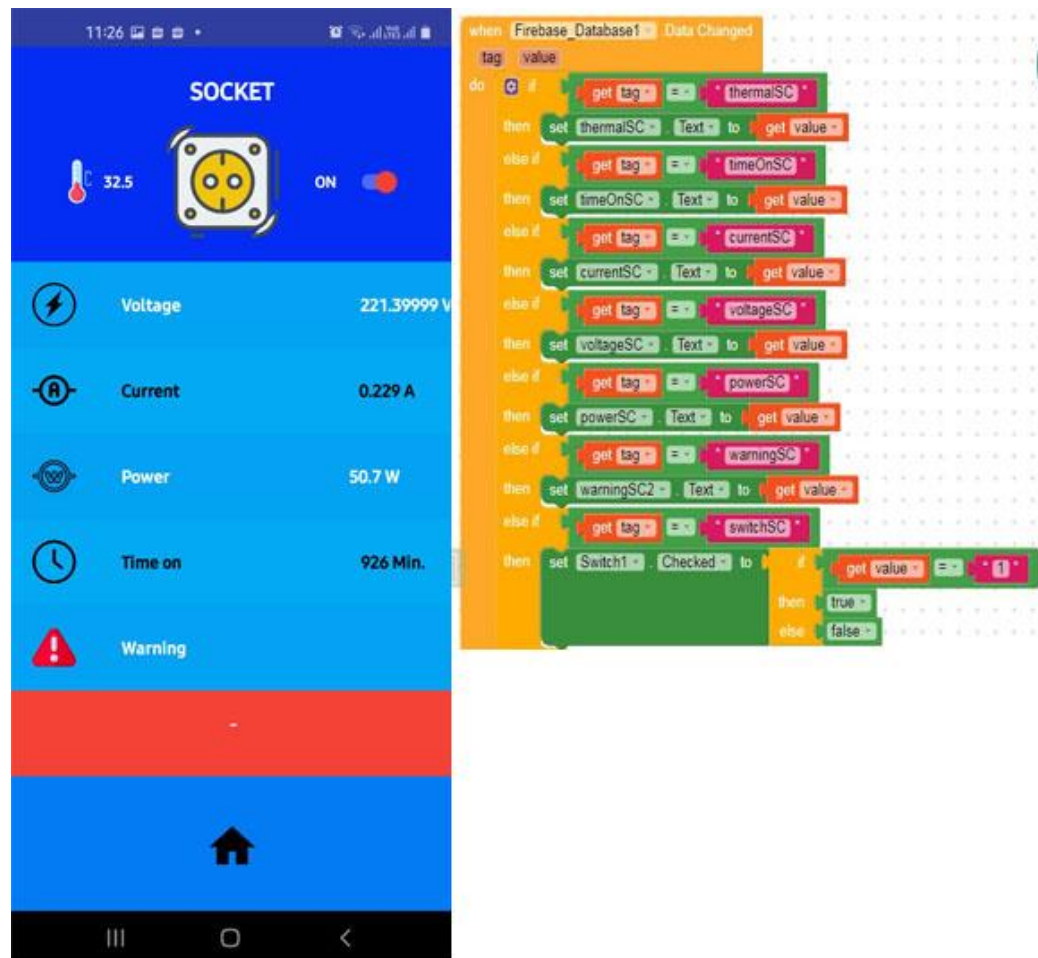
5) Layar monitor stok kontak



Penjelasan:

- Jika switch Socket ditekan ke ON, maka sistem Socket pada hardware akan diaktifkan.
- Jika switch Socket ditekan ke OFF, maka sistem Socket pada hardware akan dinonaktifkan.
- Jika icon Home ditekan, maka tampilan akan diarahkan ke layar utama / home.
- Pada saat inisialisasi screen, aplikasi akan mengambil data nilai (get value) dari firebase, yang berupa voltage, current, power, time on, warning, dan data switch.

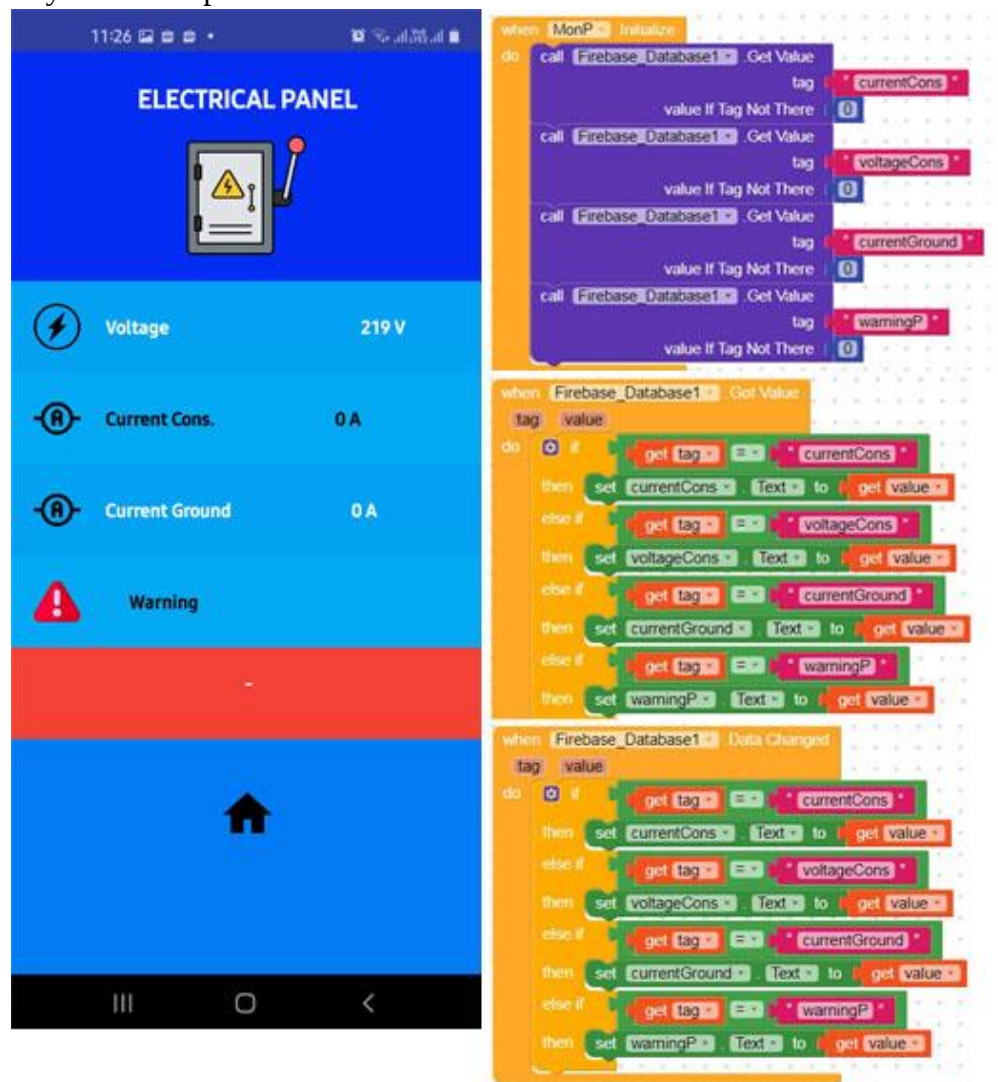




Pernjelasan:

Setelah semua data telah didapat (got value) dari firebase maka aplikasi akan mengganti data (data change) pada tampilan monitoring stok kontak.

6) Layar monitor panel.



Penjelasan:

- Jika icon Home ditekan, maka tampilan akan diarahkan ke layar utama / home.
- Pada saat inisialisasi screen, aplikasi akan mengambil data nilai (get value) dari firebase, yang berupa voltage, current consumption, current grounding, dan warning.
- Setelah semua data telah didapat (got value) dari firebase maka aplikasi akan mengganti data (data change) pada tampilan monitoring panel.

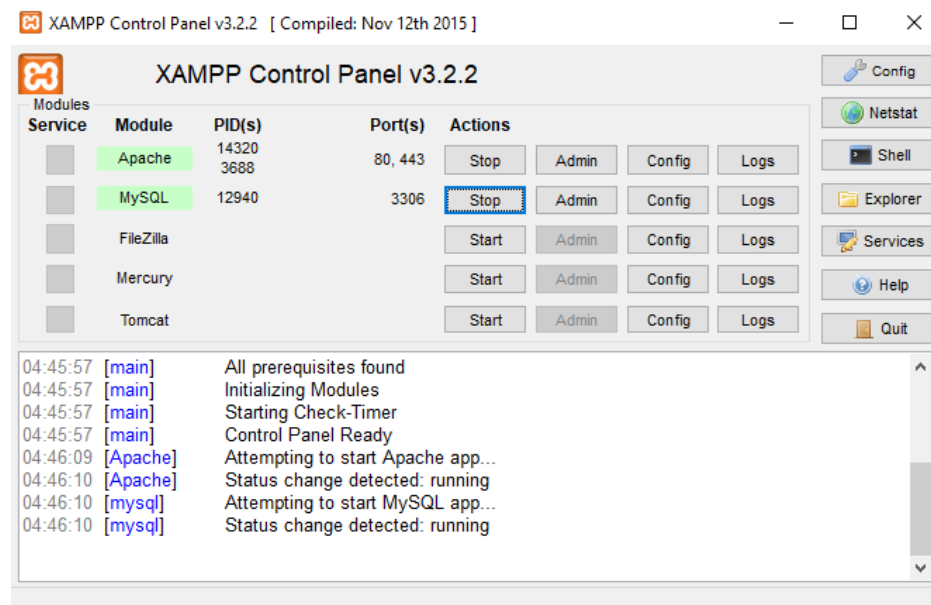
4. Pembuatan Aplikasi Website

Aplikasi website yang digunakan pada penelitian ini dibuat dengan menggunakan bahasa pemrograman PHP dengan framework Codeigniter 3.1.4. CodeIgniter merupakan aplikasi sumber terbuka yang berupa kerangka kerja PHP dengan model MVC untuk membangun situs web dinamis dengan menggunakan

PHP. CodeIgniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. Berikut langkah langkah pembuatan aplikasi website

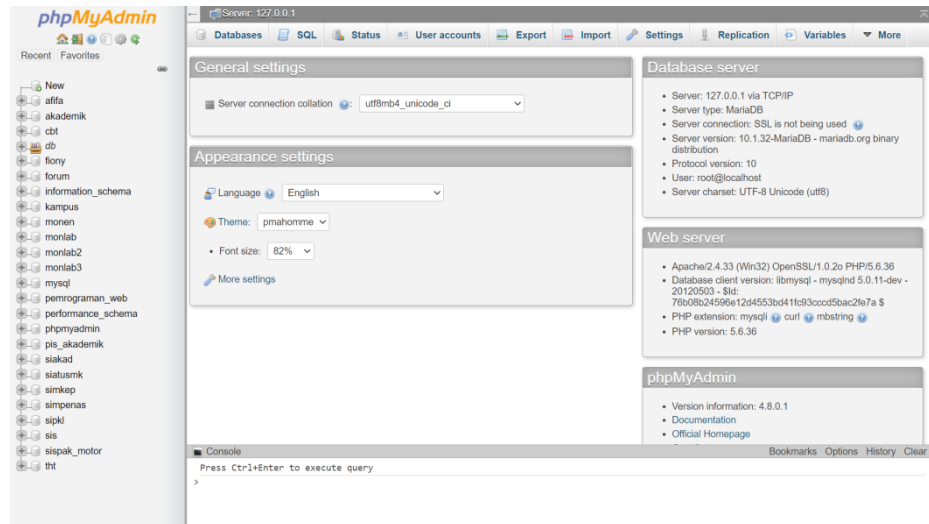
a. **Menginstal *Tools* yang Diperlukan**

hal ini mencakup sever web Apache, SQL, Phpmyadmin, web browser, text editor dan Codeingiter. Server web Apache, SQL dan phpmyadmin yang dibutuhkan bisa didapatkan dengan menginstal XAMPP, dalam penelitian ini digunakan XAMPP 3.2.2.



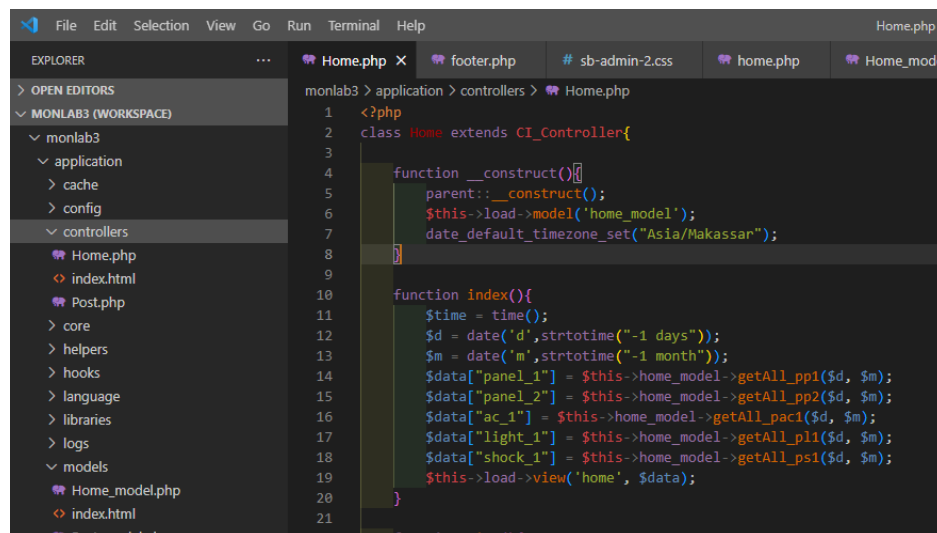
b. **Perancang dan Pembuatan Database**

pembuatan dan pengelolaan database dapat dilakukan dengan terlebih dahulu menjalankan Apache dan MySQL pada control panel XAMPP kemudian mengakses phpmyadmin dengan mengakses url pada browser <http://localhost/phpmyadmin>



c. Pembuatan *Controller* dan *Routing* Website

Controller merupakan bagian yang menjembatani *model* dan *view*. *Controller* berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web. Pembuatan *controller* dapat dilakukan pada folder `folder_projek/application/controller`, pada folder ini dilakukan pembuatan file *controller*. Aplikasi web ini menggunakan dua *controller* yaitu `Home.php` sebagai *controller view* aplikasi dan `Post.php` sebagai *controller* pengiriman data dari ESP8266



Router pada Codeigniter bertugas untuk menentukan *controller* dan *method/fungsi* yang akan dieksekusi, pembuatan *route* dapat dilakukan pada file di dalam folder `folder_projek/application/config/routes.php`

```

monlab3 > application > config > routes.php
48 |
49 | Examples: my-controller/index -> my_control
50 |           my-controller/my-method -> my_control
51 | */
52 $route['default_controller'] = 'home';
53 $route['404_override'] = '';
54 $route['translate_uri_dashes'] = FALSE;
55
56 $route['post'] = 'post';
57 $route['home'] = 'home';
58

```

d. Pembuatan *Model* (CRUD)

Model mewakili struktur data, *model* merupakan bagian yang bertugas untuk mengatur, menyiapkan, memanipulasi, dan mengorganisir data (biasanya dari basis data). Tugas yang ia lakukan meliputi memasukkan data ke basis data, pembaruan data, menghapus data, dan lain-lain. *Model* menjalankan tugasnya berdasarkan instruksi dari *controller*. Pembuatan model dapat dilakukan di dalam folder folder_projek/application/model. Aplikasi web ini menggunakan dua *model*, Home_model.php menangani pengelolaan data untuk tampilan aplikasi dan Post_model.php menangani pengelolaan data dari

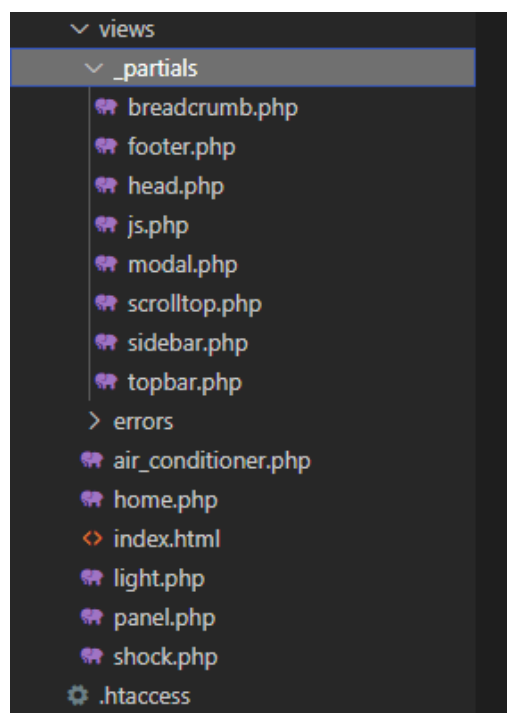
```

1  <?php
2  class Home_model extends CI_Model{
3
4      private static $_table1 = "sensor_panel_1";
5      private static $_table2 = "sensor_panel_2";
6      private static $_table3 = "power_panel_1";
7      private static $_table4 = "power_panel_2";
8      private static $_table5 = "sensor_ac_1";
9      private static $_table6 = "power_ac_1";
10     private static $_table7 = "sensor_light_1";
11     private static $_table8 = "power_light_1";
12     private static $_table9 = "sensor_shock_1";
13     private static $_table10 = "power_shock_1";
14
15     // public function getAll()
16     // {
17         return $this->db->get($this->_table)->result();
18     // }
19
20     public function getLast_sp1()
21     {

```

e. Pembuatan View

View merupakan bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web. Pembuatan view dapat dilakukan di dalam folder folder_projek/application/view



f. Hosting Aplikasi Web

Setelah pembuatan aplikasi web selesai dan dapat berjalan di dalam server lokal yang dibuat, selanjutnya adalah melakukan *hosting* secara online pada aplikasi web yang dilakukan dengan mengupload seluruh folder/file project pada hoster/ server online, sehingga aplikasi web dapat diakses secara online.

A. Hasil Pengujian Sistem

Berikut diperlihatkan tabel hasil penerimaan web server dari mikrokontroller sistem.

1. Modul AC

Modul AC ini menggunakan sensor arus PZEM-004T dan sensor temperatur DHT22 yang menjadi alat untuk monitoring sekaligus membandingkan data untuk dijadikan acuan tindakan otomatisasi pada sistem. Berikut diperlihatkan tabel hasil uji coba modul AC dengan kondisi pendingin ruangan normal dan tidak normal. Uji coba ini merupakan uji coba terbatas menggunakan beban 20 Watt dan es batu sebagai pendingin.

Tabel 1. Percobaan dengan kondisi pendingin ruangan normal.

No	Date	Time	I (A)	V (Volt)	P (W)	PF	Sw	Temp. (C)	Status
1	22/05/2022	19:05:57	0.00	0.00	0.00	0.00	OFF	0.00	Setup system
2	22/05/2022	19:06:03	0.00	0.00	0.00	0.00	OFF	0.00	Setup system
3	22/05/2022	19:06:38	0.00	216.20	0.00	0.00	OFF	32.70	System ON
4	22/05/2022	19:06:42	0.00	216.10	0.00	0.00	OFF	32.70	System ON
5	22/05/2022	19:06:48	0.09	216.10	14.10	0.71	ON	32.70	AC turn ON
6	22/05/2022	19:06:53	0.09	216.60	14.20	0.70	ON	32.70	AC turn ON
7	22/05/2022	19:06:57	0.09	216.60	14.30	0.71	ON	32.70	AC turn ON
8	22/05/2022	19:07:01	0.09	216.80	14.40	0.71	ON	32.70	AC turn ON
9	22/05/2022	19:07:05	0.09	216.60	14.30	0.71	ON	32.70	AC turn ON
10	22/05/2022	19:07:10	0.09	216.60	14.30	0.71	ON	32.70	AC turn ON
11	22/05/2022	19:07:13	0.09	216.60	14.20	0.71	ON	32.70	AC turn ON
12	22/05/2022	19:07:18	0.09	216.60	14.20	0.71	ON	32.70	AC turn ON
13	22/05/2022	19:07:22	0.09	216.60	14.10	0.71	ON	32.70	AC turn ON
14	22/05/2022	19:07:27	0.09	216.50	14.00	0.71	ON	32.70	AC turn ON
15	22/05/2022	19:07:32	0.09	216.60	13.90	0.70	ON	32.70	AC turn ON
16	22/05/2022	19:07:36	0.09	216.50	13.90	0.71	ON	32.70	AC turn ON
17	22/05/2022	19:07:41	0.09	216.70	13.80	0.71	ON	32.70	AC turn ON
18	22/05/2022	19:07:48	0.09	216.40	13.70	0.70	ON	32.80	AC turn ON
19	22/05/2022	19:07:54	0.09	216.40	13.60	0.71	ON	32.80	AC turn ON
20	22/05/2022	19:08:00	0.09	217.00	13.60	0.70	ON	32.80	AC turn ON
21	22/05/2022	19:08:07	0.09	217.00	13.50	0.71	ON	32.70	AC turn ON
22	22/05/2022	19:08:13	0.09	217.00	13.50	0.71	ON	32.70	AC turn ON

23	22/05/2022	19:08:19	0.09	217.10	13.50	0.71	ON	32.70	AC turn ON
24	22/05/2022	19:08:25	0.09	217.20	13.40	0.70	ON	32.80	AC turn ON
25	22/05/2022	19:08:32	0.09	217.20	13.30	0.70	ON	32.80	AC turn ON
26	22/05/2022	19:08:40	0.09	217.20	13.40	0.70	ON	32.80	AC turn ON
27	22/05/2022	19:08:46	0.09	217.10	13.30	0.70	ON	32.70	AC turn ON
28	22/05/2022	19:08:51	0.09	217.30	13.30	0.70	ON	32.70	AC turn ON
29	22/05/2022	19:08:56	0.09	217.30	13.30	0.70	ON	32.80	AC turn ON
30	22/05/2022	19:09:03	0.09	217.20	13.30	0.70	ON	32.80	AC turn ON
31	22/05/2022	19:09:08	0.09	217.40	13.20	0.70	ON	32.80	AC turn ON
32	22/05/2022	19:09:14	0.09	217.40	13.20	0.70	ON	32.80	AC turn ON
33	22/05/2022	19:09:20	0.09	217.50	13.20	0.70	ON	32.70	AC turn ON
34	22/05/2022	19:09:26	0.09	217.80	13.20	0.70	ON	32.80	AC turn ON
35	22/05/2022	19:09:33	0.09	217.60	13.20	0.70	ON	32.70	AC turn ON
36	22/05/2022	19:09:40	0.09	217.90	13.30	0.70	ON	32.80	AC turn ON
37	22/05/2022	19:09:46	0.09	218.00	13.20	0.70	ON	31.70	AC turn ON
38	22/05/2022	19:09:52	0.09	218.00	13.20	0.70	ON	31.70	AC turn ON
39	22/05/2022	19:09:59	0.09	218.10	13.20	0.70	ON	31.70	AC turn ON
40	22/05/2022	19:10:06	0.09	217.90	13.20	0.70	ON	30.90	AC turn ON
41	22/05/2022	19:10:12	0.09	217.50	13.20	0.70	ON	30.80	AC turn ON
42	22/05/2022	19:17:20	0.06	218.10	13.00	0.69	ON	29.40	AC turn ON
43	22/05/2022	19:17:24	0.09	218.10	13.00	0.69	ON	29.50	AC turn ON
44	22/05/2022	19:17:29	0.09	218.20	13.10	0.70	ON	29.50	AC turn ON
45	22/05/2022	19:17:34	0.09	218.10	13.10	0.70	ON	29.50	AC turn ON
46	22/05/2022	19:17:38	0.09	218.00	13.00	0.69	ON	29.40	AC turn ON
47	22/05/2022	19:17:42	0.09	218.00	12.90	0.69	ON	29.40	AC turn ON
48	22/05/2022	19:17:47	0.09	217.80	13.00	0.69	ON	29.40	AC turn ON
49	22/05/2022	19:17:52	0.09	218.10	12.90	0.70	ON	29.40	AC turn ON
50	22/05/2022	19:17:56	0.09	218.00	12.90	0.70	ON	29.40	AC turn ON
51	22/05/2022	19:18:01	0.09	218.10	12.90	0.70	ON	29.40	AC turn ON
52	22/05/2022	19:18:05	0.09	218.10	12.90	0.69	ON	29.30	AC turn ON
53	22/05/2022	19:18:12	0.09	217.50	13.00	0.70	ON	29.30	AC turn ON
54	22/05/2022	19:18:18	0.09	217.40	12.90	0.70	ON	29.30	AC turn ON
55	22/05/2022	19:18:26	0.09	217.60	12.90	0.69	ON	29.30	AC turn ON
56	22/05/2022	19:18:32	0.09	217.60	12.90	0.69	ON	29.30	AC turn ON
57	22/05/2022	19:18:38	0.09	217.70	12.90	0.69	ON	29.30	AC turn ON
58	22/05/2022	19:18:44	0.09	217.70	12.90	0.69	ON	29.30	AC turn ON
59	22/05/2022	19:18:50	0.09	217.60	12.90	0.69	ON	29.30	AC turn ON
60	22/05/2022	19:18:58	0.09	217.80	12.90	0.69	ON	29.30	AC turn ON
61	22/05/2022	19:19:06	0.09	217.60	12.80	0.69	ON	29.30	AC turn ON
62	22/05/2022	19:19:13	0.09	217.10	12.90	0.69	ON	29.30	AC turn ON
63	22/05/2022	19:19:22	0.09	217.30	12.90	0.69	ON	29.30	AC turn ON
64	22/05/2022	19:19:31	0.09	217.80	12.90	0.70	ON	29.30	AC turn ON
65	22/05/2022	19:19:37	0.09	217.90	12.80	0.69	ON	29.30	AC turn ON

66	22/05/2022	19:19:45	0.09	217.80	12.90	0.70	ON	29.20	AC turn ON
67	22/05/2022	19:19:52	0.09	217.80	12.80	0.69	ON	29.20	AC turn ON
68	22/05/2022	19:20:00	0.09	218.30	12.90	0.69	ON	29.10	AC turn ON
69	22/05/2022	19:20:06	0.09	218.60	12.90	0.69	ON	29.10	AC turn ON
70	22/05/2022	19:20:15	0.09	218.40	12.80	0.69	ON	29.10	AC turn ON
71	22/05/2022	19:20:21	0.09	218.30	12.90	0.69	ON	29.20	AC turn ON
72	22/05/2022	19:20:28	0.09	218.70	12.90	0.69	ON	29.20	AC turn ON
73	22/05/2022	19:20:38	0.09	218.70	12.90	0.69	ON	29.10	AC turn ON
74	22/05/2022	19:20:44	0.09	218.40	12.90	0.69	ON	29.20	AC turn ON
75	22/05/2022	19:20:51	0.09	218.20	12.90	0.69	ON	29.20	AC turn ON
76	22/05/2022	19:20:59	0.09	218.50	12.90	0.69	ON	29.20	AC turn ON
77	22/05/2022	19:21:06	0.09	218.30	12.90	0.70	ON	29.30	AC turn ON
78	22/05/2022	19:21:12	0.09	218.20	12.90	0.70	ON	29.30	AC turn ON
79	22/05/2022	19:21:18	0.09	218.20	12.80	0.69	ON	29.20	AC turn ON
80	22/05/2022	19:21:27	0.09	218.40	12.90	0.69	ON	29.20	AC turn ON

Berdasarkan tabel hasil uji, sesaat setelah sistem ON (baris ke 5) menunjukkan pendingin ruangan telah dinyalakan oleh sistem, pada pukul 19:06:42 dengan kondisi temperatur ruangan 32.70

4	22/05/2022	19:06:42	0.00	216.10	0.00	0.00	OFF	32.70	System ON
5	22/05/2022	19:06:48	0.09	216.10	14.10	0.71	ON	32.70	AC turn ON

Gambar 1. Kondisi pendingin ruangan dinyalakan.

Setelah sekitar 3 menit yaitu pukul 19:09:46, temperatur ruangan berubah menjadi 31.70 derajat celcius, hal ini dianggap pendingin ruangan dalam keadaan normal karena temperatur ruangan berubah menjadi lebih rendah sebelum 5 menit. Oleh karena itu penyalaan pendingin ruangan dilanjutkan.

36	22/05/2022	19:09:40	0.09	217.90	13.30	0.70	ON	32.80	AC turn ON
37	22/05/2022	19:09:46	0.09	218.00	13.20	0.70	ON	31.70	AC turn ON
38	22/05/2022	19:09:52	0.09	218.00	13.20	0.70	ON	31.70	AC turn ON

Gambar 2. Kondisi pendingin ruangan menurunkan temperatur setelah 3 menit.

Tabel 2. Percobaan dengan kondisi pendingin ruangan tidak normal.

No	Date	Time	I (A)	V (Volt)	P (W)	PF	Sw	Temp. (C)	Status
1	22/05/2022	17:36:05	0.00	218.80	0.00	0.00	OFF	33.70	System ON
2	22/05/2022	17:36:09	0.00	218.70	0.00	0.00	OFF	33.70	System ON
3	22/05/2022	17:36:16	0.00	218.60	0.00	0.00	ON	33.70	AC turn ON

4	22/05/2022	17:36:24	0.00	218.50	0.00	0.00	ON	33.80	AC turn ON
5	22/05/2022	17:36:28	0.00	218.80	4.10	0.49	ON	33.80	AC turn ON
6	22/05/2022	17:36:32	0.09	218.80	14.00	0.69	ON	33.70	AC turn ON
7	22/05/2022	17:36:36	0.09	218.60	14.30	0.70	ON	33.70	AC turn ON
8	22/05/2022	17:36:40	0.09	218.70	14.40	0.69	ON	33.70	AC turn ON
9	22/05/2022	17:36:44	0.09	218.90	14.50	0.70	ON	33.80	AC turn ON
10	22/05/2022	17:36:48	0.10	218.80	14.60	0.70	ON	33.70	AC turn ON
11	22/05/2022	17:36:52	0.10	218.90	14.60	0.69	ON	33.80	AC turn ON
12	22/05/2022	17:36:56	0.10	218.80	14.60	0.70	ON	33.70	AC turn ON
13	22/05/2022	17:37:01	0.10	218.70	14.60	0.70	ON	33.80	AC turn ON
14	22/05/2022	17:37:06	0.09	218.50	14.50	0.70	ON	33.70	AC turn ON
15	22/05/2022	17:37:13	0.09	218.30	14.40	0.69	ON	33.80	AC turn ON
16	22/05/2022	17:37:18	0.09	218.20	14.30	0.69	ON	33.70	AC turn ON
17	22/05/2022	17:37:23	0.09	218.30	14.20	0.69	ON	33.80	AC turn ON
18	22/05/2022	17:37:29	0.09	218.60	14.20	0.69	ON	33.70	AC turn ON
19	22/05/2022	17:37:35	0.09	218.80	14.10	0.69	ON	33.70	AC turn ON
20	22/05/2022	17:37:40	0.09	218.60	14.10	0.69	ON	33.80	AC turn ON
21	22/05/2022	17:37:46	0.09	218.70	14.00	0.69	ON	33.70	AC turn ON
22	22/05/2022	17:37:51	0.09	218.30	14.00	0.69	ON	33.70	AC turn ON
23	22/05/2022	17:37:58	0.09	218.40	13.90	0.69	ON	33.70	AC turn ON
24	22/05/2022	17:38:05	0.09	218.40	13.90	0.69	ON	33.70	AC turn ON
25	22/05/2022	17:38:10	0.09	218.30	13.80	0.69	ON	33.80	AC turn ON
26	22/05/2022	17:38:16	0.09	218.40	13.80	0.69	ON	33.80	AC turn ON
27	22/05/2022	17:38:21	0.09	218.40	13.70	0.69	ON	33.70	AC turn ON
28	22/05/2022	17:38:28	0.09	218.60	13.80	0.69	ON	33.80	AC turn ON
29	22/05/2022	17:38:33	0.09	218.70	13.70	0.69	ON	33.70	AC turn ON
30	22/05/2022	17:38:39	0.09	218.70	13.80	0.69	ON	33.80	AC turn ON
31	22/05/2022	17:38:44	0.09	219.00	13.80	0.69	ON	33.70	AC turn ON
32	22/05/2022	17:38:50	0.09	219.00	13.80	0.69	ON	33.80	AC turn ON
33	22/05/2022	17:38:57	0.09	219.10	13.90	0.69	ON	33.70	AC turn ON
34	22/05/2022	17:39:04	0.09	219.10	13.80	0.68	ON	33.80	AC turn ON
35	22/05/2022	17:39:10	0.09	219.30	13.80	0.68	ON	33.70	AC turn ON
36	22/05/2022	17:39:15	0.09	219.20	13.80	0.68	ON	33.70	AC turn ON
37	22/05/2022	17:39:21	0.09	219.10	13.80	0.68	ON	33.70	AC turn ON
38	22/05/2022	17:39:27	0.09	219.30	13.80	0.68	ON	33.70	AC turn ON
39	22/05/2022	17:39:34	0.09	219.50	13.80	0.69	ON	33.80	AC turn ON
40	22/05/2022	17:39:41	0.09	219.40	13.80	0.69	ON	33.70	AC turn ON
41	22/05/2022	17:40:57	0.09	219.30	13.40	0.69	ON	33.70	AC turn ON
42	22/05/2022	17:41:01	0.09	219.30	13.40	0.69	ON	33.70	AC turn ON
43	22/05/2022	17:41:06	0.09	219.30	13.30	0.68	ON	33.70	AC turn ON
44	22/05/2022	17:41:10	0.09	219.10	13.30	0.68	ON	33.70	AC turn ON
45	22/05/2022	17:41:14	0.09	218.80	13.30	0.68	ON	33.70	5'th minutes
46	22/05/2022	17:41:19	0.00	219.90	0.00	0.00	OFF	33.80	AC turn OFF

47	22/05/2022	17:41:23	0.00	219.70	0.00	0.00	OFF	33.70	AC turn OFF
48	22/05/2022	17:41:27	0.00	219.50	0.00	0.00	OFF	33.70	AC turn OFF
49	22/05/2022	17:41:31	0.00	219.50	0.00	0.00	OFF	33.80	AC turn OFF
50	22/05/2022	17:41:35	0.00	219.60	0.00	0.00	OFF	33.80	AC turn OFF
51	22/05/2022	17:41:38	0.00	219.70	0.00	0.00	OFF	33.70	AC turn OFF
52	22/05/2022	17:41:42	0.00	219.70	0.00	0.00	OFF	33.70	AC turn OFF
53	22/05/2022	17:41:46	0.00	219.40	0.00	0.00	OFF	33.80	AC turn OFF
54	22/05/2022	17:41:50	0.00	219.10	0.00	0.00	OFF	33.70	AC turn OFF
55	22/05/2022	17:41:57	0.00	219.00	0.00	0.00	OFF	33.80	AC turn OFF
56	22/05/2022	17:42:02	0.00	218.90	0.00	0.00	OFF	33.80	AC turn OFF
57	22/05/2022	17:42:08	0.00	218.90	0.00	0.00	OFF	33.80	AC turn OFF
58	22/05/2022	17:42:13	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
59	22/05/2022	17:42:19	0.00	219.00	0.00	0.00	OFF	33.70	AC turn OFF
60	22/05/2022	17:42:24	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
61	22/05/2022	17:42:31	0.00	219.10	0.00	0.00	OFF	33.70	AC turn OFF
62	22/05/2022	17:42:37	0.00	219.10	0.00	0.00	OFF	33.70	AC turn OFF
63	22/05/2022	17:42:43	0.00	219.30	0.00	0.00	OFF	33.70	AC turn OFF
64	22/05/2022	17:42:50	0.00	219.50	0.00	0.00	OFF	33.70	AC turn OFF
65	22/05/2022	17:42:55	0.00	219.40	0.00	0.00	OFF	33.70	AC turn OFF
66	22/05/2022	17:43:01	0.00	219.50	0.00	0.00	OFF	33.70	AC turn OFF
67	22/05/2022	17:43:08	0.00	219.20	0.00	0.00	OFF	33.70	AC turn OFF
68	22/05/2022	17:43:16	0.00	219.20	0.00	0.00	OFF	33.70	AC turn OFF
69	22/05/2022	17:43:21	0.00	219.10	0.00	0.00	OFF	33.70	AC turn OFF
70	22/05/2022	17:43:32	0.00	219.10	0.00	0.00	OFF	33.70	AC turn OFF
71	22/05/2022	17:43:39	0.00	219.20	0.00	0.00	OFF	33.70	AC turn OFF
72	22/05/2022	17:43:46	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
73	22/05/2022	17:43:53	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
74	22/05/2022	17:43:59	0.00	218.80	0.00	0.00	OFF	33.70	AC turn OFF
75	22/05/2022	17:44:06	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
76	22/05/2022	17:44:11	0.00	218.90	0.00	0.00	OFF	33.70	AC turn OFF
77	22/05/2022	17:44:17	0.00	219.00	0.00	0.00	OFF	33.70	AC turn OFF
78	22/05/2022	17:44:22	0.00	219.00	0.00	0.00	OFF	33.70	AC turn OFF
79	22/05/2022	17:45:32	0.00	219.00	0.00	0.00	OFF	33.60	AC turn OFF
80	22/05/2022	17:45:36	0.00	218.80	0.00	0.00	OFF	33.70	AC turn OFF
81	22/05/2022	17:45:40	0.00	218.70	0.00	0.00	OFF	33.70	AC turn OFF
82	22/05/2022	17:45:44	0.00	218.80	0.00	0.00	OFF	33.70	AC turn OFF
83	22/05/2022	17:45:49	0.00	218.70	0.00	0.00	OFF	33.70	AC turn OFF
84	22/05/2022	17:45:52	0.00	218.80	0.00	0.00	OFF	33.70	AC turn OFF

Berdasarkan tabel hasil uji, sesaat setelah sistem ON (baris ke 3) menunjukkan pendingin ruangan telah dinyalakan oleh sistem pada pukul 17: 36:16 dengan kondisi temperatur ruangan 33.70.

2	22/05/2022	17:36:09	0.00	218.70	0.00	0.00	OFF	33.70	System ON
3	22/05/2022	17:36:16	0.00	218.60	0.00	0.00	ON	33.70	AC turn ON

Gambar 3. Kondisi pendingin ruangan dinyalakan.

Setelah 5 menit yaitu pukul 17:41:19, namun temperatur ruangan tidak berubah menjadi lebih rendah dari 33.70 derajat celcius akan tetapi berubah lebih tinggi menjadi 33.80 derajat celcius , hal ini dianggap pendingin ruangan dalam keadaan tidak normal. Oleh karena itu pendingin ruangan dimatikan.

44	22/05/2022	17:41:10	0.09	219.10	13.30	0.68	ON	33.70	AC turn ON
45	22/05/2022	17:41:14	0.09	218.80	13.30	0.68	ON	33.70	5'th minutes
46	22/05/2022	17:41:19	0.00	219.90	0.00	0.00	OFF	33.80	AC turn OFF

Gambar 4. Kondisi pendingin ruangan dimatikan.

2. Modul Lampu

Modul lampu ini menggunakan sensor arus PZEM-004T dan sensor LDR yang menjadi alat untuk monitoring sekaligus membandingkan data untuk dijadikan acuan tindakan otomatisasi pada sistem. Berikut diperlihatkan tabel hasil uji coba modul lampu dengan kondisi ruangan gelap dan terang. Uji coba ini merupakan uji coba menggunakan beban lampu 20 Watt.

Tabel 3. Percobaan dengan kondisi ruangan gelap dan terang.

No	Date	Time	I (A)	V (Volt)	P (W)	PF	Sw	Room Cond.	Status
1	22/05/2022	20:01:44	0.00	217.10	0.00	0.00	OFF	Null	System ON
2	22/05/2022	20:01:51	0.00	217.40	0.00	0.00	OFF	Null	System ON
3	22/05/2022	20:01:57	0.04	217.10	13.80	0.71	ON	DARK	Lamp turn ON
4	22/05/2022	20:02:03	0.09	217.00	14.10	0.71	ON	DARK	Lamp turn ON
5	22/05/2022	20:02:09	0.09	216.80	14.30	0.71	ON	DARK	Lamp turn ON
6	22/05/2022	20:02:15	0.00	217.00	0.00	0.00	OFF	BRIGHT	Lamp turn OFF
7	22/05/2022	20:02:20	0.00	217.10	0.00	0.00	OFF	BRIGHT	Lamp turn OFF
8	22/05/2022	20:02:25	0.09	216.80	14.40	0.71	ON	DARK	Lamp turn ON
9	22/05/2022	20:02:29	0.09	217.10	14.50	0.71	ON	DARK	Lamp turn ON
10	22/05/2022	20:02:35	0.09	216.90	14.40	0.71	ON	DARK	Lamp turn ON
11	22/05/2022	20:02:40	0.00	217.10	0.70	1.00	OFF	BRIGHT	Lamp turn OFF
12	22/05/2022	20:02:45	0.00	217.00	0.00	0.00	OFF	BRIGHT	Lamp turn OFF
13	22/05/2022	20:02:50	0.00	217.00	0.00	0.00	OFF	BRIGHT	Lamp turn OFF
14	22/05/2022	20:02:54	0.09	216.70	13.80	0.67	ON	DARK	Lamp turn ON
15	22/05/2022	20:03:02	0.09	217.30	14.60	0.71	ON	DARK	Lamp turn ON
16	22/05/2022	20:03:07	0.09	217.40	0.00	0.00	OFF	Null	Resetted system
17	22/05/2022	20:03:12	0.00	217.40	0.00	0.00	OFF	Null	Resetted system
18	22/05/2022	20:03:17	0.00	217.50	0.00	0.00	OFF	Null	Resetted system
19	22/05/2022	20:03:23	0.00	217.50	0.00	0.00	OFF	Null	Resetted system
20	22/05/2022	20:03:28	0.00	216.90	0.00	0.00	OFF	Null	Resetted system
21	22/05/2022	20:03:34	0.00	217.00	0.00	0.00	OFF	Null	Resetted system
22	22/05/2022	20:03:40	0.00	216.80	0.00	0.00	OFF	Null	Resetted system
23	22/05/2022	20:03:45	0.00	216.60	0.00	0.00	OFF	Null	Resetted system
24	22/05/2022	20:03:51	0.06	216.70	14.50	0.71	ON	DARK	Lamp turn ON
25	22/05/2022	20:03:56	0.09	216.50	14.50	0.71	ON	DARK	Lamp turn ON
26	22/05/2022	20:04:02	0.09	216.70	14.40	0.71	ON	DARK	Lamp turn ON

Berdasarkan tabel hasil uji, sesaat setelah sistem ON (baris ke 3) menunjukkan ruangan dalam kondisi gelap, maka lampu dinyalakan.

2	22/05/2022	20:01:51	0.00	217.40	0.00	0.00	OFF	Null	System ON
3	22/05/2022	20:01:57	0.04	217.10	13.80	0.71	ON	DARK	Lamp turn ON
4	22/05/2022	20:02:03	0.09	217.00	14.10	0.71	ON	DARK	Lamp turn ON

Gambar 5. Kondisi ruangan gelap.

Pada data baris ke 6 menunjukkan kondisi ruangan terang, maka lampu dimatikan.

5	22/05/2022	20:02:09	0.09	216.80	14.30	0.71	ON	DARK	Lamp turn ON
6	22/05/2022	20:02:15	0.00	217.00	0.00	0.00	OFF	BRIGHT	Lamp turn OFF

Gambar 6. Kondisi ruangan terang.

3. Modul Stok Kontak

Modul stok kontak ini menggunakan sensor arus PZEM-004T dan sensor suhu DS18B20 yang menjadi alat untuk monitoring sekaligus membandingkan data untuk dijadikan acuan tindakan otomatisasi pada sistem. Berikut diperlihatkan tabel hasil uji coba modul stok kontak dengan kondisi panas berlebih (overheat) dan beban berlebih (overload). Uji coba ini merupakan uji coba terbatas menggunakan beban lampu 20 Watt, cas laptop 45 Watt, serta solder untuk memanaskan sensor suhu.

Tabel 4. Percobaan dengan kondisi kabel panas berlebih.

ID	Date	Time	I (A)	V (Volt)	P (W)	PF	Sw	Cable therm.	Status
1	22/05/2022	20:52:25	0.00	218.10	0.00	0.00	OFF	33.44	System ON
2	22/05/2022	20:52:33	0.0	217.90	14.40	0.70	ON	33.38	Socket turn ON
3	22/05/2022	20:52:40	0.09	218.00	14.40	0.70	ON	33.25	Socket turn ON
4	22/05/2022	20:52:46	0.09	218.00	14.30	0.71	ON	33.25	Socket turn ON
5	22/05/2022	20:52:53	0.09	218.10	14.30	0.70	ON	33.69	Socket turn ON
6	22/05/2022	20:52:59	0.09	217.90	14.20	0.71	ON	35.50	Socket turn ON
7	22/05/2022	20:53:06	0.09	218.20	14.20	0.70	ON	37.63	Socket turn ON
8	22/05/2022	20:53:14	0.09	218.30	14.10	0.70	ON	39.81	Socket turn ON
9	22/05/2022	20:53:20	0.09	218.20	13.90	0.70	ON	42.75	Socket turn ON
10	22/05/2022	20:53:28	0.09	217.90	14.00	0.70	ON	45.94	Socket turn ON
11	22/05/2022	20:53:35	0.09	218.00	14.00	0.70	ON	51.81	Socket turn ON
12	22/05/2022	20:53:41	0.09	218.10	14.00	0.70	ON	56.75	Socket turn ON
13	22/05/2022	20:53:48	0.09	218.00	14.00	0.71	ON	61.50	Socket turn ON
14	22/05/2022	20:53:56	0.09	217.90	14.00	0.71	ON	66.19	Socket turn ON
15	22/05/2022	20:54:09	0.00	218.00	0.00	0.00	OFF	73.63	Overheat
16	22/05/2022	20:54:15	0.00	218.00	0.00	0.00	OFF	73.63	Socket turn OFF
17	22/05/2022	20:54:21	0.00	218.10	0.00	0.00	OFF	73.63	Socket turn OFF
18	22/05/2022	20:54:26	0.00	217.90	0.00	0.00	OFF	73.63	Socket turn OFF
19	22/05/2022	20:54:31	0.00	218.30	0.00	0.00	OFF	73.63	Socket turn OFF

20	22/05/2022	20:54:35	0.00	218.40	0.00	0.00	OFF	73.63	Socket turn OFF
21	22/05/2022	20:54:42	0.00	218.40	0.00	0.00	OFF	73.63	Socket turn OFF
22	22/05/2022	20:54:47	0.00	218.30	0.00	0.00	OFF	73.63	Socket turn OFF
23	22/05/2022	20:54:54	0.00	218.10	0.00	0.00	OFF	73.63	Socket turn OFF
24	22/05/2022	20:55:02	0.00	217.60	0.00	0.00	OFF	73.63	Socket turn OFF
25	22/05/2022	20:55:11	0.00	218.20	0.00	0.00	OFF	73.63	Socket turn OFF
26	22/05/2022	20:55:16	0.00	218.30	0.00	0.00	OFF	73.63	Socket turn OFF
27	22/05/2022	20:55:23	0.00	218.10	0.00	0.00	OFF	73.63	Socket turn OFF
28	22/05/2022	20:55:28	0.00	217.90	0.00	0.00	OFF	73.63	Socket turn OFF
29	22/05/2022	20:56:38	0.00	0.00	0.00	0.00	OFF	0.00	Resetted system
30	22/05/2022	20:56:42	0.00	218.70	0.00	0.00	OFF	0.00	System ON
31	22/05/2022	20:56:47	0.00	218.80	0.00	0.00	OFF	0.00	System ON
32	22/05/2022	20:56:52	0.00	218.50	0.00	0.00	OFF	0.00	System ON
33	22/05/2022	20:56:58	0.00	218.30	0.00	0.00	OFF	0.00	System ON
34	22/05/2022	20:57:02	0.00	218.20	0.00	0.00	OFF	0.00	System ON
35	22/05/2022	20:57:07	0.00	218.60	0.00	0.00	OFF	0.00	System ON
36	22/05/2022	20:57:12	0.00	218.10	0.00	0.00	OFF	0.00	System ON
37	22/05/2022	20:57:17	0.00	218.00	0.00	0.00	OFF	0.00	System ON
38	22/05/2022	20:57:21	0.00	217.90	0.00	0.00	OFF	0.00	System ON
39	22/05/2022	20:57:26	0.00	218.00	0.00	0.00	OFF	0.00	System ON
40	22/05/2022	20:57:32	0.00	218.10	0.00	0.00	OFF	0.00	System ON
41	22/05/2022	20:57:40	0.00	218.20	0.00	0.00	OFF	0.00	System ON
42	22/05/2022	20:57:47	0.00	218.10	1.60	0.71	ON	46.75	Socket turn ON
43	22/05/2022	20:57:56	0.09	218.00	14.40	0.71	ON	46.00	Socket turn ON
44	22/05/2022	20:58:08	0.09	218.40	14.50	0.71	ON	45.00	Socket turn ON
45	22/05/2022	20:58:17	0.09	218.40	14.50	0.71	ON	44.31	Socket turn ON
46	22/05/2022	20:58:23	0.09	218.90	14.40	0.71	ON	43.75	Socket turn ON

Berdasarkan tabel hasil uji, sesaat setelah sistem ON (baris ke 2) menunjukkan stok kontak dinyalakan.

1	22/05/2022	20:52:25	0.00	218.10	0.00	0.00	OFF	33.44	System ON
2	22/05/2022	20:52:33	0.0	217.90	14.40	0.70	ON	33.38	Socket turn ON

Gambar 7. Kondisi stok kontak dinyalakan.

Pada data baris ke 15 menunjukkan kondisi kabel stok kontak mencapai 73,63 derajat celcius, telah melebihi batas maksimal suhu kabel yaitu 70 derajat celcius (overheat). Oleh karena itu stok kontak dinonaktifkan.

14	22/05/2022	20:53:56	0.09	217.90	14.00	0.71	ON	66.19	Socket turn ON
15	22/05/2022	20:54:09	0.00	218.00	0.00	0.00	OFF	73.63	Overheat
16	22/05/2022	20:54:15	0.00	218.00	0.00	0.00	OFF	73.63	Socket turn OFF

Gambar 8. Kondisi kabel overheat.

Tabel 5. Percobaan dengan kondisi beban berlebih.

No	Date	Time	I (A)	V (Volt)	P (W)	PF	Sw	Cable temp.	Status
1	22/05/2022	21:56:20	0.00	219.80	0.00	0.00	OFF	0.00	System ON
2	22/05/2022	21:56:24	0.00	220.10	0.00	0.00	OFF	0.00	Socket turn ON
3	22/05/2022	21:56:31	0.09	220.50	14.00	0.71	ON	31.94	Socket turn ON
4	22/05/2022	21:57:22	0.21	220.20	43.70	0.95	ON	31.94	Overload
5	22/05/2022	21:57:27	0.00	220.20	0.00	0.00	OFF	31.94	Soket turn OFF
6	22/05/2022	21:57:34	0.00	220.00	0.00	0.00	OFF	31.94	Soket turn OFF
7	22/05/2022	21:57:39	0.00	220.10	0.00	0.00	OFF	31.94	Soket turn OFF
8	22/05/2022	21:57:48	0.00	220.40	0.00	0.00	OFF	31.94	Soket turn OFF
9	22/05/2022	21:57:54	0.00	220.60	0.00	0.00	OFF	31.94	Soket turn OFF
10	22/05/2022	21:58:04	0.00	221.10	0.00	0.00	OFF	31.94	Soket turn OFF
11	22/05/2022	22:10:48	0.00	0.00	0.00	0.00	OFF	0.00	Resetted system
12	22/05/2022	22:10:55	0.00	0.00	0.00	0.00	OFF	0.00	Resetted system
13	22/05/2022	22:11:02	0.09	218.00	14.30	0.71	ON	31.31	Socket turn ON
14	22/05/2022	22:11:09	0.09	218.00	14.30	0.71	ON	31.31	Socket turn ON
15	22/05/2022	22:11:18	0.09	217.90	14.60	0.95	ON	31.31	Socket turn ON
16	22/05/2022	22:12:26	0.21	217.60	43.70	0.95	ON	31.31	Overload
17	22/05/2022	22:12:33	0.00	0.00	0.00	0.00	OFF	0.00	Resetted system
18	22/05/2022	22:12:37	0.00	0.00	0.00	0.00	OFF	0.00	Resetted system
19	22/05/2022	22:12:43	0.09	217.80	14.30	0.00	ON	31.25	Socket turn ON
20	22/05/2022	22:12:47	0.09	217.40	14.30	0.71	ON	31.25	Socket turn ON
21	22/05/2022	22:12:56	0.00	218.20	7.80	0.72	ON	31.25	Socket turn ON
22	22/05/2022	22:13:03	0.09	217.80	14.30	0.00	ON	31.25	Socket turn ON
23	22/05/2022	22:13:09	0.09	217.40	14.30	0.71	ON	31.25	Socket turn ON
24	22/05/2022	22:13:16	0.09	217.40	14.20	0.71	ON	31.25	Socket turn ON
25	22/05/2022	22:13:24	0.09	217.60	14.20	0.70	ON	31.25	Socket turn ON
26	22/05/2022	22:13:32	0.09	217.70	14.10	0.71	ON	31.25	Socket turn ON
27	22/05/2022	22:13:39	0.09	217.50	14.10	0.71	ON	31.25	Socket turn ON
28	22/05/2022	22:13:46	0.09	217.60	14.00	0.71	ON	31.25	Socket turn ON
29	22/05/2022	22:13:53	0.09	217.70	14.00	0.71	ON	31.25	Socket turn ON
30	22/05/2022	22:14:19	0.21	217.50	13.90	0.71	ON	31.25	Socket turn ON

31	22/05/2022	22:14:25	0.09	218.40	13.90	0.71	ON	31.25	Socket turn ON
32	22/05/2022	22:14:39	0.09	218.80	13.90	0.95	ON	31.25	Socket turn ON
33	22/05/2022	22:15:06	0.21	218.10	13.70	0.71	ON	31.25	Socket turn ON
34	22/05/2022	22:15:13	0.09	217.90	13.80	0.70	ON	31.25	Socket turn ON
35	22/05/2022	22:15:20	0.09	218.10	13.70	0.71	ON	31.25	Socket turn ON
36	22/05/2022	22:15:27	0.09	218.10	13.70	0.95	ON	31.25	Socket turn ON
37	22/05/2022	22:16:36	0.21	217.70	43.30	0.95	ON	31.25	Overload
38	22/05/2022	22:16:41	0.06	218.60	11.70	0.00	OFF	31.25	Socket turn ON
39	22/05/2022	22:16:45	0.00	218.70	0.00	0.00	OFF	31.25	Socket turn ON
40	22/05/2022	22:16:51	0.00	218.60	0.00	0.00	OFF	31.25	Socket turn ON
41	22/05/2022	22:16:55	0.00	218.50	0.00	0.00	OFF	31.25	Socket turn ON
42	22/05/2022	22:17:00	0.00	218.50	0.00	0.00	OFF	31.25	Socket turn ON
43	22/05/2022	22:17:05	0.00	218.50	0.00	0.00	OFF	31.25	Socket turn ON
44	22/05/2022	22:17:10	0.00	218.30	0.00	0.00	OFF	31.25	Socket turn ON

Berdasarkan tabel hasil uji, sesaat setelah sistem ON (baris ke 2) menunjukkan stok kontak dinyalakan.

1	22/05/2022	21:56:20	0.00	219.80	0.00	0.00	OFF	0.00	System ON
2	22/05/2022	21:56:24	0.00	220.10	0.00	0.00	OFF	0.00	Socket turn ON

Gambar 9. Kondisi stok kontak dinyalakan.

Pada data baris ke 4 menunjukkan kondisi daya beban stok kontak mencapai 43,70 watt, telah melebihi batas maksimal daya yang telah ditentukan yaitu 40 watt (overload). Oleh karena itu stok kontak dinonaktifkan.

3	22/05/2022	21:56:31	0.09	220.50	14.00	0.71	ON	31.94	Socket turn ON
4	22/05/2022	21:57:22	0.21	220.20	43.70	0.95	ON	31.94	Overload

Gambar 10. Kondisi stok kontak overload.

4. Modul Panel

Modul panel ini hanya menggunakan sensor arus PZEM-004T yang menjadi alat untuk monitoring penggunaan arus. Berikut diperlihatkan tabel hasil uji coba modul panel.

Tabel 6. Percobaan monitoring penggunaan arus.

No	Date	Time	I (A)	V (Volt)	P (W)	PF	Status
1	22/05/2022	16:42:54	0.16	219.90	23.80	0.72	Normal
2	22/05/2022	16:42:56	0.16	220.20	24.00	0.72	Normal
3	22/05/2022	16:42:58	0.16	220.20	24.30	0.72	Normal

4	22/05/2022	16:43:01	0.16	220.30	23.80	0.72	Normal
5	22/05/2022	16:43:04	0.16	220.10	23.80	0.72	Normal
6	22/05/2022	16:43:06	0.16	220.20	25.50	0.71	Normal
7	22/05/2022	16:43:09	0.16	220.10	24.50	0.72	Normal
8	22/05/2022	16:43:11	0.16	220.20	24.60	0.72	Normal
9	22/05/2022	16:43:13	0.16	220.20	24.00	0.72	Normal
10	22/05/2022	16:43:17	0.16	220.30	24.00	0.72	Normal
11	22/05/2022	16:43:21	0.16	220.20	24.40	0.71	Normal
12	22/05/2022	16:43:23	0.16	219.50	24.40	0.72	Normal
13	22/05/2022	16:43:26	0.16	219.60	24.60	0.71	Normal
14	22/05/2022	16:43:28	0.16	219.80	24.40	0.71	Normal
15	22/05/2022	16:43:30	0.16	219.90	24.10	0.71	Normal
16	22/05/2022	16:43:33	0.16	220.00	23.70	0.71	Normal
17	22/05/2022	16:43:35	0.16	220.10	23.70	0.71	Normal
18	22/05/2022	16:43:38	0.16	220.00	24.00	0.72	Normal
19	22/05/2022	16:43:40	0.16	220.10	23.90	0.71	Normal
20	22/05/2022	16:43:42	0.16	220.10	0.00	0.71	Normal
21	22/05/2022	16:43:46	0.16	220.20	24.60	0.71	Normal
22	22/05/2022	16:43:48	0.16	220.20	24.30	0.71	Normal
23	22/05/2022	16:43:51	0.16	220.20	24.30	0.71	Normal
24	22/05/2022	16:43:53	0.16	220.00	23.70	0.71	Normal
25	22/05/2022	16:43:56	0.16	220.10	24.10	0.72	Normal
26	22/05/2022	16:43:59	0.16	220.10	24.10	0.71	Normal
27	22/05/2022	16:44:01	0.16	219.90	23.80	0.72	Normal
28	22/05/2022	16:44:03	0.16	220.00	24.50	0.71	Normal
29	22/05/2022	16:44:06	0.17	220.00	23.70	0.71	Normal
30	22/05/2022	16:44:08	0.16	219.90	23.90	0.71	Normal
31	22/05/2022	16:44:11	0.16	220.70	26.30	0.71	Normal
32	22/05/2022	16:44:13	0.16	220.90	24.00	0.72	Normal
33	22/05/2022	16:44:15	0.16	220.90	24.30	0.72	Normal
34	22/05/2022	16:44:20	0.16	220.80	23.80	0.71	Normal
35	22/05/2022	16:44:22	0.16	220.90	24.30	0.72	Normal
36	22/05/2022	16:44:24	0.16	220.60	23.60	0.72	Normal
37	22/05/2022	16:44:27	0.15	220.60	23.90	0.72	Normal
38	22/05/2022	16:44:29	0.15	220.90	10.60	0.50	Normal
39	22/05/2022	16:44:32	0.10	221.00	10.90	0.50	Normal
40	22/05/2022	16:44:34	0.10	220.80	10.50	0.50	Normal
41	22/05/2022	16:44:38	0.10	220.80	10.60	0.50	Normal
42	22/05/2022	16:44:41	0.10	220.80	10.50	0.50	Normal
43	22/05/2022	16:44:43	0.10	220.90	10.40	0.49	Normal
44	22/05/2022	16:44:46	0.10	220.90	11.20	0.50	Normal
45	22/05/2022	16:44:48	0.10	220.80	10.70	0.50	Normal
46	22/05/2022	16:44:51	0.11	220.90	11.30	0.51	Normal

47	22/05/2022	16:44:53	0.10	220.80	10.70	0.50	Normal
48	22/05/2022	16:45:50	0.16	221.30	24.20	0.72	Normal
49	22/05/2022	16:45:53	0.16	221.20	23.90	0.72	Normal
50	22/05/2022	16:45:55	0.16	221.30	23.90	0.72	Normal
51	22/05/2022	16:45:58	0.16	221.10	24.00	0.72	Normal
52	22/05/2022	16:46:01	0.16	221.50	24.00	0.72	Normal
53	22/05/2022	16:46:04	0.16	221.60	24.20	0.72	Normal
54	22/05/2022	16:46:10	0.15	221.00	24.00	0.72	Normal
55	22/05/2022	16:46:14	0.16	221.10	24.40	0.72	Normal
56	22/05/2022	16:46:18	0.16	221.20	23.90	0.72	Normal
57	22/05/2022	16:46:22	0.16	221.20	24.00	0.72	Normal
58	22/05/2022	16:46:25	0.15	221.20	24.00	0.72	Normal
59	22/05/2022	16:46:28	0.15	221.10	23.90	0.72	Normal