

**APLIKASI ANDROID UNTUK PEMANTAUAN
LISTRIK PRABAYAR**



الجامعة الإسلامية
الاندونيسية

Disusun Oleh:

N a m a : Muhammad Aji Perdana
NIM : 18523057

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2022

HALAMAN PENGESAHAN DOSEN PENGUJI

**APLIKASI ANDROID UNTUK PEMANTAUAN
LISTRIK PRABAYAR****TUGAS AKHIR**

Telah dipertahankan di depan sidang pengujian sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 9 Januari 2023

Tim Penguji

Irving Vitra Paputungan, S.T., M.Sc.IT.,
Ph.D.

Anggota 1

Elyza Gustri Wahyuni, S.T., M.Cs.

Anggota 2

Zainudin Zukhri, S.T., M.IT.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Aji Perdana

NIM : 18523057

Tugas akhir dengan judul:

APLIKASI ANDROID UNTUK PEMANTAUAN LISTRIK PRABAYAR

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 14 Desember 2022

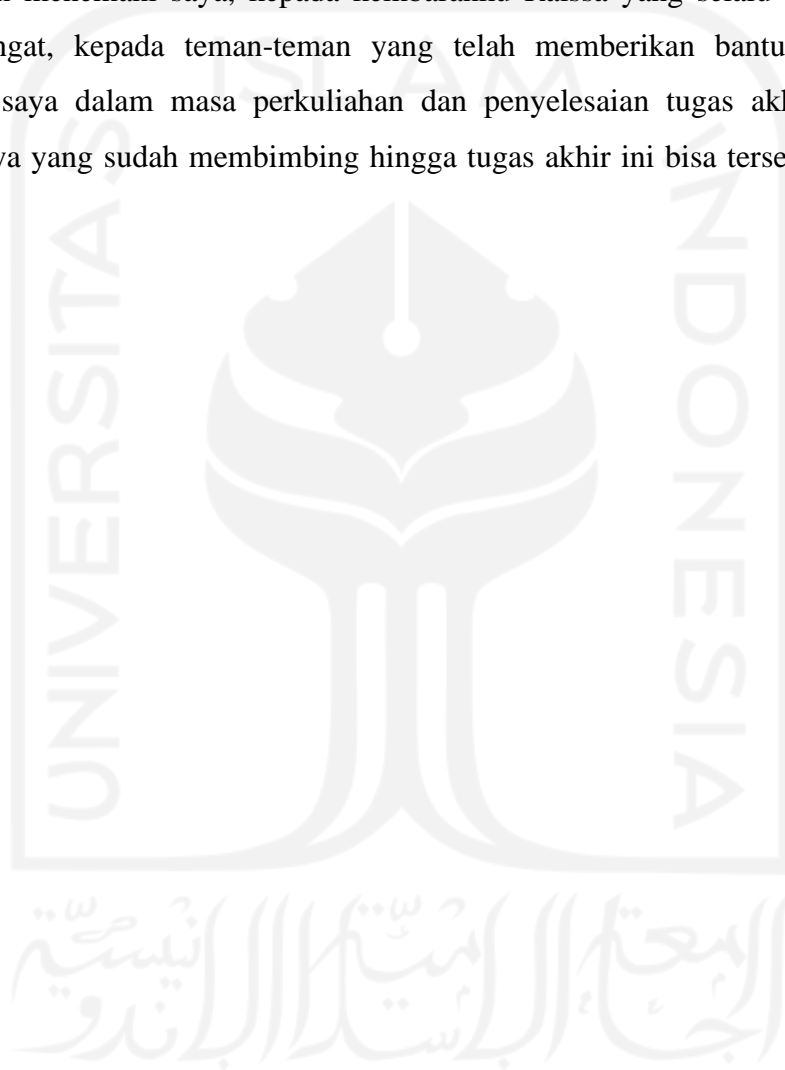


(Muhammad Aji Perdana)

HALAMAN PERSEMBAHAN

Alhamdulillah. Segala Puji Bagi Allah yang telah melimpahkan karunianya sehingga tugas akhir ini bisa terselesaikan. Saya mengucapkan terima kasih bagi semua pihak yang telah menemani, mendukung, dan membantu saya dalam menyelesaikan tugas akhir saya.

Tugas akhir ini saya persembahkan kepada Mama dan Papah yang selalu mendoakan, mendukung, dan menemani saya, kepada kembaranku Raissa yang selalu mendukung dan memberi semangat, kepada teman-teman yang telah memberikan bantuan, waktu, dan semangat bagi saya dalam masa perkuliahan dan penyelesaian tugas akhir, dan kepada pembimbing saya yang sudah membimbing hingga tugas akhir ini bisa terselesaikan dengan baik.



HALAMAN MOTO

“When we’re growing up there are all sorts of people telling us what to do when really what we need is space to work out what to do”

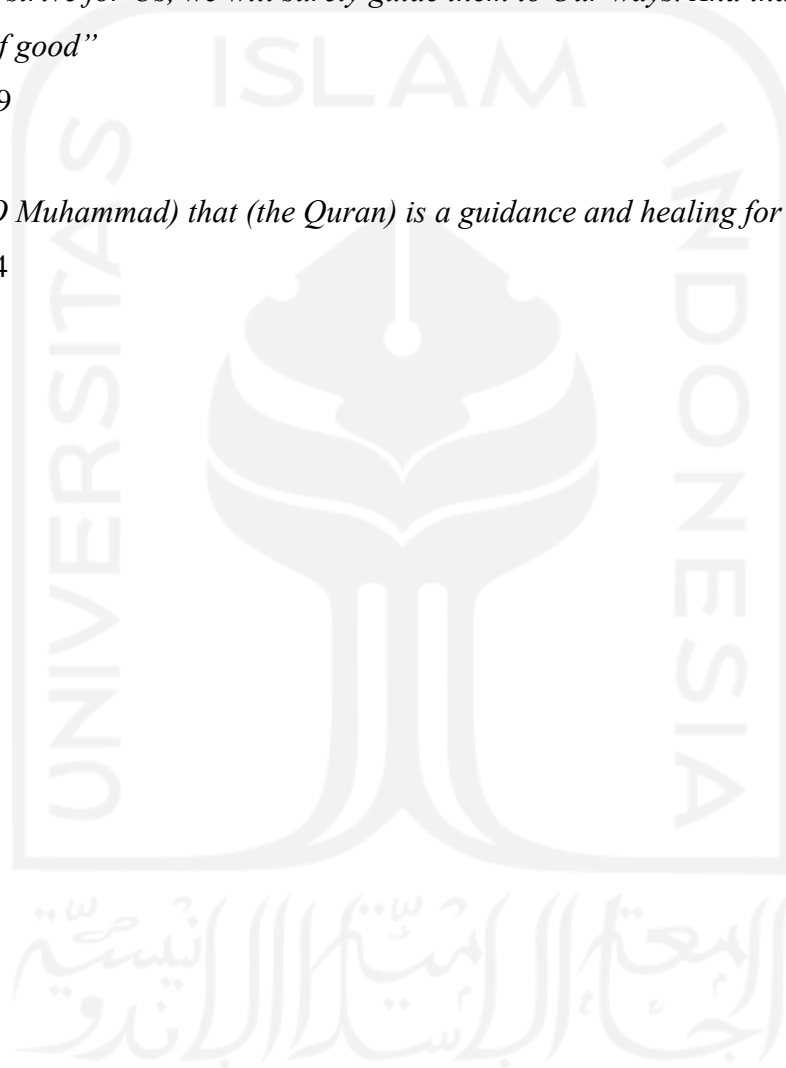
– Elliot Page

“And those who strive for Us, we will surely guide them to Our ways. And indeed Allah is with the doers of good”

– Al-Alkabut: 69

“And declare (O Muhammad) that (the Quran) is a guidance and healing for the believers”

– Al-Fussilat: 44



KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum wr. wb.

Alhamdulillah. Segala puji bagi Allah. Yang Maha Pengasih dan Maha Penyayang. Yang telah melimpahkan rahmat dan karunianya sehingga tugas akhir dengan judul “APLIKASI ANDROID UNTUK PEMANTAUAN LISTRIK PRABAYAR” ini dapat terselesaikan.

Tugas akhir ini dibuat sebagai salah satu syarat untuk meraih gelar sarjana di program studi Teknik Informatika Universitas Islam Indonesia. Tugas akhir ini diharapkan bisa memberikan manfaat yang sebesar-besarnya untuk membantu penyelesaian permasalahan dan memudahkan kehidupan dengan bantuan teknologi.

Selama penulis menyelesaikan tugas akhir ini, tentu ada banyak tantangan dan permasalahan yang dihadapi. Namun, selama proses pengerjaan tugas akhir ini, penulis mendapatkan banyak dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini, penulis mengucapkan terima kasih sebesar-besarnya kepada:

1. Pak Prof. Dr. Ir. Hari Purnomo., M.T., IPU, ASEAN.Eng., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
2. Pak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
3. Pak Irving Vitra Papatungan, S.T., M.Si., Ph.D. dan Pak Kurniawan Dwi Irianto, S.T., M.Sc., selaku dosen pembimbing.
4. Keluarga tercinta, Mama, Papah, Raissa, serta Beci yang selalu memberikan doa, semangat, dukungan, penghiburan, dan berbagai bantuan selama penulis menempuh perkuliahan dan menyelesaikan tugas akhir.
5. Teman-teman semua, ABM kids , serta Adam dan Hashfi, yang turut berperan membantu penulis selama menyelesaikan tugas akhir.

Akhir kata, *Wassalamu'alaikum wr. wb.*

Yogyakarta, 14 Desember 2022



(Muhammad Aji Perdana)

SARI

Masyarakat masih awam mengenai penggunaan listrik mereka sendiri. Hal ini diakibatkan oleh cara pengecekan penggunaan listrik yang tidak praktis. Saldo yang tersisa hanya bisa diketahui dengan cara menghampiri dan melihat meteran listrik secara langsung. Meteran listrik pada umumnya juga hanya dapat menampilkan satu jenis data pada tampilan LCD di meteran, dan untuk mengecek data-data selain saldo, pengguna harus menginput kode-kode tertentu. Kode-kode ini berbeda pada setiap meteran listrik. Hal ini sangat tidak praktis dan memakan waktu, yang menyebabkan masyarakat enggan untuk mengeceknya, padahal data tersebut berguna untuk pemantauan yang lebih terperinci. Oleh karena itu, dibutuhkan sebuah sistem untuk memudahkan masyarakat dalam pemantauan penggunaan listrik mereka. Penelitian berjudul Aplikasi Android Untuk Pemantauan Listrik Prabayar ini dilakukan untuk mengembangkan sebuah aplikasi agar masyarakat dapat dengan mudah memantau penggunaan listrik. Hanya dengan membuka aplikasi pada *smartphone*, pengguna dapat melihat data terperinci tentang penggunaan listrik mereka. Android dipilih sebagai basis aplikasi karena di Indonesia, pengguna *smartphone* Android jauh lebih banyak dibandingkan jenis *smartphone* lainnya, yang berarti akan lebih bermanfaat bagi masyarakat luas jika aplikasi dikembangkan untuk sistem operasi Android. Data yang disajikan oleh aplikasi berupa sisa saldo kWh, total penggunaan kWh, tegangan (Volt), arus (Ampere), daya (Watt), dan total harga dari jumlah kWh yang telah digunakan. Aplikasi ini juga dapat mengingatkan pengguna dengan cara mengirimkan notifikasi ke pengguna ketika saldo kWh hampir habis dan juga membantu pengguna untuk menghemat penggunaan listrik dengan notifikasi ini.

Kata kunci—Aplikasi Android, Smartphone, Pemantauan, Listrik Prabayar.

GLOSARIUM

Driver	Sebuah perangkat lunak yang berfungsi sebagai sarana komunikasi antara hardware dan software.
Framework	Kerangka kerja yang berguna untuk mempermudah pengembangan aplikasi atau web.
IDE	Singkatan dari “ <i>Integrated Development Environment</i> ”, yang berarti sebuah lingkungan pengembangan yang terintegrasi.
JSON	Singkatan dari “ <i>Javascript Object Notation</i> ”. JSON adalah sebuah format file yang digunakan untuk menyimpan dan mentransfer sebuah data.
Library	Kumpulan fungsi-fungsi atau implementasi untuk digunakan dalam kegiatan pemrograman.
Mobile	Merujuk ke jenis perangkat yang mudah dipindah-pindah.
Open Source	Kode dari sebuah perangkat lunak yang bebas untuk digunakan dan dimodifikasi oleh pengguna.
Prototype	Purwarupa
SDK	Singkatan dari “ <i>Software Development Kit</i> ”, sebuah perangkat lunak yang digunakan untuk programmer sebagai alat bantu pengembangan perangkat lunak dan sebagai API
DDoS	Singkatan dari “ <i>Distributed Denial of Service</i> ”, sebuah serangan siber terhadap suatu web atau layanan pada internet dengan cara membanjiri server dengan <i>traffic</i> palsu secara terus menerus yang menyebabkan server tidak mampu menerima <i>traffic</i> lagi.
Pop-up	Tampilan jendela kecil yang muncul pada layar yang berisi informasi tertentu.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	viii
GLOSARIUM.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penelitian	4
BAB II LANDASAN TEORI.....	5
2.1 Aplikasi <i>Mobile</i>	5
2.2 Tinjauan Metode Pengembangan Untuk Aplikasi <i>Mobile</i>	5
2.3 Tinjauan Penerapan Aplikasi <i>Mobile</i> Untuk Penelitian Sejenis.....	16
2.4 Android Studio	18
2.5 Firebase Realtime Database	18
2.6 <i>Activity Diagram</i>	18
2.7 <i>Black-box Testing</i>	19
2.8 <i>User Acceptance Test (UAT)</i>	19
2.9 <i>Internet of Things</i>	19
BAB III METODOLOGI PENELITIAN	21
3.1 Metode Pengembangan Sistem.....	21

3.2	Alur Proses Sistem	22
3.3	Analisis Kebutuhan	23
3.3.1	Kebutuhan <i>Hardware</i>	23
3.3.2	Kebutuhan <i>Output</i>	24
3.3.3	Kebutuhan <i>Software</i>	24
3.3.4	Kebutuhan Proses Sistem	24
3.4	Perancangan Sistem.....	24
3.4.1	Arsitektur Sistem.....	24
3.4.2	Rancangan Desain Sistem	25
3.4.3	Rancangan Tampilan Aplikasi	30
3.4.4	Rancangan Database.....	33
BAB IV HASIL DAN PEMBAHASAN		34
4.1	Tampilan Antarmuka.....	34
4.1.1	Halaman utama.....	34
4.1.2	Halaman Input Saldo	37
4.1.3	Halaman Pengingat Saldo	40
4.1.4	Notifikasi	42
4.2	Pengujian.....	43
4.2.1	Pengujian Aplikasi	43
4.2.2	Pengujian Keakuratan Pembacaan	44
4.2.3	Pengujian Penggunaan Kuota Data Internet.....	47
4.2.4	Pengujian Kompatibilitas Aplikasi.....	48
4.2.5	<i>User Acceptance Test (UAT)</i>	52
BAB V KESIMPULAN DAN SARAN		54
5.1	Kesimpulan.....	54
5.2	Saran	54
DAFTAR PUSTAKA		55

DAFTAR TABEL

Tabel 4.1 Hasil pengujian aplikasi.....	44
Tabel 4.2 Hasil pengujian ke-1 pembacaan kWh	45
Tabel 4.3 Hasil pengujian ke-2 pembacaan kWh	45
Tabel 4.4 Hasil pengujian ke-3 pembacaan kWh	45
Tabel 4.5 <i>Error%</i> rata-rata hasil pengujian	45
Tabel 4.6 Hasil pengujian ke-1 pembacaan data listrik	46
Tabel 4.7 Hasil pengujian ke-2 pembacaan data listrik	46
Tabel 4.8 Hasil pengujian ke-3 pembacaan data listrik	46
Tabel 4.9 Hasil perhitungan <i>error%</i> rata-rata pada pembacaan data listrik	46
Tabel 4.10 Pertanyaan kuesioner	52
Tabel 4.11 Keterangan bobot nilai.....	52
Tabel 4.12 Tabel hasil penilaian responden.....	53

DAFTAR GAMBAR

Gambar 3.1 Tahapan pengembangan dengan metode <i>waterfall</i>	21
Gambar 3.2 Alur Proses Utama Sistem	22
Gambar 3.3 Arsitektur sistem	25
Gambar 3.4 <i>Activity Diagram</i> halaman utama.....	26
Gambar 3.5 <i>Activity Diagram</i> halaman input saldo.....	27
Gambar 3.6 <i>Activity Diagram</i> halaman pengingat.....	28
Gambar 3.7 <i>Activity Diagram</i> notifikasi.....	29
Gambar 3.8 Rancangan tampilan halaman utama.....	30
Gambar 3.9 Rancangan tampilan input saldo	31
Gambar 3.10 Rancangan tampilan halaman pengingat.....	32
Gambar 3.11 Struktur <i>database</i> pada <i>software</i> Firebase Realtime Database	33
Gambar 4.1 Tampilan halaman utama	34
Gambar 4.2 Rumus harga	36
Gambar 4.3 Halaman Input Saldo.....	37
Gambar 4.4 Penginputan Saldo.....	38
Gambar 4.5 <i>Pop-up</i> input saldo berhasil.....	39
Gambar 4.6 Data kWh tersisa setelah dilakukan proses input.....	39
Gambar 4.7 Halaman pengingat saldo	40
Gambar 4.8 <i>Pop-up</i> input batas saldo berhasil	41
Gambar 4.9 Tampilan aplikasi saat saldo habis.....	41
Gambar 4.10 Notifikasi sisa saldo	42
Gambar 4.11 Notifikasi saldo habis.....	43
Gambar 4.12 Penggunaan kuota internet dari aplikasi	47
Gambar 4.13 Aplikasi berjalan pada perangkat Google Nexus 4 dengan versi Android 6.0 ..	48
Gambar 4.14 Aplikasi berjalan pada perangkat Google Nexus 4 dengan versi Android 8.0 ..	49
Gambar 4.15 Aplikasi berjalan pada perangkat Google Pixel 4 dengan versi Android 7.0	50
Gambar 4.16 Aplikasi berjalan pada perangkat Google Pixel 6 dengan versi Android 11.0 ..	51

BAB I

PENDAHULUAN

1.1 Latar Belakang

Energi listrik merupakan energi yang paling banyak dibutuhkan manusia saat ini. Menurut website Badan Pusat Statistik, penggunaan listrik per-kapita di Indonesia pada tahun 2010 adalah sebesar 0,7 MWh dan pada tahun 2020, angka tersebut sudah mencapai 1,09 MWh (Badan Pusat Statistik, 2020). Bahkan PLN mencatat data pengguna listrik di Indonesia sampai dengan Tahun 2021 adalah sebanyak 82,5 juta pelanggan. Dari data tersebut, dapat disimpulkan bahwa penggunaan listrik di Indonesia terus akan meningkat seiring dengan perkembangan teknologi di masa-masa yang akan datang dengan semakin bertambahnya perangkat-perangkat elektronik yang membutuhkan energi listrik dalam pengoperasiannya. Pemakaian listrik meningkat dengan dengan laju pertumbuhan sebesar 8,4 persen per tahun.

Penggunaan listrik Prabayar di Indonesia juga semakin meningkat karena penyediaan listrik oleh PLN lebih diarahkan menggunakan listrik Prabayar sebagai upaya untuk menutup defisit listrik dan mengurangi kerugian energi akibat pencurian listrik di Indonesia (Nugroho et al., 2017). Beberapa kelebihan dalam penggunaan listrik Prabayar antara lain pembelian listrik bisa menyesuaikan dengan kemampuan yang dimiliki tanpa memerlukan pencatatan meter, tidak ada pengenaan denda dan sanksi-sanksi lain apabila mengalami keterlambatan dalam melakukan pembayaran sebagaimana pada listrik Pasca Bayar. Namun ada beberapa kelemahan juga dari listrik Prabayar, seperti listrik yang tiba-tiba mati karena kehabisan token listrik. Bahkan ada kasus dimana meteran listrik Prabayar mengalami error yang menyebabkan tampilan pulsa listrik yang tersisa tidak berkurang meski digunakan (pojokkita.com, 2021). Hal-hal seperti ini tentunya dapat memengaruhi aktivitas pengguna yang pada saat tertentu memerlukan listrik namun terjadi kendala sehingga memengaruhi aktivitas atau kinerja.

Perkembangan teknologi yang menyebabkan peningkatan dalam pemakaian listrik belum dibarengi dengan kemudahan dalam hal melakukan pemantauan terhadap pemakaian listrik sehingga masalah dalam pemakaian listrik Prabayar masih sering muncul. Tidak ada cara praktis untuk melakukan pemantauan langsung terhadap penggunaan listrik masyarakat yang menyebabkan susah melakukan penghematan karena pengguna tidak mengetahui berapa banyak konsumsi listrik mereka (Pangestu et al., 2019). Pemantauan atau pengecekan masih dilakukan secara fisik ke alat meter yang ada sehingga sulit untuk memperkirakan saldo yang

tersisa, meskipun ada beberapa alat yang sudah disertai dengan alarm pengingat pada alat meteran listrik yang digunakan namun cara tersebut masih belum dianggap sebagai cara yang praktis dalam melakukan pemantauan pemakaian listrik Prabayar. Meteran listrik memang memiliki pengingat berupa alarm yang akan berbunyi secara terus menerus ketika saldo hampir habis, tetapi alarm ini hanya akan berguna jika pengguna berada di rumah. Jika pengguna sedang berada di luar rumah, tidak ada cara untuk memantau berapa banyak saldo listrik yang tersisa.

Berdasarkan uraian yang telah disebutkan di atas, pemantauan terhadap penggunaan atau pemakaian listrik Prabayar praktis serta akurat diperlukan agar pengguna dapat mengantisipasi masalah-masalah yang muncul. Dengan demikian diperlukan sebuah sistem yang dapat memantau penggunaan daya dari alat-alat elektronik secara real time yang dapat diakses dan digunakan dengan mudah. Cara pemantauan yang dianggap paling mudah dan praktis pada era saat ini adalah dengan menggunakan *smartphone* yang berbasis Android. Pemilihan platform Android sebagai dasar aplikasi adalah karena menurut web Statcounter (statcounter.com, 2022), pangsa pasar *smartphone* dengan sistem operasi Android di Indonesia pada bulan Mei 2022 mencapai 91.57%. Yang berarti prosentase pengguna *smartphone* Android jauh melebihi *smartphone* yang lain. Berdasarkan fakta yang telah disebutkan, sistem yang dibuat untuk platform Android akan lebih bermanfaat dibandingkan platform lain karena dapat digunakan oleh masyarakat lebih luas. Dari masalah diatas, muncul dorongan kebutuhan untuk adanya sebuah solusi untuk mengatasi ketidakpraktisan dalam pemantauan penggunaan listrik Prabayar yang telah disebutkan. Penelitian berjudul Aplikasi Android Untuk Pemantauan Listrik Prabayar ini bertujuan untuk mengembangkan aplikasi berbasis *smartphone* Android yang akan memberikan kemudahan bagi pengguna dalam melakukan pemantauan terhadap penggunaan listrik Prabayar.

Aplikasi yang dikembangkan belum dapat mengambil data langsung dari meteran listrik karena modifikasi pada meteran listrik Prabayar rumah termasuk pelanggaran hukum. Hal ini tertulis dalam Surat Perjanjian Jual Beli Tenaga Listrik (SPJBTL). Pembacaan data listrik pada aplikasi berasal dari alat elektronik yang terhubung pada terminal stopkontak yang dipasang sensor arus. Aplikasi hasil dari penelitian ini hanya berfungsi sebagai model yang diharapkan dapat diaplikasikan langsung pada meteran listrik Prabayar asli dan dapat membaca seluruh penggunaan listrik pada sebuah rumah atau tempat tinggal.

Penelitian yang mengembangkan sistem pemantauan penggunaan listrik dengan aplikasi Android sebagai sarana antarmuka untuk menampilkan data sudah pernah dilakukan

sebelumnya, tetapi fokus utama dari penelitian-penelitian tersebut lebih condong ke pengembangan hardware. Selain itu, fitur pemantauan yang disediakan oleh aplikasi-aplikasi tersebut hanya menampilkan data listrik secara real time. Belum ada penelitian yang mengembangkan aplikasi Android dengan fitur yang secara khusus ditujukan untuk memenuhi kebutuhan pemantauan listrik prabayar yang juga dapat mengingatkan pengguna mengenai saldo listrik yang tersisa.

1.2 Rumusan Masalah

Rumusan masalah yang mendasari penelitian ini adalah bagaimana cara merancang dan mengembangkan sebuah aplikasi berbasis Android untuk melakukan pemantauan penggunaan listrik prabayar yang dapat menampilkan informasi data listrik secara *real time* dan dapat mengingatkan pengguna mengenai saldo listrik yang tersisa?

1.3 Batasan Masalah

Agar pengerjaan tugas akhir ini tetap terfokus, diperlukan batasan masalah agar pengerjaan tidak melenceng dari permasalahan yang diangkat. Batasan masalah dari penelitian ini adalah:

- a. Pengembangan aplikasi hasil dari penelitian ini hanya berfungsi untuk pemantauan penggunaan listrik dan pengingat mengenai saldo listrik, bukan untuk pengisian saldo listrik.
- b. Aplikasi belum bisa mengambil data langsung dari meteran listrik, tetapi hanya dapat membaca data penggunaan listrik dari benda elektronik yang terhubung ke stop kontak yang telah dipasang alat IoT.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk merancang dan mengembangkan sebuah aplikasi untuk memantau penggunaan listrik prabayar yang berbasis Android yang dapat menampilkan informasi mengenai penggunaan listrik secara *real time* serta dapat mengingatkan pengguna dengan cara memberikan notifikasi saat saldo listrik mencapai batas yang ditentukan pengguna dan pada saat saldo habis.

1.5 Manfaat Penelitian

Manfaat dari pengembangan Aplikasi Android Untuk Pemantauan Listrik adalah:

- a. Aplikasi hasil dari penelitian ini dapat memudahkan pengguna dalam memantau penggunaan listrik prabayar.
- b. Aplikasi dapat mengingatkan pengguna ketika saldo listrik mencapai kondisi tertentu dengan cara memberikan notifikasi pada *smartphone* pengguna. Fitur notifikasi ini juga dapat membantu pengguna dalam melakukan penghematan dalam penggunaan listrik.

1.6 Sistematika Penelitian

Sistematika penulisan laporan dari penelitian ini dijelaskan pada pembahasan dibawah ini:

BAB I PENDAHULUAN

Bab ini berisi tentang gambaran umum dari permasalahan yang menjadi dasar dilakukannya penelitian. Bab ini terdiri dari pembahasan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penelitian.

BAB II LANDASAN TEORI

Bab ini menjelaskan teori yang digunakan dalam penelitian berupa konsep aplikasi mobile, tinjauan metode untuk pengembangan aplikasi mobile, tinjauan penerapan aplikasi mobile untuk penelitian sejenis, Android Studio, Firebase Realtime Database, Activity Diagram, User Acceptance Test, dan Internet of Things.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metode pengembangan, analisis kebutuhan sistem, alur proses sistem, dan perancangan sistem.

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang hasil yang telah didapat dari pengembangan sistem berupa penjelasan tampilan, dan pengujian sistem.

BAB V KESIMPULAN DAN SARAN

Bab ini membahas kesimpulan dari hasil penelitian dan saran untuk mengembangkan sistem agar menjadi lebih baik lagi.

BAB II

LANDASAN TEORI

2.1 Aplikasi *Mobile*

Aplikasi *mobile* terdiri dari kata “aplikasi” dan “*mobile*”. Aplikasi adalah suatu program di dalam perangkat elektronik seperti komputer atau handphone, yang digunakan untuk melakukan suatu pekerjaan yang diperintahkan oleh pengguna (N. K. Dewi et al., 2021), sedangkan *mobile* berarti mudah berpindah atau dipindahkan (Wahyudi, 2019). Dari penjelasan tersebut, dapat disimpulkan bahwa definisi aplikasi *mobile* adalah perangkat lunak yang berjalan di perangkat elektronik yang memiliki sistem operasi, yang bersifat portabel atau mudah untuk dipindahkan dari satu tempat ke tempat lain, contohnya adalah *smartphone*, tablet, dan *smartwatch*. Aplikasi *mobile* terdiri dari beberapa jenis, yang memiliki kekurangan dan kelebihan masing-masing.

2.2 Tinjauan Metode Pengembangan Untuk Aplikasi *Mobile*

Dalam pengembangan sistem pada perangkat *mobile*, terdapat beberapa metode pengembangan. Metode tersebut akan dibahas pada pembahasan dibawah.

a. Waterfall

Waterfall adalah jenis model pengembangan yang dilakukan dengan pendekatan sistematis, yang diawali dengan tahap analisis kebutuhan, dilanjutkan dengan perancangan sistem, pembuatan kode sistem, setelah itu pengujian, dan terakhir yaitu pemeliharaan sistem. Setiap tahap pengembangannya harus selesai terlebih dahulu dan dilakukan secara berurutan dari awal hingga akhir. Jika digambarkan, urutan tahapan pengembangan pada model ini berbentuk berbentuk menurun dari atas kebawah. Oleh karena itu, metode pengembangan ini disebut dengan Waterfall, yang berarti air terjun (Adminlp2m, 2022). Metode pengembangan jenis ini digunakan karena aplikasi yang dibuat tidak berskala besar dan tidak membutuhkan terlalu banyak orang dalam pengembangannya.

Berikut adalah beberapa penelitian yang menggunakan metode pengembangan waterfall:

Pada penelitian berjudul “Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android Studi Kasus: CV. Amanah Kalimantan Rent (Wahyudi, 2019)”, sistem yang dibuat adalah sebuah aplikasi berbasis Android yang berfungsi untuk memudahkan

pelanggan dalam penyewaan mobil dengan menyediakan informasi mengenai mobil sewaan yang tersedia. Urutan pengembangan aplikasi dari penelitian ini adalah analisis kebutuhan, analisis sistem, perancangan, implementasi, dan pengujian. Dalam pengembangan aplikasinya, digunakan metode pengembangan dengan jenis Waterfall. Metode dianggap tepat untuk digunakan karena aplikasi yang dibangun yang tidak terlalu berskala besar dan memiliki sumber daya yang terbatas.

Pada penelitian berjudul “Penerapan Metode Waterfall Pada Pengembangan Aplikasi Sistem Informasi Jasa Dan Penjualan Dengan Pemodelan Berorientasi Objek (Maulani, 2020)”, dikembangkan sebuah aplikasi sistem informasi berbasis Android yang berfungsi untuk mengintegrasikan transaksi jasa dan penjualan barang pada toko bernama Butik Husna. Tahapan pengembangan pada penelitian ini adalah analisis prosedur sistem, perancangan alur sistem menggunakan *Use Case*, *Activity Diagram*, *Sequence Diagram*, serta mockup rancangan aplikasi, dan tahap terakhirnya adalah implementasi.

Pada penelitian berjudul “Pengembangan Aplikasi COSYCALSHIP Berbasis Android untuk Pengelolaan Beasiswa Menggunakan Metode Waterfall (Widianto et al., 2020)”, dikembangkan sebuah aplikasi berbasis Android yang berfungsi untuk mencari tahu informasi seputar beasiswa seperti Lembaga yang menyediakan beasiswa, tips dan trik untuk mendapatkan beasiswa, serta klasifikasi dan pendataan terhadap penerima beasiswa. Tahapan pengembangan pada penelitian ini adalah, pertama adalah tahap identifikasi awal, yang terdiri dari studi Pustaka, identifikasi dan perumusan masalah, dan pencapaian tujuan dan manfaat penelitian, kedua adalah tahap pengembangan aplikasi yang terdiri dari analisis, desain, pengkodean, pengujian, yang ketiga adalah tahap analisis yang terdiri dari analisis tata cara penggunaan aplikasi, dan analisis manfaat aplikasi, dan yang terakhir adalah tahap penarikan kesimpulan dan saran. Metode ini dipilih karena memiliki pendekatan yang sistematis dan berurutan.

b. Agile

Metode pengembangan aplikasi berdasarkan pengerjaan yang dilakukan secara berulang dengan aturan yang sudah disepakati oleh tim pengembang, dikerjakan serempak dan terstruktur. Metode pengembangan ini memiliki jangka waktu pengerjaan yang pendek dan membutuhkan tingkat adaptasi yang tinggi oleh tim pengembang karena rawan terjadi perubahan pada aplikasi yang sedang dikembangkan yang disebabkan oleh permintaan klien.

Pada metode ini terdapat 4 *core value*. Pertama, interaksi antara pengembang dan klien lebih menjadi prioritas dibandingkan proses, karena kesuksesan pengembangan perangkat lunak sangat dipengaruhi oleh kolaborasi antara pengembang dan klien, dan membutuhkan komunikasi yang teratur antara dua belah pihak untuk mencapai hasil yang maksimal. Kedua, pengembangan software lebih diprioritaskan dibanding dokumentasi pengembangan, karena fokus dari metode ini adalah selesainya pengembangan software itu sendiri. Ketiga, kolaborasi dengan klien lebih penting dibanding kontrak awal. Produk software hasil pengembangan mungkin memiliki sedikit perbedaan dari apa yang ada di kontrak awal karena seiring pengerjaan, pengembang akan menyesuaikan produk dengan keinginan klien. Keempat, menanggapi perubahan karena permintaan klien lebih penting dari mengikuti rencana awal. Tim pengembang akan selalu diminta oleh klien untuk merevisi produknya sampai produk tersebut sesuai dengan keinginan klien, oleh karena itu, tim pengembang harus selalu siap untuk melakukan perubahan mengikuti keinginan klien (Rony Setiawan, 2021). Berikut adalah beberapa penelitian yang menggunakan metode pengembangan Agile:

Pada penelitian berjudul “Aplikasi Pembelajaran Sejarah Filsafat Berbasis Android Menggunakan Metode Pengembangan Agile (Sugara et al., 2017)”, dikembangkan sebuah aplikasi Android untuk pembelajaran sejarah filsafat yang mampu menampilkan audio visual agar pembelajaran menjadi lebih menarik. Penelitian ini menggunakan metode Agile dengan Tahapan pengembangan pada penelitian ini adalah *Rational Unified Process* (RUP). Metode RUP terdiri dari 4 tahap, yang pertama adalah *Inception*, tahap pengidentifikasian mengenai sistem apa yang akan dikembangkan, mencakup analisis kebutuhan, dan pemodelan proses bisnis menggunakan *Unified Modelling Language* (UML) berupa *Use Case*, *Activity Diagram*, dan *Class Diagram*. Kedua adalah *Elaboration*, tahap dimana pengembang melakukan desain sistem berdasarkan hasil analisis yang didapat pada tahap sebelumnya. Ketiga tahap *Construction*, yang merupakan tahap implementasi dan pengujian aplikasi hasil desain pada tahap sebelumnya, dan yang terakhir adalah *Transition*, tahap dimana aplikasi hasil pengembangan diuji, disebar, dan diajarkan cara pengoperasiannya kepada pengguna.

Pada penelitian berjudul “Pengembangan Aplikasi Augmented Reality "e-Museum" dengan Metode Agile untuk Meningkatkan Pengalaman Pengunjung Museum (Gemala Jondya et al., 2022)”, dikembangkan sebuah aplikasi berbasis Android yang dapat menampilkan informasi dan model 3D senjata-senjata bersejarah yang terdapat pada

Museum Fatahilah menggunakan teknologi *Augmented Reality*. Tahapan pengembangan pada penelitian ini adalah perumusan ide, pengumpulan data melalui observasi dan kuesioner, perancangan sistem, perancangan tampilan, dan terakhir *deploy & review*. Metode pengembangan Agile dipilih pada penelitian ini karena metode ini mengutamakan kolaborasi antara pengembang dan pengguna.

Pada penelitian berjudul "Perancangan Aplikasi "Nugas" Menggunakan Metode Design Thinking Dan Agile Development (Fahrudin & Ilyasa, 2021)", dikembangkan sebuah aplikasi berbasis Android untuk membantu mahasiswa dalam mengelola dan mengerjakan tugas kuliah. Perancangan *user interface* pada aplikasi ini menggunakan metode Design Thinking. Tahapan pengembangan pada penelitian ini adalah; *Meet & Planning*, dimana klien dan tim pengembang membahas kebutuhan pengembangan, yang kedua *Design*, dimana tampilan dirancang dan setelah itu dievaluasi oleh klien, yang ketiga *Code & Test*, dimana hasil design pada proses sebelumnya diimplementasikan menjadi sebuah produk asli lalu diuji oleh tim pengembang, keempat adalah *Release*, dimana klien mengevaluasi produk hasil pengembangan, dan terakhir adalah *Feedback*, dimana klien akan menyampaikan hasil evaluasi dan menentukan apakah produk perlu dikembangkan lagi atau sudah memuaskan. Metode pengembangan Agile digunakan pada penelitian ini untuk dapat merespon permintaan perubahan oleh klien dengan cepat sehingga juga akan mempercepat penyelesaian produk aplikasi.

c. Scrum

Metode pengembangan Scrum pada dasarnya adalah sebuah *framework* dari metode pengembangan Agile. Ada 4 fase dalam pengembangan menggunakan metode Scrum yaitu, yang pertama *Sprint*, yakni inti dari Scrum, yang merupakan sebuah fase untuk menyelesaikan satu produk yang dapat dinyatakan selesai, kedua *Sprint Planning*, yakni fase perencanaan untuk menentukan *Sprint* yang akan dijalankan dan juga untuk menentukan produk yang akan dihasilkan, ketiga *Daily Scrum*, yang merupakan aktivitas diskusi antar tim pengembang yang dilakukan setiap hari selama kurang lebih 15 menit untuk mengoptimalkan dan menginspeksi pekerjaan apa saja yang telah dilakukan oleh masing-masing anggota tim, keempat *Sprint Review*, yaitu aktivitas yang dilakukan pada akhir *Sprint*, untuk mengecek hasil pekerjaan secara keseluruhan, dan terakhir adalah *Sprint Retrospective*, sebuah kegiatan untuk merencanakan perbaikan yang akan dilakukan pada *Sprint* berikutnya (Rohatun, 2020). Berikut adalah beberapa penelitian yang menggunakan metode pengembangan Scrum:

Pada penelitian berjudul “Pengembangan Aplikasi Event Management System Berbasis Android Menggunakan Metode Scrum (Studi Kasus Organisasi Mahasiswa UPN Veteran Jakarta) (E. Dewi et al., 2021)”, dikembangkan sebuah aplikasi Android jenis *event management system* untuk memudahkan melakukan kegiatan publikasi dan pendaftaran untuk acara yang diselenggarakan oleh organisasi mahasiswa di universitas yang bersangkutan. Metode pengembangan aplikasi yang digunakan pada penelitian ini adalah metode Agile dengan *framework* Scrum. Ada 4 fase dalam pengembangan menggunakan *framework* Scrum yaitu, yang pertama *Sprint*, yakni inti dari Scrum, yang merupakan sebuah fase untuk menyelesaikan satu produk yang dapat dinyatakan selesai, kedua *Sprint Planning*, yakni fase perencanaan untuk menentukan *Sprint* yang akan dijalankan dan juga untuk menentukan produk yang akan dihasilkan, ketiga *Daily Scrum*, yang merupakan aktivitas diskusi antar tim pengembang yang dilakukan setiap hari selama kurang lebih 15 menit untuk mengoptimalkan dan menginspeksi pekerjaan apa saja yang telah dilakukan oleh masing-masing anggota tim, keempat *Sprint Review*, yaitu aktivitas yang dilakukan pada akhir *Sprint*, untuk mengecek hasil pekerjaan secara keseluruhan, dan terakhir adalah *Sprint Retrospective*, sebuah kegiatan untuk merencanakan perbaikan yang akan dilakukan pada *Sprint* berikutnya (Rohatun, 2020). Pada penelitian ini, metode Scrum dipilih karena metode ini dianggap cocok untuk pengembangan perangkat lunak jangka pendek yang memerlukan adaptabilitas.

Pada penelitian berjudul “Aplikasi Self Service Menu Menggunakan Metode Scrum Berbasis Android (Case Study: Warkobar Café Cikarang) (Prastio & Ani, 2018)”, dikembangkan sebuah aplikasi berbasis Android yang memungkinkan pelanggan Warkobar Café Cikarang untuk melakukan pemesanan menu makanan dan minuman secara mandiri melalui aplikasi tersebut. Tahapan pengembangan pada penelitian ini adalah, pertama, *Product Backlog*, yaitu pembuatan daftar kebutuhan yang didapat dari proses pengumpulan data, berupa observasi, wawancara, dan studi literatur. Kedua, *Sprint Backlog*, yaitu pembagian pekerjaan yang akan dikerjakan oleh pengembang pada fase *Sprint* nantinya. Durasi dari tiap *Sprint* berlangsung antara 1-4 minggu tergantung kesepakatan tim pengembang. Ketiga, *Sprint Planning*, yaitu perencanaan pengerjaan dari kegiatan *Sprint* berupa pemetaan kebutuhan menggunakan diagram seperti UML. Durasi pengerjaan dari tiap *Sprint* juga ditentukan pada tahap ini. Keempat, *Sprint*, yaitu pengerjaan tiap fitur yang sudah direncanakan pada tahap sebelumnya mengikuti waktu pengerjaan yang telah ditentukan pada tahap *Sprint Planning*. Tahapan pada proses ini

adalah *in-progress*, *review*, *testing*, dan *done*. Tahap terakhir pada pengembangan di penelitian ini adalah *Review* dan *Retrospective*, yaitu pengujian aplikasi oleh calon pengguna yang berfungsi untuk mencari tahu apakah terdapat kesalahan dan kekurangan pada aplikasi tersebut. Setelah itu, pada tahap *Retrospective*, feedback dari calon pengguna akan direview oleh tim pengembang, dimana hasilnya nanti akan menentukan apakah perlu dilakukan *Sprint* lanjutan atau tidak.

Pada penelitian berjudul "Implementasi Scrum Dalam Pengembangan Sistem Informasi Jasa Desain Grafis (Mutawali et al., 2020)", dikembangkan sebuah website sistem informasi untuk melakukan pemesanan jasa desain grafis untuk masyarakat di daerah Lombok Tengah. Tahap pengembangan pada penelitian ini terdiri dari 3 tahapan utama yaitu fase *Pre-Game*, *Game*, dan *Post-Game*. Pada tahap *Pre-Game*, dilakukan pembuatan *product backlog* dan pembagian pekerjaan anggota tim. Selanjutnya, pada tahap *game*, dilakukan pengembangan website mengikuti rencana yang telah dibuat pada tahap sebelumnya. Fase ini dilakukan sebanyak 4 kali dengan waktu pengerjaan 1 minggu. Pada tahap *Post-Game*, dilakukan pengujian terhadap setiap fitur pada sistem untuk menentukan apakah sistem sudah sesuai seperti yang direncanakan atau belum. Pada penelitian ini, metode pengembangan Scrum dipilih karena dianggap dapat memudahkan komunikasi antar semua pihak yang terkait dengan pengembangan dan memperlancar jalannya proyek.

d. Prototype

Metode ini mengutamakan interaksi antara pengembang dan klien dalam proses pengembangan prototype sistem sehingga kebutuhan klien akan tergambar dengan jelas. Klien akan memberikan feedback terhadap prototype yang sedang dibuat dan akan dimodifikasi mengikuti keinginan klien seiring pengerjaannya (Prasetyo, 2017).

Metode ini memiliki 6 tahap dalam pengembangannya. Tahap pertama adalah pengumpulan kebutuhan, pada tahap ini, klien dan pengembang mendiskusikan serta mendefinisikan perangkat lunak seperti apa yang ingin dibuat nanti. Tahap kedua adalah pembuatan prototype. Pada tahap ini, pengembang merancang sebuah tampilan sementara untuk menggambarkan hasil akhir perangkat lunak nantinya. Tahap ketiga adalah evaluasi prototype. Prototype yang telah dibuat akan diberikan kepada klien untuk dievaluasi kesesuaiannya dengan keinginan klien. Tahap keempat adalah pembuatan kode. Pada tahap ini, pengembang melakukan pembuatan kode mengikuti desain prototype yang telah dibuat pada tahap sebelumnya. Tahap keenam adalah pengujian. Perangkat lunak yang

telah dibuat diuji sesuai proses pengujian yang telah ditentukan. Tahap terakhir adalah evaluasi perangkat lunak. Klien akan melakukan evaluasi terhadap perangkat lunak yang telah dibuat dan menentukan apakah sudah sesuai yang diharapkan atau belum. Jika sudah sesuai, maka perangkat lunak akan difinalisasi dan siap digunakan oleh klien, jika belum, pengerjaan perangkat lunak akan kembali kepada tahap sebelumnya (Ersandi Billah, 2019). Berikut adalah beberapa penelitian yang menggunakan metode pengembangan Prototype:

Pada penelitian berjudul “Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis Website pada Credit Union Canaga Antutn (Yoko et al., 2019)”, dikembangkan sebuah aplikasi sistem informasi berbasis website untuk melakukan kegiatan simpan pinjam. Tahapan pengembangan pada penelitian ini adalah pertama, pengumpulan data untuk analisis kebutuhan, yang dilakukan dengan cara wawancara, observasi dan studi pustaka. Kedua, perancangan *Logical Record Structure* (LRS) dan *use case diagram*. Ketiga, perancangan *prototype* berdasarkan diagram yang telah dibuat pada tahap sebelumnya, yang setelah itu akan dievaluasi oleh pengguna, jika sudah sesuai, maka pengembangan akan masuk ke tahap selanjutnya. Keempat, hasil *prototype* pada tahap sebelumnya akan dikembangkan menjadi *software* produk akhir. Metode pengembangan prototype digunakan pada penelitian ini untuk memudahkan mendapatkan representasi bentuk *software* yang akan dikembangkan nantinya.

Pada penelitian “Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode *Prototype* (Aditya et al., 2021)”, dikembangkan sebuah aplikasi berbasis Android untuk melakukan monitoring kehadiran anggota/pekerja menggunakan *GPS* untuk melacak lokasi perangkat yang terinstall aplikasi ini. Tahapan pengembangan pada penelitian ini adalah, pertama, *Communication*, dimana komunikasi antara calon pengguna dan tim pengembang dilakukan untuk mendefinisikan kebutuhan, kedua, *Quick plan*, yaitu tahap perencanaan kebutuhan, ketiga, *Modelling Quick Design*, yaitu tahap untuk pembuatan desain *prototype*, keempat, pembentukan *prototype*, yaitu tahap dimana *prototype* dibuat sekaligus pengujian dan penyempurnaan, Kelima, *Deployment Delivery & Feedback*, yaitu tahap dilakukannya evaluasi *prototype* dan pembuatan produk akhir yang dapat digunakan oleh pengguna berdasarkan *prototype* yang telah dibuat.

Pada penelitian “Implementasi Aplikasi Monitoring Nilai dan Kegiatan Siswa Berbasis Android dengan Metode Prototype (Juniawan et al., 2021)”, dikembangkan sebuah aplikasi untuk memonitor nilai siswa dan untuk melakukan kegiatan siswa seperti

presensi secara online, mendaftar ekstrakurikuler, dan sebagainya. Tahap pengembangan pada penelitian ini terdiri dari, pertama, pengumpulan data dengan melakukan studi literatur dan wawancara, kedua, penggunaan *Universal Modelling Language* (UML) berupa *Activity Diagram*, *Use Case Diagram* dan *Class Diagram*. Ketiga, perancangan prototype berdasarkan hasil pemodelan UML yang telah dibuat pada tahap sebelumnya. Setelah prototype selesai, prototype akan diuji dan dievaluasi oleh pengguna dan jika masih belum sesuai dengan keinginan, pengerjaan prototype akan diulang hingga sesuai dengan keinginan pengguna. Keempat, aplikasi akan dibangun berdasarkan prototype yang telah dibuat pada tahap sebelumnya dan setelah itu diuji kinerjanya dengan metode pengujian black box. Metode pengembangan prototype dipilih pada penelitian ini karena dapat dikembangkan dengan cepat untuk mendapatkan *feedback* dari pengguna, dalam kasus ini adalah pihak sekolah.

e. RAD

Pada metode pengembangan ini, pengembangan dilakukan berdasarkan pembuatan prototype dan iterasi pengembangan berdasarkan feedback klien secara berulang hingga mencapai hasil yang diinginkan. Tahap-tahap yang terdapat pada metode pengembangan ini adalah, Pertama, menentukan kebutuhan proyek. Di tahap ini, tim pengembang dan klien akan mendiskusikan dan mendefinisikan kebutuhan serta visi produk perangkat lunak yang diinginkan. Kedua, pembuatan prototype. Pada tahap ini, tim pengembang membangun prototype untuk perangkat lunak. Semua fungsi dan komponen dikerjakan secara bersamaan dan akan terus dikonsultasikan kepada klien untuk mendapatkan feedback. Tahap ketiga adalah *Rapid Construction* dan feedback. Di tahap ini dilakukan pembuatan kode perangkat lunak, pengujian, dan evaluasi dari klien. Jika belum sesuai dengan keinginan klien, maka proses akan kembali kepada tahap sebelumnya. Tahap terakhir adalah implementasi dan finalisasi. Pada tahap ini, dilakukan pengujian terakhir yang meliputi *Usability Testing*, perbaikan *interface*, dan pengujian stabilitas perangkat lunak. Jika sudah sesuai, produk akan diserahkan kepada klien untuk digunakan (Junaedi, 2022). Berikut adalah beberapa penelitian yang menggunakan metode pengembangan RAD:

Pada penelitian berjudul “Implementasi Rapid Application Development (RAD) Model Pada Pengembangan Aplikasi Rent Car Berbasis Android (Gibran, 2018)”, dikembangkan sebuah aplikasi untuk memudahkan pengguna dalam melakukan rental mobil dalam segi pencarian lokasi penyedia rental serta untuk melakukan rental mobil.

Metode pengembangan yang digunakan pada penelitian ini adalah Rapid Application Development (RAD). Tahap pengembangan pada penelitian ini adalah, pertama, *Requirements Planning*, untuk melakukan identifikasi terhadap apa saja yang dibutuhkan serta menentukan tujuan pembuatan aplikasi itu sendiri. Kedua, *Design Workshop*, dimana pengembang melakukan proses desain berdasarkan hasil survey kepada pengguna dan penyedia jasa rental mengenai fitur apa saja yang diinginkan dan untuk memenuhi fungsionalitas aplikasi. Alur aplikasi yang dibuat pada proses ini digambarkan menggunakan diagram *Use Case*, *Activity Diagram*, dan *Sequence Diagram*. Terakhir adalah proses implementasi, dimana rancangan desain aplikasi pada tahap sebelumnya dibangun menjadi sebuah aplikasi. Pada tahap ini, terdapat 3 proses iterasi pengembangan yang dilakukan hingga memperoleh aplikasi yang diinginkan. Pada penelitian ini, metode pengembangan RAD dipilih karena dianggap lebih cepat dan berkualitas tinggi.

Pada penelitian berjudul “Implementasi *Rapid Application Development* Dalam Pengembangan Aplikasi Pelaporan Kerusakan Jalan (Suyanto & Andri, 2020)”, dikembangkan sebuah aplikasi berbasis Android untuk menyampaikan laporan mengenai kerusakan jalan, yang akan dioperasikan oleh masyarakat, dan aplikasi berbasis *web* untuk mengelola hasil laporan dari masyarakat, yang akan dioperasikan oleh pemerintah. Tahap pengembangan dari penelitian ini adalah; pertama, perencanaan. Pada tahap ini, dilakukan pendefinisian kebutuhan pengembangan. Kedua, *Workshop* Desain RAD. Pada tahap ini, dibuat sebuah model menggunakan UML berupa *Use Case Diagram*, *Behavior Diagram*, dan *Interaction Diagram* untuk dijadikan acuan dalam pengembangan aplikasi nantinya. Ketiga, Implementasi. Pada tahap ini, aplikasi dikembangkan berdasarkan model yang telah dibuat pada tahap sebelumnya. Metode pengembangan RAD digunakan pada penelitian ini karena metode ini mengedepankan kecepatan dalam proses pengembangannya dan dapat selesai dalam waktu yang relatif singkat.

Pada penelitian berjudul “Pembangunan Sistem Informasi Perpustakaan Berbasis Web Menggunakan Metode Rapid Application Development (RAD) (Studi pada : SMK Negeri 11 Malang) (Suyanto & Andri, 2020)” dikembangkan sebuah *website* sistem informasi untuk peminjaman dan pengelolaan buku perpustakaan. Tahap pengembangan pada penelitian ini adalah; pertama, *Requirement Planning*, yaitu tahap dilakukannya analisis kebutuhan sistem yang terdiri dari kegiatan studi literatur, observasi, dan wawancara. Kedua, *Workshop* Desain RAD, yaitu tahap pengerjaan iterasi pengembangan yang terdiri dari 2 iterasi yang menghasilkan *Use Case Diagram*, *Class Diagram*, dan

rancangan antarmuka. Ketiga, Implementasi, yaitu tahap dibangunnya sistem berdasarkan perancangan yang dilakukan pada tahap sebelumnya. Pada tahap ini juga dilakukan pengujian sistem menggunakan *Black Box Testing* dan *User Acceptance Testing (UAT)*. Metode pengembangan RAD digunakan pada penelitian ini karena dianggap dapat mempersingkat waktu pengembangan dibandingkan metode tradisional.

Dari tinjauan penelitian diatas, tiap metode pengembangan memiliki kelebihan dan kekurangan masing-masing yang akan dijelaskan pada tabel 2.1 dibawah:

Tabel 2.1 Perbandingan kelebihan dan kekurangan masing-masing metode pengembangan

No.	Metode Pengembangan	Kelebihan	Kekurangan
1.	Waterfall	<ul style="list-style-type: none"> - Tahapan proses pengembangan teratur dan mudah diaplikasikan - Cocok digunakan untuk pengembangan software yang sudah memiliki kebutuhan yang jelas - Pembuatan dokumen pengembangan akan lebih mudah dan rapi karena setiap tahapan sudah terdefinisi jelas sejak awal (Priscilla & Zulfachmi, 2021) 	<ul style="list-style-type: none"> - Tahapan pengembangan tidak fleksibel karena tidak memungkinkan proses pengembangan kembali ke tahap sebelumnya, yang berarti jika ada perubahan kebutuhan, proses pengembangan harus diulang dari awal. - Jika terdapat kekurangan pada tahap planning dan design akan memakan waktu yang lama karena metode ini hamper tidak memiliki toleransi kesalahan - Pengerjaan proyek yang melibatkan banyak anggota akan menjadi tidak efisien karena harus menunggu penyelesaian tahap sebelumnya (Priscilla & Zulfachmi, 2021)
2	Agile	<ul style="list-style-type: none"> - Pengembang dapat melakukan perubahan berdasarkan permintaan klien - Permintaan klien akan cepat diimplementasikan 	<ul style="list-style-type: none"> - Tidak cocok untuk tim pengembang berskala besar - Rawan terjadi perubahan dalam pengembangannya

		<ul style="list-style-type: none"> - Proses pengembangan yang fleksibel (cias.co, 2021). 	<ul style="list-style-type: none"> - Susah dalam melakukan dokumentasi pengembangan (Adani, 2020).
3	Scrum	<ul style="list-style-type: none"> - Dapat mengatasi perubahan kebutuhan dengan mudah - Pengukuran produktivitas lebih mudah dilakukan dengan <i>Sprint meeting</i> - (Sameen Mirza & Datta, 2019) - Memprioritaskan kepuasan klien dengan penyelesaian produk tahap awal dengan cepat (Esa et al., 2021). 	
4	Prototype	<ul style="list-style-type: none"> - Penyesuaian produk dengan keinginan klien akan lebih mudah - Waktu pengembangan lebih singkat - Pendefinisian kebutuhan lebih mudah karena partisipasi aktif dari klien (Priscilla & Zulfachmi, 2021) 	<ul style="list-style-type: none"> - Setiap masukan yang menyebabkan perubahan terhadap prototype akan mengharuskan programmer untuk menyesuaikan aplikasi, dimana hal ini akan menambah beban programmer - Proses analisis terlalu singkat - Karena singkatnya waktu pengembangan, programmer kemungkinan besar akan melakukan kompromi terhadap kualitas (Priscilla & Zulfachmi, 2021)
5	Rapid Application Development (RAD)	<ul style="list-style-type: none"> - Cocok digunakan untuk proyek yang hanya memerlukan waktu singkat - Lebih efektif dari metode pengembangan sekuensial seperti <i>waterfall</i> karena langsung memenuhi kebutuhan pelanggan 	<ul style="list-style-type: none"> - Rawan terjadi perubahan pada kebutuhan aplikasi - Memerlukan tim pengembang dengan skill yang mumpuni - Perubahan yang terjadi ditengah-tengah proses pengembangan mengharuskan pembuatan kontrak baru

		- Pengembangan lebih singkat dan efisien (Priscilla & Zulfachmi, 2021)	dengan klien (Priscilla & Zulfachmi, 2021)
--	--	--	--

Dari tinjauan penelitian mengenai metode pengembangan aplikasi diatas, dilakukan pertimbangan dalam menentukan metode apa yang akan digunakan pada penelitian ini. Dan didapat kesimpulan bahwa metode yang paling tepat untuk digunakan pada penelitian ini adalah metode *Waterfall* karena aplikasi yang dikembangkan hanya berskala kecil, tidak membutuhkan tim dalam pengembangannya dan kebutuhan sudah terdefinisi dengan jelas sejak awal.

2.3 Tinjauan Penerapan Aplikasi *Mobile* Untuk Penelitian Sejenis

Terdapat sejumlah penelitian terdahulu yang mengangkat topik sejenis, yaitu penelitian terkait pemantauan listrik berbasis Android. Berikut ini adalah tinjauan mengenai beberapa penelitian sebelumnya:

Pertama, penelitian yang berjudul “Rancang Bangun Sistem Monitoring dan Kendali Daya Listrik pada Rumah Kos Menggunakan NodeMCU dan Firebase Berbasis Android (Furqon et al., 2019)”, membahas perancangan dan pengembangan sebuah sistem manajemen penggunaan listrik yang dapat memutus aliran listrik sekaligus untuk memantau penggunaan listrik pada kamar kos. Pada penelitian ini, fokus dari penelitian adalah perancangan hardware untuk monitoring listrik dan data hasil pembacaan akan ditampilkan pada sebuah aplikasi Android. Data yang ditampilkan berupa data penggunaan arus, daya, tegangan listrik, energi, dan tagihan listrik per kamar. Database yang digunakan untuk penyimpanan data adalah Firebase Realtime Database. Pada penelitian ini, Firebase dipilih sebagai database karena dapat diakses dari jauh. Hal ini memungkinkan pemantauan listrik untuk dilakukan di mana saja selama ada koneksi internet.

Kedua, pada penelitian dengan judul “Perancangan Sistem Monitoring Dan Kontroling Penggunaan Daya Listrik Berbasis Android (Mitha Djaksana et al., 2020)”, dikembangkan sebuah sistem pemantauan listrik berbasis Android dengan fitur untuk mengontrol aliran listrik dengan telegram chatbot sebagai tool untuk mengonfigurasi sistem. Sistem yang dikembangkan dapat memutus aliran listrik ke benda elektronik yang terhubung ke prototype sistem. Data listrik ditampilkan pada sebuah aplikasi Android. Untuk data yang ditampilkan, aplikasi menampilkan data tegangan, arus, daya, energi serta tarif listrik. Database yang

digunakan adalah Firebase Realtime Database karena dapat digunakan di berbagai platform seperti Android dan iOS.

Ketiga, penelitian dengan judul “Rancang Bangun Alat Monitoring Daya Listrik Berbasis IoT Menggunakan Firebase (William et al., 2022)” Sistem yang dikembangkan pada penelitian ini berfungsi untuk memantau penggunaan listrik. Data listrik ditampilkan melalui aplikasi Android. Data yang ditampilkan pada aplikasi berupa tegangan, arus, daya, energi dan estimasi biaya pemakaian. Pada abstrak disebutkan bahwa aplikasi tersebut akan memiliki fitur untuk mengingatkan tentang sisa saldo, tetapi pada perancangan dan hasil, fitur ini tidak ditampilkan dan tidak dibahas sama sekali. Firebase Realtime Database digunakan sebagai database aplikasi karena database ini bersifat real time, yang merupakan kebutuhan wajib untuk pemantauan data listrik yang selalu berubah dari waktu ke waktu.

Keempat, pada penelitian berjudul “Prototipe Sistem Monitoring Biaya Penggunaan Listrik Pada Rumah Kos Berbasis IoT (Hamamni et al., 2021)”, sistem yang dikembangkan berfungsi untuk monitoring penggunaan listrik pada kamar kos serta dapat memutus sambungan listrik menggunakan sebuah komponen hardware bernama *Relay Shield*. Data ditampilkan melalui aplikasi Blynk, sebuah aplikasi yang dapat secara simpel menghubungkan sebuah proyek IoT yang kodenya dibuat pada Arduino IDE agar dapat dioperasikan melalui aplikasinya sendiri. tersedia pada Android. Data yang ditampilkan berupa daya, estimasi biaya, dan saldo. Tidak ada penggunaan database eksternal pada sistem ini karena hal ini telah disediakan oleh Blynk dan terkonfigurasi secara otomatis.

Kelima, pada penelitian berjudul “Pengembangan Aplikasi Monitoring Pemakaian Daya Listrik Rumah Development of Home Electricity Monitoring Application (Simangunsong & Handayani, 2020), dikembangkan sebuah aplikasi berbasis Android untuk memantau penggunaan listrik rumah. Aplikasi dibuat menggunakan sebuah software bernama MIT App Inventor, sebuah aplikasi web yang dapat secara otomatis membuat aplikasi tanpa harus melakukan coding. Data yang ditampilkan pada aplikasi ini berupa daya dan arus listrik. Aplikasi ini dapat menampilkan statistik penggunaan harian, mingguan, serta bulanan. Untuk menggunakan aplikasi, pengguna harus melakukan login terlebih dahulu. Database yang digunakan pada aplikasi ini adalah MySQL.

Dari penelitian-penelitian yang telah disebutkan diatas, didapatkan acuan mengenai komponen-komponen dan fitur apa saja yang akan digunakan dan diimplementasikan untuk aplikasi hasil dari penelitian ini.

Aplikasi hasil penelitian ini akan menggunakan Firebase Realtime Database sebagai *database* karena bersifat *real time* dan integrasinya dengan Arduino IDE dan Android Studio yang memudahkan untuk melakukan pembuatan kode pada perangkat-perangkat IoT Arduino dan pembuatan kode untuk aplikasi Android. Aplikasi akan mengimplementasikan fitur berupa tampilan hasil pembacaan data listrik berupa tegangan, arus, daya, harga, saldo digunakan, dan saldo tersisa, fitur menginput saldo yang akan berkurang seiring penggunaan, fitur reset saldo untuk mengulang kembali perhitungan saldo dari awal sesuai saldo awal yang telah diinput, dan fitur pengingat saldo untuk mengingatkan pengguna jika saldo mencapai batas yang ditentukan oleh pengguna. Fitur-fitur tersebut dipilih untuk diimplementasikan karena menurut tinjauan dari penelitian-penelitian diatas, fitur-fitur tersebut belum ditemukan, dan keberadaannya akan membantu pengguna dalam melakukan pemantauan listrik.

2.4 Android Studio

Android Studio adalah IDE yang dikhususkan untuk pengembangan aplikasi Android yang diluncurkan pada 16 Mei 2013 pada acara Google I/O 2013. Android Studio berisi tools lengkap untuk mendesain, menguji, men-debug, dan memprofilkan aplikasi Android. Android Studio menggunakan Gradle untuk pengelolaan proyek. Gradle adalah sebuah toolkit yang mengelola dependensi software serta memungkinkan pengembang untuk membuat logika build sendiri untuk proyek mereka (Prithviraj Biswas, 2021).

2.5 Firebase Realtime Database

Firebase Realtime Database adalah sebuah layanan *cloud database* yang dibuat oleh Google. Data disimpan dalam format JSON dan disinkronkan secara *real time* ke aplikasi yang terhubung. Keuntungan dari hal ini adalah memudahkan pengelolaan *database* dengan skala besar. Semua pengguna pada seluruh platform akan menerima perubahan yang terjadi pada *database* secara *real time* dan otomatis (Dicoding, 2020).

2.6 Activity Diagram

Activity Diagram adalah bagian dari diagram *Universal Modelling Language* (UML) yang menggambarkan rancangan aktivitas yang terjadi antara pengguna dan sistem atau dapat juga didefinisikan sebagai aliran kerja sebuah sistem. *Activity Diagram* juga dapat digunakan untuk mengelompokkan aliran tampilan dari sistem. Diagram ini memiliki komponen dengan bermacam-macam bentuk yang dihubungkan dengan tanda panah. Panah tersebut

menunjukkan alur urutan aktivitas yang terjadi dari awal sistem dijalankan hingga akhir (Rizky, 2019).

2.7 Black-box Testing

Black-box Testing adalah sebuah metode pengujian perangkat lunak yang digunakan untuk menguji perangkat lunak tanpa mengetahui kode dari program tersebut.

Pengujian didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh customer. Pengujian ini meninjau fungsi-fungsi dari aplikasi, apakah sudah sesuai dengan yang seharusnya atau belum. Struktur dan fungsi internalnya tidak dipelajari. Dengan demikian penguji perlu mengetahui apa saja yang dapat dilakukan sistem, dan fungsinya, tetapi tidak tahu cara kerjanya secara internal.

Pengujian ini bertujuan untuk memeriksa, setelah tahap akhir proyek, apakah perangkat lunak atau aplikasi berfungsi dengan baik, serta dapat mengatasi input penggunaannya secara efisien. Penguji menyiapkan skenario berdasarkan jalur pengguna yang berbeda yang dapat diikuti pada sistem yang akan diuji. Pengujian ini dapat dilakukan oleh pengguna, tim marketing bahkan *user* sebagai kusionernya.

2.8 User Acceptance Test (UAT)

UAT adalah aktivitas pengujian yang bertujuan untuk mengetahui apakah produk yang telah dikembangkan sesuai dengan kebutuhan user. Tujuan utama dari dilakukannya UAT adalah untuk memastikan bahwa pengguna merasa puas saat menggunakan produk tersebut dan dapat menyelesaikan masalah utama dari pengguna. Yang membedakan pengujian ini dengan pengujian lainnya adalah pada pengujian ini, yang melakukan pengujian terhadap sistem adalah pengguna akhir, bukan tim pengembang (Simbolon, 2021).

2.9 Internet of Things

Istilah *Internet of Things* (IoT) umumnya mengacu pada keadaan ketika sebuah objek yang biasanya tidak terhubung ke jaringan dapat terhubung ke jaringan internet, memungkinkan objek-objek ini menghasilkan data maupun berinteraksi satu sama lain dengan campur tangan minimal dari manusia (Singhania, n.d.). Definisi lain dari IoT adalah jaringan komprehensif antar objek-objek cerdas yang memiliki kemampuan untuk mengatur, berbagi informasi, data dan sumber daya secara otomatis, serta mampu bereaksi dan bertindak secara adaptif dalam menghadapi situasi dan perubahan lingkungan (Madakam et al., 2015). Tujuan dari IoT adalah

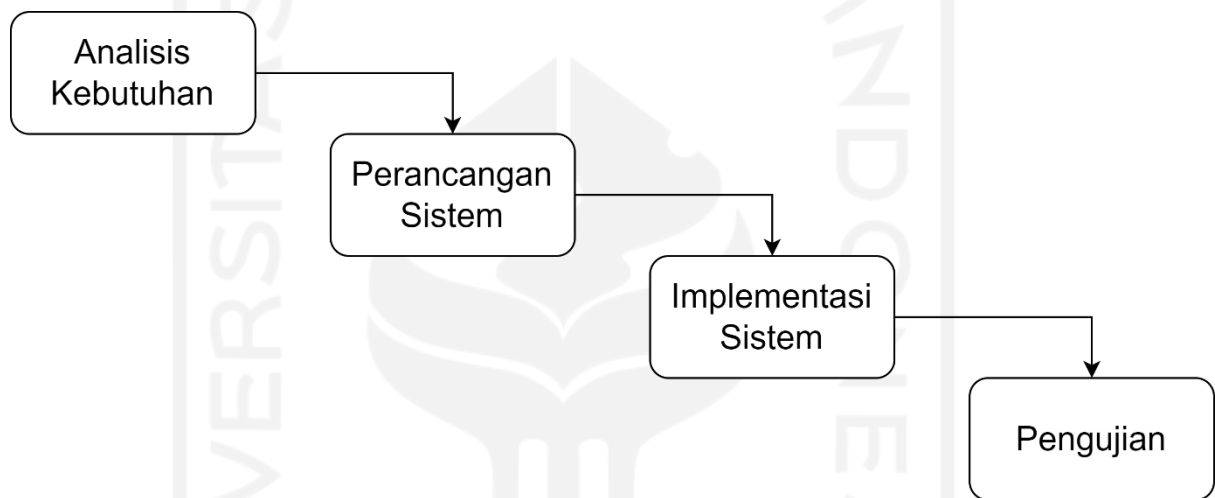
untuk meningkatkan fungsi dari jaringan internet untuk berbagai macam alat untuk mempermudah manusia dalam kegiatan sehari-hari.

IoT sendiri terbentuk dari beberapa komponen dasar, yaitu (Amera P. Safira, 2021), *Artificial Intelligence* — komponen ini berfungsi sebagai “otak” sebuah perangkat. Teknologi AI ini bekerja sesuai dengan fungsi dari perangkat itu sendiri, *Konektivitas* — Tanpa konektivitas, komponen IoT tidak dapat terhubung dengan jaringan yang menyebabkan sistem tidak berjalan sebagaimana mestinya, *Sensor* — Sensor bekerja untuk mengumpulkan data dari lingkungan sekitar. Sensor juga sering disebut sebagai ‘detektor’ karena fungsi utamanya adalah mendeteksi perubahan sekecil apa pun yang terjadi di lingkungan sekitarnya, *Active engagement* — *engagement* yang diterapkan oleh teknologi pada umumnya adalah tipe *engagement* pasif. IoT menerapkan keterlibatan secara aktif dalam berbagai produk, dan layanan yang tersedia, *Perangkat berukuran kecil* — di masa yang semakin modern ini, banyak perangkat berukuran kecil yang mampu melakukan banyak hal, salah satunya adalah benda elektronik yang sering kita gunakan sehari-hari yaitu *smartphone*.

BAB III METODOLOGI PENELITIAN

3.1 Metode Pengembangan Sistem

Dalam pengembangan sistem pada penelitian Aplikasi Android Untuk Pemantauan Listrik Prabayar, metode pengembangan yang digunakan adalah metode Waterfall. Tahapan pengembangan sistem digambarkan pada Gambar 3.1 dibawah ini



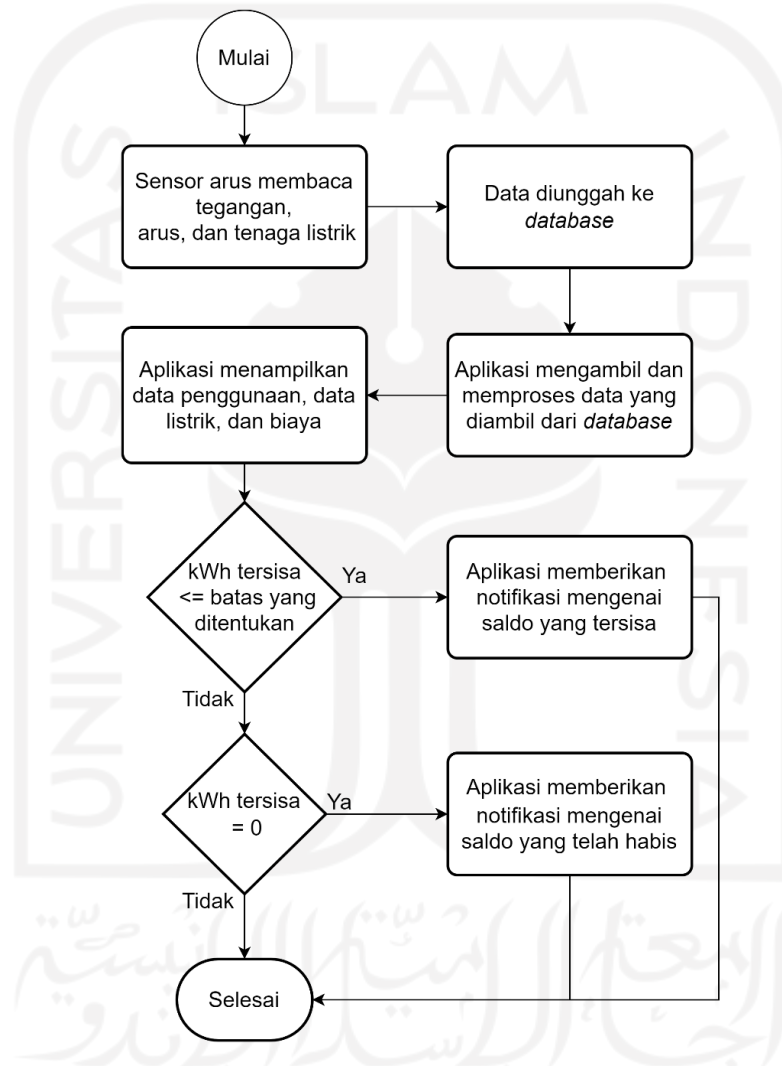
Gambar 3.1 Tahapan pengembangan dengan metode Waterfall

Berikut adalah penjelasan mengenai tahapan pengembangan pada Gambar 3.1:

- a. Analisis kebutuhan. Pada tahap ini, dilakukan analisis untuk menemukan kebutuhan apa saja yang diperlukan dalam pengembangan Aplikasi Android Untuk Pemantauan Listrik Prabayar.
- b. Perancangan sistem. Pada tahap ini, sistem mulai dikembangkan yang dimulai dari perancangan alur proses sistem, arsitektur sistem, dan perancangan desain sistem.
- c. Implementasi sistem. Pada tahap ini, dilakukan pembuatan kode untuk membuat aplikasi berdasarkan perancangan yang telah dibuat sebelumnya pada tahap perancangan sistem.
- d. Pengujian sistem. Pada tahap ini, dilakukan pengujian terhadap fungsionalitas sistem apakah sudah bekerja sesuai yang diharapkan atau tidak.

3.2 Alur Proses Sistem

Aplikasi pada penelitian ini dibuat dengan tujuan untuk memantau penggunaan listrik, menampilkan harga dari jumlah kWh yang telah digunakan, dan mengingatkan pengguna dengan cara memberikan notifikasi ketika saldo listrik mencapai batas yang ditentukan oleh pengguna. Sistem ini memiliki proses yang akan dijelaskan pada Gambar 3.2 dan pembahasan di bawah.



Gambar 3.2 Alur Proses Utama Sistem

Seperti yang tercantum dalam Gambar 3.2, Pada proses pertama, dilakukan pembacaan terhadap data arus, tegangan, dan daya listrik dari alat elektronik yang telah dicolok ke terminal stopkontak yang sudah dihubungkan dengan sensor arus PZEM-004T. Sensor ini terhubung dengan modul ESP32 yang menggunakan daya dari komputer. Pembuatan kode untuk menjalankan proses ini dilakukan pada *software* Arduino IDE. Kebutuhan pada proses ini

adalah sensor arus PZEM-004T, modul ESP32, terminal stopkontak, komputer, dan Arduino IDE.

Untuk proses selanjutnya, modul ESP32 mengunggah data ke *database* Firebase Realtime Database. Pembuatan kode untuk menjalankan proses ini dilakukan pada *software* Arduino IDE. Kebutuhan pada proses ini adalah modul ESP32 dan *software* Firebase Realtime Database.

Setelah itu, aplikasi mengambil dan memproses data dari *database* untuk ditampilkan pada antarmuka aplikasi serta melakukan perhitungan untuk biaya dari kWh yang telah digunakan dan sisa saldo. Kode untuk menjalankan proses ini dibuat menggunakan *software* Android Studio. Kebutuhan pada proses ini adalah Android Studio dan Firebase Realtime Database.

Pada proses berikutnya, aplikasi menampilkan data yang telah diambil dari database berupa tegangan, arus, daya, saldo terpakai (kWh), sisa saldo (kWh), dan total harga berdasarkan penggunaan kWh secara *real time*. Kode untuk menjalankan proses ini dibuat menggunakan *software* Android Studio. Kebutuhan pada proses ini adalah Android Studio dan Firebase Realtime Database.

Selanjutnya, jika saldo kWh mencapai batas yang telah ditentukan, aplikasi akan mengirimkan notifikasi kepada pengguna yang memberitahukan jumlah saldo listrik yang tersisa. Kode untuk menjalankan proses ini dibuat menggunakan *software* Android Studio. Kebutuhan pada proses ini adalah Android Studio.

Terakhir, jika saldo kWh = 0, aplikasi akan mengirimkan notifikasi yang memberitahukan pengguna bahwa saldo telah habis. Kode untuk menjalankan proses ini dibuat menggunakan *software* Android Studio. Kebutuhan pada proses ini adalah Android Studio.

3.3 Analisis Kebutuhan

Berdasarkan hasil analisis yang dilakukan dari proses di atas, didapatkan kebutuhan-kebutuhan berikut untuk mengembangkan sistem.

3.3.1 Kebutuhan *Hardware*

Kebutuhan *hardware* dari sistem adalah sebagai berikut:

- a. Mikrokontroler ESP32
- b. Sensor arus PZEM-004T
- c. Terminal Stopkontak
- d. Komputer

3.3.2 Kebutuhan *Output*

Kebutuhan *output* dari sistem adalah sebagai berikut:

- a. Tampilan mengenai data listrik.
- b. Notifikasi mengenai sisa saldo listrik.
- c. Notifikasi mengenai saldo listrik habis.

3.3.3 Kebutuhan *Software*

Kebutuhan *software* dari sistem adalah sebagai berikut:

- a. Arduino IDE
- b. Android Studio
- c. Firebase Realtime Database

3.3.4 Kebutuhan Proses Sistem

Kebutuhan proses dari sistem adalah sebagai berikut:

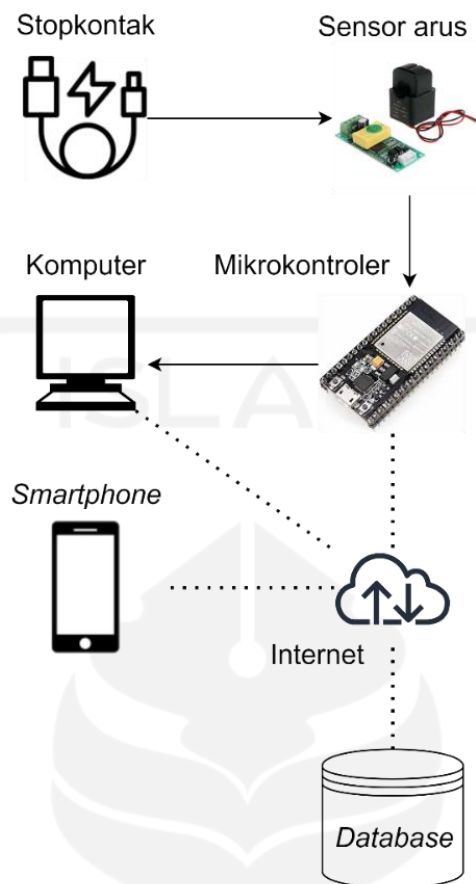
- a. Menampilkan data listrik
- b. Menginput saldo
- c. Menampilkan notifikasi
- d. *Reset* penggunaan kWh

3.4 Perancangan Sistem

Pada penelitian ini, fokus dari penelitian adalah pengembangan aplikasi, bukan pembuatan hardware untuk pemantauan listrik prabayar, Jadi, pada perancangan desain sistem, perancangan yang dilakukan adalah untuk diimplementasikan pada hasil yang berupa sebuah aplikasi. Proses perancangan aplikasi dijelaskan pada pembahasan berikut.

3.4.1 Arsitektur Sistem

Arsitektur sistem digambarkan pada Gambar 3.3 di bawah ini:



Gambar 3.3 Arsitektur sistem

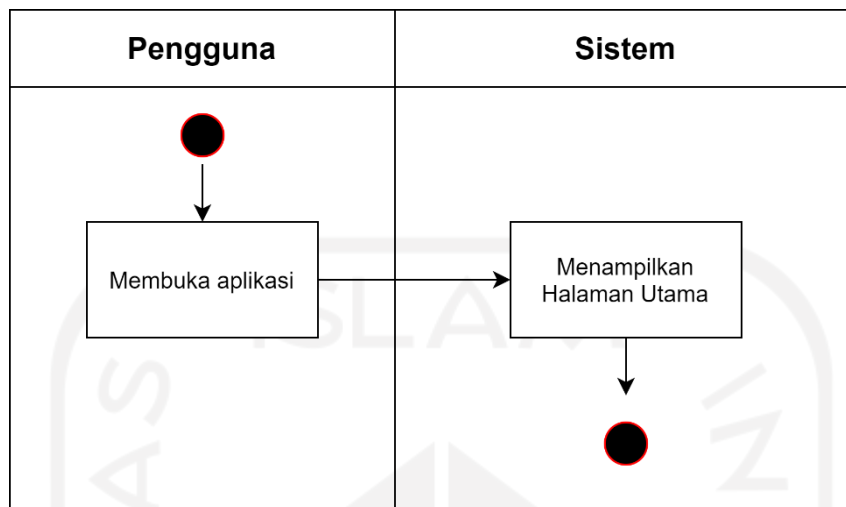
Arsitektur sistem seperti yang ditunjukkan pada Gambar 3.3 akan dijelaskan pada pembahasan di bawah:

- Sensor arus terpasang pada stopkontak, sensor ini membaca data listrik dari alat elektronik yang terhubung.
- Hasil pembacaan data listrik diproses oleh mikrokontroler yang setelah itu mengunggah data ke database melalui internet. Mikrokontroler harus selalu terhubung ke komputer karena komputer bertindak sebagai sumber daya untuk menjalankan mikrokontroler.
- Aplikasi yang sudah terinstal pada *smartphone* mengambil data yang telah diunggah ke database dan menampilkannya di halaman utama aplikasi.

3.4.2 Rancangan Desain Sistem

Perancangan desain sistem dilakukan dengan membuat *Activity Diagram*. diagram tersebut akan dijelaskan pada penjelasan di bawah ini.

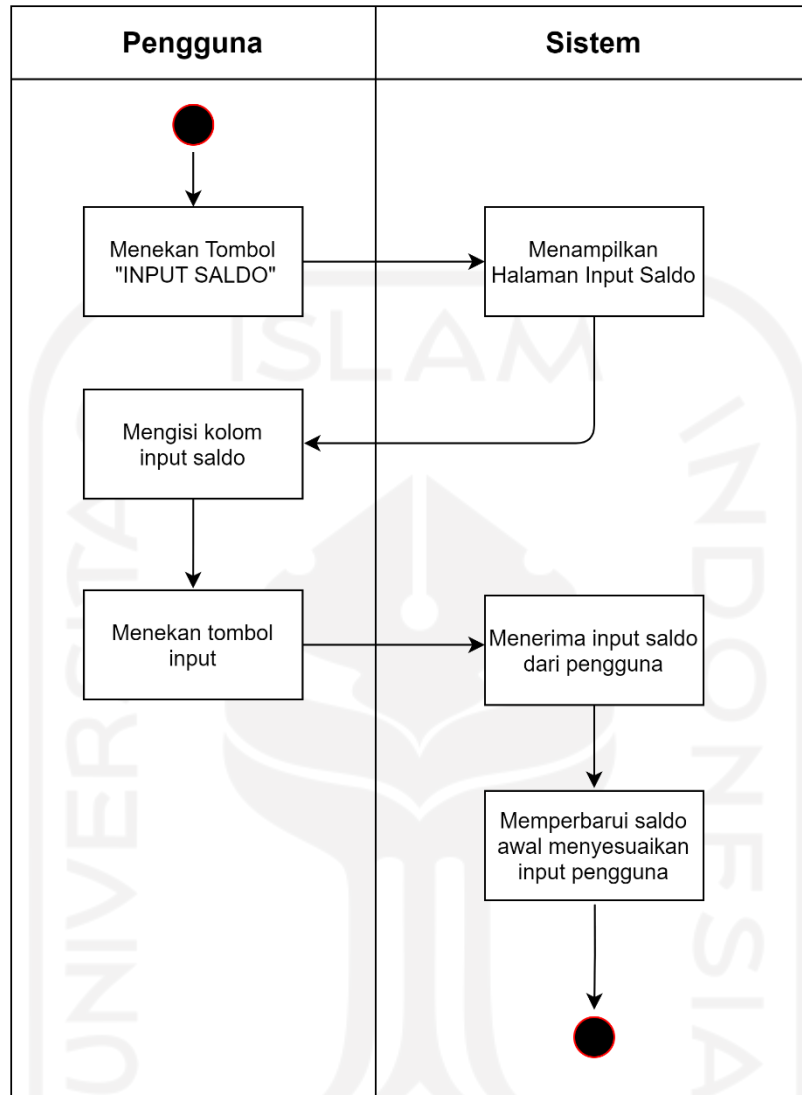
a. Halaman Utama

Gambar 3.4 *Activity Diagram* halaman utama

menampilkan interaksi antara pengguna dan sistem pada halaman utama. Aktivitas yang terjadi antara pengguna dan sistem adalah sebagai berikut:

1. Pengguna membuka aplikasi
2. Sistem menampilkan halaman utama yang menyajikan data pembacaan data listrik
3. Jika saldo kWh tersisa kurang dari batas yang ditentukan, aplikasi akan memberikan notifikasi kepada pengguna yang berisi tentang sisa saldo kWh.
4. Jika saldo kWh mencapai nilai 0, aplikasi akan memberikan notifikasi kepada pengguna yang memberitahukan bahwa saldo telah habis.

b. Halaman Input Saldo

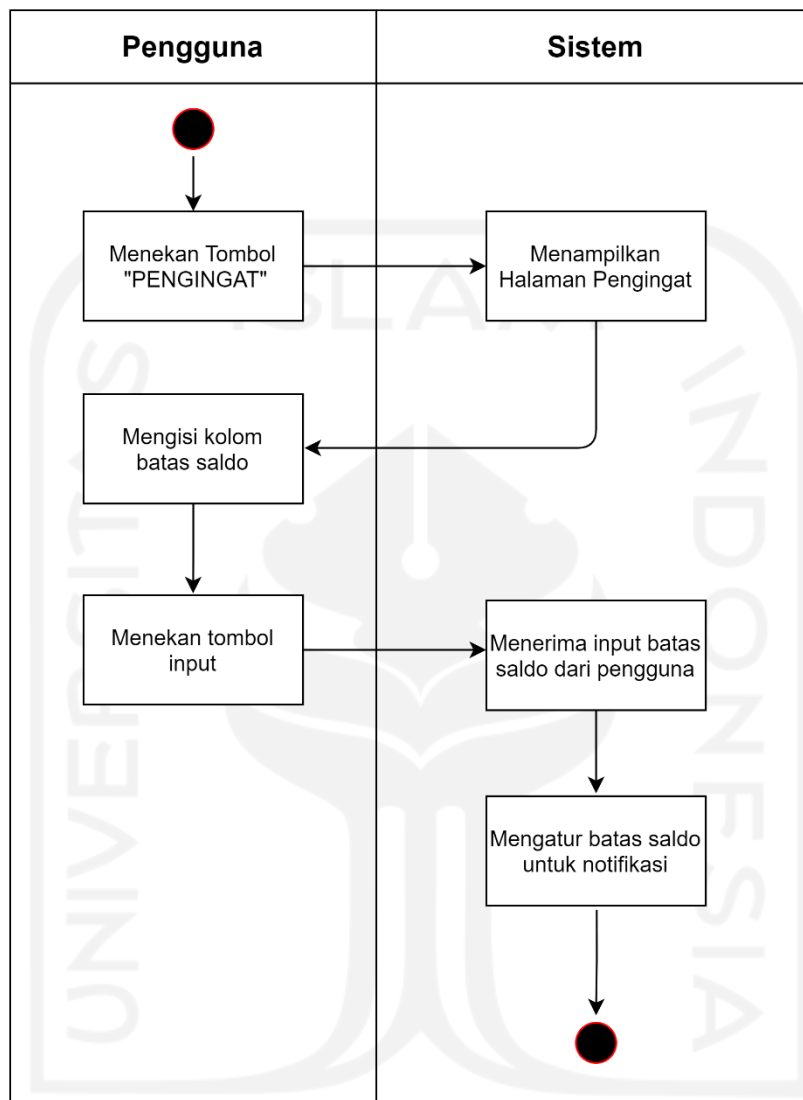


Gambar 3.5 Activity Diagram halaman input saldo

Gambar 3.5 menampilkan interaksi antara pengguna dan sistem pada halaman utama. Aktivitas yang terjadi antara pengguna dan sistem adalah sebagai berikut:

1. Pengguna menekan tombol "INPUT SALDO"
2. Sistem menampilkan halaman Input Saldo yang berfungsi untuk menginput data saldo kWh
3. Pengguna mengisi kolom input saldo yang tersedia pada halaman
4. Pengguna menekan tombol input
5. Sistem menerima dan memproses data saldo yang baru saja diinput oleh pengguna
6. Sistem memperbarui nilai saldo kWh pada aplikasi.

c. Halaman Peningat



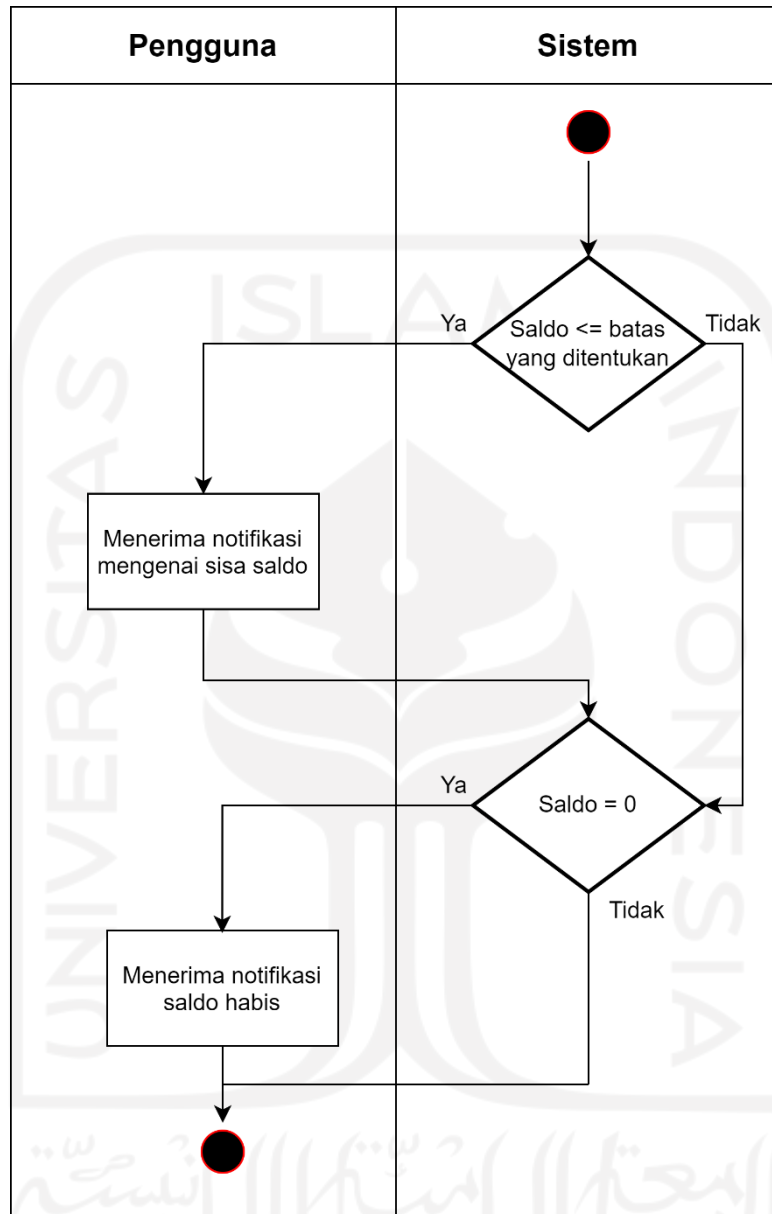
Gambar 3.6 Activity Diagram halaman peringatan

Gambar 3.6 menampilkan interaksi antara pengguna dan sistem pada halaman Peringatan.

Aktivitas yang terjadi antara pengguna dan sistem adalah sebagai berikut:

1. Pengguna menekan tombol "PENGINGAT"
2. Sistem menampilkan halaman Peringatan
3. Pengguna mengisi kolom batas saldo yang tersedia pada halaman
4. Pengguna menekan tombol input
5. Sistem menerima dan memproses data batas saldo yang baru saja diinput oleh pengguna

6. Sistem menjadikan data yang telah diinput pengguna sebagai kondisi notifikasi
- d. Notifikasi



Gambar 3.7 Activity Diagram notifikasi

Gambar 3.8 menampilkan interaksi antara pengguna dan sistem pada fitur notifikasi. Aktivitas yang terjadi antara pengguna dan sistem adalah sebagai berikut:

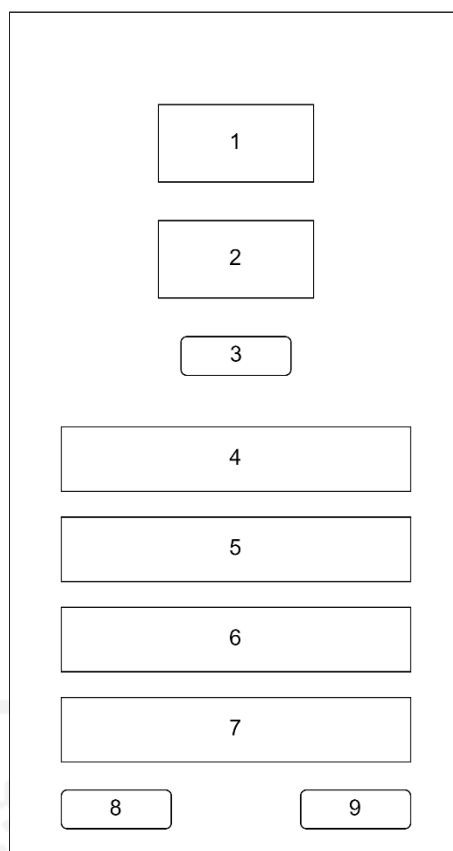
1. Jika saldo kWh tersisa kurang dari batas yang ditentukan, aplikasi akan memberikan notifikasi kepada pengguna yang berisi tentang sisa saldo kWh.
2. Jika saldo kWh mencapai nilai 0, aplikasi akan memberikan notifikasi kepada pengguna yang memberitahukan bahwa saldo telah habis.

3.4.3 Rancangan Tampilan Aplikasi

Perancangan tampilan aplikasi bertujuan untuk menggambarkan bagaimana interface aplikasi yang akan dibuat. Tampilan didesain sepraktis mungkin agar pengguna dapat langsung mengakses kegunaan utama dari aplikasi ini yaitu untuk memantau penggunaan dan data listrik. Berikut adalah rancangan tampilan dari aplikasi.

a. Rancangan halaman utama

Rancangan halaman utama ditunjukkan melalui Gambar 3.9 berikut ini.



Gambar 3.9 Rancangan tampilan halaman utama

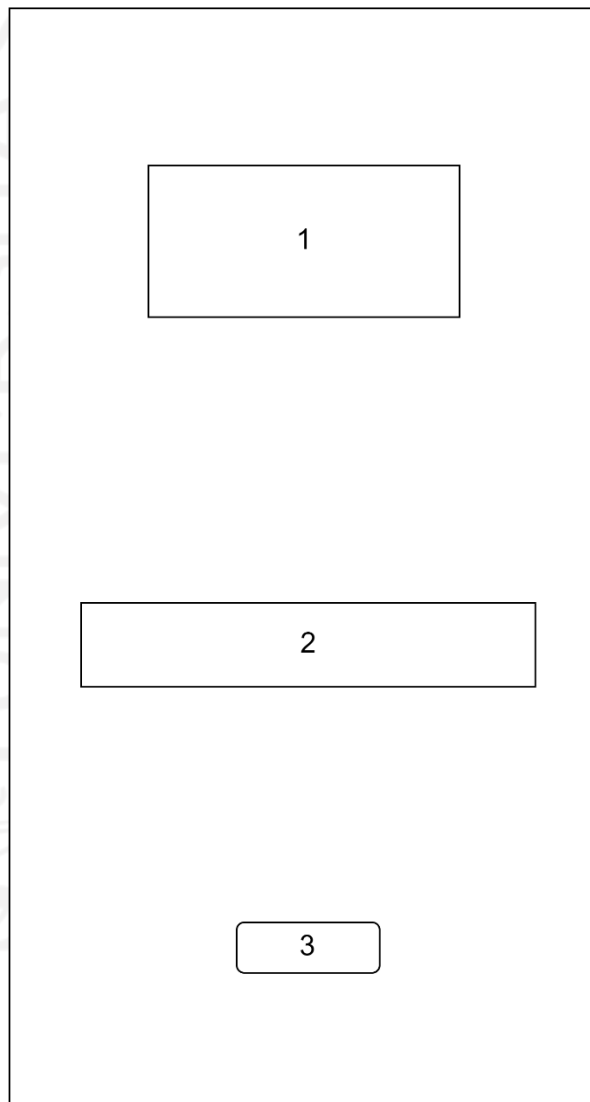
Elemen yang ditampilkan pada halaman utama adalah:

1. Saldo Terpakai (kWh)
2. Sisa Saldo (kWh)
3. Tombol “RESET”
4. Tegangan (V)
5. Arus (A)

6. Daya (Watt)
7. Harga (Rp)
8. Tombol “INPUT SALDO”
9. Tombol “PENGINGAT”

b. Rancangan halaman Input Saldo

Rancangan halaman input saldo ditunjukkan melalui Gambar 3.10 berikut ini.



Gambar 3.10 Rancangan tampilan input saldo

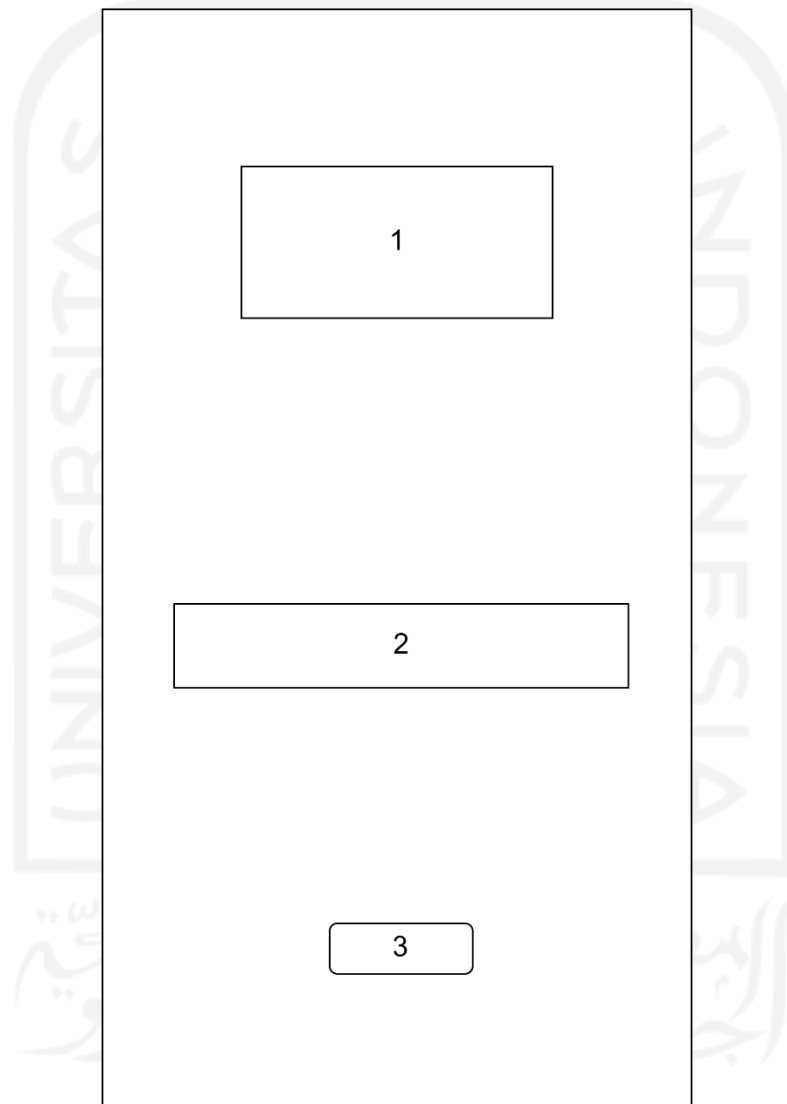
Elemen yang ditampilkan pada halaman input saldo adalah:

1. Teks bertuliskan “INPUT SALDO LISTRIK”

2. Kolom penginputan saldo kWh
3. Tombol “INPUT”

c. Rancangan halaman Peningat

Rancangan halaman pengingat ditunjukkan melalui Gambar 3.11 berikut ini.



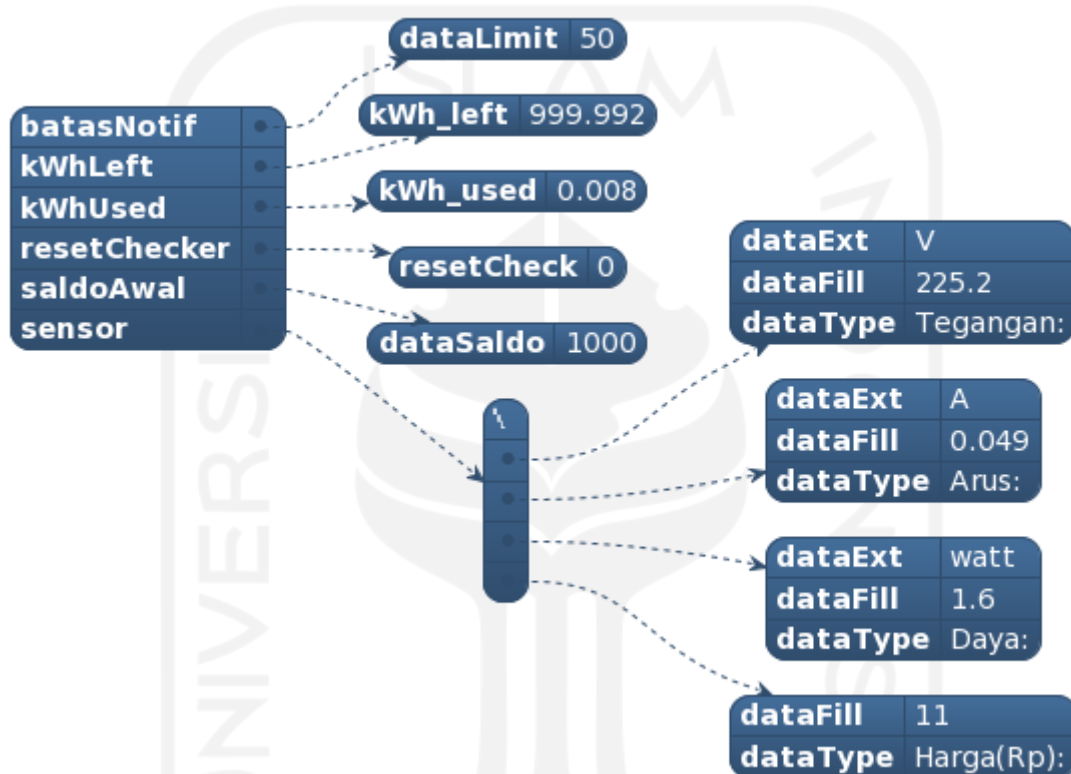
Gambar 3.11 Rancangan tampilan halaman pengingat

Elemen yang ditampilkan pada halaman pengingat adalah:

1. Teks bertuliskan “PENGINGAT SALDO”
2. Kolom penginputan batas saldo
3. Tombol “INPUT”

3.4.4 Rancangan Database

Database yang digunakan untuk penyimpanan data pada sistem ini adalah Firebase Realtime Database. Struktur dari database yang digunakan oleh aplikasi ditampilkan pada Gambar 3.12 di bawah.



Gambar 3.12 Struktur *database* pada *software* Firebase Realtime Database

BAB IV

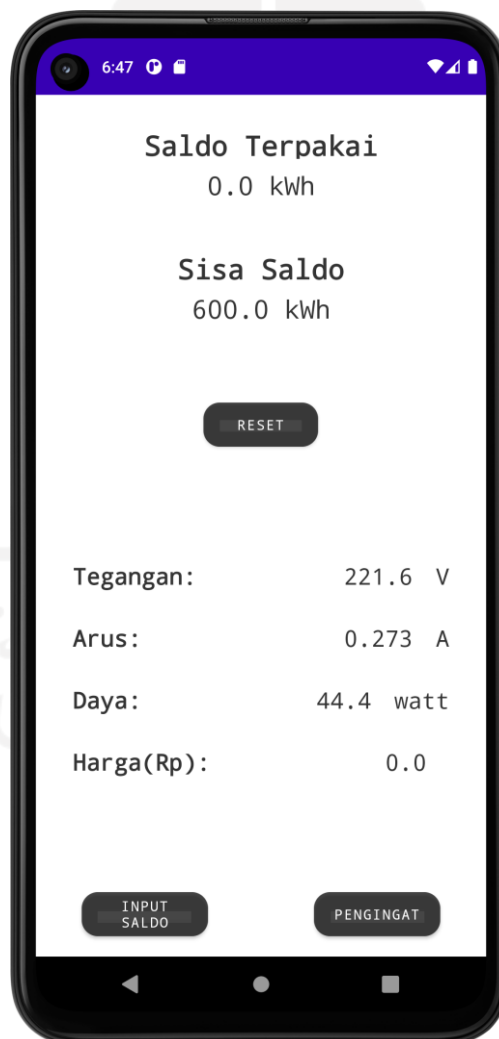
HASIL DAN PEMBAHASAN

4.1 Tampilan Antarmuka

Terdapat empat macam tampilan antarmuka pada aplikasi pemantauan listrik prabayar ini. Empat tampilan tersebut adalah halaman utama, halaman input saldo, halaman pengingat saldo, dan notifikasi.

4.1.1 Halaman utama

Gambar 4.1 di bawah menampilkan tampilan halaman utama pada aplikasi.



Gambar 4.1 Tampilan halaman utama

Seperti yang ditunjukkan pada Gambar 4.1 Tampilan halaman utama, pada halaman utama, data yang ditampilkan adalah “Saldo Terpakai”, “Sisa Saldo”, “Tegangan”, “Arus”, “Daya” dan “Harga (Rp)”. Pada bagian bawah halaman utama, Terdapat dua tombol yaitu tombol “INPUT SALDO” dan tombol “PENGINGAT”. Tombol “INPUT SALDO” berfungsi untuk masuk ke halaman untuk penginputan saldo (Halaman Input Saldo) dan tombol “PENGINGAT” yang berfungsi untuk masuk ke halaman penginputan batas saldo untuk notifikasi sisa saldo (Halaman Peningkat). Pada bagian tengah halaman utama, terdapat tombol “RESET” yang berfungsi untuk mereset data “Saldo Terpakai” dan data “Sisa Saldo”.

Data yang ditampilkan pada halaman utama akan dijelaskan pada pembahasan berikut:

Saldo Terpakai

Informasi yang ditampilkan pada data “Saldo Terpakai” berisi jumlah total penggunaan energi yang telah digunakan oleh pengguna dalam satuan kWh. Pembacaan pada data ini akan terus bertambah seiring penggunaan listrik. Informasi yang ditampilkan pada data ini adalah informasi utama yang dapat digunakan pengguna untuk memantau jumlah energi listrik yang telah ia gunakan.

Sisa Saldo

Informasi yang ditampilkan pada data “Sisa Saldo” berisi jumlah saldo listrik yang masih dimiliki pengguna pada layanan listrik Prabayarannya dalam satuan kWh. Informasi yang muncul pada data ini menyesuaikan dengan total saldo yang telah lebih dulu diinput oleh pengguna. Untuk itu, pengguna harus masuk terlebih dahulu ke halaman “Input Saldo” dan menginput saldo. Setelah saldo diinput, data yang muncul pada “Sisa Saldo” akan otomatis terus berkurang seiring besarnya penggunaan energi listrik.

Tegangan

Informasi yang ditampilkan pada data “Tegangan” berisi hasil pembacaan tegangan dari seluruh alat elektronik yang terhubung ke stopkontak dan ditampilkan dalam satuan Volt (V). Nilai yang ditampilkan pada data ini memberikan informasi kepada pengguna mengenai besarnya tegangan dan dapat digunakan untuk menjadi salah satu informasi yang bisa dikalkulasikan bersama arus untuk memprediksi besar energi yang digunakan dalam satuan kWh.

Arus

Informasi yang ditampilkan pada data “Arus” berisi hasil pembacaan arus dari seluruh alat elektronik yang terhubung ke stopkontak dan ditampilkan dalam satuan Ampere (A). Nilai yang ditampilkan pada data ini memberikan informasi kepada pengguna mengenai besar arus yang mengalir dan dapat digunakan untuk memantau apakah arus tidak melebihi batas maksimal. Data arus juga berguna untuk memprediksi besar konsumsi listrik dalam satuan kWh ketika dikalkulasikan bersama tegangan.

Daya

Informasi yang ditampilkan pada data “Daya” berisi pembacaan daya dari seluruh alat elektronik yang terhubung ke terminal stopkontak. Nilai yang ditampilkan pada data ini dapat digunakan oleh pengguna untuk memantau besarnya daya alat-alat elektronik tertentu serta besarnya daya secara keseluruhan yang berguna untuk memperkirakan besarnya penggunaan listrik dan juga dapat menjadi pertimbangan bagi pengguna untuk menghemat penggunaan listrik sesuai saldo yang tersisa.

Harga

Informasi yang ditampilkan pada data “Harga” berisi perhitungan tarif untuk total kWh kumulatif yang telah digunakan dan ditampilkan dengan satuan mata uang Rupiah (Rp). Informasi dari data harga ini dapat digunakan oleh pengguna untuk memantau banyaknya penggunaan listrik yang telah ia gunakan dalam satuan yang lebih mudah diperhitungkan secara awam yaitu besarnya biaya dalam Rupiah. Rumus perhitungan harga ini dapat dilihat pada Gambar 4.2 di bawah.

$$\begin{aligned} \text{Harga per kWh} &= \text{Rp.1444,70} \\ \text{Harga penggunaan} &= \text{kWh digunakan} \times \text{harga per kWh} \end{aligned}$$

Gambar 4.2 Rumus harga

Harga per kWh di atas didapat berdasarkan tarif per kWh untuk pelanggan golongan tarif R-1 (1300 VA) (Kumparan, 2021). Tarif untuk golongan ini dijadikan patokan karena menurut statistik PLN tahun 2021, pengguna listrik untuk golongan ini adalah yang terbanyak di Indonesia (PLN, 2021).

4.1.2 Halaman Input Saldo

Gambar 4.3 di bawah menampilkan tampilan halaman input saldo pada aplikasi.



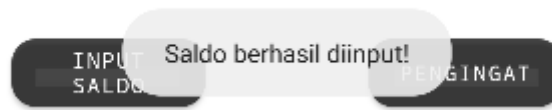
Gambar 4.3 Halaman Input Saldo

Seperti yang ditunjukkan pada Gambar 4.3, fitur ini memungkinkan pengguna untuk menginput saldo pada aplikasi. *Text box* pada halaman ini hanya dapat menerima jenis input berupa angka untuk mengurangi potensi kesalahan input dari pengguna. Contoh penginputan saldo ditunjukkan pada Gambar 4.4.

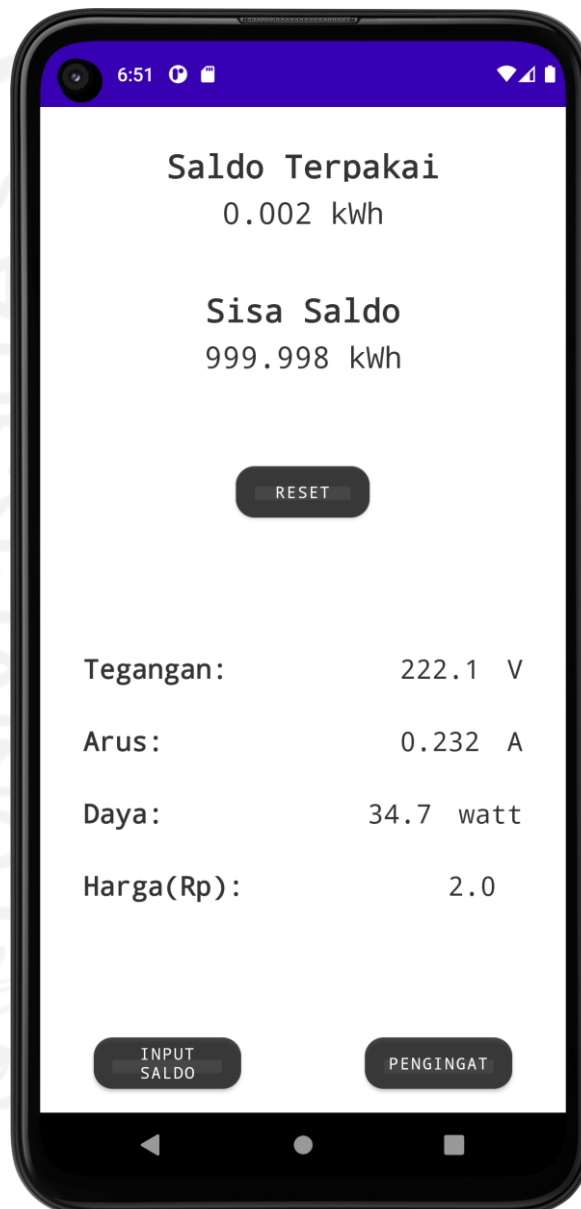


Gambar 4.4 Penginputan Saldo

Setelah pengguna menekan tombol input, akan ada *pop-up* yang muncul yang memberitahukan bahwa input saldo telah berhasil dan pengguna akan langsung dikembalikan ke halaman utama. Tampilan *pop-up* yang muncul saat saldo berhasil diinput ditunjukkan pada Gambar 4.5 dan tampilan saldo tersisa setelah dilakukan input ditunjukkan pada Gambar 4.6.



Gambar 4.5 *Pop-up* input saldo berhasil



Gambar 4.6 Data kWh tersisa setelah dilakukan proses input

4.1.3 Halaman Peningat Saldo

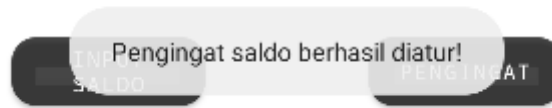
Tampilan Halaman pengingat saldo ditunjukkan pada Gambar 4.7.



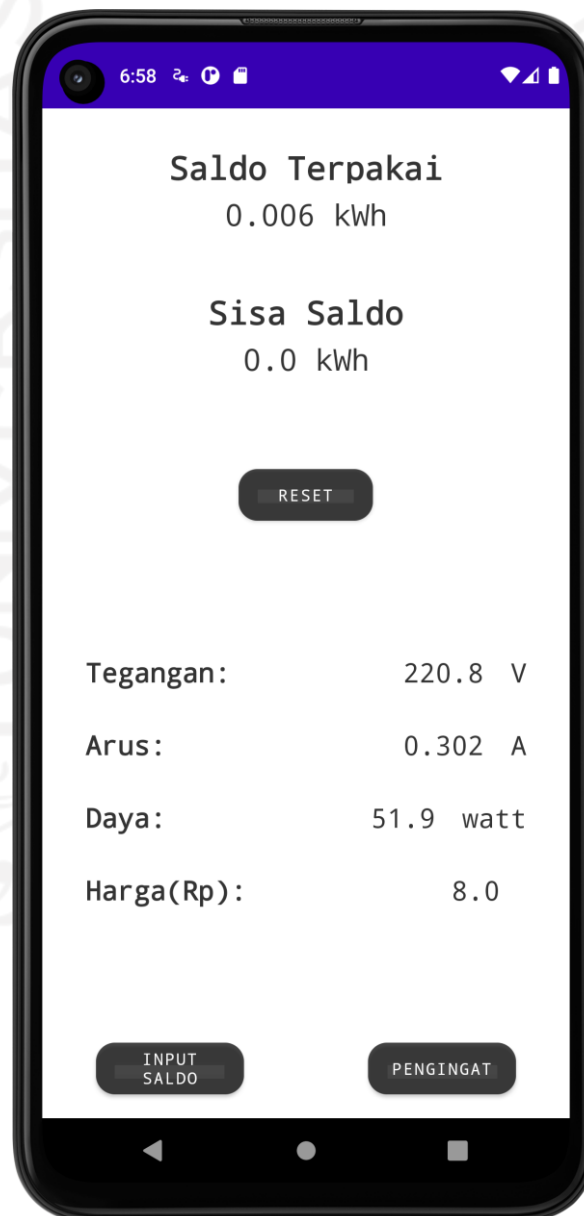
Gambar 4.7 Halaman pengingat saldo

Halaman pengingat saldo berfungsi untuk menginput batas sisa saldo agar aplikasi memberikan notifikasi sisa saldo ke pengguna. Seperti pada halaman input saldo, *text box* pada halaman pengingat juga hanya mendukung penginputan untuk data numerik. Setelah penginputan batas saldo berhasil, akan muncul sebuah *pop-up* yang memberitahukan bahwa

penginputan telah berhasil dan pengguna akan dikembalikan ke halaman utama. Tampilan saat *pop-up* muncul ditunjukkan pada Gambar 4.8.



Gambar 4.8 *Pop-up* input batas saldo berhasil



Gambar 4.9 Tampilan aplikasi saat saldo habis

Pada saat saldo kWh habis atau = 0, data pada “Sisa Saldo” akan tetap bernilai “0” sampai pengguna menginput saldo yang baru. Tampilan saat saldo habis ditunjukkan pada Gambar 4.9.

4.1.4 Notifikasi

Ada dua jenis notifikasi yang dapat diberikan oleh aplikasi, notifikasi tersebut akan ditunjukkan oleh Gambar 4.10 juga dijelaskan pada pembahasan berikut.

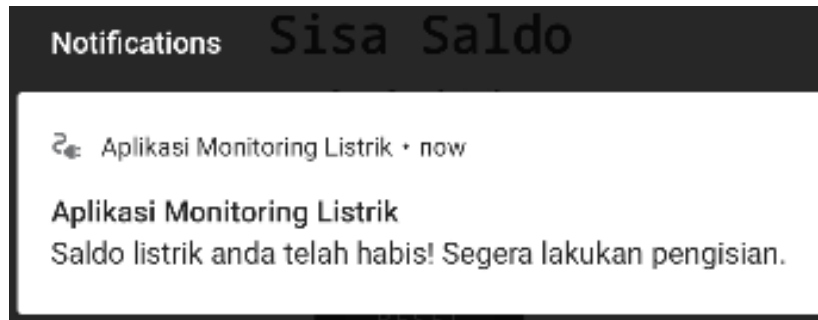
- a. Notifikasi sisa saldo



Gambar 4.10 Notifikasi sisa saldo

Untuk notifikasi sisa saldo yang ditampilkan pada Gambar 4.10, informasi yang ditampilkan adalah berupa sisa kWh. Notifikasi akan muncul jika data “Sisa Saldo” mencapai batas yang telah ditentukan pada halaman pengingat. Batas saldo pada halaman pengingat harus diinput terlebih dahulu, karena jika tidak, notifikasi tidak akan muncul karena kondisinya tidak terpenuhi. Notifikasi ini akan terus muncul seiring konsumsi kWh jika sisa saldo melewati batas yang ditentukan pada fitur pengingat (dengan konsumsi kWh minimum sebesar 0.001 kWh) jika jumlah saldo tidak diperbarui. Notifikasi ini bersifat tidak intrusif (tidak muncul tampilan *pop-up* di layar dan tidak berbunyi, karena jika tidak demikian, pengguna akan terganggu karena seringnya notifikasi tersebut muncul. Hal ini dibuat sedemikian rupa agar pengguna tidak terganggu dengan notifikasi tersebut serta agar tidak lupa untuk mengisi saldo listrik.

b. Notifikasi saldo habis



Gambar 4.11 Notifikasi saldo habis

Notifikasi seperti pada Gambar 4.11 akan diberikan kepada pengguna jika data “Sisa Saldo” = 0. Notifikasi ini bersifat *onGoing*, yang berarti notifikasi ini akan terus ada dan tidak bisa dihapus oleh pengguna selama sisa saldo tidak diperbarui. Jika jumlah saldo sudah tidak sama dengan 0, notifikasi akan terhapus secara otomatis. Notifikasi ini berjenis intrusif (muncul *pop-up* di layar dan mengeluarkan bunyi) dan hanya terjadi 1 kali.

4.2 Pengujian

Pengujian dilakukan untuk mengetahui apakah aplikasi hasil penelitian sudah sesuai dengan yang direncanakan atau tidak. Pengujian yang dilakukan adalah berupa pengujian aplikasi dan pengujian keakuratan pembacaan data listrik.

4.2.1 Pengujian Aplikasi

Pada tahap ini, pengujian dilakukan dengan tujuan untuk mengetahui apakah fitur-fitur pada aplikasi sudah berjalan sebagaimana mestinya atau belum. Metode pengujian yang digunakan untuk menguji aplikasi adalah pengujian *black box*. Setiap kasus pengujian dilakukan 10 kali. Hasil pengujian ini dijelaskan pada Tabel 4.1 di bawah:

Tabel 4.1 Hasil pengujian aplikasi.

Skenario pengujian	Kasus pengujian	Hasil yang diharapkan	Hasil pengujian
Hasil pembacaan data listrik	Membuka dan menutup aplikasi dan mengecek apakah data listrik muncul pada aplikasi	Data berhasil ditampilkan pada antarmuka aplikasi	Berhasil 10/10 kali
Input saldo listrik	Menginput saldo listrik pada halaman input saldo dan mengecek apakah data kWh tersisa pada halaman utama menyesuaikan data yang baru diinput	Data berubah sesuai saldo yang diinput	Berhasil 10/10 kali
Input batas saldo untuk notifikasi	Menginput sebuah nilai untuk batas saldo lalu mengurangi saldo sampai batas yang telah diinput	Notifikasi batas saldo ditampilkan oleh aplikasi yang berarti batas saldo berhasil ditentukan	Berhasil 10/10 kali
Notifikasi pada saat "Sisa Saldo" \leq batas saldo yang ditentukan	Menggunakan fitur input saldo untuk mengatur agar data "Sisa Saldo" \leq saldo yang ditentukan	Notifikasi muncul pada <i>smartphone</i> pengguna	Berhasil 10/10 kali
Notifikasi saat saldo habis	Menggunakan fitur "Input Saldo" agar data saldo awal = "Saldo Terpakai" yang menyebabkan "Sisa Saldo" menjadi = 0	Notifikasi saldo habis muncul pada <i>smartphone</i> pengguna	Berhasil 10/10 kali
Reset penggunaan saldo	Menekan tombol "RESET" pada halaman utama	Data penggunaan kWh ter- <i>reset</i> dan nilai saldo kembali seperti saldo awal sebelum penggunaan	Berhasil 10/10 kali

Berdasarkan hasil pengujian pada Tabel 4.1, didapat kesimpulan bahwa fitur aplikasi bekerja seperti yang diinginkan dengan tingkat keberhasilan 100%.

4.2.2 Pengujian Keakuratan Pembacaan

Pada tahap ini, pengujian dilakukan dengan tujuan untuk mengetes tingkat keakuratan data listrik yang ditampilkan pada aplikasi. Metode pengujian yang digunakan pada tahap ini adalah membandingkan hasil pembacaan pada aplikasi dengan power meter komersial pada saat bersamaan dalam 1 jam. Alat elektronik yang digunakan pada pengujian ini adalah sebuah televisi *LED* dengan ukuran layar 32 inci, mesin cuci bukaan depan, kipas angin, dan pengisi

daya sebuah laptop. Rumus yang digunakan (IndeedEditorialTeam, 2022) untuk menghitung *error%* ditampilkan pada ((4. 1) dibawah.

$$Error\% = \frac{kWh \text{ Power Meter} - kWh \text{ Aplikasi}}{kWh \text{ Power Meter}} \cdot 100\% \quad (4.1)$$

Tabel 4.2 berikut menunjukkan hasil pengujian untuk pembacaan kWh (energi):

Tabel 4.2 Hasil pengujian ke-1 pembacaan kWh

Waktu	kWh pada aplikasi	kWh pada power meter	<i>Error</i> (%)
15 Menit	0,052	0,053	1,88
30 Menit	0,104	0,105	0,95
45 Menit	0,154	0,154	0
60 Menit	0,197	0,197	0

Tabel 4.3 Hasil pengujian ke-2 pembacaan kWh

Waktu	kWh pada aplikasi	kWh pada power meter	<i>Error</i> (%)
15 Menit	0,039	0,039	0
30 Menit	0,080	0,080	0
45 Menit	0,114	0,113	0,88
60 Menit	0,148	0,148	0

Tabel 4.4 Hasil pengujian ke-3 pembacaan kWh

Waktu	kWh pada aplikasi	kWh pada power meter	<i>Error</i> (%)
15 Menit	0,029	0,029	0
30 Menit	0,066	0,066	0
45 Menit	0,102	0,102	0
60 Menit	0,139	0,139	0

Tabel 4.5 *Error%* rata-rata hasil pengujian

Waktu	<i>Error</i> (%) total
Pengujian 1	0,707
Pengujian 2	0,22
Pengujian 3	0
Rata-rata	0,309

Dari data hasil pengujian pada Tabel 4.5 di atas, didapat kesimpulan bahwa hasil perhitungan penggunaan kWh yang ditampilkan pada aplikasi dapat terbilang cukup akurat dengan *error%* rata-rata 0,309%. Tabel 4.6 di bawah adalah tabel hasil pengujian untuk pembacaan tegangan (V), arus (A), dan daya (Watt):

Tabel 4.6 Hasil pengujian ke-1 pembacaan data listrik

Jenis	Data pada aplikasi	Data pada power meter	Error(%)
Tegangan (V)	225,8	225,7	0,044
Arus (A)	0,813	0,814	0,112
Daya (W)	154,8	155,2	0,257

Tabel 4.7 Hasil pengujian ke-2 pembacaan data listrik

Jenis	Data pada aplikasi	Data pada power meter	Error(%)
Tegangan (V)	223,8	224,5	0,311
Arus (A)	1,079	1,068	1,02
Daya (W)	78,6	78,8	0,25

Tabel 4.8 Hasil pengujian ke-3 pembacaan data listrik

Jenis	Data pada aplikasi	Data pada power meter	Error(%)
Tegangan (V)	230,4	228,6	0,787
Arus (A)	1,156	1,144	1,04
Daya (W)	204,1	201,7	1,18

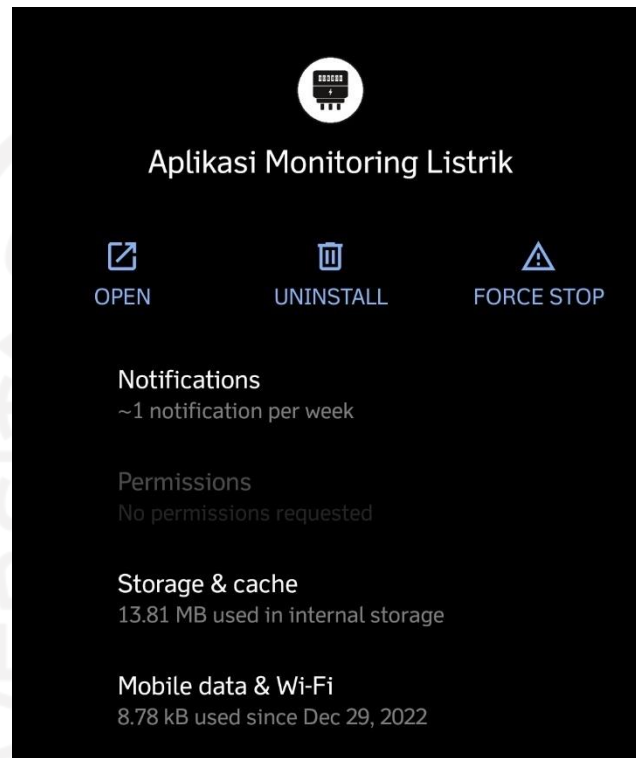
Tabel 4.9 Hasil perhitungan *error%* rata-rata pada pembacaan data listrik

Jenis	Error(%)
Tegangan (V)	0,388
Arus (A)	0,724
Daya (W)	0,562
Rata-rata	0,558

Dari data pada Tabel 4.9, didapat kesimpulan bahwa hasil pembacaan data listrik yang ditampilkan pada aplikasi dapat dianggap cukup akurat, dengan *error%* rata-rata 0,558%.

4.2.3 Pengujian Penggunaan Kuota Data Internet

Aplikasi dicoba dijalankan pada *smartphone* secara terus-menerus selama 1 bulan, dan penggunaan kuota internet dari aplikasi hanya terhitung sebesar 8.78kb seperti yang ditunjukkan pada Gambar 4.13 dibawah.

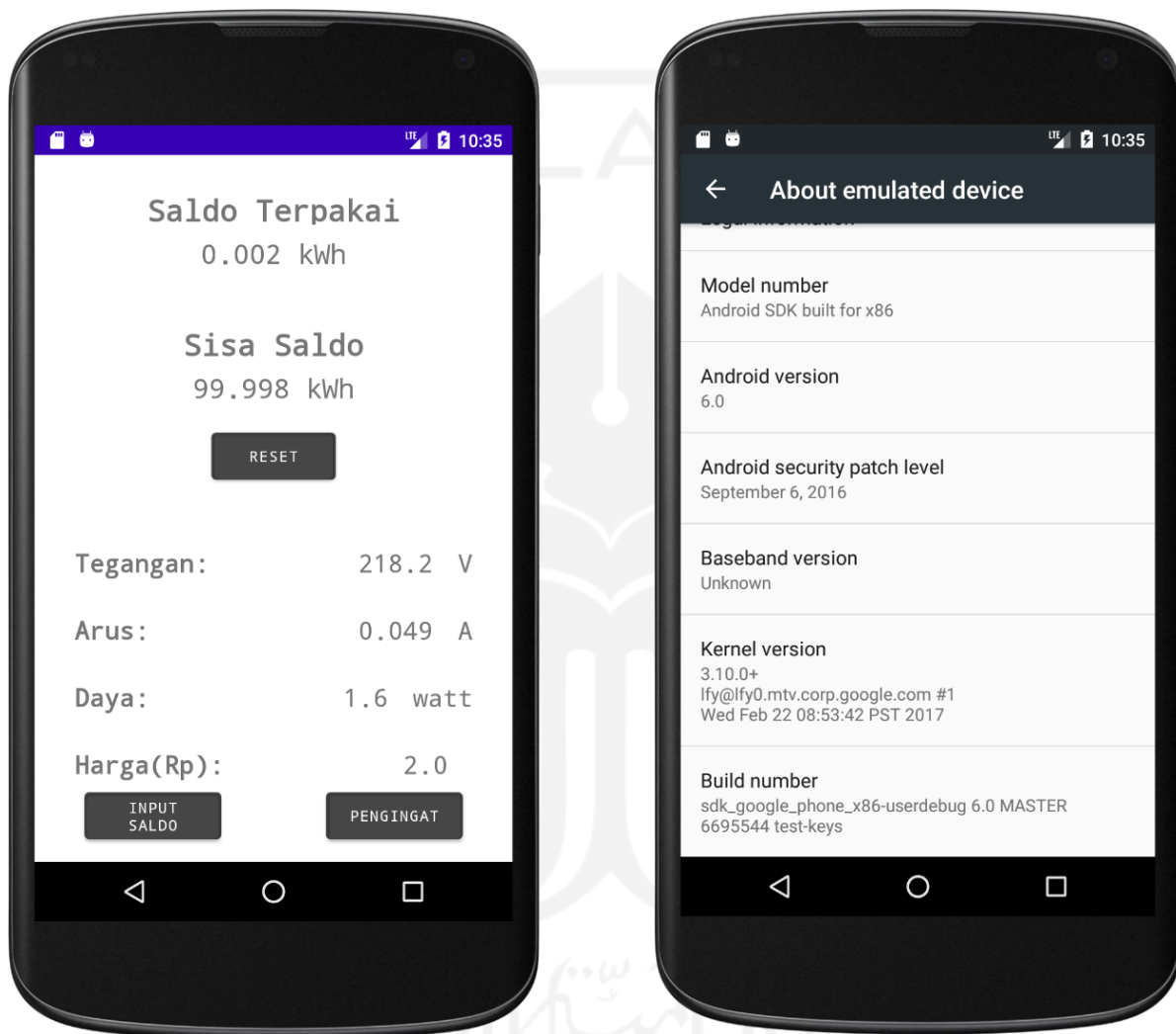


Gambar 4.12 Penggunaan kuota internet dari aplikasi

Berdasarkan pengujian diatas, pengguna tidak perlu khawatir mengenai penggunaan kuota internet dalam penggunaan aplikasi ini karena penggunaan data kuota internetnya sangat kecil.

4.2.4 Pengujian Kompatibilitas Aplikasi

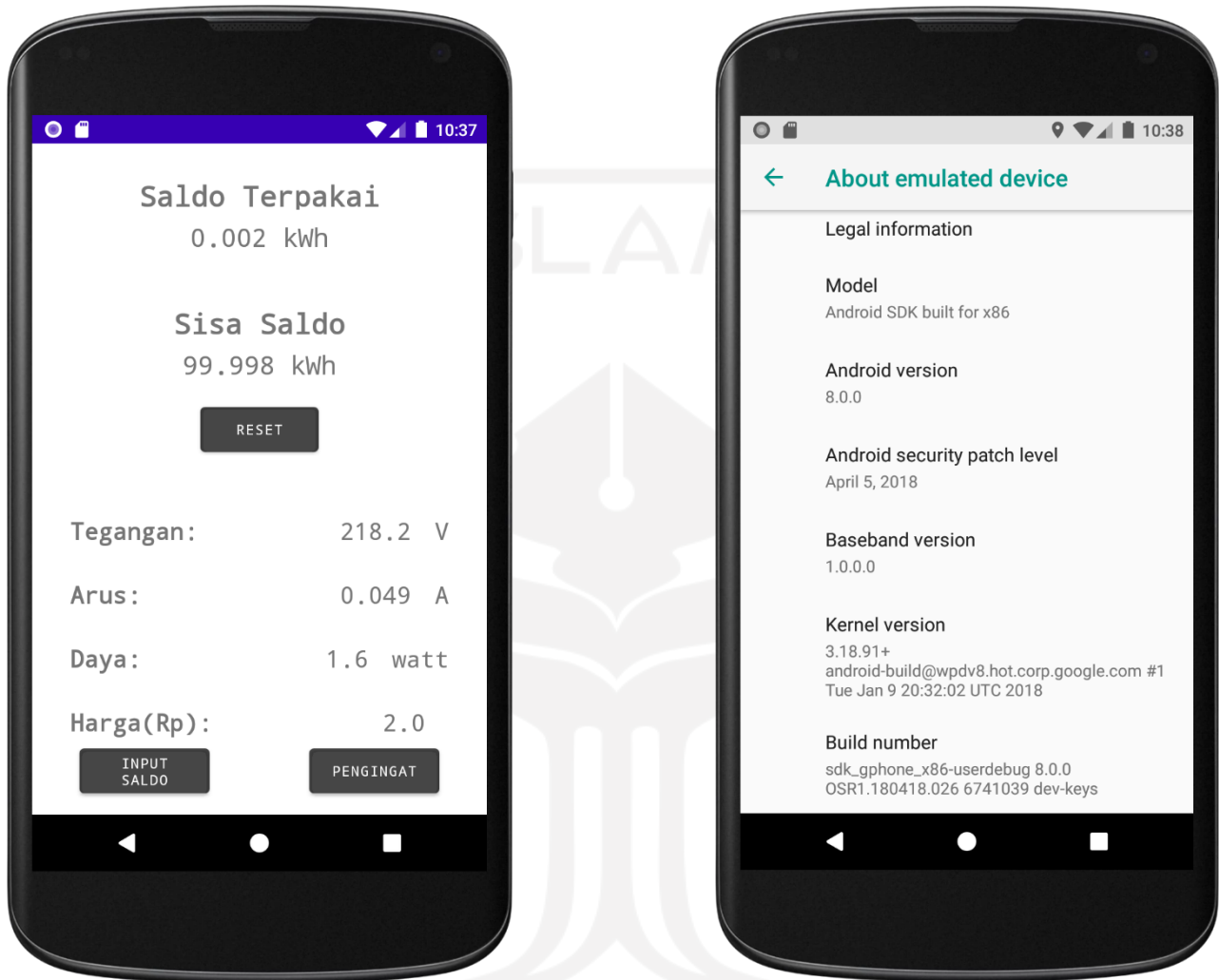
Perangkat: Google Nexus 4
Versi Android: 6.0 Marshmallow



Gambar 4.14 Aplikasi berjalan pada perangkat Google Nexus 4 dengan versi Android 6.0

Perangkat: Google Nexus 4

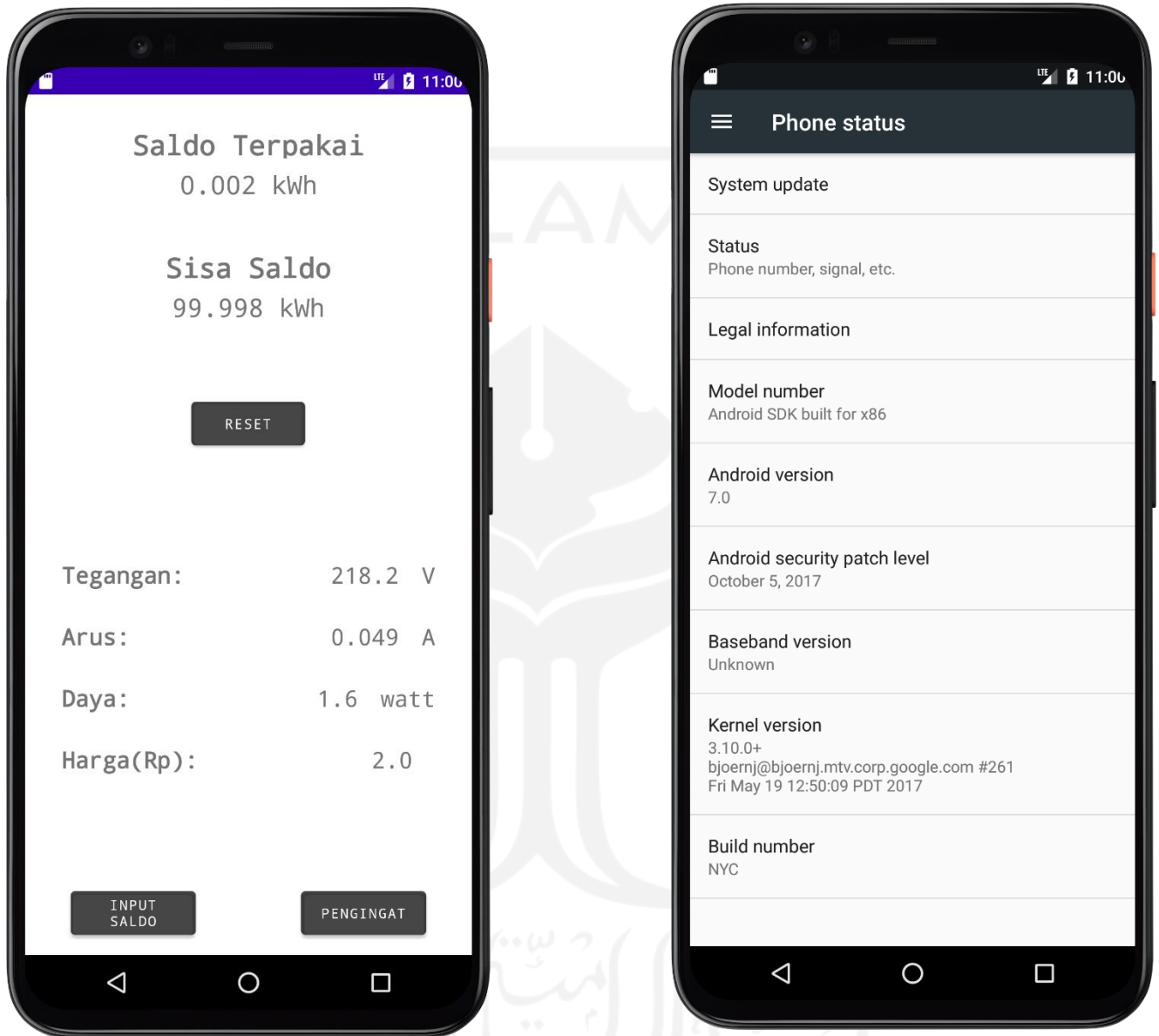
Versi Android: 8.0 Oreo



Gambar 4.15 Aplikasi berjalan pada perangkat Google Nexus 4 dengan versi Android 8.0

Perangkat: Google Pixel 4

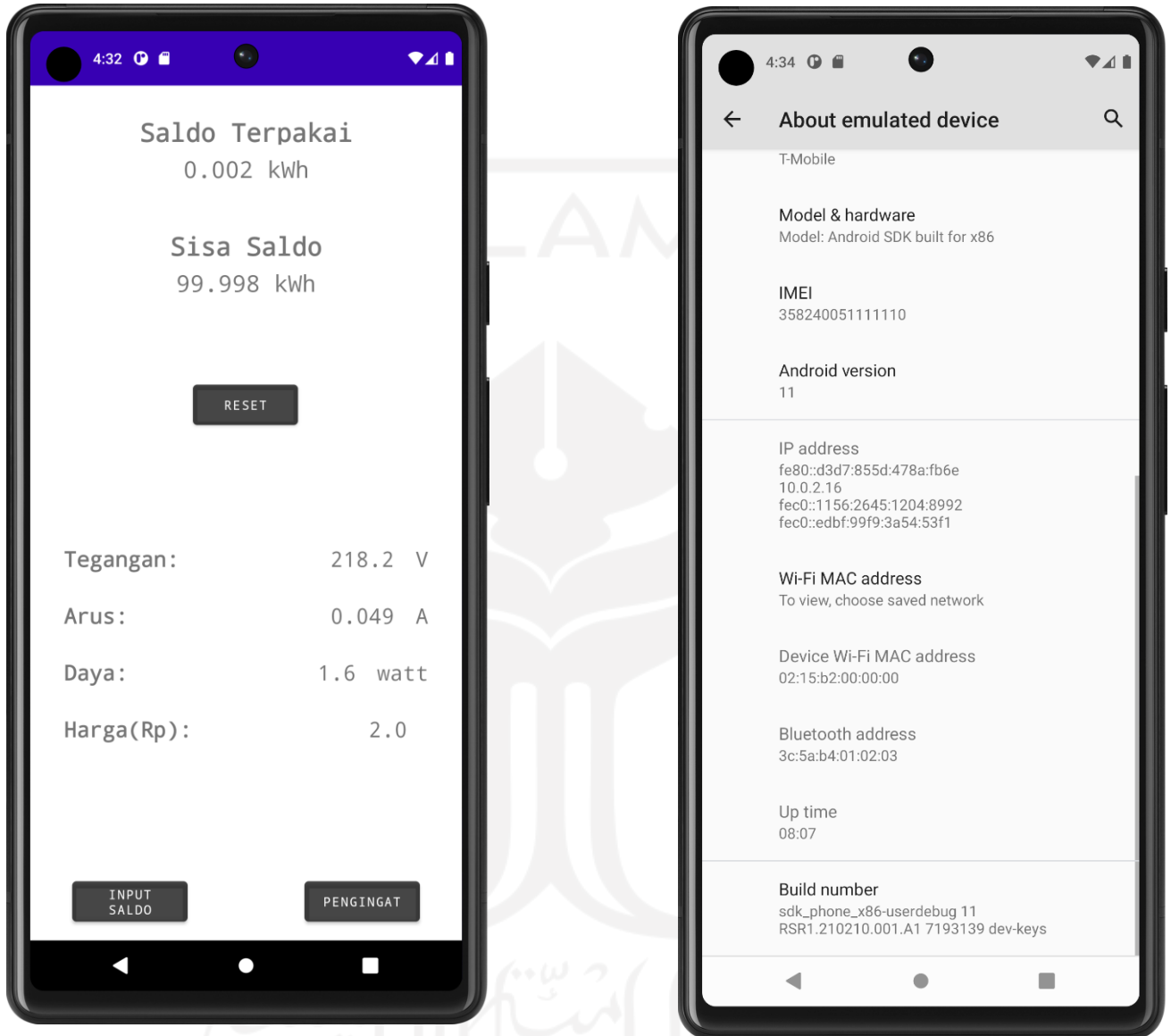
Versi Android: 7.0 Nougat



Gambar 4.16 Aplikasi berjalan pada perangkat Google Pixel 4 dengan versi Android 7.0

Perangkat: Google Pixel 6

Versi Android: 11.0



Gambar 4.17 Aplikasi berjalan pada perangkat Google Pixel 6 dengan versi Android 11.0

Pengujian kompatibilitas di atas dilakukan pada emulator bawaan Android Studio dengan perangkat dan versi Android yang terbilang cukup lama. Seperti pada yang ditampilkan pada Gambar 4.12, Pada perangkat Google Nexus 4, yang rilis pada tahun 2012, yang berjalan di Android versi 6.0 yang rilis pada tahun 2015 hingga device baru dengan versi Android yang baru juga seperti pada gambar 4.15, dengan perangkat Google Pixel 6, yang rilis pada tahun 2021, yang berjalan di Android versi 11.0, yang rilis pada tahun 2020, Dapat disimpulkan

bahwa aplikasi ini tidak akan bermasalah berjalan di bermacam-macam perangkat dan versi sistem operasi Android.

4.2.5 User Acceptance Test (UAT)

Pengujian ini diujicobakan kepada 10 orang responden yang menggunakan listrik prabayar di tempat tinggal mereka, pengujian dilakukan menggunakan *software* Zoom secara remote dengan menggunakan fitur remote control pada fitur share screen yang terdapat pada software tersebut. Responden akan diberikan kontrol terhadap cursor dan setelah itu dihimbau untuk menguji fitur-fitur yang terdapat pada aplikasi yang dijalankan menggunakan emulator Android Studio. Setelah selesai melakukan pengujian, responden akan diberikan link untuk mengisi kuesioner terkait penilaian sistem menggunakan Google Form. Pertanyaan yang terdapat pada kuesioner termuat pada tabel 4.10 di bawah. Pada bagian akhir kuesioner juga disediakan kolom saran untuk diisi oleh pengguna setelah melakukan pengujian aplikasi.

Tabel 4.10 Pertanyaan kuesioner

No	Pertanyaan
1	Apakah fitur aplikasi bekerja dengan baik?
2	Apakah data monitoring yang ditampilkan pada aplikasi sudah cukup lengkap?
3	Apakah aplikasi ini membantu dalam pemantauan penggunaan listrik prabayar jika diimplementasikan ke dunia nyata?

Bobot penilaian untuk pertanyaan-pertanyaan tersebut terdapat pada tabel 4.11 di bawah:

Tabel 4.11 Keterangan bobot nilai

Nilai	Keterangan
1	Sangat tidak setuju
2	Tidak setuju
3	Ragu
4	Setuju
5	Sangat setuju

Tabel 4.12 Tabel hasil penilaian responden

Pertanyaan Ke	Penilaian Responden					Bobot Penilaian/Bobot Maksimal
	1 (x1)	2 (x2)	3 (x3)	4 (x4)	5 (x5)	
1	-	-	-	5(x4) = 20	5(x5) = 25	45/50
2	-	-	-	6(x4) = 24	4(x5) = 20	44/50
3	-	-	-	6(x4) = 24	4(x5) = 25	44/50

Dari data pada tabel 4.12 di atas, didapatkan penilaian bobot total dari responden sebesar 133/150. Dari hasil tersebut, dihitung persentase nilai rata-ratanya dengan menggunakan rumus (4. 2) (Rice, 2020) di bawah ini:

$$\text{Persentase Kepuasan} = \frac{\text{Jumlah Bobot Penilaian Responden}}{\text{Nilai Bobot Total}} \cdot 100\% \quad (4.2)$$

Dari rumus diatas, didapat hasil nilai persentase kepuasan responden terhadap aplikasi sebesar $133 / 150 \times 100 = 88,6\%$.

Di bagian saran pada kuesioner, dari 10 responden pengujian, 5 dari 10 responden memberikan saran agar aplikasi dapat menampilkan penggunaan listrik dalam jangka waktu tertentu, contohnya seperti penggunaan per-hari, per-minggu, dan per-bulan. 3 dari 10 responden memberikan saran agar aplikasi dapat berjalan otomatis saat device dinyalakan, agar notifikasi dapat muncul tanpa pengguna harus membuka aplikasi terlebih dahulu. 2 dari 10 responden tidak mengisi kolom saran.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian pengembangan Aplikasi Android Untuk Pemantauan Listrik Prabayar, didapatkan kesimpulan mengenai aplikasi yang telah dikembangkan, yaitu:

- a. Aplikasi dapat menjalankan fitur-fitur utamanya dengan baik. Fitur-fitur tersebut adalah pemantauan data listrik, penginputan saldo listrik, penginputan batas saldo untuk menampilkan notifikasi, dan reset penggunaan saldo.
- b. Hasil perhitungan penggunaan kWh yang ditampilkan pada aplikasi terbilang cukup akurat dengan *error%* rata-rata 0,309%.
- c. Aplikasi dapat berjalan hampir di semua perangkat dengan sistem operasi Android yang beredar pada zaman sekarang.
- d. Aplikasi dianggap cukup membantu dalam pemantauan listrik prabayar serta untuk membantu melakukan penghematan penggunaan listrik oleh pengguna dengan persentase kepuasan terhadap aplikasi sebesar 88,6%.

5.2 Saran

Berdasarkan hasil pengujian yang dilakukan, didapat saran untuk hasil penelitian sebagai berikut:

- a. Membuat aplikasi dapat berjalan secara otomatis pada saat *device* dinyalakan.
- b. Penambahan fitur untuk melihat penggunaan kWh per hari, per minggu, dan per bulan.

DAFTAR PUSTAKA

- Adani, R. M. (2020, August). *Agile Adalah Metode Development yang Perlu Kalian Ketahui*.
<https://www.sekawanmedia.co.id/blog/metode-agile-development/>.
- Aditya, R., Handrianus Pranatawijaya, V., Bagus Adidyana Anugrah Putra, P., Hendrik Timang, J., Palangkaraya, K., & Tengah, K. (2021). Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype. In *JOINTECOMS (Journal of Information Technology and Computer Science)* p-ISSN: xxxx-xxxx (Vol. 1, Issue 1).
- Adminlp2m. (2022, June 7). *Metode Waterfall – Definisi dan Tahap-tahap Pelaksanaannya*.
<https://lp2m.uma.ac.id/2022/06/07/metode-waterfall-definisi-dan-tahap-tahap-pelaksanaannya/#:~:Text=Metode%20Waterfall%20merupakan%20pendekatan%20SDLC,Desain%2C%20dan%20implementasi%20pada%20sistem.>
- Badan Pusat Statistik. (2020). *Konsumsi Listrik per Kapita (MWH/Kapita)*.
https://www.bps.go.id/indikator/indikator/view_data/0000/Data/1156/Sdgs_7/4.
- cias.co. (2021). *Apa itu Agile? Mengenal Agile, Tujuan, serta Kelebihan dan Kekurangannya*.
<https://www.cias.co/post/apa-itu-agile-mengenal-agile-tujuan-serta-kelebihan-dan-kekurangannya>.
- Dewi, E., Listianti, S., Zaidiah, A., & Isnainiyah, I. N. (2021). Pengembangan Aplikasi Event Management System Berbasis Android Menggunakan Metode Scrum (Studi Kasus Organisasi Mahasiswa UPN Veteran Jakarta). In *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA) Jakarta-Indonesia*.
- Dewi, N. K., Harira Irawan, B., Fitry, E., Putra, A. S., & Jakarta, S. M. (2021). *Konsep Aplikasi E-Dakwah Untuk Generasi Milenial Jakarta*.
- Dicoding. (2020, November 25). *Apa itu Firebase? Pengertian, Jenis-Jenis, dan Fungsi Kegunaannya*.
<https://www.dicoding.com/blog/apa-itu-firebase-pengertian-jenis-jenis-dan-fungsi-kegunaannya/#:~:Text=Firebase%20Realtime%20Database%20adalah%20database,Ke%20setiap%20user%20yang%20terkoneksi.>
- Ersandi Billah. (2019, January 15). *Tahapan-Tahapan SDLC Protoype*.
<https://medium.com/@ersandibillah03/sdlc-prototype-8a3323c1ca33>.
- Esa, S., Buana, T., Atrinawati, L. H., Gilvy, M., & Putra, L. (2021). *Penerapan Metode Agile Untuk Membangun Sistem Informasi Monitoring Santri Pondok Modern Asy-Syifa Balikpapan*.

- Fahrudin, R., & Ilyasa, R. (2021). Perancangan Aplikasi “NUGAS” Menggunakan Metode Design Thinking Dan Agile Development. In *Reza Ilyasa Jurnal Ilmiah Teknologi Informasi Terapan* (Vol. 8, Issue 1).
- Furqon, A., Prasetijo, A. B., & Widiyanto, E. D. (2019). *Rancang Bangun Sistem Monitoring dan Kendali Daya Listrik pada Rumah Kos Menggunakan NodeMCU dan Firebase Berbasis Android*.
- Gemala Jondya, A., Josh, J., Pradipta Saputro, D., & Christopher Sungkharisma, L. (2022). Pengembangan Aplikasi Augmented Reality “e-Museum” dengan Metode Agile untuk Meningkatkan Pengalaman Pengunjung Museum. *Journal of Information System Research*, 3(4). <https://doi.org/10.47065/josh.v3i4.1746>
- Gibran, G. (2018). *Implementasi Rapid Application Development (RAD) Model Pada Pengembangan Aplikasi Rent Car Berbasis Android*.
- Hamamni, K., Mukhsim, M., & Siswanto, D. (2021). Prototipe Sistem Monitoring Biaya Penggunaan Listrik Pada Rumah Kos Berbasis IoT. *JASEE Journal of Application and Science on Electrical Engineering*, 1(02), 35–46. <https://doi.org/10.31328/jasee.v1i02.12>
- IndeedEditorialTeam. (2022, October). *How To Calculate Error (With Steps, Example and Types)*. <https://www.indeed.com/career-advice/career-development/how-to-calculate-error>.
- Junaedi, N. L. (2022, August 16). *Informasi lengkap Rapid Application Development dan tahapannya*. <https://www.ekrut.com/media/rapid-application-development-adalah>.
- Juniawan, F. P., Sylfania, D. Y., Rian Chrisna Putra, R., & Sulaiman, R. (2021). Implementasi Aplikasi Monitoring Nilai dan Kegiatan Siswa Berbasis Android dengan Metode Prototype. *Jurnal Komtika (Komputasi Dan Informatika)*, 5(1), 26–34. <https://doi.org/10.31603/komtika.v5i1.5119>
- Kumpanan. (2021, November 23). *Isi Token Listrik 100 Ribu Berapa KWh? Ini Penjelasannya!* <https://kumpanan.com/berita-update/isi-token-listrik-100-ribu-berapa-kwh-ini-penjelasannya-1wyMkkvqIHK/Full>.
- Maulani, J. (2020). *Penerapan Metode Waterfall Pada Pengembangan Aplikasi Sistem Informasi Jasa Dan Penjualan Dengan Pemodelan Berorientasi Objek*.
- Mitha Djaksana, Y., Eresha Jl Raya Puspipetek, S., Serpong, K., & Tangerang Selatan, K. (2020). Perancangan Sistem Monitoring Dan Kontroling Penggunaan Daya Listrik Berbasis Android. *Jurnal Sistem Informasi Dan Teknologi Informasi*, 2(3), 13–24.

- Mutawali, L., Fathoni, B. K., & Asyari, H. (2020). *Implementasi Scrum Dalam Pengembangan Sistem Informasi Jasa Desain Grafis*.
- Nugroho, S. B., Zusman, E., Nakano, R., Takahashi, K., Koakutsu, K., Kaswanto, R. L., Arifin, N., Munandar, A., Arifin, H. S., Muchtar, M., Gomi, K., & Fujita, T. (2017). The Effect of Prepaid Electricity System on Household Energy Consumption - The Case of Bogor, Indonesia. *Procedia Engineering*, 198, 642–653. <https://doi.org/10.1016/j.proeng.2017.07.117>
- Pangestu, A. D., Ardianto, F., & Alfaresi, B. (2019). *Sistem Monitoring Beban Listrik Berbasis Arduino NodeMCU ESP8266*. 4(1).
- PLN. (2021). *Statistik-PLN-2021-Unaudited-21.2.22*.
- pojokkita.com. (2021, November 11). *Pulsa Listrik Prabayar Tidak Berkurang*. <https://Pojokkita.Com/Pulsa-Listrik-Prabayar-Tidak-Berkurang-02406952>.
- Prasetyo, H. (2017). *Prototyping Untuk Analisis Kebutuhan Dan Perancangan Sistem Monitoring Mentoring (Studi Kasus: Program Asistensi Agama Islam Fakultas Teknologi Industri Universitas Islam Indonesia)*.
- Prastio, C. E., & Ani, N. (2018). *Aplikasi Self Service Menu Menggunakan Metode Scrum Berbasis Android (Case Study : Warkobar Café Cikarang)*. 11(2). <https://teknologi.id>,
- Priscilla, T., & Zulfachmi. (2021). *Survey Paper: Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD)*.
- Prithviraj Biswas. (2021). *What is Android Studio?* <https://Www.Loginradius.Com/Blog/Engineering/Introduction-to-Android-Studio/>.
- Rice, R. (2020). *What is the Value of User Acceptance Testing?* Shorturl.at/UvGQV.
- Rizky, M. (2019). *UML Diagram : Activity Diagram*. <https://Socs.Binus.Ac.Id/2019/11/22/Uml-Diagram-Activity-Diagram/>.
- Rohatun, N. (2020, December 21). *Scrum? Yuk Kenal Lebih Dekat Dengan Metode Scrum*. <https://Www.Mii.Co.Id/Insight/Listing/2020/12/21/04/05/Mengenal%20Scrum#:~:Text=Secara%20definisi%2C%20berdasarkan%20Scrum%20Guide,Namun%20cukup%20sulis%20untuk%20dikuasai>.
- Rony Setiawan. (2021, October 2). *Konsep Agile pada Software Development*. <https://Www.Dicoding.Com/Blog/Konsep-Agile-Pada-Software-Development/#:~:Text=Apa%20itu%20agile%3F,Kolaborasi%20secara%20terstruktur%20dan%20terorganisir>.

- Sameen Mirza, M., & Datta, S. (2019). Strengths and Weakness of Traditional and Agile Processes - A Systematic Review. *Journal of Software*, 14(5), 209–219. <https://doi.org/10.17706/jsw.14.5.209-219>
- Simangunsong, K., & Handayani, R. (2020). *Pengembangan Aplikasi Monitoring Pemakaian Daya Listrik Rumah Development Of Home Electricity Monitoring Application*.
- Simbolon, N. (2021, April). *Pengujian UAT (User Acceptance Test) dalam desain produk*. <https://www.ekrut.com/media/uat-adalah>.
- statcounter.com. (2022, May). *Mobile Operating System Market Share Indonesia*. <https://gs.statcounter.com/os-market-share/mobile/indonesia>.
- Sugara, E. P. A., Perdana, A., & Subrata, A. (2017). *Aplikasi Pembelajaran Sejarah Filsafat Berbasis Android Menggunakan Metode Pengembangan Agile*. <https://doi.org/10.31227/osf.io/b5kua>
- Suyanto, & Andri. (2020). *Implementasi Rapid Application Development Dalam Pengembangan Aplikasi Pelaporan Kerusakan Jalan*.
- Wahyudi, J. (2019). *Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android Studi Kasus: CV. Amanah Kalimantan RENT*.
- Widianto, H., Pratama, A. P., & Laksmi, A. P. (2020). Pengembangan Aplikasi COSYCALSHIP Berbasis Android untuk Pengelolaan Beasiswa Menggunakan Metode Waterfall. In *Journal of Advances in Information and Industrial Technology (JAIIT)* (Vol. 2, Issue 2).
- William, J., Widodo, A., Kholis, N., & Rakhmawati, L. (2022). *Rancang Bangun Alat Monitoring Daya Listrik Berbasis IoT Menggunakan Firebase*.
- Yoko, P., Adwiya, R., & Nugraha, W. (2019). *Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis Website pada Credit Union Canaga Antutn*.