

PENGEMBANGAN SISTEM PEMANTAUAN DAN PENGENDALIAN DAYA LISTRIK PENYEWA KOS BERBASIS LORA DENGAN INTEGRASI ANDROID

Yenniwarti Rafsyam¹, Benny Nixon², Khomala Ernia Putri^{*3}, Molliyana Tota Angelica⁴, Rifqi Fuadi Hasani⁵, Shita Fitria Nurjihan⁶, Irwan Prasetya⁷

^{1,2,3,4,5,6,7}Politeknik Negeri Jakarta, Depok

Email: ¹yenniwarti.rafisyam@elektro.pnj.ac.id, ²benny.nixon@elektro.pnj.ac.id,

³khomala.ernia.putri.te21@mhs.pnj.ac.id, ⁴molliyana.tota.angelica.te21@mhs.pnj.ac.id,

⁵rifqi.fuadi.hasani@elektro.pnj.ac.id, ⁶sita.fitria.nurjihan@elektro.pnj.ac.id, ⁷irwan.prasetya@elektro.pnj.ac.id

*Penulis Korespondensi

(Naskah masuk: 12 Maret 2025, diterima untuk diterbitkan: 13 Juli 2025)

Abstrak

Pengelolaan daya listrik di kamar kos sering kali menimbulkan ketidakadilan karena biaya listrik disamaratakan, meski pemakaian berbeda. Untuk mengatasi masalah ini, dirancang sebuah sistem pemantauan daya listrik dengan teknologi LoRa. Sistem ini memanfaatkan mikrokontroler ESP32, sensor PZEM-004T, LoRa, dan Arduino IDE. Data penggunaan daya listrik yang dikumpulkan oleh sensor dikirim ke pemilik kos melalui jaringan LoRa. Data penggunaan daya listrik dikirim melalui jaringan LoRa hingga 150 meter (RSSI -107 dBm) dalam kondisi *NON-LOS* dan 600 meter (RSSI -111 dBm) dalam kondisi *LOS*, kemudian diintegrasikan ke aplikasi Android untuk pengelolaan dan pemantauan daya listrik. Hasil pengujian menunjukkan akurasi 99%, menjadikan sistem ini solusi efektif dan efisien untuk pengelolaan daya listrik di kamar kos.

Kata kunci: *android, daya listrik, ESP32, firebase, LoRa, PZEM-004T*

DEVELOPMENT OF LORA-BASED BOARDING HOUSE TENANT ELECTRICITY MONITORING AND CONTROL SYSTEM WITH ANDROID INTEGRATION

Abstract

Electricity management in boarding rooms is often unfair because electricity costs are shared equally, even though each room uses different amounts of electricity. To address this issue, an electricity monitoring system was developed using LoRa technology. The system uses an ESP32 microcontroller, a PZEM-004T sensor, LoRa, and Arduino IDE. The sensor collects electricity usage data and sends it to the landlord via the LoRa network. This data can be transmitted up to 150 meters (RSSI -107 dBm) in *NON-LOS* conditions and 600 meters (RSSI -111 dBm) in *LOS* conditions. The system is also integrated with an Android application for easier management and monitoring. Testing results show an accuracy of 99%, making this system an effective and efficient solution for managing electricity in boarding rooms.

Keywords: *android, electric power, ESP32, firebase, LoRa, PZEM-004T*

1. PENDAHULUAN

Manajemen daya listrik di lingkungan kamar kos merupakan aspek penting yang sering kali menghadapi berbagai tantangan, terutama terkait efisiensi dan pengendalian penggunaan. Salah satu masalah utama yang sering muncul adalah penggunaan listrik yang tidak terkontrol di setiap kamar, yang mengakibatkan peningkatan biaya listrik. Selain itu, penerapan tarif listrik yang seragam untuk semua kamar, meskipun tingkat konsumsi listrik bervariasi, menimbulkan ketidakadilan bagi para penyewa kos (Dwi Alfian, 2021).

Penelitian oleh Dwi Alfian berfokus pada manajemen daya listrik di hunian multi-unit, namun tidak mengimplementasikan teknologi LoRa, yang membatasi jangkauan dan efisiensi transmisi data. Sebagai solusi yang lebih optimal, pengembangan sistem kendali dan pemantauan daya listrik di kamar kos berbasis teknologi LoRa menjadi sangat diperlukan. Sistem ini tidak hanya membantu pemilik kos dalam memonitor penggunaan listrik secara lebih efisien, tetapi juga memberikan keuntungan bagi penyewa dengan memastikan bahwa biaya listrik yang mereka bayar sesuai dengan konsumsi sebenarnya. Dengan demikian, sistem yang andal dan berdaya guna tinggi ini diharapkan dapat

meningkatkan efisiensi dan keadilan dalam pengelolaan daya listrik di lingkungan kamar kos, memberikan dampak positif bagi semua pihak yang terlibat.

Penelitian lain oleh Suga dan Nurwarsito menunjukkan bahwa sistem yang tidak terintegrasi dengan aplikasi Android kurang efektif bagi pengguna (Suga & Nurwarsito, 2021). Oleh karena itu, diperlukan sistem yang dapat memantau konsumsi daya listrik secara *real-time* dan mengelola penggunaan daya secara lebih optimal, serta terintegrasi dengan aplikasi yang mudah digunakan. Penggunaan daya listrik dalam berbagai aplikasi, baik industri maupun rumah tangga, memerlukan pemahaman yang mendalam tentang komponen listrik, termasuk Daya Nyata (P) atau *Real Power*, yang dihitung dengan mengalikan nilai rms tegangan dan arus dengan faktor daya (PF) (Setiaji et al., 2022). Sedangkan untuk perhitungan energi listrik Daya Nyata (P) dikalikan dengan waktu dalam jam (T) (Huda, 2021).

Pengukuran konsumsi energi listrik biasanya dilakukan dengan menggunakan kWh meter, yang berperan penting dalam transaksi listrik dan harus memenuhi standar PT. PLN (Manda Karina et al., 2022). Dalam konteks rumah kos, daya listrik sebesar 1.300 VA adalah pilihan umum karena sesuai untuk kebutuhan rumah tangga dan usaha kecil (Hasanudin, 2021), dengan tarif listrik sebesar Rp 1.444,70 per kWh pada tahun 2024, belum termasuk Pajak Penerangan Jalan (PPJ).

Teknologi komunikasi nirkabel seperti LoRa SX1278, yang mampu mendukung pengiriman data dengan efisiensi konsumsi daya (Herlukman et al., 2024) sering digunakan dalam aplikasi IoT bersama sensor PZEM-004T untuk mengukur parameter listrik seperti daya, arus, maupun energi (Adiwiranto et al., 2022). Mikrokontroler ESP32, yang didukung oleh WiFi dan Bluetooth, juga populer dalam aplikasi IoT (Nizam et al., 2022) sering dipadukan dengan relay untuk pengendalian arus (Santosa & Nugroho, 2021).

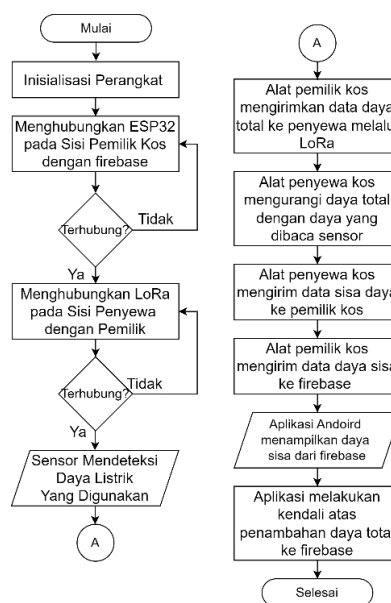
Liquid Crystal Display (LCD) digunakan untuk menampilkan informasi pada perangkat elektronik dengan merefleksikan cahaya dari lampu latar (Lutfi Tsabi et al., 2020), sementara buzzer berfungsi sebagai indikator suara. Perangkat ini memerlukan catu daya, atau adaptor, yang mengubah arus listrik AC menjadi DC untuk memastikan suplai energi yang sesuai (Suga & Nurwarsito, 2021). Arduino IDE memudahkan pengendalian perangkat dalam proyek mikrokontroler melalui pemrograman menggunakan bahasa JAVA dan pustaka C/C++ (Nur Alfian & Ramadhan, 2022). Sistem operasi Android memungkinkan modifikasi dan pengembangan aplikasi secara bebas (Gunadi et al., 2020). Aplikasi Android ini juga terintegrasi dengan Firebase, sebuah *platform database* yang menyediakan informasi secara *real-time* (Sugiyatno, 2023).

2. METODE PENELITIAN

Metode penelitian melibatkan perancangan dan implementasi sistem menggunakan mikrokontroler ESP32, sensor PZEM-004T, dan LoRa, dengan program yang direalisasikan menggunakan Arduino IDE dan Android Studio.

2.1 Perancangan Umum Sistem

Perancangan sistem ini mencakup perangkat keras dan aplikasi Android yang bekerja secara terpadu, dengan alur proses yang dapat dilihat pada Gambar 1. Sistem dimulai dengan inialisasi perangkat dan menghubungkan mikrokontroler ESP32 milik pemilik kos ke Firebase, yang berperan penting dalam menyimpan dan mengakses data secara *real-time*. Setelah itu, modul LoRa pada setiap perangkat dikoneksikan untuk memastikan komunikasi nirkabel jarak jauh. Ketika koneksi LoRa sudah terbentuk, sensor PZEM-004T mulai mendeteksi dan mengukur daya listrik yang digunakan. Data tersebut kemudian dikirim ke Firebase dan dapat diakses melalui aplikasi Android, memungkinkan pemilik kos untuk memantau dan mengelola konsumsi daya listrik secara lebih efektif.

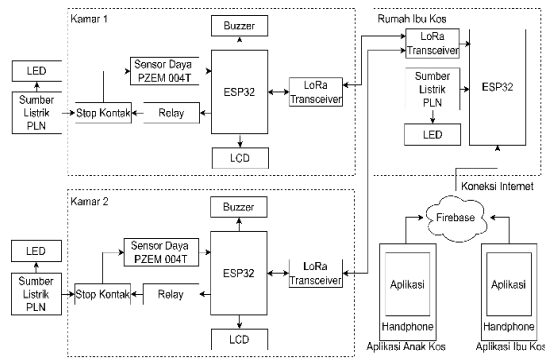


Gambar 1 Flowchart Sistem

Selama sistem ini berjalan alat pemilik kos akan mengirimkan data daya total ke penyewa kos melalui LoRa kemudian data daya total ini akan dikalkulasikan dengan daya yang terbaca pada sensor PZEM-004T. Hasil ini akan dikirimkan kembali ke alat pemilik kos dan dikirimkan ke Firebase. Data di Firebase dapat dibaca melalui aplikasi android pengguna serta aplikasi ini dapat melakukan penambahan daya total yang nantinya akan menambah nilai daya pada alat penyewa kos.

Selanjutnya sistem akan dirancang seperti diagram blok yang ada pada Gambar 2.

Diagram blok pada Gambar 2 menunjukkan komponen yang diperlukan serta hubungannya dengan komponen lain. Alat pada sisi pemilik kos menggunakan ESP32 sebagai mikrokontroler yang terhubung ke firebase dimana alat ini akan membaca nilai daya total yang ada di Firebase.



Gambar 2 Diagram Blok Sistem Pemantauan Dan Pengendalian Daya Listrik Penyewa Kos

Selanjutnya data yang didapat akan dikirimkan ke alat pada sisi penyewa melalui LoRa. Pada alat sisi penyewa kos menggunakan mikrokontroler ESP32 yang terhubung dengan komponen LoRa.

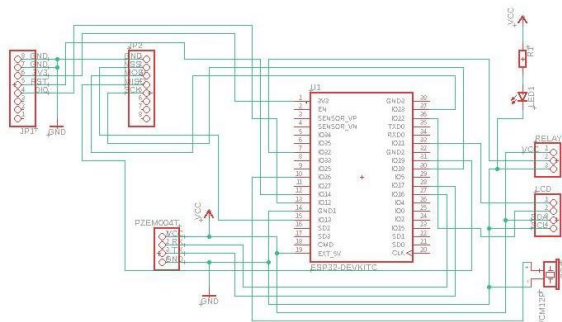
Pada alat ini akan menerima data berupa daya total yang dikirimkan oleh alat pemilik kos melalui LoRa. Data yang didapatkan akan dikurangi dengan pembacaan sensor PZEM-004T yang terhubung dengan ESP32 dan ditampilkan pada LCD.

Pada LCD selain menampilkan daya sisa dari alat penyewa kos, akan menampilkan juga tegangan, arus dan daya yang digunakan. Hasil pengurangan ini akan dikirimkan kembali ke alat pemilik kos untuk dikirimkan ke Firebase.

Ketika daya sisa kurang dari 1kWh relay pada alat penyewa kos akan memutuskan arus listrik pada stop kontak dan memberikan notifikasi berupa buzzer dan notifikasi pada aplikasi android.

2.1.1 Perancangan Alat Penyewa Kos

Perancangan alat penyewa kos menggunakan ESP32 sebagai mikrokontroler. Skematik alat penyewa kos dapat dilihat pada Gambar 3.



Gambar 3 Skematik alat penyewa kos

ESP32 ini akan terhubung dengan sensor PZEM-004T dan LoRa SX1278. Untuk memutuskan arus pada stop kontak ESP32 terhubung dengan relay

dan ESP32 terhubung dengan buzzer dan LED sebagai indikator. Selanjutnya untuk menampilkan data untuk pengguna ESP32 terhubung dengan LCD i2c 16x2. Hubungan pin sensor dan komponen lainnya dengan ESP32 dapat dilihat pada Tabel 1.

Tabel 1. Hubungan Pin ESP32 dengan Komponen

Komponen	Pin	Definisi Nama Pin	Keterangan
LoRa SX1278	12,19	DIO0, MISO	INPUT
LoRa SX1278	13,14,18,23	SS, RST, SCK, MOSI	OUTPUT
PZEM-004T	16,17	RX, TX	INPUT
LCD I2C	21,22	SDA, SCL	OUTPUT
Relay	32	IN1	OUTPUT
Buzzer	26	Positif	OUTPUT

2.1.2 Perancangan Aplikasi

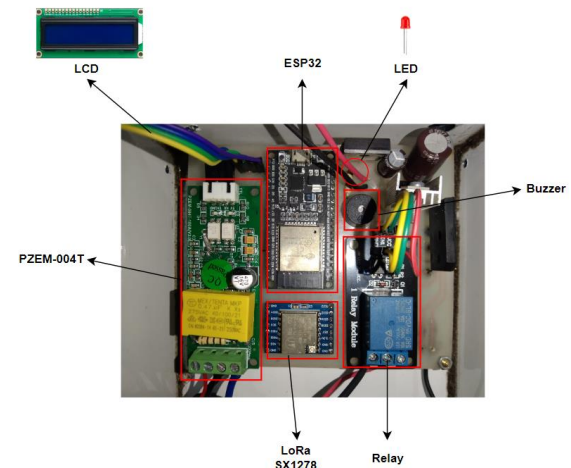
Perancangan aplikasi yang dibuat untuk memonitor daya yang digunakan oleh penyewa kos dimulai dari merancang bagaimana menghubungkan aplikasi dengan database yang digunakan yaitu firebase realtime database. Selanjutnya membuat tampilan pada aplikasi dimulai dari tampilan autentikasi berupa *login* dan *register*. Lalu untuk menampilkan daya yang tersisa dibuat tampilan utama dari aplikasi yang akan dibuat.

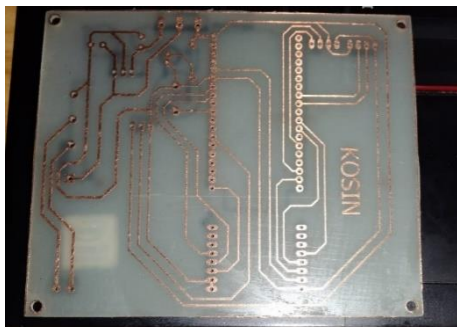
Dalam tampilan utama ini juga terdapat jumlah deposit yang ditambahkan oleh pemilik kos. Pada tampilan utama ini juga terdapat pilihan paket-paket yang dapat dipilih oleh penyewa kos sesuai dengan deposit yang dimiliki. Untuk melihat riwayat pembelian ini penyewa kos dapat menekan tombol riwayat dan akan diarahkan ke tampilan riwayat dan untuk keluar dari akun penyewa kos terdapat tombol *logout* di tampilan utama. Aplikasi ini juga memiliki fitur notifikasi ketika daya yang tersisa berada di ambang batas yaitu kurang dari 1 kWh.

2.2 Realisasi Sistem

2.2.1 Realisasi Alat Penyewa Kos

Pada tahap ini, hasil perancangan *layout* rangkaian alat *transceiver* direalisasikan dan dicetak pada PCB.



Gambar 4 Tampak Atas Rangkaian *Transceiver* Penyewa KosGambar 5 Tampak Bawah Hasil Cetak *Layout PCB*

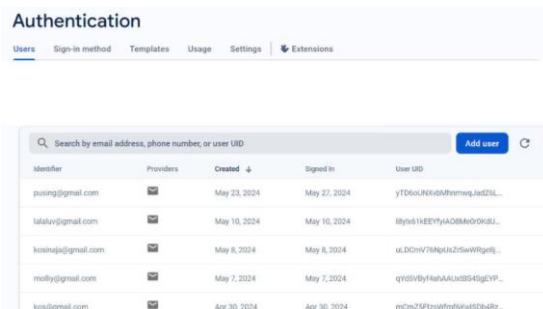
2.2.2 Realisasi Aplikasi

Pada tahap realisasi aplikasi, pertama aplikasi harus dapat terhubung ke *database* Firebase Realtime Database. Firebase Realtime Database digunakan untuk menyimpan data secara instan, sehingga memungkinkan data yang diterima untuk diproses dengan cepat dan pengaturannya mudah. Untuk membuat realtime database ini dimulai dari membuat project di Firebase Console isi nama project Firebase, selanjutnya pada tampilan *dashboard* pilih layanan Realtime Database untuk membuat *database*. Lalu pastikan izin untuk membaca dan edit diberikan pada tab Rules. Berikut merupakan Realtime Database yang sudah berhasil dibuat dapat dilihat pada Gambar 6.



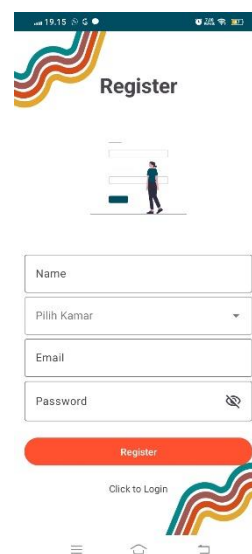
Gambar 6 Realisasi Firebase Realtime Database

Selanjutnya untuk membuat fitur autentikasi pada aplikasi, harus terlebih dahulu diatur pengaturan *Sign In* pada Firebase. Pada menu authentication pilih tab *Sign-in method*, tambahkan metode *login* berupa *Email/Password*. Berikut tampilan ketika *sign-in method* berhasil di atur dapat dilihat pada Gambar 7.



Gambar 7 Realisasi Firebase Authentication

Setelah Firebase siap digunakan, tahapan berikutnya adalah realisasi pada sisi aplikasi android. Aplikasi android yang dibuat menggunakan software Android Studio. Untuk mengambil ataupun mengubah data yang ada pada firebase, perlu melakukan konfigurasi pada android studio. Setelah berhasil terhubung ke firebase aplikasi dapat menggunakan fitur autentikasi berupa *Email/Password* yang telah dibuat. Untuk fitur ini perlu dibuat sistem *register* untuk membuat akun. Tampilan untuk fitur *register* dapat dilihat pada Gambar 8.



Gambar 8 Tampilan Register



Gambar 9 Tampilan Login

Seperti yang dapat dilihat pada Gambar 8 Pada fitur register ini pengguna harus memasukkan nama, nomor kos yang akan ditempati, email dan *password*. Selanjutnya ketika semua data terisi pengguna dapat menekan tombol “*Register*” atau jika sudah memiliki akun maka pengguna dapat langsung masuk ke tampilan login dengan menekan “*Click to Login*”.

Selanjutnya ketika pengguna sudah memiliki akun, dapat memasukkan email dan *password* yang sudah didaftarkan pada tampilan login seperti pada Gambar 9.

Lalu setelah proses autentikasi, pengguna akan diarahkan ke tampilan utama dari aplikasi ini. Tampilan utama dari aplikasi ini dapat dilihat pada Gambar 10.



Gambar 10 Tampilan Utama Aplikasi

Pada Gambar 10 tampilan utama aplikasi ini terdapat nilai daya dan deposit yang tersisa, daftar paket yang dapat dibeli oleh pengguna berikut dengan harga dan jumlah kWh yang akan didapat. Selanjutnya terdapat tombol riwayat untuk melihat riwayat pembelian yang sudah dilakukan pengguna

dan tombol *logout* untuk keluar dari akun yang digunakan oleh pengguna penyewa kos.

Terakhir untuk menampilkan riwayat pembelian daya dibuat tampilan riwayat pada aplikasi. Tampilan riwayat dapat dilihat pada Gambar 11.



Gambar 11 Tampilan Riwayat

Tampilan riwayat, seperti yang terlihat pada Gambar 11, menunjukkan riwayat pembelian yang dilakukan oleh penyewa kos. Data yang ditampilkan mencakup nilai kWh yang dibeli, harga paket kWh tersebut, serta *timestamp* yang mencatat waktu pembelian varian paket listrik dengan format tahun-bulan-tanggal jam:menit:detik.

3. HASIL DAN PEMBAHASAN

3.1. Pengujian Sensor PZEM-004T

Pengujian Sensor PZEM-004T dilakukan untuk mengetahui apakah nilai yang terbaca oleh sensor sesuai dengan rumus atau teori yang mendukung. Untuk melakukan pengujian pada rangkaian Sensor PZEM-004T dibutuhkan alat – alat sebagai berikut :

1. Rangkaian alat
2. Laptop
3. Beban Listrik

Prosedur atau langkah-langkah pengujian yang dilakukan pada pengujian sensor PZEM-004T adalah sebagai berikut :

1. Menyiapkan sensor PZEM-004T yang telah diprogram untuk mengukur tegangan (V), arus (I), daya (W), faktor daya (PF), dan energi (kWh).
2. Selanjutnya, menghubungkan beban listrik ke terminal input pada sensor PZEM-004T.
3. Terakhir, mencatat hasil pengukuran dari sensor dan membandingkannya dengan teori yang relevan pada Tabel 2 dan Gambar 12.

Hasil dari pengujian yang telah dilakukan pada sensor PZEM-004T dapat dilihat dalam Tabel 2. Tabel ini menyajikan data yang diperoleh dari proses pengujian dan Tabel tersebut telah dikonversi

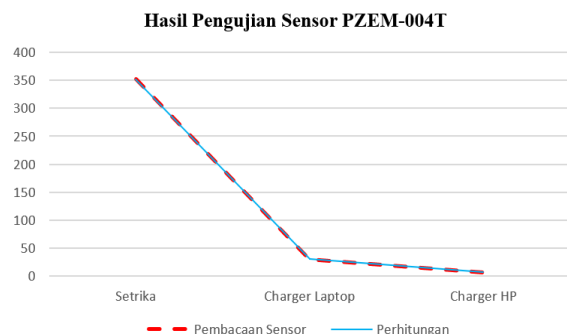
menjadi grafik untuk mempermudah *interpretasi* data dan memperjelas perbedaan yang ada. Grafik dapat dilihat pada Gambar 12.

Berdasarkan hasil percobaan pada Tabel 2, hasil perhitungan didapat dari perhitungan rumus daya nyata (W) yaitu tegangan (V) dan arus (I) dikalikan dengan faktor daya (PF) sedangkan untuk perhitungan energi listrik yaitu daya nyata (W) dikalikan dengan waktu (Jam) kemudian dibagi seribu dikarenakan dalam orde yang lebih tinggi (kWh). Dari hasil pengukuran dan perhitungan dapat dianalisis persentase perbedaan yang didapat dengan

Tabel 2. Hasil Pengujian Sensor PZEM-004T

Beban Listrik	V	I	PF	Daya (W)		Energi (kWh)	
				Pembacaan	Perhitungan	Pembacaan	Perhitungan
Setrika	216	1.63	1	352	352.08	0.352	0.35208
Charger Laptop	221	0.29	0.480	30	30.76	0.03	0.0376
Charger HP	221	0.06	0.520	7	6.89	0.007	0.00689

Berdasarkan analisis, rata-rata kesalahan atau *error rate* yang diperoleh adalah 1,3409, yang menunjukkan tingkat akurasi hampir 99%, sesuai dengan spesifikasi ketepatan ukur sensor PZEM-004T. Selain itu, faktor karakteristik beban juga mempengaruhi hasil pengukuran, terutama pada *charger* HP yang memiliki faktor daya lebih rendah, sehingga dapat memengaruhi pembacaan daya aktif.



Gambar 12 Grafik Hasil Pengujian Sensor PZEM-004T

Dari Grafik hasil pengujian sensor PZEM-004T dapat dilihat bahwa pengujian terdapat 3 beban yang digunakan pada saat pengujian yaitu setrika, charger laptop dan charger Hp. Dari ketiga beban tersebut dapat dilihat daya yang paling besar yaitu setrika dan dapat dilihat dari grafik bahwa perbandingan hasil pembacaan sensor dengan hasil perhitungan hampir setara dan tidak menunjukkan perbedaan yang berarti persentase perbedaan sangat kecil.

3.2. Pengujian Sinyal LoRa

Pengujian sinyal LoRa merupakan pengujian yang dilakukan untuk mengetahui seberapa jauh jarak sinyal dari LoRa *node* pada sisi penyewa kos kepada LoRa *gateway* sebagai pemilik kos.

Untuk melakukan pengujian pada rangkaian sistem dibutuhkan alat – alat sebagai berikut :

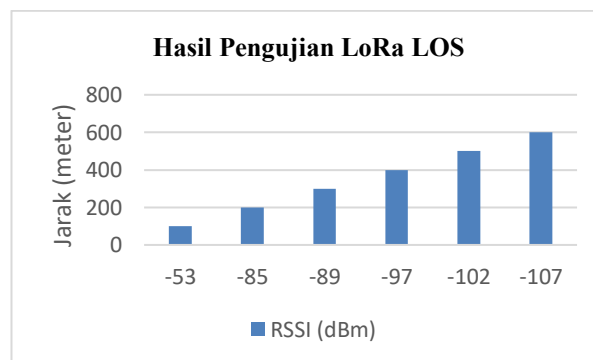
cara hasil perhitungan dikurangkan dengan hasil pembacaan kemudian dibagi dengan hasil pembacaan dan dikalikan dengan seratus. Dari pengujian yang dilakukan, diperoleh bahwa persentase perbedaan hasil pengujian sensor dengan beban setrika adalah 0,0227%. Untuk beban *charger* laptop, persentase perbedaannya mencapai 2,5%. Sedangkan untuk beban *charger* HP, persentase perbedaannya adalah 1,5%.

1. Rangkaian alat penyewa dan pemilik kos
2. Laptop
3. Kabel

Prosedur pengujian yang dilakukan untuk mengukur jarak sinyal LoRa dengan titik awal adalah Laboratorium Telekomunikasi. Adapun langkah-langkah pengujian sebagai berikut:

1. Menyiapkan alat *transceiver* penyewa kos maupun pemilik kos.
2. Menghubungkan alat *transceiver* dengan daya listrik.
3. Melakukan pengukuran di beberapa titik.
4. Mendapatkan data dan mencatat hasil.

Dari hasil pengujian sinyal LoRa *Line of Sight* yang telah dilakukan dapat diperoleh hasil seperti pada Gambar 13.

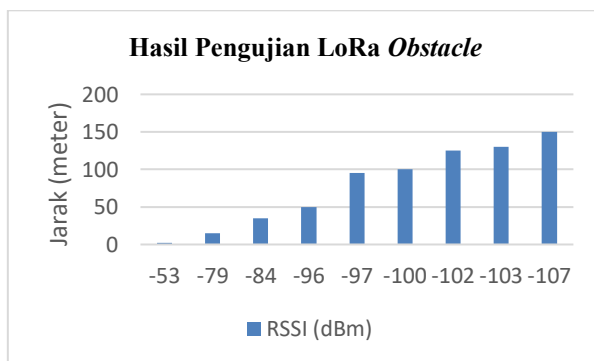


Gambar 13 Grafik Hasil Pengujian Sinyal LoRa Kondisi LOS

Menurut TIPPHON (Noval et al., 2023) jika level RSSI > -70 dBm, maka nilai RSSI tersebut sangat bagus jika berada di antara -70 dBm dan -85 dBm, maka nilai RSSI tersebut bagus jika berada di antara -86 dBm dan -100 dBm, maka nilai RSSI tersebut sedang dan jika di bawah -100 dBm, maka nilai RSSI tersebut dianggap buruk.

Berdasarkan kesimpulan dari grafik yang ditampilkan pada Gambar 13, terlihat bahwa nilai RSSI (*Received Signal Strength Indicator*) meningkat seiring dengan bertambahnya jarak. Hasil pengujian menunjukkan bahwa pada jarak 50 meter, nilai RSSI sangat baik, sedangkan pada jarak 100 hingga 200 meter, nilai RSSI masih tergolong baik. Pada jarak 300 meter, nilai RSSI dikategorikan sedang, sementara pada jarak 400 hingga 600 meter, nilai RSSI mengalami penurunan dan dikategorikan buruk.

Dari analisis ini, dapat disimpulkan bahwa dalam kondisi LOS (*Line of Sight*) di area terbuka, pengiriman data masih dapat diterima dengan baik hingga jarak 600 meter, dengan nilai RSSI mencapai -111 dBm. Namun, pada jarak yang melebihi 600 meter, kemampuan LoRa untuk menerima sinyal mulai menurun secara signifikan, sehingga sinyal menjadi tidak dapat diterima dengan baik.



Gambar 14 Grafik Hasil Pengujian Sinyal LoRa Kondisi Obstacle

Berdasarkan hasil pengujian yang ditampilkan dalam grafik pada Gambar 14, terlihat bahwa nilai RSSI sangat baik pada jarak 2 meter. Nilai RSSI yang masih sangat baik juga ditemukan pada jarak antara 15 hingga 35 meter. Namun, seiring bertambahnya jarak, nilai RSSI mulai menurun. Pada jarak 50 hingga 95 meter, nilai RSSI tergolong sedang, dan pada jarak 100 hingga 150 meter, nilai RSSI menjadi buruk.

Dari analisis ini, dapat disimpulkan bahwa dalam kondisi *NON-LOS* (*Non-Line of Sight*) atau *Obstacle*, yaitu ketika ada hambatan seperti gedung atau pepohonan yang menghalangi jalur transmisi, pengiriman data masih dapat diterima dengan cukup baik hingga jarak 150 meter, dengan nilai RSSI sebesar -107 dBm. Namun, pada jarak yang lebih jauh, seperti 250 meter, kemampuan LoRa untuk menerima sinyal mengalami penurunan yang signifikan, dan pada titik ini, LoRa sudah tidak mampu menerima sinyal.

3.3. Pengujian Aplikasi Android

Pada pengujian ini, *smartphone* yang digunakan telah terinstal aplikasi yang diperlukan, sehingga pengujian dapat dilanjutkan dengan membuka aplikasi tersebut. Semua perangkat transceiver, baik

milik penyewa kos maupun pemilik kos, dapat terhubung dengan baik. Selain itu, perangkat pemilik kos juga berhasil terhubung dengan Firebase, sehingga informasi mengenai nilai deposit dan sisa daya dapat ditampilkan di aplikasi Android dengan lancar.

Hasil pengujian telah dilakukan menggunakan *smartphone* yang sudah terinstal aplikasi, maka didapatkan hasil sebagai berikut:

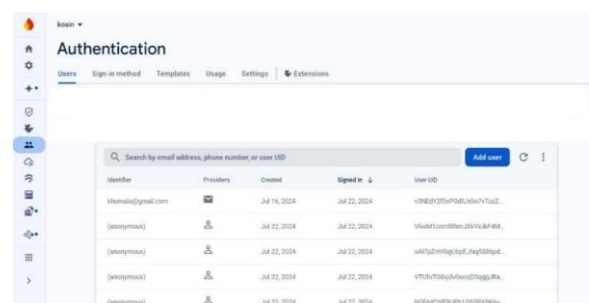
a. Pengujian fitur *authentication* dengan melakukan proses login

Dapat dilihat pada Gambar 15 dan Gambar 16 bahwa proses login berhasil dilakukan dengan aplikasi sesuai dengan akun yang telah terdaftar di Firebase serta aktivitas login berjalan tanpa adanya masalah.



Gambar 15 Pengujian Fitur Login

Saat pengguna ingin login, data yang diperlukan mencakup email, kata sandi, serta nomor ruangan atau kamar pengguna. Data yang diinput oleh pengguna saat registrasi akan disimpan di Firebase Authentication.



Gambar 16 Firebase Authentication Login

b. Pengujian *Main Activity*

Dapat dilihat pada Gambar 17 bahwa dalam pengujian *Main Activity* berjalan dengan baik tanpa adanya *error*. Tombol beli menyesuaikan dengan saldo yang tersedia jika saldo tidak cukup untuk pilihan paket listrik maka tombol beli tidak menyala.



Gambar 17 Pengujian Main Activity

c. Pengujian Riwayat Pembelian

Berdasarkan Gambar 18, aktivitas *Purchase History* berfungsi dengan baik tanpa error. Halaman ini menampilkan jumlah kWh yang dibeli, bersama dengan harganya dan waktu pembelian.



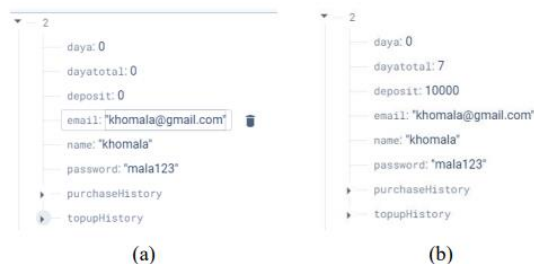
Gambar 18 Pengujian Purchase History Activity

d. Pengujian menampilkan data aktivitas pembelian daya

Gambar 19 menggambarkan bahwa data pengguna kos berhasil disimpan setelah proses registrasi selesai dilakukan. Proses penyimpanan ini mencakup beberapa komponen penting dari sistem basis data yang digunakan. Data pengguna disimpan dengan aman di Firebase Authentication, termasuk informasi penting seperti email dan password. Selain itu, data terkait penggunaan kos, seperti daya, total daya, deposit, serta data riwayat pembelian dan riwayat top up, juga disimpan dalam Firebase Realtime Database.

Firebase Realtime Database menyimpan data ini dalam format yang telah terstruktur dengan baik, memastikan bahwa informasi mengenai daya listrik, transaksi keuangan, serta detail pengguna dapat diakses dan dikelola dengan mudah dan efisien.

Dengan sistem penyimpanan ini, informasi dapat diperbarui secara real-time, memungkinkan pemilik kos dan penyewa untuk memantau dan mengelola data mereka dengan lebih baik.



Gambar 19 Data Penyewa Kos di Realtime Database (a) Data awal (b) Data Setelah Ditambahkan Deposit



Gambar 20 Pengujian Menampilkan Data Aktivitas Pembelian Daya (a) Sebelum Ditambahkan Deposit (b) Setelah Ditambahkan Deposit

Setelah pemilik kos menambahkan deposit, penyewa kos dapat melakukan pembelian sesuai dengan saldo deposit yang tersedia. Gambar 20 memperlihatkan bahwa aplikasi Android berhasil menampilkan data sisa daya dan saldo deposit yang diperoleh dari Firebase.

e. Pengujian Notifikasi

Jika daya listrik pengguna hampir habis yaitu di bawah 1 kWh aplikasi akan menampilkan notifikasi.



Gambar 21 Pengujian Notifikasi

4. KESIMPULAN

Berdasarkan hasil pembahasan dapat diambil suatu kesimpulan bahwa:

1. Perangkat *transceiver* yang dirancang untuk mengirim data hasil pembacaan sensor berupa daya listrik dari penyewa kos dan diterima pemilik kos, kemudian data tersebut diteruskan ke Firebase dan diolah untuk aplikasi. Komponen yang digunakan adalah Mikrokontroler, LoRa, Sensor PZEM-004T, Relay, LED, LCD, dan Buzzer.
2. Dalam realisasi alat *transceiver* pada pemilik dan penyewa digunakan LoRa. LoRa *node* dan LoRa *gateway* serta mikrokontroler ESP32. LoRa *node* mengirimkan data penggunaan daya listrik dari sensor PZEM-004T yang telah dikonversi menjadi sisa daya kepada LoRa *gateway*. Data ini kemudian diteruskan ke Firebase. Alat *transceiver* bekerja dengan baik dalam mengirim data hasil pembacaan sensor dengan akurasi sensor PZEM-004T mencapai hampir 99%, dengan rata-rata *error rate* sekitar 1% dari tiga pengujian beban listrik.
3. Aplikasi bekerja dengan baik dapat menampilkan sisa daya listrik yang sesuai, informasi pengguna, serta dapat melakukan pembelian paket listrik dan melihat riwayat pembelian juga menampilkan notifikasi jika daya hampir habis.

Penulis menyarankan kedepannya sistem ini dikembangkan dan dikombinasikan dengan sistem lain seperti sistem keamanan serta bisa memanfaatkan lebih banyak fitur untuk sistem manajemen kos secara keseluruhan.

DAFTAR PUSTAKA

- ADIWIRANTO, M. N., WALUYO, C. B., SUDIBYA, B., TEKNOLOGI, I., & ADISUTJIPTO, D. 2022. Prototipe Sistem Monitoring Konsumsi Energi Listrik Serta Estimasi Biaya Pada Peralatan Rumah Tangga Berbasis Internet Of Things. *Jurnal Edukasi Elektro*, 6(1), 32–41. <https://journal.uny.ac.id/index.php/jee>
- DWI ALFIAN, R. 2021. Rancang Bangun Alat Monitoring Pemakaian Tarif Listrik Dan Kontrol Daya Listrik Pada Rumah Kos Berbasis Internet Of Things Subuh Isnur Haryudo, Unit Three Kartini, Nur Kholis. *Jurnal Teknik Elektro*, 10(3), 661–670.
- GUNADI, R. J., TANONE, R., & BEEH, Y. R. 2020. Penerapan Firebase Cloud Storage Pada Aplikasi Mobile Android Untuk Melakukan Penyimpanan Image Lahan Pertanian. (*Jurnal Teknologi Informasi*) Vol.4, No.2, Desember, 4(2), 282–291.
- HASANUDIN, A. 2021. Analisis Kepuasan Konsumen Terhadap Pelayanan listrik Prabayar Dan Pascabayar Desa Banjarrejo kecamatan Batanghari Kabupaten Lampung Timur.
- HERLUKMAN, S., WIDIATMOKO, D., & KHOLID, F. 2024. Penerapan Canggih Lora Sx1278 Pada Senjata Robot. *Nucleus Journal*, 3(1), 39–47.
- HUDA, M. 2021. Analysis Of Electric Energy Consumption On Induction Motors In Production Water Treatment Installation Ii Company Regional Drinking Water (Perumda) Semarang City L.
- LUTFI TSABI, M., IRZAL ISMAIL S.T., M.T., S. J., & SULARSA S.T., M.T., A. 2020. Perancangan Sistem Otomatisasi Pemberian Pakan Kucing Menggunakan Penjadwalan Berbasis Mikrokontroler. *E-Proceeding of Applied Science*, 6(2), 3450–3459.
- MANDA KARINA, J., ANISAH, S., & HAMDANI. 2022. Studi Komparasi Kwh Meter Pascabayar Dengan Kwh Meter Prabayar Tentang Akurasi Pengukuran Terhadap Tarif Listrik Yang Bervariasi. *Jurnal Darma Agung*, Vol 30 No(2022), 488–506.
- NIZAM, M., YUANA, H., & WULANSARI, Z. 2022. Mikrokontroler Esp 32 Sebagai Alat Monitoring Pintu Berbasis Web. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 6, Issue 2).
- NUR ALFAN, A., & RAMADHAN, V. 2022. Prototype Detektor Gas Dan Monitoring Suhu Berbasis Arduino Uno. *Jurnal PROSISKO*, 9(2), 61–69.
- SANTOSA, S. P., & NUGROHO, R. M. W. 2021. Rancang Bangun Alat Pintu Geser Otomatis Menggunakan Motor Dc 24 V. *Jurnal Ilmiah Elektrokrisna Vol 9 No 1*, 9(2), 38–45.
- SETIAJI, N., SUMPENA, I. M., SUGIHARTO, A., TEKNIK ELEKTRO, J., DIRGANTARA MARSEKAL SURYADARMA ABSTRAK, U., KUNCI, K., SEMU, D., AKTIF, D., & DAYA, K. 2022. Analisis Konsumsi Daya Dan Distribusi Tenaga Listrik.
- SUGA, M. I., & NURWARSITO, H. 2021. Sistem Monitoring KWH Meter berbasis Modul Komunikasi LoRa. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 5(4), 1257–1266. <http://j-ptiik.ub.ac.id>
- SUGIYATNO. 2023. Pengiriman Informasi Real Time Menggunakan Teknologi Database Firebase pada Aplikasi Mobile Android Abstrak (Bahasa Indonesia). In *Jurnal Informatika Komputer, Bisnis dan Manajemen* (Vol. 21, Issue 2).

Halaman ini sengaja dikosongkan