

RANCANG BANGUN SISTEM MONITORING KONSUMSI ENERGI LISTRIK BERBASIS PROGRESSIVE WEB APP DENGAN REKOMENDASI OPTIMASI PENGGUNA

PROYEK AKHIR

Disusun Oleh:

M Zaini Ridha

3312301106

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya
Teknik Informatika



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM**

2026

KATA PENGANTAR

Puji dan syukur saya panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga saya dapat menyelesaikan Proyek Akhir saya dengan baik dan tepat waktu. Laporan proyek akhir ini disusun sebagai salah satu syarat dalam menyelesaikan program studi Teknik Informatika. Dalam penyusunan proyek akhir ini, penulis menyadari bahwa banyak pihak yang telah memberikan bantuan dan dukungan. Dengan demikian, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Bapak Muchamad Fajri Amirul Nasrullah, [S.ST.](#), [M.Sc.](#), selaku dosen pembimbing yang telah memberikan arahan, bimbingan dan motivasi selama proses penyelesaian proyek akhir.
2. Ibu Yeni Rokhayati, [S.Si.](#), [M.Sc.](#), selaku ketua program studi teknik informatika yang telah memberikan dukungan serta kebijakan akademik selama proses penyelesaian studi.
3. Kedua orang tua dan keluarga yang selalu memberikan dukungan dan semangat kepada penulis.
4. Teman-teman yang turut membantu dan saling mendukung selama proses penyelesaian proyek akhir berlangsung.

Penulis menyadari bahwa laporan proyek akhir ini masih banyak kekurangan dan jauh dari sempurna. Maka penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan di masa akan datang. Semoga laporan proyek akhir ini dapat memberikan manfaat bagi pembaca dan menjadi referensi yang berguna di kemudian hari.

DAFTAR ISI

KATA PENGANTAR.....	2
DAFTAR ISI.....	3
DAFTAR TABEL.....	5
DAFTAR GAMBAR.....	6
DAFTAR LAMPIRAN.....	8
ABSTRAK.....	9
BAB I PENDAHULUAN.....	11
1.1. Latar Belakang.....	11
1.2. Rumusan Masalah.....	13
1.3. Tujuan.....	13
1.4. Batasan Masalah.....	13
1.5. Manfaat.....	15
1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek.....	16
BAB II TINJAUAN PUSTAKA.....	17
2.1. Tinjauan Solusi dan Sistem Terkait.....	17
2.2. Landasan Konsep dan Teknologi.....	19
2.2.1. Monitoring Konsumsi Energi Listrik.....	19
2.2.2. Progressive Web App (PWA).....	19
2.2.3. Sistem Monitoring Berbasis Website.....	20
2.2.4. Rekomendasi Optimasi Penggunaan Energi.....	21
2.2.5. NestJs Framework.....	22
2.2.6. Quasar Framework.....	22
2.2.7. Relational Database, MySQL, dan Sequelize ORM.....	22
2.2.8. Web Push Notification.....	23
2.3. Metode Pengembangan Produk.....	23
BAB III ANALISIS DAN PERANCANGAN.....	25
3.1. Analisis Kebutuhan.....	25
3.1.1 Proses Bisnis Berjalan.....	25
3.1.2 Gambaran Umum Sistem.....	26
3.2. Perancangan.....	26
3.2.1 Kebutuhan Fungsional.....	26
3.2.2 Kebutuhan Non-Fungsional.....	28
3.2.3 Diagram Use Case.....	28
3.2.4 Skenario Use Case.....	29
3.2.5 Activity diagram.....	40

3.2.6 ER Diagram.....	54
3.2.7 Perancangan Antarmuka (Wireframe).....	55
3.2.8 Perancangan Metode Simple Additive Whiting (SAW).....	64
BAB IV IMPLEMENTASI DAN PENGUJIAN.....	69
4.1. Hasil Implementasi.....	69
4.1.1 Instalasi Aplikasi.....	69
4.1.2 Fitur Aplikasi.....	72
4.1.3 Alur Kerja Aplikasi.....	80
4.2. Pengujian.....	82
4.2.1 Pengujian Fungsional.....	82
BAB V KESIMPULAN.....	84
5.1 Kesimpulan.....	84
5.2 Saran.....	84
DAFTAR PUSTAKA.....	86
DAFTAR LAMPIRAN.....	88

DAFTAR TABEL

Table 2.1 Tinjauan Aplikasi Sejenis.....	18
Tabel 3.1 Kebutuhan Fungsional.....	26
Tabel 3.2 Kebutuhan Non Fungsional.....	28
Table 3.3 Use Case Scenario Registrasi Pengguna.....	29
Table 3.4 Use Case Scenario Login.....	31
Table 3.5 Use Case Scenario Mengelola Profil & Informasi Rumah.....	32
Table 3.6 Use Case Scenario Mengelola Golongan Tarif.....	33
Table 3.7 Use Case Scenario Mengelola Kategori Perangkat.....	34
Tabel 3.8 Use Case Scenario Mengelola Perangkat.....	35
Table 3.9 Use Case Scenario Mencatat Pemakaian Harian.....	36
Table 3.10 Use Case Scenario Melihat Dashboard Analisis.....	37
Table 3.11 Use Case Scenario Melihat DSS Smart Insights.....	38
Table 3.12 Use Case Scenario Menerima Notifikasi.....	40
Table 3.13 Penentuan Kriteria.....	65
Tabel 4.1 Dependensi dan Tools yang Digunakan.....	70
Tabel 4.2 Pengujian Fungsional.....	83
Tabel 1 Lampiran Hasil Kuesioner Pengguna Listrik Rumah Tangga.....	88

DAFTAR GAMBAR

Gambar 2.1 Progressive Web App.....	20
Gambar 2.2 Proses Input dan Output Rekomendasi Optimasi Energi.....	22
Gambar 2.3 Metode Waterfall.....	23
Gambar 3.1 Proses Bisnis Berjalan.....	25
Gambar 3.2 Gambaran Umum Sistem.....	26
Gambar 3.3 Use Case Diagram.....	29
Gambar 3.4 Activity Diagram Registrasi Pengguna.....	41
Gambar 3.5 Activity Diagram Login.....	42
Gambar 3.6 Activity Diagram Mengelola Profil & Informasi Rumah.....	43
Gambar 3.7 Activity Diagram Mengelola Golongan Tarif.....	44
Gambar 3.8 Activity Diagram Mengelola Kategori Perangkat.....	45
Gambar 3.9 Activity Diagram Mengelola Perangkat.....	46
Gambar 3.10 Activity Diagram Mencatat Pemakaian Perangkat Harian.....	48
Gambar 3.11 Activity Diagram Melihat Dashboard Analysis & Download Laporan.....	49
Gambar 3.12 Activity Diagram Melihat DSS Smart Insights.....	51
Gambar 3.13 Activity Diagram Menerima Notifikasi.....	53
Gambar 3.14 ER Diagram.....	54
Gambar 3.15 Antarmuka Registrasi Pengguna.....	55
Gambar 3.16 Antarmuka Login.....	56
Gambar 3.16 Antarmuka Mengelola Mengelola Profil & Informasi Rumah.....	57
Gambar 3.17 Antarmuka Mengelola Tarif.....	58
Gambar 3.18 Antarmuka Mengelola Kategori Perangkat.....	59
Gambar 3.19 Antarmuka Mengelola Perangkat Listrik.....	60
Gambar 3.20 Antarmuka Mencatat Pemakaian Perangkat Harian.....	61
Gambar 3.21 Antarmuka Melihat Dashboard Analisis & Download Laporan.....	62
Gambar 3.22 Antarmuka Melihat DSS Smart Insight.....	63
Gambar 3.23 Antarmuka Menerima Notifikasi.....	64
Gambar 4.1 Halaman Registrasi dan Login.....	72
Gambar 4.2 Halaman Dashboard.....	73
Gambar 4.3 Halaman Manajemen Rumah Tangga.....	74
Gambar 4.4 Halaman Manajemen Perangkat Listrik.....	75
Gambar 4.5 Halaman Pemantauan dan Simulasi Penggunaan.....	75
Gambar 4.6 Halaman Notifikasi.....	76
Gambar 4.7 Halaman Profil Pengguna.....	77

Gambar 4.8 Halaman Kategori Perangkat.....	77
Gambar 4.9 Halaman Golongan Tarif.....	78
Gambar 4.10 Gambar Web App Manifest.....	79
Gambar 4.11 Gambar Service Worker.....	79
Gambar 4.12 Gambar Cache Storage.....	80
Gambar 4.13 Halaman Masuk.....	81
Gambar 4.14 Halaman Masuk.....	82

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement.....	88
Lampiran B. Link Produk.....	89
Lampiran C. Dokumen Pengujian.....	89

ABSTRAK

Konsumsi energi listrik rumah tangga di Indonesia terus meningkat seiring bertambahnya jumlah peralatan elektronik yang digunakan sehari-hari, namun sebagian besar masyarakat belum memiliki alat bantu yang praktis untuk memantau dan mengendalikan pemakaian listrik mereka secara efisien. Pencatatan pemakaian yang masih dilakukan secara manual menyebabkan kurangnya kesadaran terhadap pola konsumsi energi yang boros sehingga berdampak pada tingginya tagihan listrik bulanan. Proyek akhir ini bertujuan untuk merancang dan membangun sistem monitoring konsumsi energi listrik berbasis *Progressive Web App* (PWA) bernama Voltaris yang dilengkapi fitur rekomendasi optimasi bagi pengguna rumah tangga. Pengembangan sistem menggunakan metode *Waterfall* yang meliputi tahap analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Arsitektur sistem dibangun dengan pendekatan *client-server* menggunakan NestJS sebagai backend API dan Quasar *Framework* (Vue.js 3) sebagai *frontend* PWA, dengan MySQL sebagai basis data relasional. Sistem ini memungkinkan pengguna mencatat peralatan elektronik beserta durasi pemakaian hariannya, menghitung estimasi konsumsi energi dalam kWh dan biaya dalam Rupiah berdasarkan golongan tarif PLN, serta menerima rekomendasi optimasi secara otomatis melalui mekanisme *Decision Support System* (DSS) apabila terdeteksi pola pemakaian yang tidak efisien. Aplikasi juga menyediakan fitur ekspor laporan pemakaian energi dalam format PDF dan notifikasi *Web Push* sebagai media penyampaian rekomendasi penghematan. Hasil pengujian *black-box* menunjukkan bahwa seluruh fungsionalitas sistem berjalan sesuai spesifikasi kebutuhan. Sistem ini diharapkan dapat membantu masyarakat dalam mengoptimalkan penggunaan energi listrik sehingga mampu menekan biaya tagihan listrik bulanan secara signifikan.

Kata Kunci: Monitoring Energi Listrik, *Progressive Web App*, *Decision Support System*, Optimasi Konsumsi, *Waterfall*

BAB I PENDAHULUAN

1.1. Latar Belakang

Pertumbuhan populasi dan perkembangan teknologi yang pesat telah mendorong peningkatan pada konsumsi energi secara signifikan. Salah satu sektor yang penyumbang konsumsi listrik terbesar adalah sektor rumah tangga. Sektor ini mengalami peningkatan konsumsi energi, dimana listrik digunakan untuk mendukung kegiatan sehari-hari, seperti memenuhi kebutuhan air hangat, menjalankan pompa air dan digunakan untuk berbagai penggunaan peralatan elektronik lainnya.

Di Indonesia, konsumsi energi listrik menunjukkan peningkatan setiap tahunnya. Berdasarkan data Badan Pusat Statistika (BPS), konsumsi energi listrik per kapita meningkat dari 0,84 MWh per kapita pada tahun 2013 menjadi 1,20 MWh per kapita pada tahun 2022. Sejalan dengan peningkatan tersebut, peningkatan terjadi pada rumah tangga yang mendapatkan akses terhadap kelistrikan, di mana persentase pada tahun 2013 adalah 80,51% dan meningkat menjadi 99,63% pada tahun 2022 (Badan Pusat Statistik, n.d). Peningkatan ini didorong oleh pembangunan infrastruktur kelistrikan di berbagai wilayah.

Tingginya konsumsi energi oleh rumah tangga mencerminkan tingginya tingkat ketergantungan masyarakat terhadap listrik untuk memenuhi kebutuhan sehari-hari seperti penerangan. Listrik juga digunakan untuk mendukung kenyamanan dan produktivitas mulai dari pendingin ruangan, hingga penggunaan perangkat elektronik lainnya. Peningkatan konsumsi ini tidak lepas dari pengaruh peningkatan populasi, urbanisasi, dan daya beli masyarakat terhadap peralatan listrik (Jamilatun et al., 2025). Dampak dari tingginya konsumsi energi tersebut dapat dirasakan mulai dari meningkatnya tagihan listrik perbulan sampai meningkatnya emisi karbon yang berdampak pada lingkungan. Oleh karena itu, upaya dalam meningkatkan efisiensi penggunaan energi listrik menjadi sangat penting untuk mendukung keberlanjutan lingkungan.

Salah satu teknologi yang dapat memudahkan akses dalam sistem monitoring adalah *Progressive Web App* (PWA). Menurut Amrullah et al (2021) *Progressive Web App* (PWA) adalah konsep yang menyatukan bagian terbaik dari web dan *native apps*. Dimana PWA menggunakan kemampuan web modern yang memberikan pengalaman pengguna seperti menggunakan aplikasi *native*, pwa dapat dijalankan saat sedang *offline* dan mendukung *splash screen* dan *push notification*.

Penelitian yang dilakukan oleh Bimanatara et al (2025) menyatakan bahwa penerapan PWA pada sistem pesan antar makanan mampu memberikan kemudahan akses melalui fitur instalasi lintas platform, akses offline dan push notifications. Hasil penelitian menunjukkan bahwa PWA merupakan teknologi yang relevan untuk diterapkan di sistem yang membutuhkan akses cepat, fleksibel, dan mudah diakses,

Penelitian sebelumnya yang dilakukan oleh Karuniawan (2024) mengembangkan sistem monitoring konsumsi energi listrik berbasis IoT. Sistem tersebut menampilkan data pengguna energi secara *real-time*. Namun sistem belum mendukung rekomendasi penggunaan energi dan belum memanfaatkan teknologi berbasis *Progressive Web App* (PWA) yang dapat diakses melalui berbagai platform tanpa memerlukan aplikasi tambahan. Oleh karena itu penelitian ini berfokus untuk membuat sistem monitoring konsumsi energi listrik yang dapat memberikan rekomendasi optimasi penggunaan listrik kepada pengguna dan dirancang berbasis PWA sehingga dapat diakses dengan mudah tanpa memerlukan instalasi khusus.

Pengembangan sistem ini juga sejalan dengan *Sustainable Development Goals* (SDGs) poin ke-7 yaitu energi bersih dan terjangkau. Dengan adanya sistem ini diharapkan masyarakat dapat lebih bijak dalam menggunakan energi listrik, mengurangi pemborosan penggunaan listrik dan mendukung upaya pelestarian lingkungan.

1.2. Rumusan Masalah

Berdasarkan latar belakang masalah yang telah di bahas diatas, maka rumusan masalah pada penelitian ini adalah sebagai berikut.

1. Bagaimana merancang sistem monitoring konsumsi energi listrik pada rumah tangga yang memberikan rekomendasi optimasi penggunaan energi listrik kepada pengguna?
2. Bagaimana merancang sistem dan membangun sistem monitoring berbasis *Progressive Web App* (PWA)?
3. Bagaimana memastikan fungsional aplikasi dapat berjalan sesuai kebutuhan?

1.3. Tujuan

Berdasarkan rumusan masalah di atas, maka tujuan dari penelitian ini adalah sebagai berikut.

1. Merancang dan membangun sistem monitoring konsumsi energi listrik pada rumah tangga yang memberikan rekomendasi optimasi penggunaan energi listrik kepada pengguna.
2. Merancang dan membangun sistem monitoring berbasis *Progressive Web App* (PWA).
3. Memastikan fungsional aplikasi dapat berjalan sesuai kebutuhan.

1.4. Batasan Masalah

Batasan masalah pada penelitian ini digunakan sebagai acuan dalam menghindari keluasan pembahasan dan menjaga penelitian tetap pada jalur yang sesuai dengan tujuan. Maka batasan masalah pada penelitian ini adalah sebagai berikut:

1. Sistem yang dikembangkan berbentuk *Progressive Web App* (PWA) berbasis web, tidak mendukung versi mobile atau desktop. Sehingga dapat diakses melalui browser dan tidak memerlukan instalasi tambahan.

2. Sistem tidak terhubung langsung secara *real-time* dengan kWh meter fisik atau sensor IoT (*hardware/smart meter*), melainkan input data penggunaan daya (Watt) dan durasi penggunaan perangkat dilakukan secara mandiri oleh pengguna.
3. Analisis rekomendasi penghematan energi pada *Decision Support System* (DSS) menggunakan aturan heuristik sederhana (*rule-based*).
4. Data penelitian dan simulasi dibatasi pada lingkup penggunaan peralatan elektronik skala rumah tangga (residensial) dan tidak mencakup kebutuhan monitoring energi industri atau gedung perkantoran besar.
5. Perhitungan estimasi biaya listrik mengacu pada data golongan tarif PLN yang diinput oleh Administrator pada sistem, serta tidak melakukan sinkronisasi otomatis secara *real-time* dengan API tarif PLN pusat.
6. Sistem hanya berfokus pada monitoring dan tidak mencakup transaksi pembayaran tagihan listrik secara langsung dan tidak terintegrasi dengan *Payment Gateway* pihak ketiga.
7. Laporan pemakaian daya yang dihasilkan sistem dibatasi pada bentuk ringkasan statistik (total kWh, estimasi biaya rupiah, rata-rata harian, dan log aktivitas) yang dapat diunduh dalam format dokumen PDF dengan rentang tanggal yang dipilih oleh pengguna.
8. Hak akses (role) pada sistem dibatasi menjadi dua peran utama, yaitu Admin (untuk mengelola master data seperti kategori perangkat dan kelas tarif) dan *User* (untuk mengelola data rumah tangga, perangkat elektronik, dan mencatat pemakaian).

1.5. Manfaat

Manfaat dari penelitian “Sistem Monitoring dan Rekomendasi Optimasi Konsumsi Energi Listrik Rumah Tangga Berbasis Progressive Web App (PWA)” terdiri atas:

1. Manfaat Teoritis

- a. Penelitian dapat menjadi referensi untuk pengembangan sistem yang berkaitan dengan sistem monitoring dan rekomendasi optimasi.
- b. Menambah wawasan mengenai teknologi berbasis *progressive web app* (PWA) dalam sistem monitoring.

2. Manfaat Praktis

- a. Membantu pengguna rumah tangga dalam memantau penggunaan energi listrik.
- b. Memberikan rekomendasi penggunaan energi listrik yang lebih efisien untuk mengurangi perilaku boros.
- c. Mempermudah dalam mengakses monitoring melalui aplikasi berbasis website tanpa memerlukan proses instalasi tambahan.
- d. Mendukung upaya penghematan energi dan pengurangan biaya tagihan listrik.

3. Kontribusi terhadap *Sustainable Development Goals* (SDGs)

- a. Penelitian berkontribusi dalam SDGs Goal 7, yaitu *Affordable and clean energy* yang berfokus pada penyediaan energi bersih, terjangkau dan berkelanjutan. Sistem membantu dalam pemantauan dan penghematan energi listrik, sehingga pengguna dapat mengurangi pemakaian listrik yang berlebihan.
- b. Kemudian, penelitian ini juga mendukung SDGs Goal 12, yaitu *responsible consumption and production*. Dimana sistem menyediakan fitur monitoring dan rekomendasi penggunaan listrik yang diharapkan mampu membantu masyarakat dalam menggunakan energi listrik secara lebih bijak.

1.6. Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Proyek akhir ini dikerjakan secara individu tanpa mengambil studi kasus dari perusahaan atau instansi tertentu. Namun, pelaksanaan proyek tetap tidak lepas dari kolaborasi yang dilakukan terhadap berbagai pihak untuk mendukung proses pengembangan sistem dan memastikan bahwa sistem telah memberikan solusi dari permasalahan.

Bentuk kolaborasi yang dilakukan selama pelaksanaan proyek diantaranya adalah sebagai berikut:

1. Melakukan konsultasi dengan dosen pembimbing untuk memperoleh arahan terkait pengembangan sistem secara keseluruhan, penyusunan dokumentasi pengembangan sistem dan luaran lainnya.
2. Melakukan diskusi dengan rekan-rekan mahasiswa untuk bertukar pendapat dan memperoleh masukan terkait implementasi fitur dan proses pengujian sistem, untuk memastikan aplikasi yang dikembangkan sudah optimal.
3. Pengumpulan umpan balik dari calon pengguna yang dilakukan sebagai bahan evaluasi untuk meningkatkan fungsionalitas dan kenyamanan pengguna dalam menggunakan aplikasi.

BAB II TINJAUAN PUSTAKA

2.1. Tinjauan Solusi dan Sistem Terkait

Penelitian terkait dengan sistem monitoring konsumsi energi listrik telah dikembangkan sebelumnya dengan berbagai pendekatan. Tinjauan terhadap solusi dan sistem terkait dilakukan untuk memahami fitur, teknologi, serta kelebihan dan keterbatasan dari sistem-sistem terdahulu, sehingga dapat dijadikan sebagai acuan dalam pengembangan sistem ini.

Penelitian pertama dilakukan oleh Zaen et al. (2021) dengan judul “Sistem Monitoring Konsumsi Energi Rumah Tangga Berbasis Web”, menghasilkan sebuah sistem monitorin yang berjalan secara real-time dengan akurasi sensor yang tinggi, selain itu sistem ini menyediakan fitur untuk menampilkan data tagihan listrik dan dapat diakses melalui *pc* maupun *smartphone*. Namun penelitian ini masih memiliki kekurangan karena tidak memiliki fitur rekomendasi penghematan energi, sistem yang hanya berfokus pada monitoring dan hanya sebatas website biasa belum mendukung PWA sehingga tidak bisa diakses secara *offline*.

Penelitian kedua dilakukan oleh Satya et al. (2024) dengan judul “Sistem Perhitungan Pemakaian Listrik Rumah Tangga Berbasis Android”, menghasilkan sebuah sistem monitoring konsumsi energi listrik secara real-time, yang dilengkapi oleh fitur perhitungan biaya listrik dan sistem rekomendasi efisiensi energi listrik. Namun sistem ini masih memiliki kekurangan, yaitu karena hanya menyediakan versi Android maka sistem memerlukan instalasi aplikasi.

Penelitian ketiga dilakukan oleh Manik et al. (2021) dengan judul “Penerapan *Internet of Things* (IoT) Pada Sistem Monitoring Pemakaian Daya Listrik Rumah Tangga Berbasis Mikrokontroler dan Website”, menghasilkan sebuah sistem monitoring konsumsi listrik secara real-time dengan akurasi sensor yang cukup baik, pengiriman data yang stabil dan mendukung dalam pemantauan perangkat yang boros. Namun sistem ini masih memiliki kekurangan, yaitu tidak tersedianya sistem rekomendasi, sistem hanya berfokus pada monitoring, serta

sistem juga merupakan website biasa dan tidak mendukung PWA sehingga tidak mendukung akses *offline*.

Table 2.1 Tinjauan Aplikasi Sejenis

Pengembang/ Tahun	Nama Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
Zaen et al. (2021)	Sistem Monitoring Pemakaian Energi Listrik Rumah Tangga Berbasis web	Monitoring konsumsi energi listrik real-time, monitoring tegangan, arus, data, frekuensi energi, informasi tagihan listrik	NodeMCU ESP8266, Sensor PZEM-004T, PHP, MySQL, Web	Monitoring real-time, akurasi sensor tinggi, dan menyediakan informasi tagihan listrik.	Belum memiliki fitur rekomendasi
Satya et al (2024)	Sistem Perhitungan Pemakaian Listrik Rumah Tangga Berbasis Android	Monitoring penggunaan listrik, perhitungan biaya, laporan konsumsi energi, rekomendasi efisiensi energi, notifikasi penggunaan listrik.	Android	Mudah digunakan dan menyediakan fitur rekomendasi.	Hanya tersedia di platform Android
Manik et al. (2021)	Sistem Monitoring Pemakaian Daya Listrik Rumah Tangga Berbasis Mikrokontroler dan Website	Monitoring penggunaan daya listrik real-time, dashboard monitoring, analisis penggunaan daya, estimasi tarif listrik, cetak tagihan.	Arduino Uno R3, Ethernet Shield, W5100, Sensor SCT-013, PHP, MySQLm Website, IoT	Monitoring real-time dan dapat diakses melalui website	Belum memiliki fitur rekomendasi
Proyek ini	Voltaris	Rekomendasi penggunaan listrik, push notifikasi dan dashboard pemantauan.	Quasar, VueJs, dan NestJs	Mendukung sistem rekomendasi pemakaian listrik, dan menggunakan PWA sehingga dapat diakses secara <i>offline</i> .	-

2.2. Landasan Konsep dan Teknologi

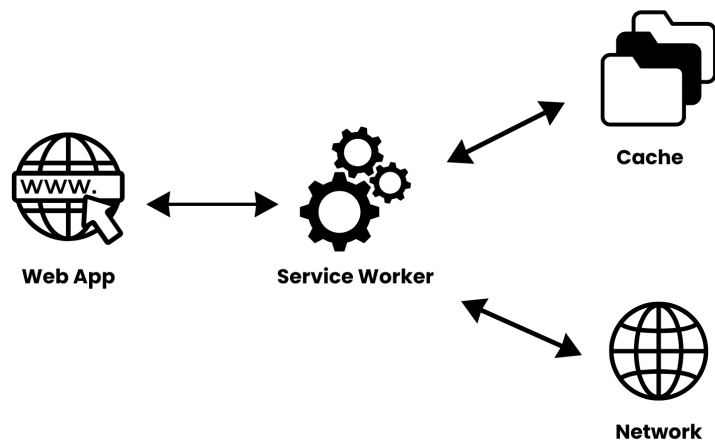
Aplikasi Voltari merupakan aplikasi yang dikembangkan untuk membantu pengguna dalam memantau konsumsi listrik rumah tangga dan memberikan rekomendasi optimasi penggunaan, bagian ini adalah konsep-konsep dasar yang mendukung pengembangan proyek “Rancang Bangun Sistem Monitoring Konsumsi Energi Listrik Berbasis *Progressive web App* dengan Rekomendasi Optimasi Pengguna”. Teori yang ditulis merupakan teori yang berkaitan langsung proyek yang dikerjakan.

2.2.1. Monitoring Konsumsi Energi Listrik

Monitoring konsumsi energi listrik adalah sistem yang melakukan pengukuran, pencatatan, dan pemantauan penggunaan energi listrik yang berfungsi menghasilkan informasi mengenai konsumsi daya, energi, dan biaya penggunaan listrik (Atmanto et al., 2025). Sistem monitoring merupakan bagian penting yang mendukung pengguna dalam melakukan pemantauan dan pengendalian energi listrik yang digunakan.

2.2.2. *Progressive Web App* (PWA)

Progressive Web App (PWA) merupakan aplikasi berbasis website yang memanfaatkan *Service Worker* dan *Web App Manifest*, sehingga PWA dapat mendukung instalasi, diakses secara *offline* dan memberikan pengalaman seperti menggunakan aplikasi *native* (Amrullah et al., 2021). Dalam penerapannya pada aplikasi Voltaris, PWA memberikan solusi untuk penggunaan aplikasi dari berbagai platform dan memberikan pengalaman pengguna yang baik, dengan hanya satu *code*.



Gambar 2.1 *Progressive Web App*

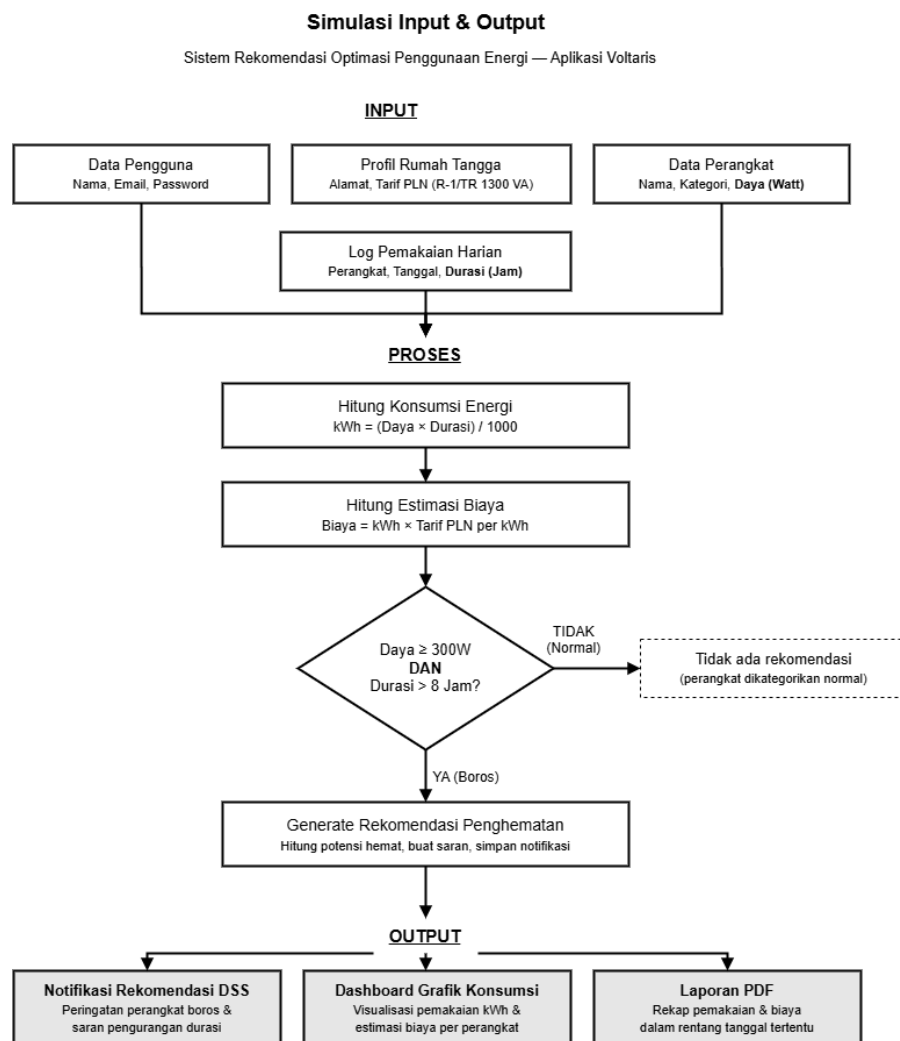
Gambar 2.1 merupakan arsitektur *Service Worker* pada *Progressive Web App* (PWA). *Service worker* berfungsi sebagai perantara antara aplikasi, *cache* dan *network*. Ketika aplikasi meminta data, maka *service worker* akan memeriksa data pada *cache*. Jika data tersedia, maka data akan dikirim ke aplikasi tanpa memerlukan akses *server*. Sedangkan, jika data tidak tersedia, maka *service worker* akan mengambil data melalui *network*, kemudian menyimpan ke dalam *cache* untuk efisiensi pengambilan data berikutnya. Mekanisme ini memungkinkan aplikasi memiliki performa yang lebih cepat dan dapat diakses secara *offline*.

2.2.3. Sistem Monitoring Berbasis Website

Sistem monitoring berbasis website merupakan sistem yang digunakan untuk melakukan pemantauan, pengolahan dan penyajian informasi, memungkinkan pengguna untuk mengakses informasi dari berbagai perangkat dan kapan saja guna pengambilan keputusan yang efisien (Septanto & Hidayatullah, 2022).

2.2.4. Rekomendasi Optimasi Penggunaan Energi

Rekomendasi optimasi penggunaan energi adalah metode pemberian saran yang bertujuan untuk meningkatkan efisiensi penggunaan listrik berbasis data. Rekomendasi yang diberikan merupakan data penggunaan listrik berlebih, pengaturan penggunaan perangkat elektronik dan usaha penghematan energi untuk menurunkan biaya penggunaan listrik (Atmanto et al., 2025).



Contoh Simulasi:

Input: Perangkat = AC Kamar (350 Watt), Durasi = 10 Jam, Tarif PLN = R-1/TR 1300 VA (Rp 1.444,70/kWh)
Proses: $kWh = (350 \times 10) / 1000 = 3,5 \text{ kWh}$ → $Biaya = 3,5 \times 1.444,70 = \text{Rp } 5.056/\text{hari}$ → Kriteria: $350W \geq 300W \checkmark$ DAN $10 \text{ Jam} > 8 \text{ Jam} \checkmark = \text{BOROS}$
Output: Notifikasi → "AC Kamar terdeteksi boros. Kurangi durasi menjadi 6–8 jam untuk menghemat hingga Rp 1.011/hari"

Gambar 2.2 Proses *Input* dan *Output* Rekomendasi Optimasi Energi

2.2.5. NestJs Framework

NestJS merupakan framework *backend* berbasis TypeScript yang dibangun di atas Node.js. NestJS menerapkan konsep modular, *dependency injection*, dan arsitektur yang terinspirasi dari Angular sehingga memudahkan pengembangan aplikasi *server-side* yang terstruktur, *scalable*, dan mudah dipelihara.

2.2.6. Quasar Framework

Quasar merupakan *framework frontend* berbasis [Vue.js](https://vuejs.org/) yang dapat digunakan dalam pembuatan aplikasi website, *mobile*, *desktop*, maupun *Progressive Web App* (PWA) dari satu baris *code*. Quasar memiliki fitur *reactive data binding*, *component-based architecture*, dan *state management* yang mempermudah dalam mengolah data aplikasi quasar juga mendukung mode pengembangan seperti *Single Page Application* (SPA), *Server-Side Rendering* (SSR), *Progressive Web Application* (PWA), aplikasi *mobile* melalui Cordova atau Capacitor serta aplikasi dekstop melalui Electron. Fitur-fitur tersebut membantu pengembang dalam membuat aplikasi yang konsisten, mudah dipelihara dan dapat dijalankan di berbagai platform sekaligus.

2.2.7. Relational Database, MySQL, dan Sequelize ORM

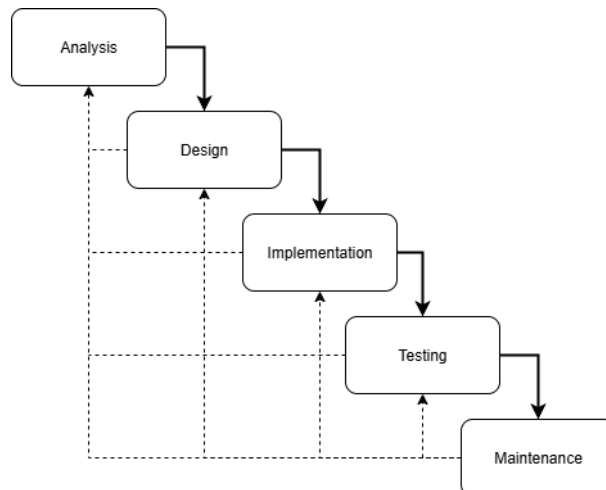
Relational Database merupakan sistem basis data yang menyimpan data dalam bentuk tabel yang saling berhubungan melalui relasi tertentu. MySQL adalah salah satu sistem manajemen basis data relasional (RDBMS) yang banyak digunakan karena memiliki performa tinggi, bersifat *open source*, dan mendukung berbagai platform. *Sequelize* merupakan *Object Relational Mapping* (ORM) untuk Node.js yang digunakan untuk mempermudah interaksi aplikasi dengan *database* relasional menggunakan objek JavaScript atau TypeScript.

2.2.8. Web Push Notification

Web Push Notification merupakan teknologi yang memungkinkan website atau aplikasi web mengirimkan notifikasi kepada pengguna meskipun aplikasi sedang tidak dibuka. Teknologi ini memanfaatkan Push API dan *Service Worker* untuk mengirimkan informasi secara *real-time* sehingga dapat meningkatkan keterlibatan pengguna dan mempercepat penyampaian informasi penting.

2.3. Metode Pengembangan Produk

Metode pengembangan perangkat lunak yang digunakan dalam proyek ini adalah Waterfall. Waterfall merupakan metode pengembangan perangkat lunak yang dilakukan secara berurutan dan sistematis, dimulai dari tahap analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan. Setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahapan berikutnya sehingga proses pengembangan menjadi lebih terstruktur dan mudah dikendalikan (Willyansah et al., 2025).



Gambar 2.3 Metode Waterfall

Tahapan-tahapan dalam metode Waterfall beserta rencana kegiatan yang akan dilakukan adalah sebagai berikut:

1. Analisis Kebutuhan

Tahap analisis kebutuhan merupakan tahap yang dilakukan untuk menganalisis kebutuhan dari sistem. Pada tahap ini akan dilakukan pengumpulan kebutuhan pengguna untuk memenuhi dan menyelesaikan masalah yang dihadapi.

2. Perancangan

Tahap perancangan merupakan tahap dilakukannya perencanaan terhadap kebutuhan fungsional yang telah ditentukan sebelumnya. Pada tahap ini dilakukan perancangan diagram *Use Case*, *Activity* diagram, *Entity Relationship Diagram* (ERD) dan rancangan antarmuka untuk memudahkan dalam implementasi sistem yang akan dilakukan.

3. Implementasi

Tahap Implementasi merupakan tahap dilakukannya implementasi terhadap rancangan-rancangan yang telah dilakukan. Pada tahapan ini peneliti melakukan implementasi aplikasi menjadi *code* dengan membangun *Frontend* Quasar *framework* dan *Backend* Nest Js, sehingga aplikasi dapat berjalan sesuai dengan analisis dan perancangan yang telah dilakukan.

4. Pengujian

Tahap pengujian merupakan tahap dilakukannya pengujian untuk memastikan bahwa aplikasi yang dibangun telah sesuai dengan kebutuhan pengguna. Pada tahap ini peneliti menggunakan *Blackbox* testing dengan *test case* untuk memastikan bahwa seluruh fungsional telah berjalan sesuai dengan standar yang telah ditentukan.

5. Pemeliharaan

Tahap pemeliharaan merupakan tahap yang dilakukan secara berkelanjutan untuk memastikan bahwa sistem sudah berjalan sesuai dengan kebutuhan pengguna

BAB III ANALISIS DAN PERANCANGAN

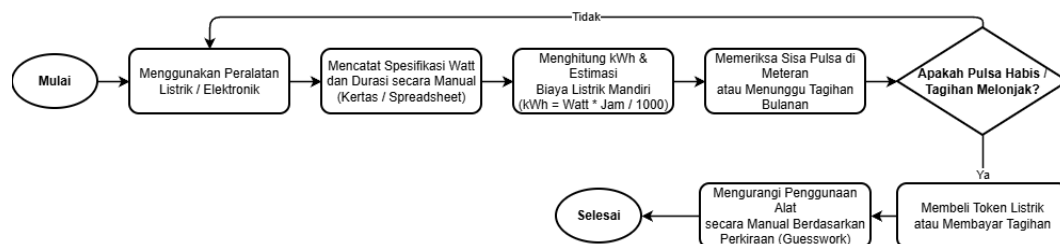
3.1. Analisis Kebutuhan

3.1.1 Proses Bisnis Berjalan

Berdasarkan hasil observasi yang telah dilakukan, pada saat ini proses pemantauan pemakaian listrik masih berjalan secara manual. Pengguna menggunakan listrik atau peralatan elektronik, kemudian pengguna melakukan pencatatan data yang mencakup durasi pemakaian secara manual, kemudian pengguna akan menghitung dengan menggunakan biaya listrik per watt secara manual, jika tagihan melonjak pengguna akan membayar tagihan dan mengurangi penggunaan daya listrik. Semua proses ini dilakukan dengan pencatatan dan perhitungan secara manual.

Hasil observasi yang telah dilakukan di dukung dengan data hasil kuesioner terhadap 6 responden mengenai penggunaan listrik rumah tangga. Sebanyak 83,3% responden mengalami kesulitan menghitung estimasi biaya listrik per perangkat elektronik secara mandiri, kemudian 83,3% responden mengaku sering terkejut dengan tagihan listrik perbulan. Sebanyak 83,3% responden menyatakan bahwa perangkat monitoring penggunaan listrik merupakan kebutuhan yang penting.

Berdasarkan hasil observasi dan kuesioner, disimpulkan bahwa proses monitoring dan pengelolaan penggunaan listrik rumah tangga masih dilakukan secara manual, belum ada sistem yang memberikan informasi penggunaan listrik secara terintegrasi. Maka diperlukannya sebuah sistem yang mampu memantau, menghitung dan memberikan rekomendasi penghematan konsumsi energi.

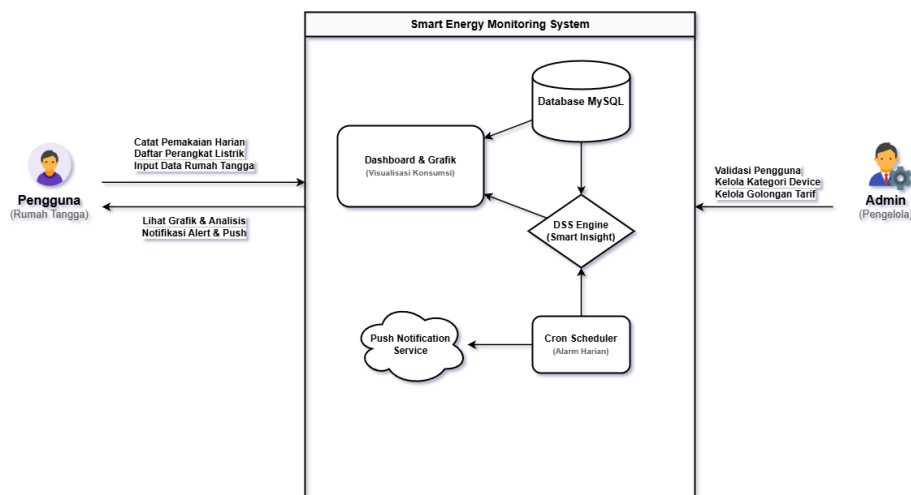


Gambar 3.1 Proses Bisnis Berjalan

3.1.2 Gambaran Umum Sistem

Voltaris merupakan sistem yang dikembangkan untuk membantu pengguna dalam melakukan monitoring penggunaan listrik rumahan dan memberikan rekomendasi penggunaan energi listrik. Sistem dirancang untuk menjawab permasalahan efisiensi konsumsi energi listrik. Pada Voltaris terdapat dua jenis aktor, yang pertama adalah pengguna dan yang kedua adalah admin.

Pengguna dapat melakukan pencatatan pemakaian harian listrik dan melihat grafik, mendapatkan notifikasi dan mendapatkan rekomendasi penggunaan listrik. Sementara itu, admin dapat melakukan validasi pengguna, mengelola kategori dan mengelola golongan tarif. Gambaran umum sistem Voltaris dapat dilihat pada Gambar 3.2 di bawah ini.



Gambar 3.2 Gambaran Umum Sistem

3.2. Perancangan

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional yang terdapat dalam Voltaris terdiri atas 10 kebutuhan fungsional yang dapat dilihat pada Tabel 3.1 berikut.

Tabel 3.1 Kebutuhan Fungsional

User/Rumah Tangga	FR-01	Registrasi Pengguna : Pengguna mendaftar dengan mengisi formulir registrasi. Sistem memvalidasi kelengkapan data di <i>frontend</i> , dan memverifikasi keunikan <i>username/email</i> di <i>backend</i> sebelum
-------------------	-------	---

		melakukan enkripsi (<i>hashing</i>) <i>password</i> ke database.
User/Rumah Tangga & Admin	FR-02	Login : Proses autentikasi menggunakan <i>username/email</i> dan <i>password</i> . Setelah sukses divalidasi oleh <i>backend</i> , sistem akan menyimpan token autentikasi (JWT) dan mengarahkan aktor ke dashboard sesuai role mereka.
	FR-03	Mengelola Profil & Informasi Rumah : Mengubah informasi profil/ <i>password</i> pengguna, serta menambah, mengedit, atau menghapus informasi (seperti rumah utama, kos, dll.).
Admin	FR-04	Mengelola Golongan Tarif : Fitur kelola tarif dasar listrik per kWh berdasarkan kapasitas daya listrik PLN (VA) untuk perhitungan biaya pemakaian.
	FR-05	Mengelola Kategori Perangkat : Mengelompokkan alat elektronik (misal: "Pendingin", "Pencahaya") agar manajemen perangkat lebih teratur.
User/Rumah Tangga	FR-06	Mengelola Perangkat Listrik : Mendaftarkan peralatan elektronik beserta dayanya (<i>watt</i>) ke dalam suatu rumah tangga, serta mengontrol statusnya (aktif/nonaktif secara virtual).
	FR-07	Mencatat Pemakaian Perangkat Harian : Menginput durasi operasional perangkat harian. Sistem secara otomatis menghitung konsumsi energi dalam kWh serta estimasi rupiah berdasarkan tarif yang berlaku.
	FR-08	Melihat Dashboard Analisis & Download Laporan : Visualisasi grafik pemakaian energi harian/bulanan beserta fitur ekspor laporan berkala ke format PDF.
	FR-09	Melihat DSS Smart Insight : Sistem Pendukung Keputusan menggunakan algoritma SAW (<i>Simple Additive Weighting</i>) untuk merangking prioritas penghematan energi dari perangkat terboros.
	FR-10	Menerima Notifikasi (Web Push) : Pengiriman push notification secara <i>real-time</i> via <i>Service Worker</i> jika akumulasi pemakaian energi harian/mingguan melampaui batas wajar target bulanan (<i>threshold</i>) yang dipicu oleh Cron Job di server.

3.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional yang terdapat dalam Voltaris dapat dilihat pada Tabel 3.2 berikut.

Tabel 3.2 Kebutuhan Non Fungsional

NFR-01	Keamanan (<i>Security</i>)	Sistem harus menjaga kerahasiaan data pengguna dengan menerapkan mekanisme autentikasi dan penyimpanan data yang aman.
NFR-02		Sistem harus menggunakan mekanisme autentikasi untuk membatasi akses ke layanan sistem.
NFR-03	Kegunaan (<i>Usability</i>)	Sistem harus dapat diakses dan digunakan dengan baik pada berbagai ukuran layar, baik desktop maupun perangkat <i>mobile</i> .
NFR-04		Sistem harus mampu memberikan notifikasi kepada pengguna secara <i>real-time</i> terkait informasi penting yang terjadi pada sistem.
NFR-05	Kompabilitas (<i>Compatibility</i>)	Sistem harus dapat berjalan pada browser modern seperti Google Chrome, Mozilla Firefox, dan Microsoft Edge.

3.2.3 Diagram Use Case

Pada Gambar 3.2 ditunjukkan *Use Case Diagram*. Diagram ini menggambarkan interaksi antara *User/Rumah Tangga* dan *System Admin* dengan sistem. User dapat mengelola data rumah tangga, perangkat listrik, pemakaian energi, analisis konsumsi, dan notifikasi. Sedangkan *System Admin* bertugas mengelola kategori perangkat dan golongan tarif. Diagram ini juga menunjukkan hubungan antar use case yang mendukung proses utama dalam sistem.



Gambar 3.3 Use Case Diagram

3.2.4 Skenario Use Case

3.2.4.1 Use Case Scenario Registrasi Pengguna

Skenario *use case* register pada Tabel 3.3 menggambarkan proses pembuatan akun baru oleh pengguna. Proses ini dimulai ketika pengguna mengakses halaman registrasi hingga akun berhasil dibuat dan dapat digunakan untuk *login* ke dalam sistem.

Table 3.3 Use Case Scenario Registrasi Pengguna

Kode Use Case	UC001
Nama Use Case	Registrasi Pengguna

Aktor	<i>User/Rumah Tangga</i>
Deskripsi	<i>User</i> dapat membuat akun baru untuk mengakses sistem
Kondisi Awal	<i>User</i> berada pada halaman registrasi
Kondisi Akhir	<i>User</i> berhasil melakukan registrasi
Alur Kejadian Normal	
Aktor	Sistem
1. <i>User</i> mengakses halaman registrasi 3. Mengisi data registrasi 4. Menekan tombol register	2. Menampilkan halaman formulir registrasi 5. Memvalidasi data yang di input 6. Memeriksa keunikan <i>username</i> dan email 7. Menyimpan data pengguna 8. Menampilkan notifikasi registrasi berhasil 9. Mengarahkan ke halaman login
Alur kejadian Alternatif	
1. Pengguna menekan tombol register namun terdapat data yg belum diisi, maka sistem akan menampilkan pesan kesalahan. 2. Pengguna memasukan <i>password</i> yang tidak sesuai, sistem akan menampilkan pesan kesalahan. 3. Pengguna menggunakan <i>username</i> yang sudah di daftar, maka sistem akan menampilkan pesan bawa username telah digunakan.	

3.2.4.2 Use Case Scenario Login

Skenario *use case login* pada Tabel 3.4 menggambarkan proses autentikasi pengguna dan admin untuk mengakses sistem. Proses ini dimulai ketika pengguna memasukkan kredensial *login* hingga berhasil masuk ke halaman *dashboard* sesuai hak akses yang dimiliki.

Table 3.4 *Use Case Scenario* Login

Kode Use Case	UC002
Nama Use Case	Login
Aktor	<i>User/Rumah Tangga & Admin</i>
Deskripsi	Pengguna melakukan autentikasi untuk masuk kedalam sistem
Kondisi Awal	Pengguna memiliki akun yang telah didaftarkan
Kondisi Akhir	Pengguna berhasil masuk ke dashboard
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman <i>login</i> 3. Memasukan email dan <i>password</i> 4. Menekan tombol <i>login</i>	2. Menampilkan halaman <i>login</i> 5. Memvalidasi data <i>login</i> 6. Memeriksa <i>username</i> dan <i>password</i> 7. Membuat token autentikasi 8. Mengarahkan pengguna ke <i>dashboard</i>
Alur kejadian Alternatif	
1. Pengguna menekan tombol <i>login</i> dan sistem akan menampilkan pesan gagal login apabila email atau <i>password</i> tidak sesuai	

3.2.4.3 *Use Case Scenario* Mengelola Profil & Informasi Rumah

Skenario *use case* mengelola profil dan informasi rumah pada Tabel 3.5 menggambarkan proses pengelolaan data akun dan rumah

tangga oleh pengguna. Proses ini dimulai ketika pengguna mengakses halaman profil hingga perubahan data berhasil disimpan oleh sistem.

Table 3.5 *Use Case Scenario* Mengelola Profil & Informasi Rumah

Kode Use Case	UC003
Nama Use Case	Mengelola Profil & Informasi Rumah
Aktor	<i>User</i> /Rumah Tangga
Deskripsi	<i>User</i> dapat mengelola data profil akun dan informasi tentang kapasitas listrik rumah tangga
Kondisi Awal	<i>User</i> telah <i>login</i> ke dalam sistem
Kondisi Akhir	Data profil atau informasi kapasitas listrik rumah telah berhasil diperbarui
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman profil akun atau halaman informasi rumah 3. Memilih aksi ubah data 4. Mengisi dan memperbarui data 5. Menekan tombol simpan	2. Menampilkan halaman informasi profil dan data informasi rumah 6. Memvalidasi data yang telah di input 7. Menyimpan perubahan data ke <i>database</i> 8. Menampilkan notifikasi berhasil
Alur kejadian Alternatif	
1. <i>User</i> menyimpan data yang belum lengkap dan sistem menampilkan pesan kesalahan 2. <i>User</i> memasukan data dengan format tidak valid dan sistem menampilkan pesan kesalahan	

3.2.4.4 Use Case Scenario Mengelola Golongan Tarif

Skenario *use case* mengelola profil dan informasi rumah pada Tabel 3.6 menggambarkan proses pengelolaan data akun dan rumah tangga oleh pengguna. Proses ini dimulai ketika pengguna mengakses halaman profil hingga perubahan data berhasil disimpan oleh sistem.

Table 3.6 *Use Case Scenario* Mengelola Golongan Tarif

Kode Use Case	UC004
Nama Use Case	Mengelola Golongan Tarif
Aktor	Admin
Deskripsi	Admin dapat menambah, mengubah, dan menghapus data golongan tarif listrik.
Kondisi Awal	Admin telah login ke dalam sistem
Kondisi Akhir	Data golongan tarif berhasil diperbarui
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman <i>System Management</i> .	2. Menampilkan halaman golongan tarif
3. Memilih aksi tambah, ubah data atau menghapus data	
4. Mengisi dan memperbarui data	
5. Menekan tombol simpan	6. Memvalidasi data yang telah di input
	7. Menyimpan perubahan data ke <i>database</i>
	8. Menampilkan notifikasi berhasil
Alur kejadian Alternatif	

1. Admin menyimpan data yang belum lengkap dan sistem menampilkan pesan kesalahan
2. Admin memasukan data dengan format tidak valid dan sistem menampilkan pesan kesalahan

3.2.4.5 Use Case Scenario Mengelola Kategori Perangkat

Skenario *use case* mengelola profil dan informasi rumah pada Tabel 3.7 menggambarkan proses pengelolaan data akun dan rumah tangga oleh pengguna. Proses ini dimulai ketika pengguna mengakses halaman profil hingga perubahan data berhasil disimpan oleh sistem.

Table 3.7 Use Case Scenario Mengelola Kategori Perangkat

Kode Use Case	UC005
Nama Use Case	Mengelola Kategori Perangkat
Aktor	Admin
Deskripsi	Admin dapat menambah, mengubah, dan menghapus data kategori perangkat listrik
Kondisi Awal	Admin telah <i>login</i> ke dalam sistem
Kondisi Akhir	Data kategori perangkat berhasil diperbarui
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman <i>System Management</i> . 3. Memilih aksi tambah, ubah data atau menghapus data 4. Mengisi dan memperbarui data 5. Menekan tombol simpan	2. Menampilkan halaman kategori perangkat 6. Memvalidasi data yang telah di

input
7. Menyimpan perubahan data ke database
8. Menampilkan notifikasi berhasil
Alur kejadian Alternatif
1. Admin menyimpan data yang belum lengkap dan sistem menampilkan pesan kesalahan 2. Admin memasukan data dengan format tidak valid dan sistem menampilkan pesan kesalahan

3.2.4.6 Use Case Scenario Mengelola Perangkat Listrik

Skenario *use case* mengelola perangkat pada Tabel 3.8 menggambarkan proses pengelolaan perangkat listrik oleh pengguna. Proses ini dimulai ketika pengguna mengakses halaman perangkat hingga data perangkat berhasil ditambahkan, diubah, dihapus, atau diperbarui statusnya.

Tabel 3.8 Use Case Scenario Mengelola Perangkat

Kode Use Case	UC006
Nama Use Case	Mengelola Perangkat
Aktor	User/Rumah Tangga
Deskripsi	User dapat menambah, mengubah, menghapus dan mengelola status perangkat listrik
Kondisi Awal	User telah <i>login</i> ke dalam sistem
Kondisi Akhir	Data perangkat berhasil diperbarui
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman perangkat listrik	2. Menampilkan daftar perangkat yang sudah terdaftar
3. Memilih aksi tambah,	

ubah data atau menghapus data 4. Mengisi dan memperbarui data 5. Menekan tombol simpan	6. Memvalidasi data yang telah di input 7. Menyimpan perubahan data ke database 8. Menampilkan notifikasi berhasil
Alur kejadian Alternatif	
1. <i>User</i> menyimpan data yang belum lengkap dan sistem menampilkan pesan kesalahan 2. <i>User</i> memasukan data dengan format tidak valid dan sistem menampilkan pesan kesalahan	

3.2.4.7 Use Case Scenario Mencatat Pemakaian Perangkat Harian

Skenario *use case* mencatat pemakaian harian pada Tabel 3.9 menggambarkan proses pencatatan penggunaan perangkat listrik oleh pengguna. Proses ini dimulai ketika pengguna mengisi data penggunaan perangkat hingga data pemakaian berhasil disimpan oleh sistem.

Table 3.9 *Use Case Scenario* Mencatat Pemakaian Harian

Kode Use Case	UC007
Nama Use Case	Mencatat Pemakaian Harian
Aktor	<i>User</i> /Rumah Tangga
Deskripsi	<i>User</i> dapat mencatat durasi penggunaan perangkat listrik setiap hari
Kondisi Awal	<i>User</i> memiliki daftar perangkat yang sudah terdaftar
Kondisi Akhir	Data Pemakaian berhasil di simpan
Alur Kejadian Normal	

Aktor	Sistem
1. Memilih fitur pencatatan penggunaan perangkat	2. Menampilkan daftar perangkat yang sudah terdaftar
3. Mengisi tanggal dan durasi penggunaan	4. Menghitung estimasi konsumsi listrik
5. Menekan tombol simpan	6. Memvalidasi data yang telah di input
	7. Menyimpan perubahan data ke database
	8. Menampilkan notifikasi berhasil
Alur kejadian Alternatif	
1. <i>User</i> tidak mengisi durasi penggunaan dan sistem menampilkan pesan kesalahan	

3.2.4.8 Use Case Scenario Melihat Dashboard Analisis & Download Laporan

Skenario use case melihat dashboard analisis pada Tabel 3.10 menggambarkan proses pemantauan konsumsi energi listrik oleh pengguna. Proses ini dimulai ketika pengguna mengakses *dashboard* hingga sistem menampilkan informasi analisis dan laporan penggunaan energi.

Table 3.10 Use Case Scenario Melihat Dashboard Analisis

Kode Use Case	UC008
Nama Use Case	Melihat Dashboard Analisis
Aktor	<i>User</i> /Rumah Tangga
Deskripsi	<i>User</i> dapat melihat analisis konsumsi energi dan mengunduh laporan.
Kondisi Awal	<i>User</i> telah <i>login</i> ke dalam sistem

Kondisi Akhir	Informasi analisis atau laporan berhasil ditampilkan
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman <i>dashboard</i> 3. Memilih menu <i>download</i> laporan 5. Memilih rentang waktu dan menekan tombol <i>download</i>	2. Menampilkan data analisis dan grafik konsumsi energi 4. Menampilkan pilihan rentang waktu laporan 6. Membuat laporan PDF 7. Mengunduh laporan dan menampilkan notifikasi berhasil
Alur kejadian Alternatif	
1. <i>User</i> memilih rentang waktu tanpa data dan sistem hanya akan menampilkan data kosong 2. <i>User</i> memilih waktu yang tidak valid dan sistem menampilkan pesan kesalahan	

3.2.4.9 Use Case Scenario Melihat DSS Smart Insights

Skenario *use case* DSS *Smart Insights* pada Tabel 3.11 menggambarkan proses pemberian rekomendasi penghematan energi berdasarkan metode SAW. Proses ini dimulai ketika pengguna mengakses halaman analisis hingga sistem menampilkan hasil rekomendasi dan simulasi penghematan energi.

Table 3.11 *Use Case Scenario* Melihat DSS Smart Insights

Kode Use Case	UC009
Nama Use Case	Melihat DSS Smart Insights
Aktor	<i>User</i> /Rumah Tangga
Deskripsi	<i>User</i> dapat melihat rekomendasi penghematan energi dan melakukan

	simulasi penggunaan perangkat
Kondisi Awal	<i>User</i> memiliki data perangkat dan penggunaan listrik
Kondisi Akhir	Hasil rekomendasi dan simulasi berhasil ditampilkan
Alur Kejadian Normal	
Aktor	Sistem
1. Mengakses halaman DSS <i>Smart Insights</i> 4. Melihat detail perhitungan atau membuka <i>Eco-Simulator</i> 6. Menjalankan simulasi optimasi	2. Melakukan perhitungan metode SAW 3. Menampilkan hasil prioritas perangkat 5. Menampilkan detail matriks dan simulasi 7. Menghitung estimasi penghematan energi 8. Menampilkan hasil simulasi dan notifikasi berhasil
Alur kejadian Alternatif	
1. <i>User</i> belum memiliki data yang cukup untuk dianalisis dan Sistem menampilkan informasi bahwa analisis tidak dapat dilakukan.	

3.2.4.10 Use Case Scenario Menerima Notifikasi & Push Notification

Skenario *use case* menerima notifikasi pada Tabel 3.12 menggambarkan proses pengiriman dan penerimaan notifikasi oleh pengguna. Proses ini dimulai ketika sistem mendeteksi kondisi tertentu hingga notifikasi berhasil diterima dan dapat diakses oleh pengguna.

Table 3.12 *Use Case Scenario* Menerima Notifikasi

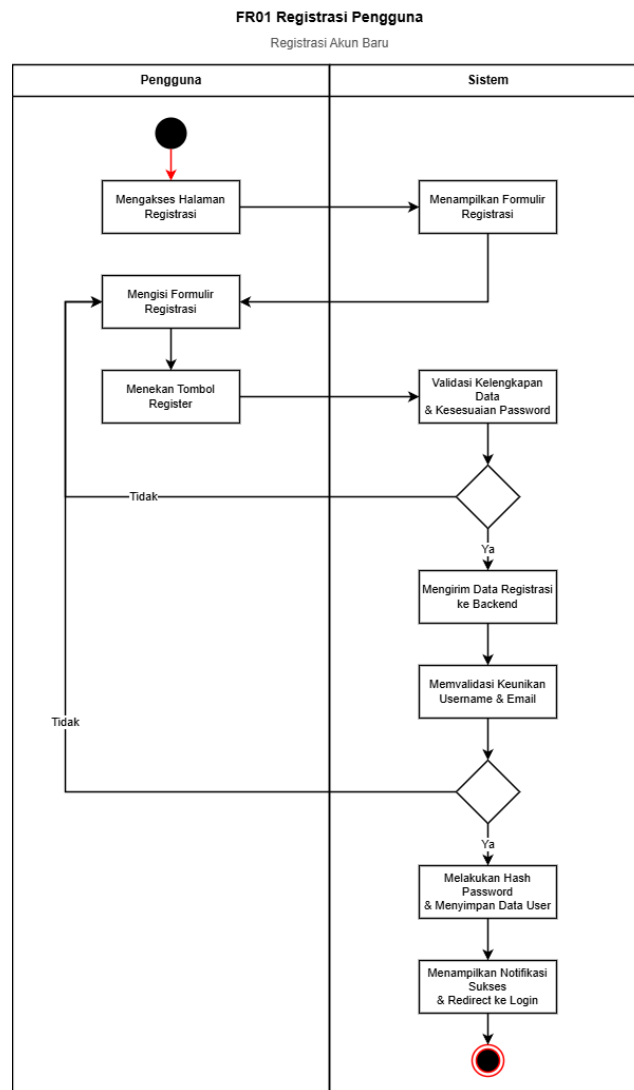
Kode Use Case	UC010
Nama Use Case	Menerima Notifikasi
Aktor	<i>User</i> /Rumah Tangga
Deskripsi	<i>User</i> menerima notifikasi terkait penggunaan energi dan rekomendasi sistem
Kondisi Awal	<i>User</i> telah mengaktifkan layanan notifikasi
Kondisi Akhir	Notifikasi berhasil diterima dan dapat diakses oleh user
Alur Kejadian Normal	
Aktor	Sistem
4. Menerima notifikasi 5. Membuka notifikasi	1. Menjalankan proses pengecekan kondisi notifikasi 2. membuat data notifikasi baru 3. mengirimkan push notification ke perangkat user 6. Mengarahkan user ke halaman yang sesuai
Alur kejadian Alternatif	
1. Tidak terdapat kondisi yang memenuhi kriteria notifikasi 2. Perangkat user belum terdaftar untuk menerima notifikasi	

3.2.5 *Activity diagram*

3.2.5.1 Registrasi Pengguna

Gambar 3.4 menunjukkan *activity diagram* registrasi pengguna pada sistem. Proses dimulai ketika pengguna membuka halaman *registrasi*.

Sistem kemudian menampilkan formulir registrasi yang harus diisi oleh pengguna. Setelah seluruh data diisi, pengguna menekan tombol *register* untuk mengirimkan data ke sistem.



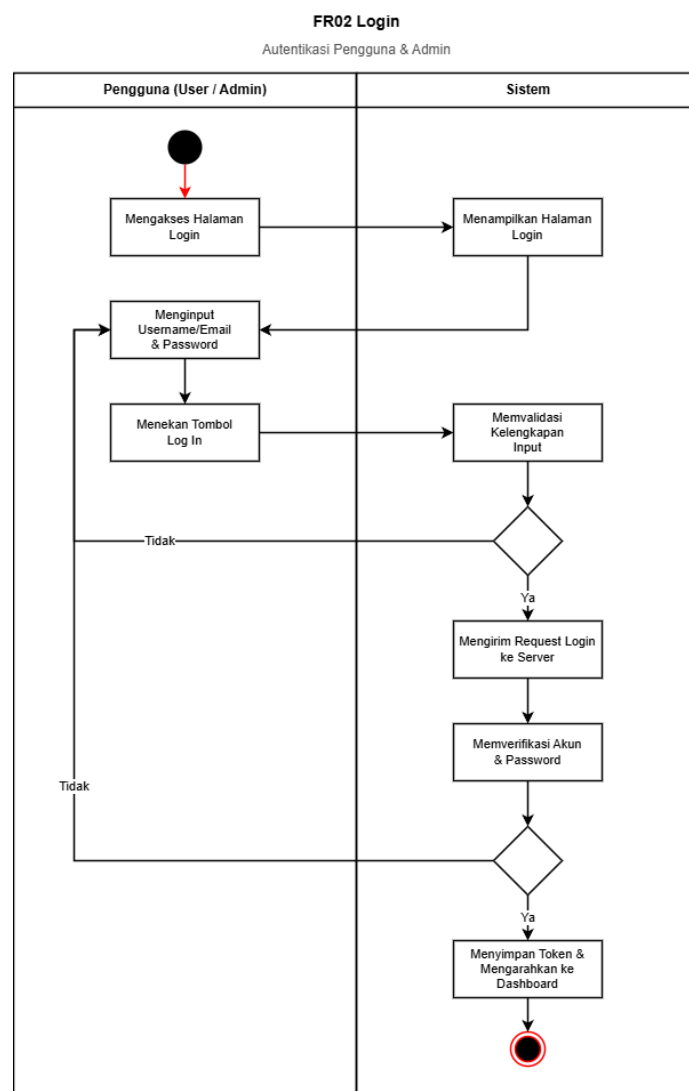
Gambar 3.4 *Activity Diagram* Registrasi Pengguna

3.2.5.2 Login

Pada Gambar 3.5 ditunjukkan *activity diagram* proses *login* pengguna dan admin. Proses dimulai ketika pengguna mengakses halaman *login*, kemudian sistem menampilkan formulir *login*. Pengguna memasukkan email dan *password*, lalu menekan tombol *login*.

Sistem melakukan validasi terhadap kelengkapan data yang dimasukkan. Jika data belum lengkap, pengguna diminta untuk melengkapi kembali data *login*. Selanjutnya sistem memverifikasi akun dan password yang diberikan. Apabila data *login* tidak sesuai, pengguna diminta untuk memasukkan kembali kredensial yang benar.

Jika proses verifikasi berhasil, sistem membuat token autentikasi dan mengarahkan pengguna ke halaman dashboard sesuai hak akses yang dimiliki.

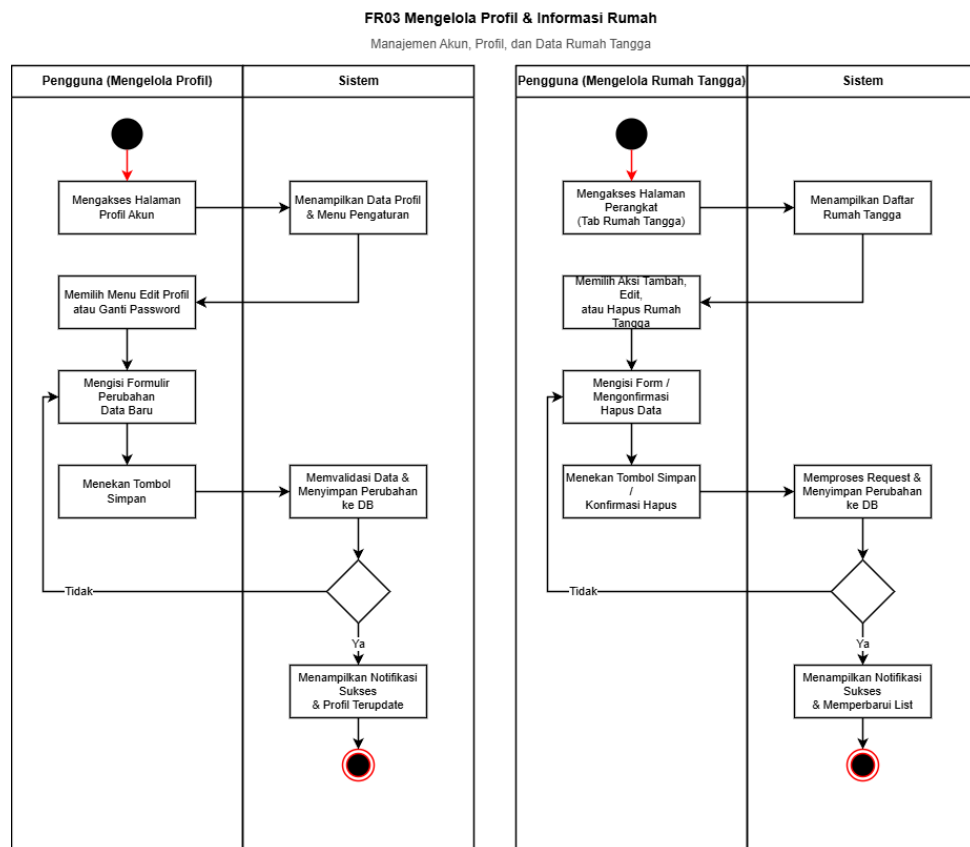


Gambar 3.5 Activity Diagram Login

3.2.5.3 Mengelola Profil & Informasi Rumah

Pada Gambar 3.6. ditunjukkan *activity diagram* pengelolaan profil dan informasi rumah tangga. Pada pengelolaan profil, pengguna dapat mengakses halaman profil untuk mengubah data akun atau mengganti *password*. Sistem kemudian memvalidasi data yang dimasukkan dan menyimpan perubahan apabila data valid.

Selain itu, pengguna juga dapat mengelola data rumah tangga melalui halaman yang tersedia. Pengguna dapat menambahkan, mengubah, atau menghapus data rumah tangga. Sistem akan memproses setiap perubahan yang dilakukan dan memperbarui data yang tersimpan. Setelah proses berhasil, sistem menampilkan notifikasi dan memperbarui informasi yang ditampilkan kepada pengguna.

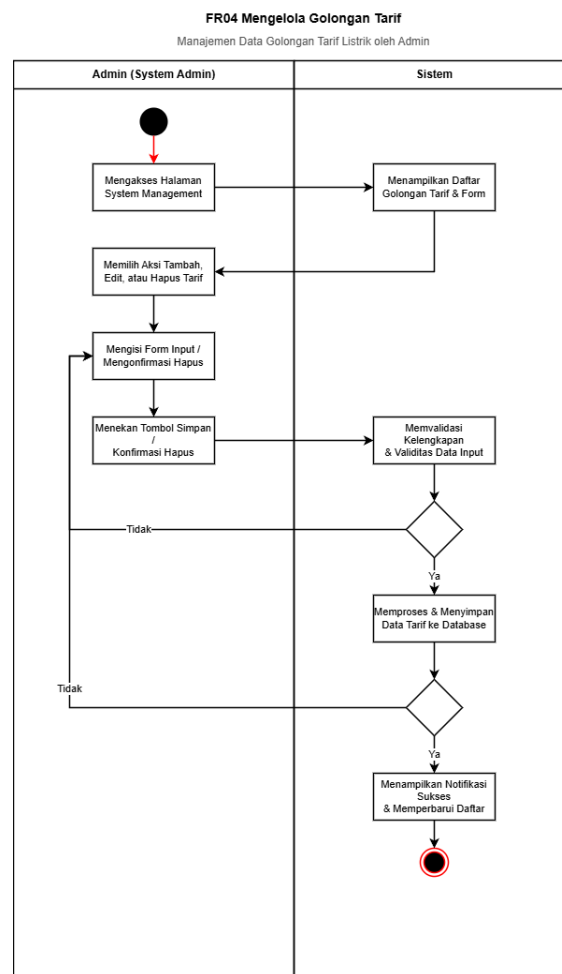


Gambar 3.6 *Activity Diagram* Mengelola Profil & Informasi Rumah

3.2.5.4 Mengelola Golongan Tarif

Pada Gambar 3.7 ditunjukkan *activity diagram* pengelolaan golongan tarif oleh *System Admin*. Proses dimulai ketika admin mengakses halaman *System Management*, kemudian sistem menampilkan daftar golongan tarif yang tersedia. Admin dapat melakukan penambahan, perubahan, atau penghapusan data tarif sesuai kebutuhan.

Setelah data diinputkan atau aksi dipilih, sistem melakukan validasi terhadap data yang diberikan. Jika data valid, sistem akan memproses dan menyimpan perubahan ke dalam basis data. Setelah proses berhasil, sistem menampilkan notifikasi sukses dan memperbarui daftar golongan tarif yang ditampilkan.

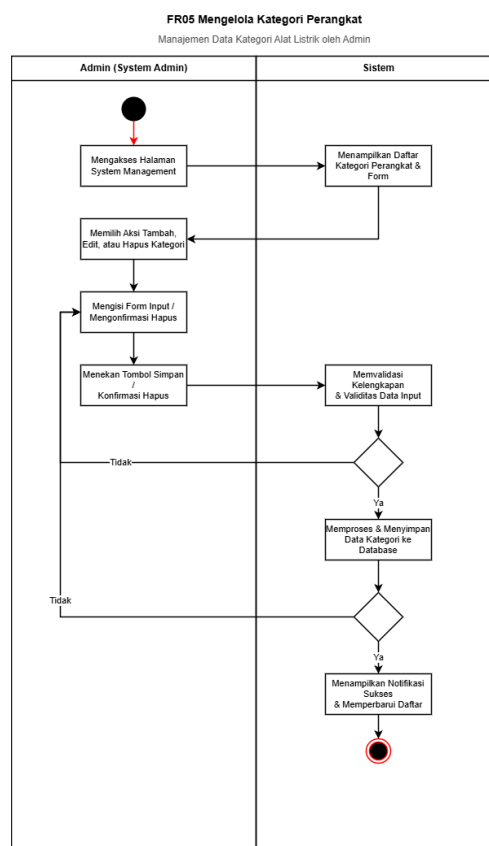


Gambar 3.7 *Activity Diagram* Mengelola Golongan Tarif

3.2.5.5 Mengelola Kategori Perangkat

Pada Gambar 3.8 ditunjukkan *activity diagram* pengelolaan kategori perangkat oleh *System Admin*. Proses dimulai ketika admin mengakses halaman *System Management* dan sistem menampilkan daftar kategori perangkat yang tersedia. Admin dapat melakukan penambahan, perubahan, atau penghapusan kategori perangkat sesuai kebutuhan.

Setelah data diinputkan atau aksi dipilih, sistem melakukan validasi terhadap data yang diberikan. Jika data valid, sistem akan memproses dan menyimpan perubahan ke dalam basis data. Setelah proses berhasil, sistem menampilkan notifikasi sukses dan memperbarui daftar kategori perangkat yang ditampilkan.

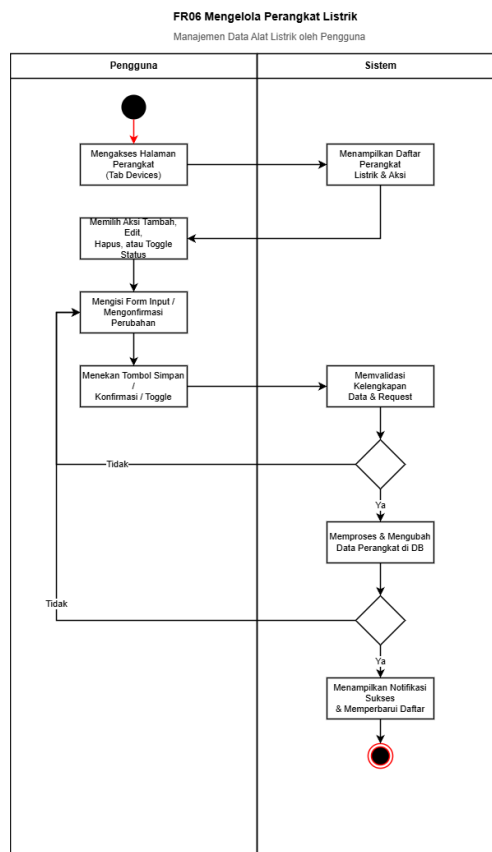


Gambar 3.8 *Activity Diagram* Mengelola Kategori Perangkat

3.2.5.6 Mengelola Perangkat

Pada Gambar 3.9 ditunjukkan *activity diagram* pengelolaan perangkat listrik oleh pengguna. Proses dimulai ketika pengguna mengakses halaman perangkat listrik, kemudian sistem menampilkan daftar perangkat yang telah terdaftar beserta aksi yang dapat dilakukan. Pengguna dapat menambahkan, mengubah, menghapus, atau mengubah status perangkat sesuai kebutuhan.

Setelah pengguna mengisi data atau melakukan aksi tertentu, sistem melakukan validasi terhadap data dan permintaan yang diberikan. Jika valid, sistem akan memproses perubahan dan memperbarui data perangkat pada basis data. Setelah proses berhasil, sistem menampilkan notifikasi sukses dan memperbarui daftar perangkat yang ditampilkan.

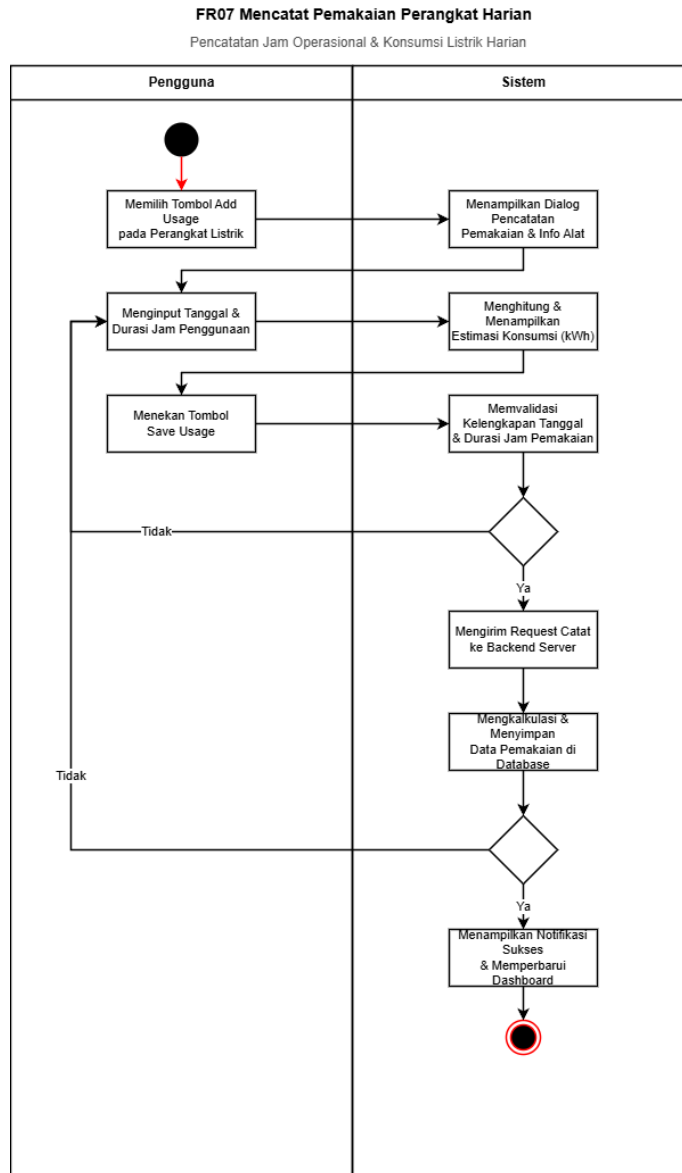


Gambar 3.9 *Activity Diagram* Mengelola Perangkat

3.2.5.7 Mencatat Pemakaian Perangkat Harian

Pada Gambar 3.10 ditunjukkan *activity diagram* pencatatan pemakaian perangkat harian. Proses dimulai ketika pengguna memilih fitur pencatatan penggunaan pada perangkat listrik yang diinginkan. Sistem kemudian menampilkan formulir pencatatan, dan pengguna mengisi tanggal serta durasi penggunaan perangkat.

Selanjutnya sistem menghitung estimasi konsumsi listrik berdasarkan data yang dimasukkan dan melakukan validasi terhadap input pengguna. Jika data valid, sistem akan menyimpan data pemakaian ke dalam basis data. Setelah proses berhasil, sistem menampilkan notifikasi sukses dan memperbarui informasi pada dashboard.

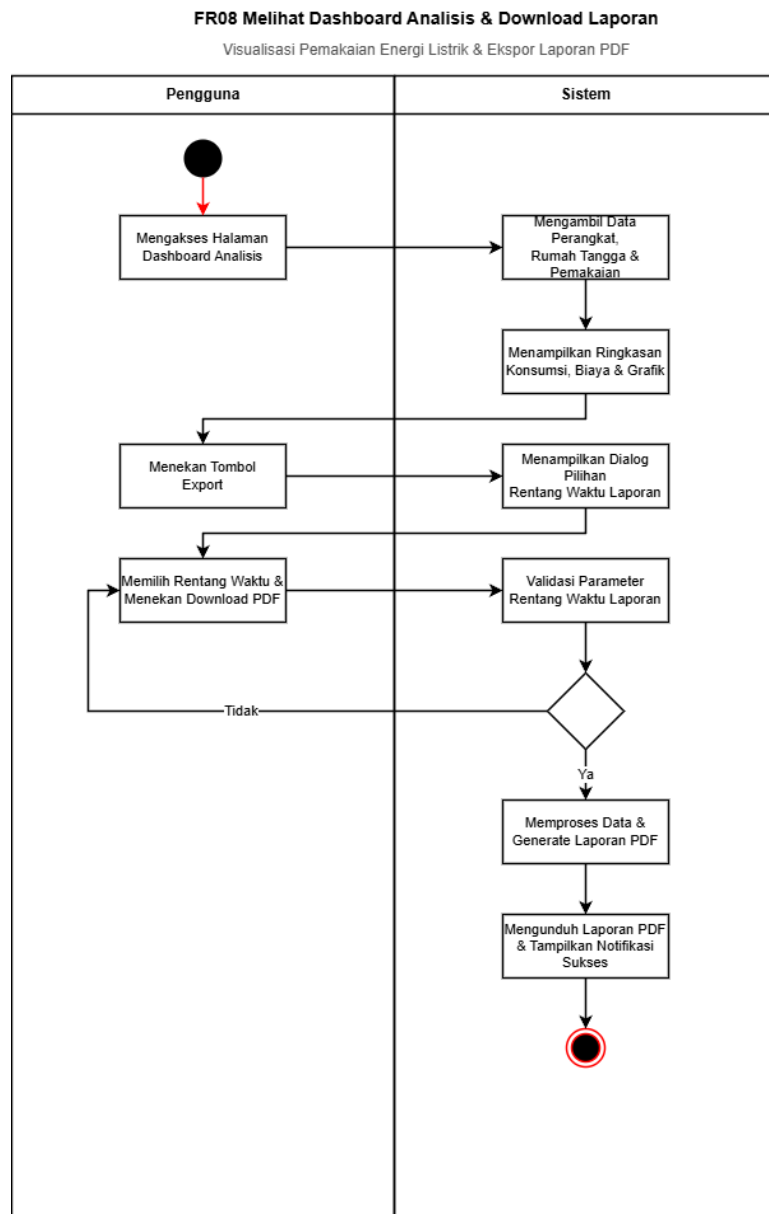


Gambar 3.10 *Activity Diagram* Mencatat Pemakaian Perangkat Harian

3.2.5.8 Melihat *Dashboard Analisis & Download Laporan*

Pada Gambar 3.11 ditunjukkan *activity diagram dashboard* analisis dan ekspor laporan. Proses dimulai ketika pengguna mengakses halaman *dashboard*. Sistem kemudian mengambil dan menampilkan data perangkat, rumah tangga, penggunaan energi, serta ringkasan konsumsi dalam bentuk informasi dan grafik.

Pengguna dapat melakukan ekspor laporan dengan memilih rentang waktu yang diinginkan. Sistem akan memvalidasi parameter laporan, kemudian memproses data dan menghasilkan laporan dalam format PDF. Setelah proses selesai, sistem mengunduh laporan PDF dan menampilkan notifikasi bahwa proses ekspor berhasil.

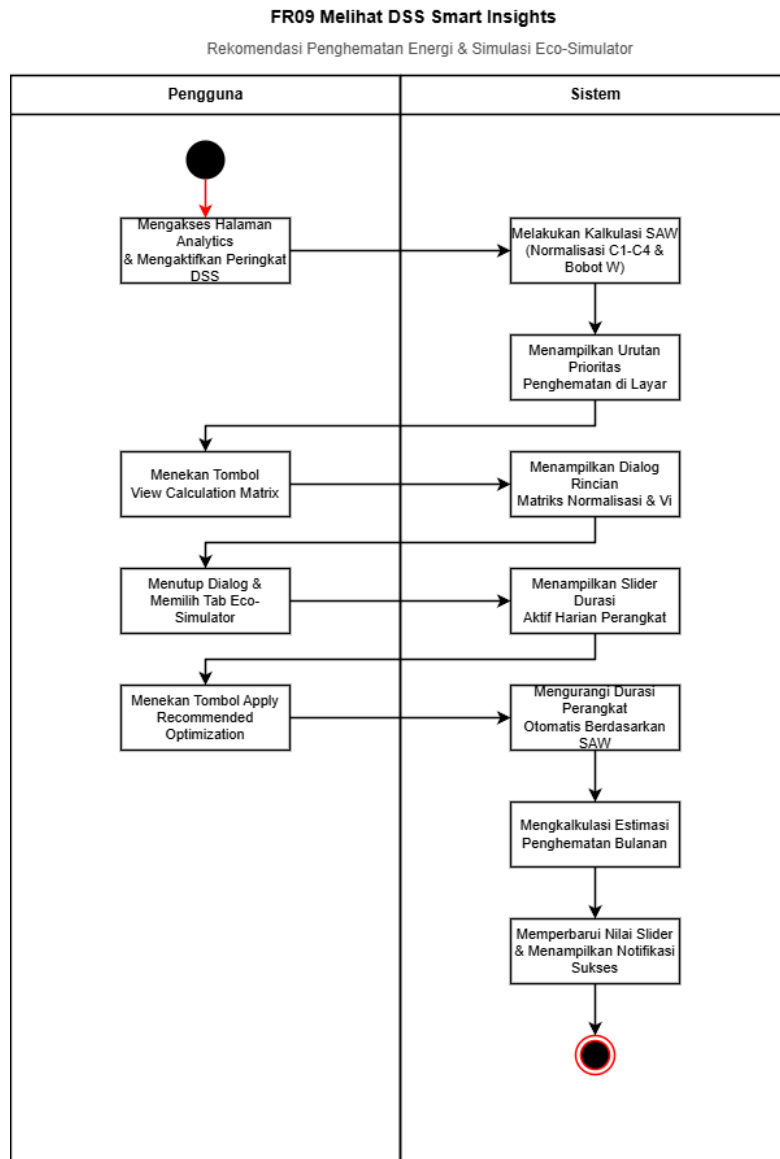


Gambar 3.11 Activity Diagram Melihat Dashboard Analisis & Download Laporan

3.2.5.9 Melihat DSS *Smart Insight*

Pada Gambar 3.12 ditunjukkan *activity diagram* DSS *Smart Insights* yang digunakan untuk memberikan rekomendasi penghematan energi dan simulasi optimasi penggunaan perangkat. Proses dimulai ketika pengguna mengakses halaman analisis dan mengelola perangkat listrik. Sistem kemudian melakukan perhitungan menggunakan metode SAW dan menampilkan urutan prioritas perangkat yang berpotensi dioptimalkan.

Pengguna dapat melihat detail perhitungan yang dilakukan sistem atau menggunakan fitur *Eco-Simulator* untuk melakukan simulasi pengurangan durasi penggunaan perangkat. Selanjutnya sistem menghitung estimasi penghematan energi berdasarkan parameter yang dipilih dan menampilkan hasil simulasi kepada pengguna. Setelah proses selesai, sistem memperbarui data simulasi dan menampilkan notifikasi keberhasilan.



Gambar 3.12 *Activity Diagram* Melihat DSS *Smart Insights*

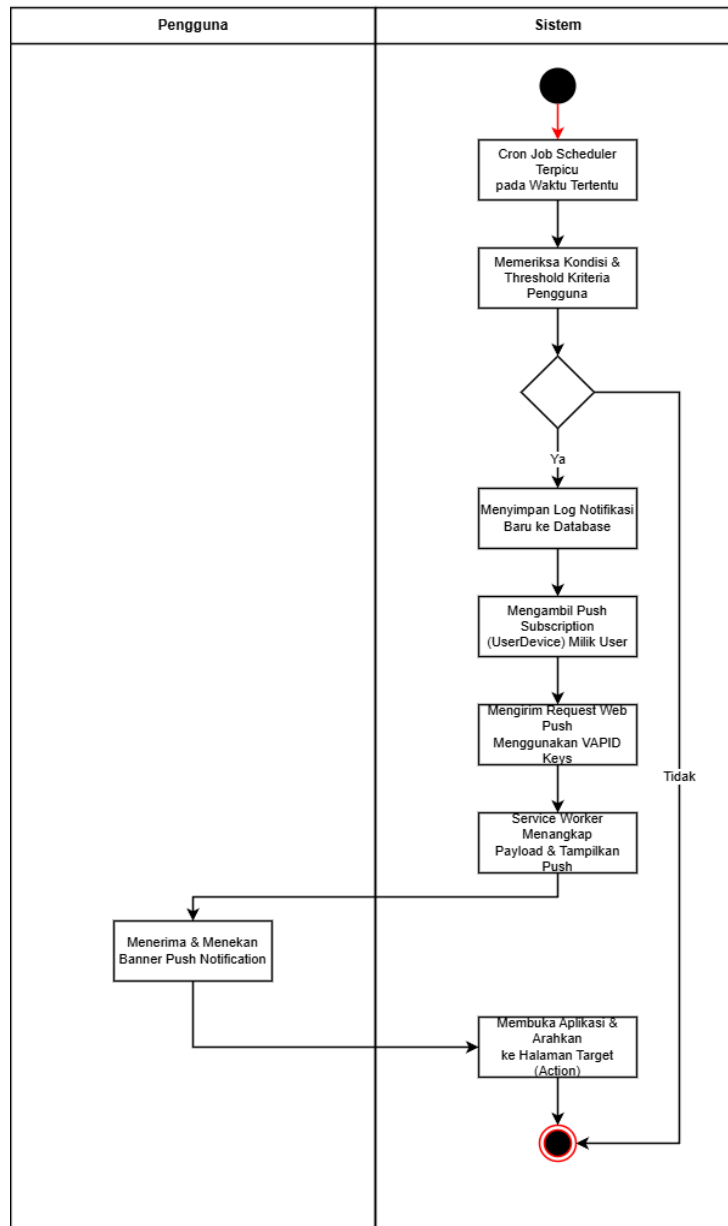
3.2.5.10 Menerima Notifikasi

Pada Gambar 3.13 ditunjukkan *activity diagram* proses notifikasi dan push notification. Proses dimulai ketika sistem menjalankan *scheduler* pada waktu tertentu untuk memeriksa kondisi dan kriteria notifikasi yang telah ditentukan. Jika kondisi terpenuhi, sistem membuat data notifikasi dan mengirimkan push notification ke perangkat pengguna yang telah terdaftar.

Selanjutnya pengguna menerima notifikasi pada perangkat yang digunakan. Apabila notifikasi dipilih, sistem akan mengarahkan pengguna ke halaman yang sesuai dengan informasi atau tindakan yang terkait. Dengan demikian, pengguna dapat memperoleh informasi penting secara langsung tanpa harus membuka aplikasi terlebih dahulu.

FR10 Menerima Notifikasi & Push Notification

Pemicu Cron Job, Pengiriman Web Push & Interaksi Pengguna

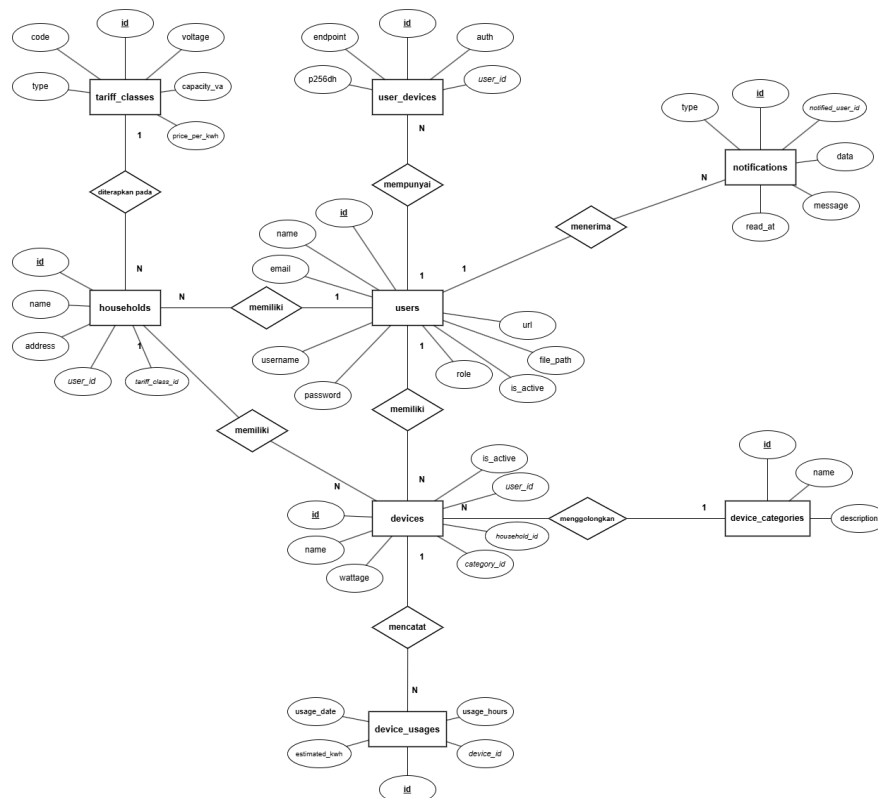


Gambar 3.13 *Activity Diagram* Menerima Notifikasi

3.2.6 ER Diagram

Pada Gambar 3.14 ditunjukkan *Entity Relationship Diagram* (ERD) yang menggambarkan struktur basis data pada sistem monitoring dan analisis konsumsi energi listrik. Entitas utama pada sistem adalah *users* yang berhubungan dengan *households*, *devices*, *notifications*, dan *user_devices*. Setiap pengguna dapat memiliki lebih dari satu rumah tangga dan perangkat listrik yang digunakan untuk mencatat konsumsi energi.

Data perangkat listrik dikelompokkan berdasarkan *device_categories* dan setiap perangkat dapat memiliki banyak data penggunaan yang disimpan pada entitas *device_usages*. Selain itu, sistem juga menyediakan fitur notifikasi yang direpresentasikan oleh entitas *notifications* serta pengelolaan perangkat pengguna melalui entitas *user_devices*. Untuk kebutuhan perhitungan biaya listrik, setiap rumah tangga dikaitkan dengan golongan tarif yang disimpan pada entitas *tariff_classes*.

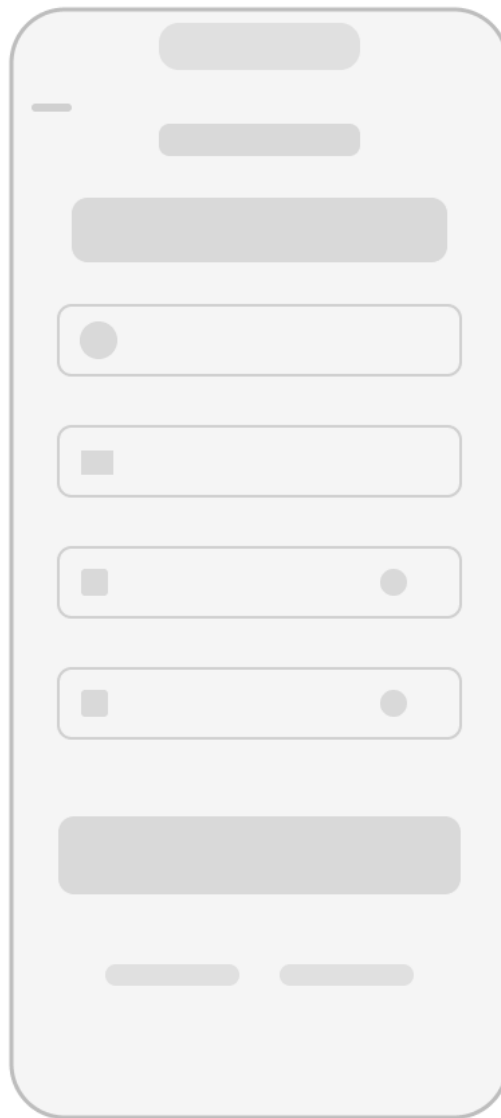


Gambar 3.14 ER Diagram

3.2.7 Perancangan Antarmuka (*Wireframe*)

3.2.7.1 Antarmuka Registrasi Pengguna

Antarmuka dari fungsional registrasi pengguna dapat dilihat pada Gambar 3.15 dibawah ini.



Gambar 3.15 Antarmuka Registrasi Pengguna

3.2.7.2 Antarmuka *Login*

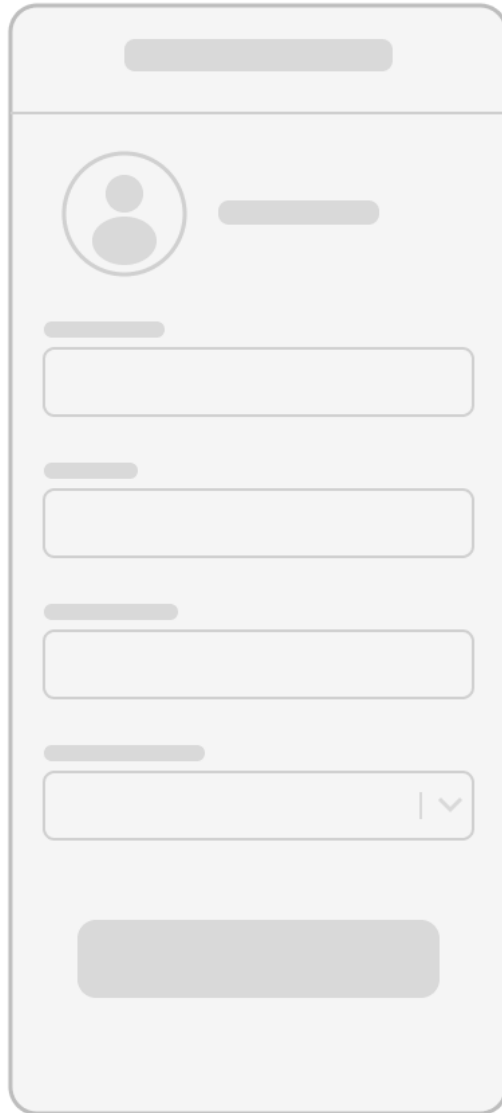
Antarmuka dari fungsional *login* dapat dilihat pada Gambar 3.16 dibawah ini.



Gambar 3.16 Antarmuka *Login*

3.2.7.3 Antarmuka Mengelola Profil & Informasi Rumah

Antarmuka dari fungsional mengelola profil dan informasi rumahan dapat dilihat pada Gambar 3.17 dibawah ini.



Gambar 3.16 Antarmuka Mengelola Mengelola Profil & Informasi Rumah

3.2.7.4 Antarmuka Mengelola Tarif

Antarmuka dari fungsional mengelola tarif dapat dilihat pada Gambar 3.17 dibawah ini.



Gambar 3.17 Antarmuka Mengelola Tarif

3.2.7.5 Antarmuka Mengelola Kategori Perangkat

Antarmuka dari fungsional mengelola kategori perangkat dapat dilihat pada Gambar 3.18 dibawah ini.



Gambar 3.18 Antarmuka Mengelola Kategori Perangkat

3.2.7.6 Antarmuka Mengelola Perangkat Listrik

Antarmuka dari fungsional mengelola mengelola perangkat listrik dapat dilihat pada Gambar 3.19 dibawah ini.



Gambar 3.19 Antarmuka Mengelola Perangkat Listrik

3.2.7.7 Antarmuka Mencatat Pemakaian Perangkat Harian

Antarmuka dari fungsional mencatat pemakaian perangkat listrik dapat dilihat pada Gambar 3.20 dibawah ini.



Gambar 3.20 Antarmuka Mencatat Pemakaian Perangkat Harian

3.2.7.8 Antarmuka Melihat *Dashboard Analisis & Download Laporan*

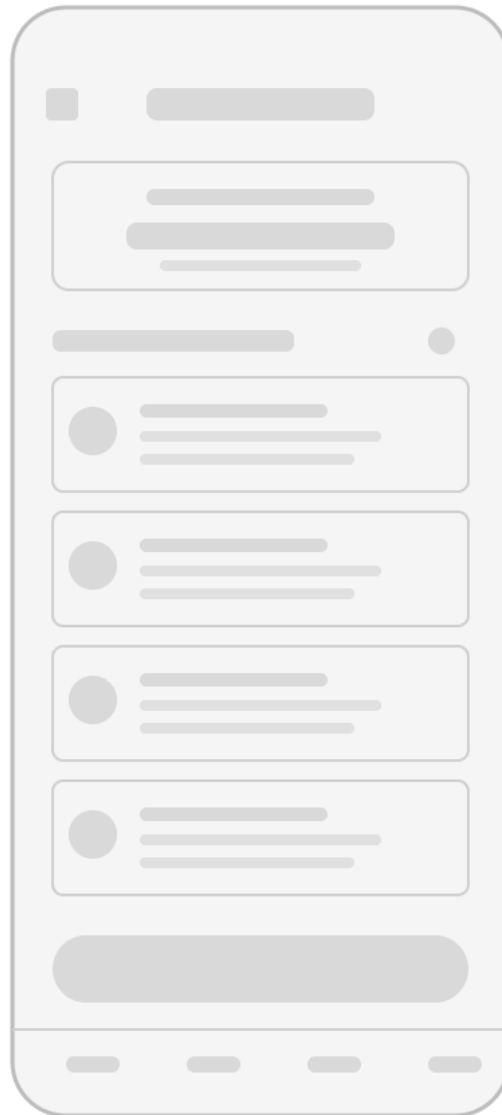
Antarmuka dari fungsional melihat *dashboard* dan *download* laporan dapat dilihat pada Gambar 3.21 dibawah ini.



Gambar 3.21 Antarmuka Melihat *Dashboard Analisis & Download*
Laporan

3.2.7.9 Antarmuka Melihat DSS *Smart Insight*

Antarmuka dari fungsional melihat DSS *Smart Insight* dapat dilihat pada Gambar 3.22 dibawah ini.



Gambar 3.22 Antarmuka Melihat DSS *Smart Insight*

3.2.7.10 Antarmuka Menerima Notifikasi

Antarmuka dari fungsional menerima notifikasi dapat dilihat pada Gambar 3.23 dibawah ini.



Gambar 3.23 Antarmuka Menerima Notifikasi

3.2.8 Perancangan Metode Simple Additive Whiting (SAW)

Metode *Simple Additive Weighting* (SAW), yang sering dikenal sebagai metode penjumlahan terbobot, merupakan salah satu algoritma pengambilan keputusan multi-kriteria (Multi-Criteria Decision Making / MCDM). Konseptual

dasar dari metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif di semua kriteria. Dalam pengembangan aplikasi Voltaris, metode SAW diterapkan sebagai landasan ilmiah Sistem Pendukung Keputusan (Decision Support System / DSS) untuk menentukan skala prioritas tindakan penghematan energi listrik pada peralatan elektronik rumah tangga.

Dalam penelitian ini, implementasi metode SAW dilakukan melalui serangkaian tahapan yang bertujuan untuk menghasilkan nilai preferensi sebagai dasar penentuan prioritas rekomendasi penghematan energi. Tahapan tersebut meliputi penentuan kriteria (C1–C4), penentuan bobot kriteria, perhitungan normalisasi matriks keputusan, serta perhitungan nilai preferensi dan perangkingan alternatif. Uraian dari masing-masing tahapan disajikan pada subbab berikut.

3.2.8.1 Penentuan Kriteria C1-C4

Untuk menentukan prioritas rekomendasi penghematan energi listrik, ditetapkan empat kriteria yang digunakan sebagai dasar penilaian dalam metode SAW. Kriteria-kriteria tersebut disajikan pada Tabel 3.13.

Table 3.13 Penentuan Kriteria

Kode	Nama Kriteria	Tipe Kriteria	Satuan	Sumber Data	Alasan Penggunaan
C1	Konsumsi Energi Bulanan	Benefit	kWh	Log pemakaian bulanan (device_usages)	Konsumsi energi berbanding lurus dengan tagihan biaya. kWh terbesar memberi ruang hemat finansial terbesar.
C2	Daya Listrik Perangkat	Benefit	Watt	Profil teknis alat (devices.wattage)	Perangkat berdaya besar berkontribusi langsung pada beban puncak

					daya rumah tangga.
C3	Durasi Aktif Harian	Benefit	Jam	Kalkulasi (kWh * 1000) / Watt	Durasi aktif adalah variabel perilaku pengguna yang paling realistis untuk dikendalikan/dikurangi.
C4	Tingkat Urgensi Alat	Cost	Skala 1-5	Klasifikasi nama/kategori perangkat	Melindungi kenyamanan dan utilitas kritis (misal kulkas) agar tidak dikurangi secara ekstrim.

**Catatan skala urgensi kriteria C4: Skala 5 (Sangat Tinggi / Kulkas), Skala 4 (Tinggi / Pompa Air, Heater), Skala 3 (Sedang / AC, Kipas), Skala 2 (Rendah / TV, PC, Laptop), Skala 1 (Sangat Rendah / Lampu, Charger, dll).*

3.2.8.2 Penentuan Bobot Kriteria

Pada penelitian ini, bobot ditentukan berdasarkan tingkat kontribusi setiap kriteria terhadap potensi penghematan energi listrik pada peralatan elektronik rumah tangga. Vektor bobot yang digunakan adalah:

$$W = [w_1, w_2, w_3, w_4] = [0.35, 0.25, 0.20, 0.20]$$

Jumlah total bobot adalah **0.35 + 0.25 + 0.20 + 0.20 = 1.00 (100%)**. Adapun alasan pemberian bobot pada masing-masing kriteria dijelaskan sebagai berikut:

a. Bobot C1 (kWh) sebesar 35% (0.35)

Energi riil dalam kWh adalah penentu utama nominal rupiah tagihan listrik bulanan. Penghematan pada perangkat berkonsumsi kWh terbesar akan memberikan dampak finansial yang paling signifikan.

b. Bobot C2 (Watt) sebesar 25% (0.25)

Daya listrik menentukan beban puncak rumah tangga. Mengendalikan perangkat berdaya besar membantu mengurangi risiko turunnya pembatas daya (MCB) dan laju pemakaian energi harian.

c. Bobot C3 (Durasi Jam) sebesar 20% (0.20)

Waktu pemakaian adalah variabel perilaku pengguna yang paling dinamis. Intervensi perilaku untuk mematikan perangkat yang tidak perlu secara langsung mengurangi jam aktif pemakaian.

d. Bobot C4 (Urgensi Alat) sebesar 20% (0.20)

Berfungsi sebagai penyeimbang aspek kenyamanan dan utilitas dasar rumah tangga, memastikan program efisiensi energi tidak mengganggu kelangsungan fungsional rumah tangga.

3.2.8.3 Perhitungan Normalisasi

Setelah nilai kriteria dan bobot ditentukan, langkah selanjutnya dalam metode SAW adalah melakukan normalisasi matriks keputusan. Proses normalisasi bertujuan untuk menyamakan skala nilai antar kriteria yang memiliki satuan berbeda sehingga dapat dibandingkan dalam proses perhitungan. Hasil normalisasi dinyatakan dalam matriks R dengan rentang nilai antara 0 sampai 1.

Pada metode SAW, proses normalisasi dilakukan menggunakan rumus yang berbeda berdasarkan tipe kriteria. Untuk kriteria bertipe benefit digunakan rumus:

$$\text{Criteria Benefit (C1, C2, C3)} : r_{ij} = x_{ij} / \max(x_{ij})$$

sedangkan untuk kriteria bertipe cost digunakan rumus:

$$\text{Criteria Cost (C4)} : r_{ij} = \min(x_{ij}) / x_{ij}$$

Keterangan: x_{ij} adalah nilai asli alternatif ke-i pada kriteria ke-j, $\max(x_{ij})$ adalah nilai maksimum pada kriteria ke-j dari seluruh alternatif, dan $\min(x_{ij})$ adalah nilai minimum pada kriteria ke-j dari seluruh alternatif.

3.2.8.4 Perhitungan Nilai Preferensi dan Ranking

Setelah proses normalisasi selesai dilakukan, langkah berikutnya dalam metode SAW adalah menghitung nilai preferensi (V_i) untuk setiap alternatif ke-i diperoleh dari hasil penjumlahan perkalian antara nilai normalisasi setiap kriteria (r_{ij}) dengan bobot kriteria yang bersesuaian (w_j), yang dirumuskan sebagai berikut:

$$V_i = (0.35 * r_{i1}) + (0.25 * r_{i2}) + (0.20 * r_{i3}) + (0.20 * r_{i4})$$

Setelah nilai preferensi seluruh alternatif diperoleh, dilakukan proses perangkingan dengan mengurutkan nilai preferensi dari yang terbesar hingga terkecil. Alternatif dengan nilai preferensi tertinggi menunjukkan perangkat yang memiliki prioritas terbesar berdasarkan kombinasi seluruh kriteria yang digunakan dalam metode SAW. Hasil perangkingan tersebut kemudian digunakan sebagai dasar dalam pemberian rekomendasi penghematan energi listrik.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

4.1.1 Instalasi Aplikasi

Voltaris merupakan sistem monitoring konsumsi energi listrik rumah tangga berbasis web yang dilengkapi dengan fitur rekomendasi optimasi pengguna melalui mekanisme *Decision Support System* (DSS). Sistem dirancang untuk membantu pengguna dalam mencatat peralatan elektronik, memantau pemakaian daya listrik harian, menghitung estimasi biaya berdasarkan tarif PLN, menerima rekomendasi penghematan energi, serta mengeksport laporan pemakaian dalam format PDF secara terintegrasi. Dalam implementasinya, Voltaris dikembangkan melalui dua komponen utama, yaitu Frontend yang berfungsi sebagai antarmuka pengguna dan Backend yang berfungsi sebagai pengelolaan logika bisnis, pengolahan data, dan penyedia layanan API.

Frontend dibangun menggunakan *framework* Quasar berbasis Vue.js 3 dengan TypeScript, yang mendukung tampilan responsif, interaktif, serta dapat diinstal sebagai *Progressive Web App* (PWA). Visualisasi data konsumsi energi ditampilkan dalam bentuk grafik menggunakan library Chart.js.

Sisi *backend* menggunakan *framework* NestJS dengan TypeScript dan *runtime* Bun, terhubung ke *database* MySQL melalui *Sequelize* sebagai ORM untuk pengelolaan data relasional. Penyimpanan file statis memanfaatkan layanan AWS S3.

Selain itu, sistem dilengkapi dengan fitur ekspor laporan dalam format PDF menggunakan Puppeteer yang merender template HTML melalui *Headless Chromium*, serta fitur notifikasi rekomendasi hemat energi yang dikirimkan ke pengguna melalui *library* Web Push.

Proses implementasi sistem dilakukan melalui beberapa tahapan, yaitu mempersiapkan lingkungan pengembangan, melakukan konfigurasi basis data, instalasi dependensi aplikasi, migrasi database, dan menjalankan layanan *frontend* serta *backend*. Berikut merupakan alur dalam melakukan implementasi sistem:

1. Melakukan kloning (*clone*) repository proyek.
2. Melakukan konfigurasi file environment (.env dan .env.development) untuk koneksi database MySQL, layanan penyimpanan file AWS S3, konfigurasi VAPID Key untuk Web Push, dan Secret Key untuk JWT.
3. Membuat database MySQL bernama tulis_nama_db yang akan digunakan oleh sistem.
4. Menginstal seluruh dependensi Backend menggunakan Bun dan dependensi Frontend menggunakan npm.
5. Menjalankan migrasi database dan seeder data awal (golongan tarif listrik PLN).
6. Menjalankan migrasi database.
7. Menjalankan server *Backend*.
8. Melakukan pengujian akses aplikasi melalui browser untuk memastikan seluruh fitur telah berjalan dengan baik.

Tabel 4.1 Dependensi dan *Tools* yang Digunakan

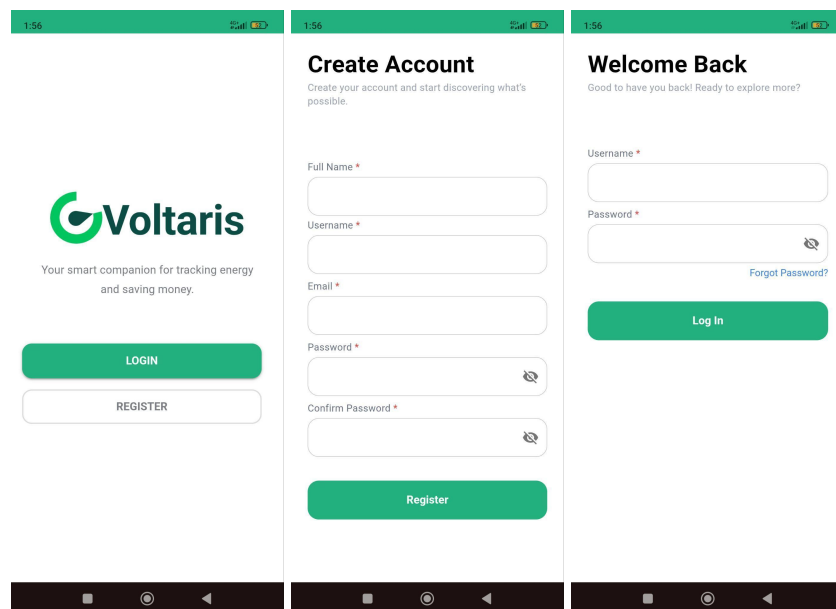
No	Software / Depedensi	Versi	Fungsi
1	Node.js	v20+	Menjalankan <i>package</i> manager npm dan proses build aplikasi
2	Bun	v1.x	<i>Runtime backend</i> untuk menjalankan server NestJS
3	MySQL	v8.0	Sistem manajemen basis data relasional
4	npm	$\geq 6.13.4$	Mengelola <i>package</i> dan <i>dependency</i> frontend
5	Quasar CLI	v2.x	Menjalankan dan melakukan <i>build frontend</i> PWA
6	Git	Terbaru	<i>Version control</i> dan <i>clone repository</i>

7	NestJS	v11.x	<i>Framework backend</i> untuk membangun API RESTful
8	Sequelize	v6.x	ORM yang menghubungkan aplikasi dengan database MySQL
9	Vue.js	v3.x	<i>Framework</i> JavaScript untuk membangun antarmuka <i>frontend</i>
10	Quasar Framework	v2.x	<i>Framework</i> UI berbasis Vue.js dengan dukungan PWA
11	Chart.js	v4.x	<i>Library</i> visualisasi data konsumsi energi dalam bentuk grafik
12	Puppeteer	v24.x	<i>Library Headless Chromium</i> untuk rendering ekspor laporan PDF
13	Web Push	v3.x	<i>Library</i> pengiriman notifikasi Web Push ke <i>browser</i> pengguna
14	AWS SDK	v2.x	Integrasi layanan penyimpanan file AWS S3
15	Passport & JWT	v0.7 / v11.x	Autentikasi dan otorisasi pengguna berbasis JSON Web Token
16	Axios	v1.x	HTTP <i>client</i> untuk komunikasi <i>frontend</i> ke <i>backend</i> API
17	TypeScript	v5.x	Bahasa pemrograman dengan keamanan tipe data pada <i>frontend</i> dan <i>backend</i>

4.1.2 Fitur Aplikasi

4.1.2.1 Halaman Registrasi dan *Login*

Gambar 4.1 merupakan gambar halaman registrasi dan login berfungsi sebagai gerbang masuk bagi pengguna untuk dapat mengakses fitur-fitur personalisasi di dalam aplikasi. Halaman registrasi mewajibkan pengguna baru mengisi nama lengkap, *username*, email, dan kata sandi. Setelah akun terdaftar, pengguna dapat melakukan autentikasi melalui halaman *login* menggunakan email/username dan kata sandi yang telah didaftarkan untuk mendapatkan hak akses ke dalam sistem.

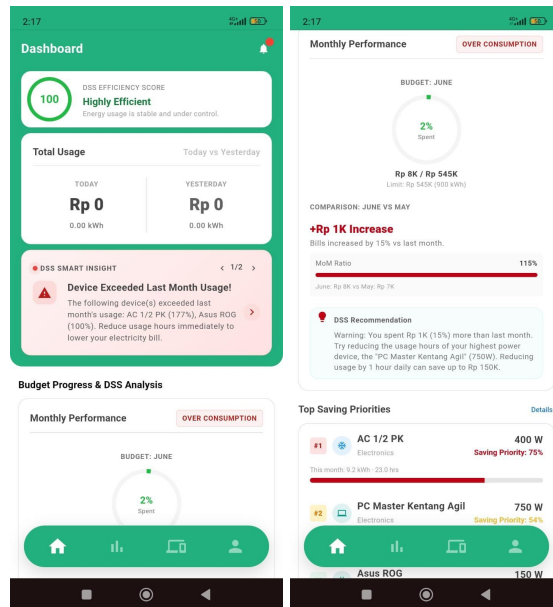


Gambar 4.1 Halaman Registrasi dan Login

4.1.2.2 Halaman *Dashboard* Utama

Gambar 4.2 merupakan gambar halaman dashboard yang merupakan beranda utama setelah pengguna berhasil masuk ke dalam sistem. Halaman ini menyajikan visualisasi ringkasan data yang informatif dan dinamis, meliputi total estimasi konsumsi daya (kWh) dan perkiraan biaya listrik bulanan dari seluruh rumah tangga yang dikelola, grafik tren

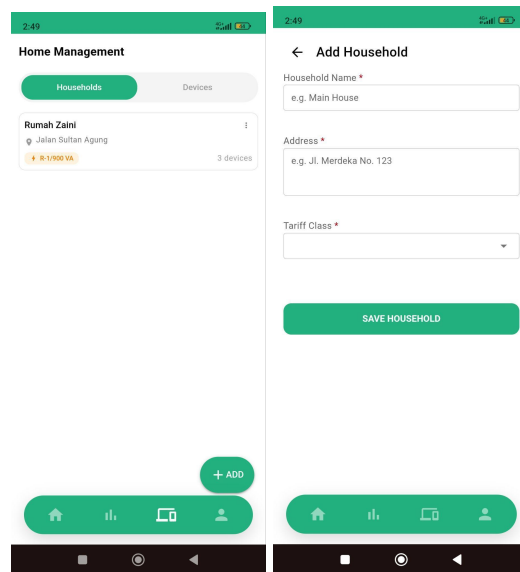
konsumsi listrik dari waktu ke waktu, serta status aktif/non aktif dari perangkat-perangkat listrik yang terdaftar.



Gambar 4.2 Halaman Dashboard

4.1.2.3 Halaman Manajemen Rumah Tangga

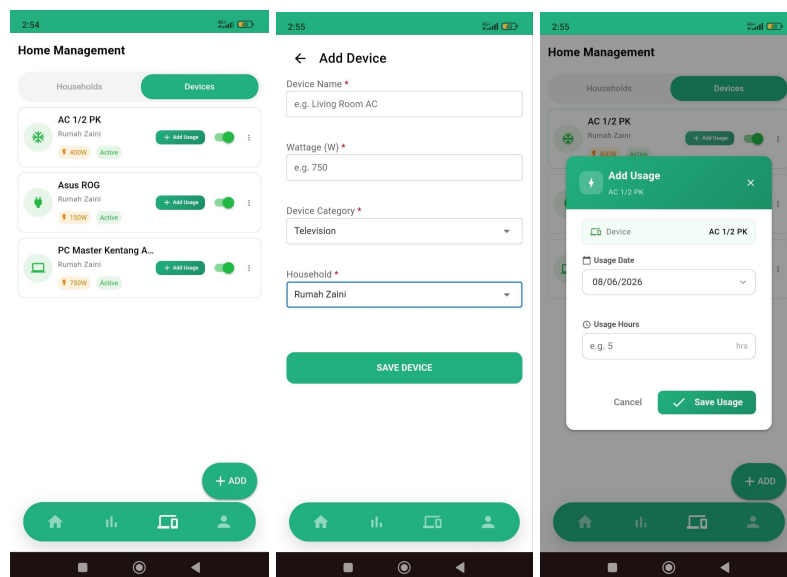
Gambar 4.3 merupakan gambar halaman manajemen rumah tangga pada halaman ini pengguna dapat mengelola data properti atau unit rumah tangga yang dipantau (misalnya: rumah pribadi, kantor, atau kos). Pada halaman ini, pengguna dapat menambahkan rumah tangga baru dengan menginput nama lokasi dan memilih golongan tarif listrik PLN yang berlaku, serta memperbarui atau menghapus data rumah tangga tersebut sesuai kebutuhan.



Gambar 4.3 Halaman Manajemen Rumah Tangga

4.1.2.4 Halaman Manajemen Perangkat Listrik

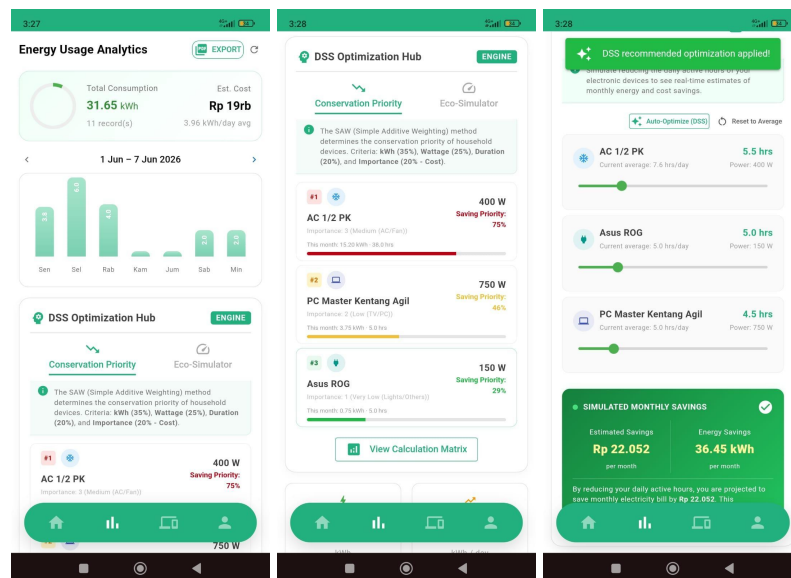
Gambar 4.4 merupakan gambar halaman daftar perangkat listrik yang terdaftar pada suatu rumah tangga. Halaman ini menampilkan detail nama alat elektronik, kategori perangkat, daya listrik nominal (*Watt*), serta menyediakan tombol untuk menambah perangkat baru, mengubah informasi detail perangkat, maupun menghapusnya dari sistem.



Gambar 4.4 Halaman Manajemen Perangkat Listrik

4.1.2.5 Halaman Pemantauan dan Simulasi Penggunaan

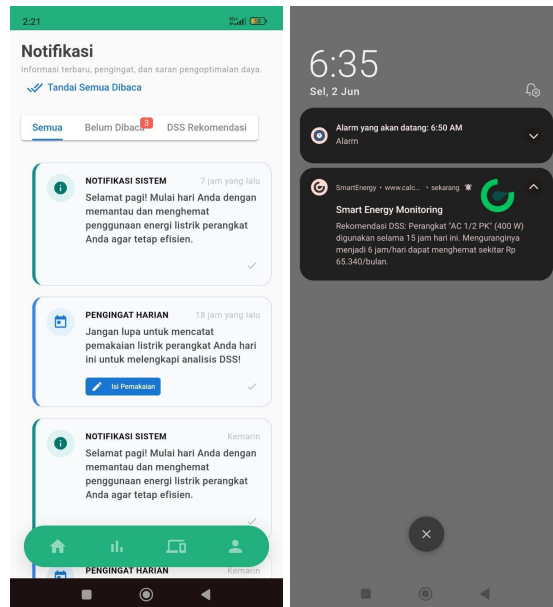
Gambar 4.5 merupakan tampilan halaman simulasi dan estimasi konsumsi listrik. Pengguna dapat menginputkan atau menggeser estimasi durasi penggunaan harian (dalam jam) dari masing-masing alat elektronik. Sistem secara otomatis menghitung dan menampilkan estimasi total konsumsi energi harian/bulanan (kWh) serta akumulasi biaya rupiah yang harus dibayarkan berdasarkan tarif PLN yang aktif..



Gambar 4.5 Halaman Pemantauan dan Simulasi Penggunaan

4.1.2.6 Halaman Notifikasi

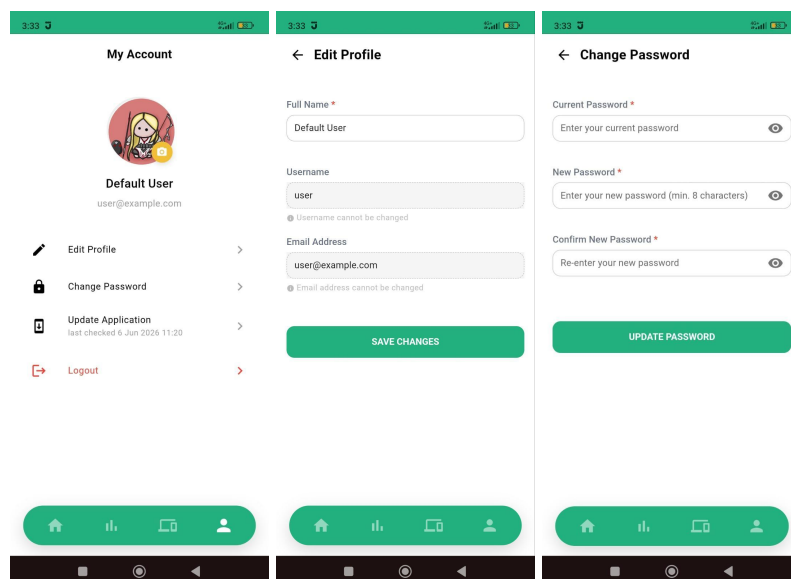
Gambar 4.6 merupakan gambar halaman daftar pemberitahuan sistem. Halaman ini menampilkan riwayat pesan masuk dan peringatan otomatis dari aplikasi, seperti peringatan konsumsi listrik yang melebihi batas rata-rata harian, kiat-kiat penghematan energi, serta informasi pemeliharaan sistem.



Gambar 4.6 Halaman Notifikasi

4.1.2.7 Halaman Profil Pengguna

Gambar 4.7 merupakan halaman pengaturan profil pengguna. Pada halaman ini, pengguna dapat memperbarui foto profil, mengubah informasi identitas akun seperti nama lengkap, username, dan alamat email, melakukan pergantian kata sandi lama, serta keluar (*logout*) dari sesi aplikasi.



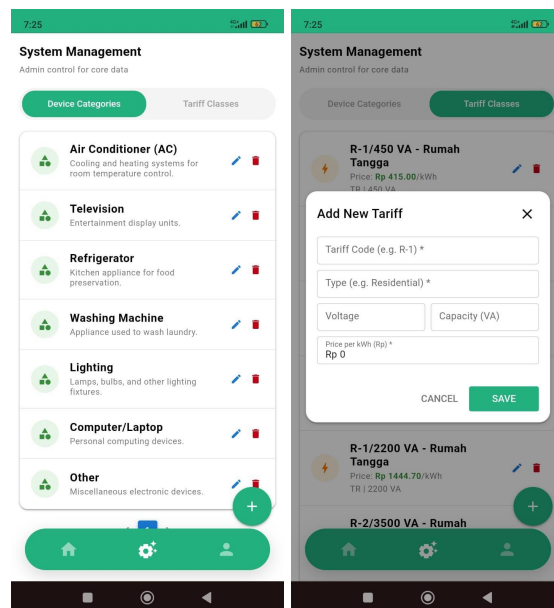
Gambar 4.7 Halaman Profil Pengguna

4.1.2.6 Halaman Manajemen Admin

Halaman Manajemen Admin merupakan antarmuka khusus yang hanya dapat diakses oleh pengguna dengan peran administrator. Halaman ini memiliki dua tab pengelolaan utama:

1. Tab Kategori Perangkat

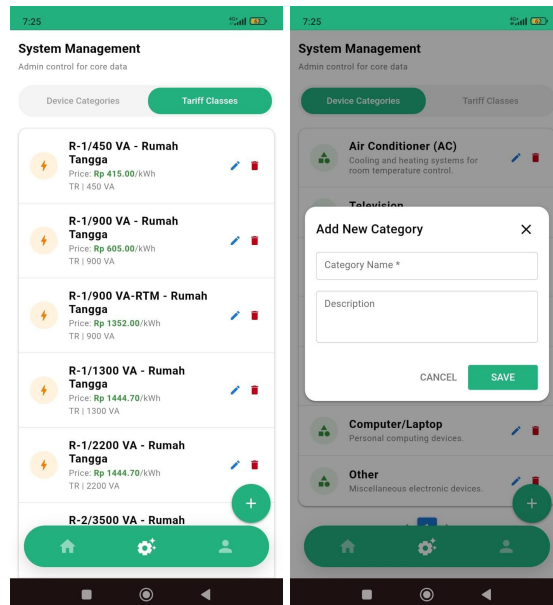
Digunakan admin untuk mengelola kategori perangkat elektronik (misalnya: Pencahayaan, Pendingin Ruangan, Hiburan) yang nantinya akan menjadi pilihan saat pengguna mendaftarkan perangkat mereka.



Gambar 4.8 Halaman Kategori Perangkat

2. Tab Golongan Tarif

Digunakan admin untuk mengelola basis data golongan tarif listrik PLN (seperti R-1/TR 900 VA, R-1/TR 1300 VA beserta tarif per kWh terbarunya). Data tarif ini menjadi acuan rumus perhitungan estimasi biaya di seluruh akun pengguna.



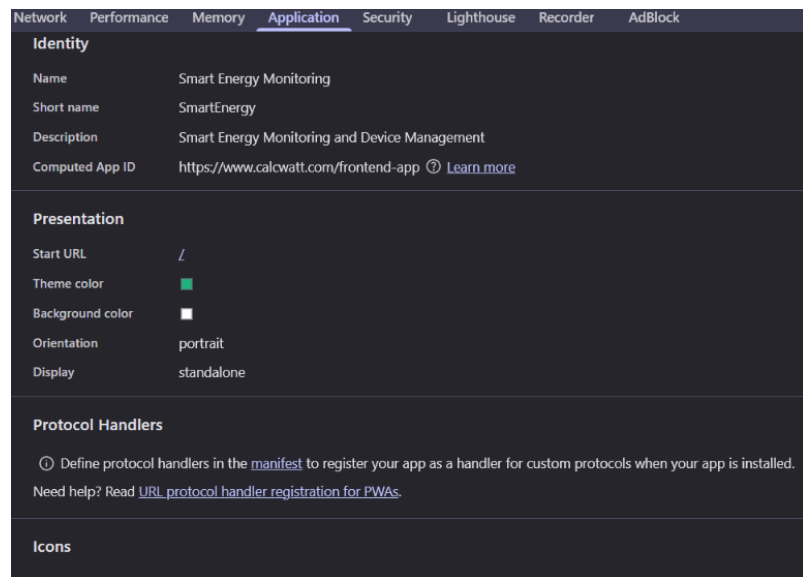
Gambar 4.9 Halaman Golongan Tarif

4.1.2.7 Fitur PWA (*Manifest, Service workers, Chacing*)

Penerapan *Progressive Web App* (PWA) pada aplikasi Smart Energy Monitoring bertujuan agar aplikasi dapat diinstal langsung di perangkat pengguna (HP/Laptop) serta dapat diakses dalam kondisi tanpa koneksi internet (offline). Fitur ini ditopang oleh tiga pilar utama:

1. *Web App Manifest*

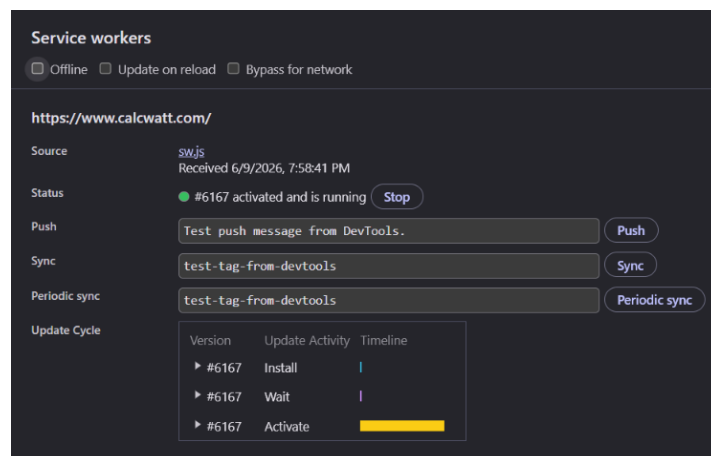
Gambar 4.8 merupakan halaman pemeriksaan *manifest* aplikasi PWA pada Google Chrome *Developer Tools*. Pada halaman ini, penguji dapat memverifikasi nama lengkap aplikasi (Smart Energy Monitoring), nama pendek (SmartEnergy), ikon pintasan, warna tema, serta memastikan konfigurasi mode tampilan (display) berjalan dalam format standalone layaknya aplikasi *mobile native*.



Gambar 4.10 Gambar *Web App Manifest*

2. *Service Workers*

Gambar 4.9 merupakan halaman status *service worker* aplikasi PWA pada Google Chrome *Developer Tools*. Pada halaman ini, penguji dapat memastikan bahwa berkas skrip *service worker* (sw.js) telah terdaftar secara sukses di *browser* dan berstatus aktif (*activated and is running*) untuk menangani penerimaan notifikasi push serta intersepsi jaringan secara *realtime*.



Gambar 4.11 Gambar *Service Worker*

3. *Cache Storage (Caching)*

Gambar 4.10 merupakan halaman penyimpanan cache storage aplikasi PWA pada Google Chrome *Developer Tools*. Pada halaman ini, penguji dapat melihat folder penyimpanan data backend (*api-data-cache*) yang menyimpan riwayat respon data API secara lokal agar halaman utama dashboard pemakaian energi tetap dapat menampilkan informasi sebelumnya saat dijalankan dalam kondisi tanpa internet (*offline*).



The screenshot shows the 'Cache Storage' tab in Chrome Developer Tools for the URL <https://www.calcwatt.com>. It displays a table of cached data under the 'Default bucket'. The table has four columns: '#', 'Name', 'Response-Type', and 'Content-Type'. It lists eight entries, all with a 'cors' response type and 'application/json' content type.

#	Name	Response-Type	Content-Type
0	/api/v1/auth/profile	cors	application/json
1	/api/v1/device-categories	cors	application/json
2	/api/v1/device-categories?page=1&limit=10	cors	application/json
3	/api/v1/devices	cors	application/json
4	/api/v1/devices/1/device-usages	cors	application/json
5	/api/v1/devices/2/device-usages	cors	application/json
6	/api/v1/devices/3/device-usages	cors	application/json
7	/api/v1/devices/4/device-usages	cors	application/json

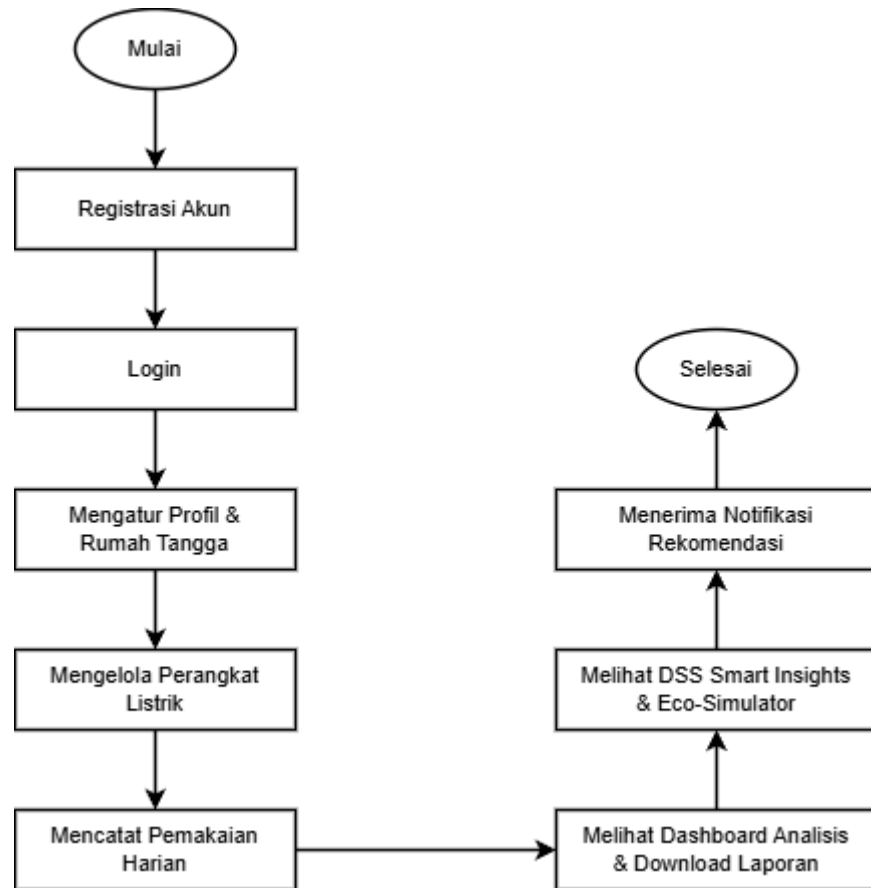
Gambar 4.12 Gambar *Cache Storage*

4.1.3 Alur Kerja Aplikasi

4.1.3.1 Alur Penggunaan Aplikasi Role User

Gambar 4.12 memperlihatkan alur penggunaan aplikasi Voltaris oleh pengguna rumah tangga. Langkah awal yang dilakukan adalah membuat akun melalui halaman registrasi, kemudian melakukan login untuk mengakses sistem. Setelah terautentikasi, pengguna melengkapi data profil dan mendaftarkan rumah tangga beserta golongan tarif listrik PLN yang sesuai. Selanjutnya, pengguna meregistrasikan perangkat-perangkat elektronik yang dimiliki dan mulai mencatat durasi pemakaian harian pada setiap perangkat, di mana sistem secara otomatis mengkalkulasi estimasi konsumsi dalam kWh. Pengguna kemudian dapat memantau ringkasan konsumsi dan biaya melalui dashboard analisis serta mengunduh laporan dalam format PDF. Fitur DSS Smart Insights menyajikan prioritas penghematan berdasarkan metode SAW dan *Eco-Simulator* untuk

mensimulasikan potensi penghematan biaya. Secara berkala, sistem juga mengirimkan notifikasi rekomendasi optimasi melalui Web Push Notification apabila terdeteksi pola pemakaian yang tidak efisien.



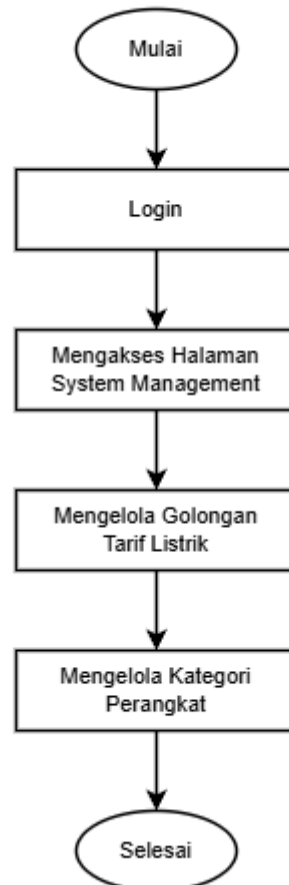
Gambar 4.13 Halaman Masuk

4.1.3.2 Alur Penggunaan Aplikasi Role Admin

Gambar 4.13 memperlihatkan alur penggunaan aplikasi Voltaris oleh System Admin. Tahap pertama yang dilalui adalah melakukan login menggunakan akun administrator, kemudian admin diarahkan ke halaman System Management. Pada halaman tersebut, admin bertanggung jawab mengelola data golongan tarif listrik PLN yang meliputi nama golongan, daya (VA), dan harga per kWh data ini menjadi acuan kalkulasi biaya bagi seluruh pengguna di dalam sistem. Selain itu, admin juga mengelola data kategori perangkat elektronik yang nantinya digunakan oleh pengguna

sebagai opsi klasifikasi saat mendaftarkan alat listrik miliknya.

Alur Penggunaan Aplikasi Admin



Gambar 4.14 Halaman Masuk

4.2. Pengujian

4.2.1 Pengujian Fungsional

Pengujian fungsional dilakukan untuk memastikan bahwa seluruh fungsional telah berjalan sesuai dengan standar yang telah ditetapkan, pengujian dilakukan menggunakan metode *BlackBox Testing* dengan membuat skenario *test case* dan menyesuaikan dengan kebutuhan fungsional. Hasil dari pengujian fungsional dapat dilihat pada Tabel 4.2 dibawah ini.

Tabel 4.2 Pengujian Fungsional

Fungsional	Total <i>Test Case</i>	Total Berhasil	Total Gagal	<i>Persentase Berhasil</i>
Tambah & Kelola Akun (Autentikasi)	8	8	0	100%
Manajemen Rumah Tangga	3	3	0	100%
Manajemen Perangkat Elektronik	4	4	0	100%
Pencatatan Pemakaian	3	3	0	100%
Sistem Pendukung Keputusan (DSS)	2	2	0	100%
Ekspor Laporan PDF	2	2	0	100%
Notifikasi & Logout	2	2	0	100%
Total	24	24	0	100%

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, pembangunan, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa tujuan proyek akhir untuk membangun sistem monitoring konsumsi energi listrik rumah tangga berbasis *Progressive Web App* (PWA) yang dilengkapi fitur rekomendasi optimasi energi telah tercapai. Sistem yang dikembangkan tidak terhubung langsung dengan perangkat keras atau sensor *IoT*, melainkan data pemakaian perangkat elektronik diinput secara mandiri oleh pengguna. Data tersebut kemudian diolah oleh sistem untuk menghasilkan estimasi konsumsi energi (kWh) dan biaya listrik berdasarkan tarif PLN, serta memberikan rekomendasi penghematan energi melalui modul *Decision Support System* (DSS) berbasis aturan heuristik.

Hasil pengujian menggunakan metode *Black Box Testing* menunjukkan bahwa sebanyak 24 *test case* yang telah dirancang berhasil dijalankan tanpa ditemukan kegagalan. Hal ini membuktikan bahwa seluruh fungsionalitas sistem telah berjalan dengan baik dan sesuai dengan kebutuhan yang telah ditetapkan.

5.2 Saran

Berdasarkan hasil pengembangan dan pengujian yang telah dilakukan, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan sistem selanjutnya. :

1. Mengintegrasikan sistem dengan sensor *IoT* atau smart meter agar data konsumsi energi dapat terbaca secara otomatis dan *real-time*, sehingga tidak lagi bergantung pada input manual oleh pengguna.
2. Menambahkan fitur notifikasi aktif melalui email atau aplikasi pesan untuk memberikan peringatan kepada pengguna ketika konsumsi energi mendekati atau melebihi batas yang telah ditetapkan.
3. Mengembangkan metode rekomendasi pada modul DSS dengan pendekatan yang lebih adaptif, seperti pemanfaatan algoritma *machine*

learning, sehingga rekomendasi yang diberikan dapat lebih personal dan sesuai dengan pola konsumsi masing-masing pengguna.

DAFTAR PUSTAKA

- Amrullah, A., Salim, Y., & Rachman Manga, A. (2021). Implementasi Progressive Web App Terhadap Aplikasi E-Commerce Sebagai Solusi Untuk Meningkatkan Kinerja Aplikasi Berbasis Web INFORMASI ARTIKEL ABSTRAK. *Buletin Sistem Informasi Dan Teknologi Islam*, 2(3), 213–221. <https://jurnal.fikom.umi.ac.id/index.php/BUSITI/article/view/912>
- Atmanto, D. A., Nanditama, R. W., Suteddy, W., & Adiwilaga, A. (2025). Sistem Monitoring Konsumsi Energi Listrik Berbasis IoT Menggunakan Fuzzy Logic Mamdani. *TELKA - Telekomunikasi Elektronika Komputasi Dan Kontrol*, 11(2), 151–166. <https://doi.org/10.15575/telka.v11n2.151-166>
- Badan Pusat Statistik. (n.d.). Konsumsi listrik per kapita. <https://www.bps.go.id/id/statistics-table/2/MTE1NiMy/konsumsi-listrik-per-kapita.html>
- Badan Pusat Statistik. (n.d.). Rasio elektrifikasi. <https://www.bps.go.id/id/statistics-table/2/MTE1NSMy/rasio-elektrifikasi.html>
- Bimanatara, C. K. M., Akbar, F. A., & Puspaningrum, E. Y. (2025). IMPLEMENTASI PROGRESSIVE WEB APPLICATION (PWA) DALAM PENGEMBANGAN SISTEM PESAN- ANTAR MAKANAN (STUDI KASUS: WIRAWIRI BOJONEGORO). *JITET (Jurnal Informatika Dan Teknik Elektro Terapan)*, 13(2), 185–196. <https://doi.org/http://dx.doi.org/10.23960/jitet.v13i2.6132>
- Dolly H. Manik, Reza Nandika, P. G. (2021). *PENERAPAN INTERNET OF THINGS (IOT) PADA SISTEM MONITORING PEMAKAIAN DAYA LISTRIK RUMAH TANGGA BERBASIS MIKROKONTROLER DAN WEBSITE*. 4(2), 255–261. <https://doi.org/https://doi.org/10.33373/sigmateknika.v4i2.3618>
- Jamilatun, S., Rhomadoni, F. R., Astuti, E., Wardhana, B. S., Idris, M., & Auliasari, P. A. (2025). Peran Manajemen Energi terhadap Efisiensi Konsumsi Listrik Rumah Tangga di Indonesia. *Prosiding Semnastek*.
- Karuniawan, A. E. (2024). Sistem Monitoring Konsumsi Energi Listrik kWh Meter Secara Real Time pada Rumah Tangga Berbasis IoT. *Jurnal Elektro Kontrol (ELKON)*, 4(1), 15-24.
- Satya, R., Putra, B., Agustina, S. B., & Vernanda, D. (2024). Sistem Perhitungan Pemakaian Listrik Rumah Tangga Berbasis Android. *INVENTOR: Jurnal Inovasi*

Dan Tren Pendidikan Teknologi Informasi, 2(2), 61–66.
<https://doi.org/https://doi.org/10.37630/inventor.v2i2.1336>

Septanto, H., & Hidayatullah, A. (2022). Perancangan Sistem Informasi Monitoring Proyek Berbasis Web Untuk Mendukung Implementasi Paperless Office. *Jurnal Tera*, 2(2), 34–43. <https://doi.org/10.59832/jt.v2i2.130>

Siska Ludfi Zean, Solekhan, I. A. R. (2021). SISTEM MONITORING PEMAKAIAN ENERGI LISTRIK RUMAH TANGGA. *ELKON*, 01(01), 15–24.

Willyansah, W., Ayu, F., & Muhammad, M. (2025). Implementasi Sistem Informasi Monitoring Laboratorium Komputer Berbasis Web Menggunakan Metode Waterfall. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 7(1), 166–171. <https://doi.org/10.47233/jteksis.v7i1.1753>

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement

Kuesioner ini disebarikan kepada 5 orang pengguna listrik rumah tangga yang aktif memantau penggunaan energi listrik. Pengumpulan data dilakukan secara online via zoom dan menggunakan google form untuk pengisian kuesioner pada tanggal 01 Januari 2026.

Skala penilaian yang digunakan adalah Skala Likert 5 poin dengan keterangan sebagai berikut:

- SS** = Sangat Setuju
- S** = Setuju
- N** = Netral
- TS** = Tidak Setuju
- STS** = Sangat Tidak Setuju

Tabel 1 Lampiran Hasil Kuesioner Pengguna Listrik Rumah Tangga

No	Pertanyaan	SS	S	N	TS	STS
1.	Saya masih mencatat pemakaian listrik secara manual (catatan kertas, ingatan, atau tidak mencatat sama sekali)	3	2	1		
2.	Saya kesulitan menghitung estimasi biaya listrik per perangkat elektronik secara mandiri	2	3	1		
3.	Saya sering terkejut ketika tagihan listrik bulanan lebih tinggi dari perkiraan	3	2		1	
4.	Saya tidak mengetahui perangkat mana yang paling banyak mengkonsumsi listrik di rumah	2	2	2		

5.	Saya tidak memiliki cara untuk memantau pola pemakaian listrik harian secara real-time	3	2	1		
6.	Saya membutuhkan notifikasi atau peringatan otomatis ketika pemakaian listrik tergolong boros	4	1	1		
7.	Saya membutuhkan saran/rekomendasi untuk menghemat penggunaan listrik berdasarkan data pemakaian	3	2	1		
8.	Saya membutuhkan laporan rekap pemakaian listrik dalam bentuk dokumen (PDF)	2	2	2		
9.	Apakah menurut Anda dibutuhkan sistem informasi untuk memantau dan mengoptimalkan penggunaan energi listrik rumah tangga?	4	1	1		

Lampiran B. Link Produk

Repository GitHub : <https://github.com/n3eversurrender/frontend-app.git>
<https://github.com/n3eversurrender/backend-api.git>

Lampiran C. Dokumen Pengujian

Lampiran Pengujian Fungsional :  Dokumentasi Pengujian Fungsional.pdf