

Internet of Things Applications in Android-Based Smart Homes

Aplikasi Internet of Things Pada Rumah Pintar Berbasis Android

Daniel Alexander Runtu, Pinrolivic Manembu, Xaverius Najooan

Dept. of Electrical Engineering, Sam Ratulangi University Manado, Kampus Bahu St., 95115, Indonesia

e-mails: danielruntu141@gmail.com, pmanembu@unsrat.ac.id, xnajooan@unsrat.ac.id

Received: 14 March 2024; revised: 07 December 2024; accepted: 03 March 2025

Abstract — *The advancement of technology has significantly impacted daily life by incorporating Internet of Things (IoT) and voice-based personal assistants. However, the complexity of using personal assistants to control household electronic devices has increased alongside IoT advancements. The surge in electricity consumption due to modern lifestyles also presents a challenge. In this context, this research proposes the development of an Android application utilizing IoT to control household electronic devices automatically or remotely. Thus, this application is expected to assist users in reducing electricity bills and mitigating the environmental impact of excessive electricity production.*

Key words — *Android, Internet of Things, Smart House.*

Abstrak — Perkembangan teknologi telah membawa dampak signifikan dalam memudahkan kehidupan sehari-hari melalui penerapan Internet of Things (IoT) dan asisten pribadi berbasis suara. Namun, kompleksitas penggunaan asisten pribadi untuk mengontrol alat elektronik rumah telah meningkat seiring dengan kemajuan IoT. Peningkatan konsumsi energi listrik sebagai akibat dari gaya hidup modern juga menjadi tantangan yang perlu diatasi. Dalam konteks ini, penelitian ini mengusulkan pengembangan aplikasi Android yang menggunakan IoT untuk mengontrol perangkat elektronik rumah secara otomatis atau jarak jauh. Dengan demikian, aplikasi ini diharapkan dapat membantu pengguna menghemat biaya tagihan listrik dan mengurangi dampak lingkungan yang dihasilkan dari produksi energi listrik yang berlebihan.

Kata kunci — *Android, Internet of Things, Rumah Pintar.*

I. PENDAHULUAN

Perkembangan teknologi saat ini dapat meliputi hampir seluruh hal yang ada di masyarakat. Hal ini berguna untuk mempermudah dan menghemat waktu dalam membantu pekerjaan manusia dan menanggulangi beberapa masalah. Dari perkembangan teknologi yang terjadi muncul penerapan dan terciptanya alat-alat elektronik yang dapat digunakan secara otomatis atau jarak jauh.

Dalam beberapa tahun terakhir, *Internet of Things* sudah menjadi teknologi yang populer. *Internet of Things* mengacu pada koneksi perangkat yang berbeda-beda seperti sensor, alat elektronik, dan perangkat elektronik yang dapat berkomunikasi satu sama lain melalui internet. Konsep IoT telah menawarkan banyak peluang baru untuk meningkatkan kualitas hidup masyarakat melalui solusi pintar untuk rumah dan kehidupan sehari-hari.

Peningkatan penggunaan alat elektronik di rumah seiring dengan perkembangan teknologi dan gaya hidup modern telah mengakibatkan konsumsi energi listrik yang semakin meningkat. Hal ini dapat menyebabkan kenaikan biaya tagihan listrik dan juga menimbulkan masalah lingkungan karena produksi listrik yang lebih besar dapat menyebabkan emisi gas rumah kaca yang lebih tinggi.

Untuk mengatasi masalah ini, penggunaan *Internet of Things* (IoT) dalam memantau konsumsi energi listrik di rumah telah menjadi solusi yang semakin populer. IoT adalah teknologi yang memungkinkan perangkat untuk terhubung dan berkomunikasi melalui jaringan internet, sehingga dapat melakukan tugas-tugas secara otomatis dan cerdas.

Dalam konteks tersebut, penelitian ini bertujuan untuk mengembangkan sebuah aplikasi IoT pada rumah pintar berbasis platform Android. Aplikasi ini dirancang untuk memberikan kontrol yang efisien dan pemantauan terhadap berbagai perangkat elektronik yang umumnya ditemukan di rumah, serta memantau kondisi lingkungan rumah secara real-time. Selain itu, aplikasi ini juga akan memungkinkan pengguna untuk memantau penggunaan energi secara detail, dengan tujuan untuk meningkatkan kesadaran akan konsumsi energi dan mendorong penggunaan energi yang lebih efisien di tingkat rumah tangga.

Berdasarkan hal tersebut, untuk mempermudah manusia dalam mengontrol perangkat elektronik menggunakan smartphone melalui jaringan internet diperlukan aplikasi *Android* untuk mengontrol perangkat elektronik rumah agar dapat digunakan secara otomatis ataupun jarak jauh dan juga memantau kondisi lingkungan rumah serta penggunaan energi yang digunakan, maka dibuatlah “Aplikasi IoT pada Rumah Pintar berbasis *Android*”.

A. Penelitian Terkait

- 1) “Rancang Bangun Aplikasi Asisten Pribadi Pengontrol Alat Elektronik Rumah Berbasis IoT” yang membahas mengenai aplikasi yang dapat mengontrol alat elektronik rumah menggunakan perintah suara.
- 2) “Aplikasi Smartphone Pengontrol Lampu Rumah menggunakan Internet of Things”, yang membahas mengenai mematikan dan menghidupkan lampu rumah dari jarak jauh menggunakan aplikasi smartphone.
- 3) Suraidi, Shinta Nathania (2019) dengan judul “Sistem Pengendali Smart-Kontak dengan Aplikasi Android dan

Web”, dimana penelitian ini membahas mengenai bagaimana cara mengendalikan smart-kontak melalui aplikasi android dan aplikasi web.

- 4) Tracy M. Kereh, Sukandar Sawidin, Yohanis S. Rompon, Deitje S. Pongoh (2022) dengan judul “*Prototype Sistem Kontrol Peralatan Listrik dengan Aplikasi Android Voice Controller*”, penelitian ini membahas mengenai mengontrol peralatan listrik menggunakan perintah suara sesuai dengan yang pengguna ucapkan.
- 5) Dedi Putra (2022) dengan judul “*Rancang Bangun Smarthome Pengamanan Kelistrikan Berbasis Android*”, penelitian ini membahas tentang merancang dan mengimplementasikan sebuah sistem *Smarthome* yang dapat meningkatkan efisiensi dan kenyamanan dalam mengelola perangkat rumah, dengan fokus pada keamanan dan penghematan energi.
- 6) Tatak Prihadi Prihantoro (2021) dengan judul “*Sistem Smart Home IoT di Perumahan Ndalem Parikesit*”, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem *Smarthome* berbasis NodeMCU dan WiFi, yang dapat dikendalikan melalui smartphone untuk meningkatkan efisiensi, kenyamanan, keamanan, dan penghematan energi dalam pengelolaan perangkat rumah secara adaptif dan cerdas.
- 7) Aan Restu Mukti, Widyanto, Chairul Mukmin, Elpan Randa Kasih (2022) dengan judul “*Perancangan Smart Home menggunakan Konsep Internet of Things (IoT) berbasis Microcontroller*”, Penelitian ini bertujuan merancang sistem *smart home* berbasis Internet of Things (IoT) menggunakan modul relay dan koneksi *WiFi* untuk mengendalikan perangkat rumah melalui aplikasi *Android*, guna mengatasi kendala manualitas dan meningkatkan pemahaman serta penerapan konsep rumah pintar yang terintegrasi dan efisien.
- 8) Irvandi (2022) dengan judul “*Perancangan Prototype Alat Monitoring Listrik pada Rumah Tangga berbasis IoT*”, Penelitian ini bertujuan membangun alat monitoring listrik rumah tangga berbasis IoT yang mengumpulkan data penggunaan daya secara real-time, sehingga pengguna dapat mengidentifikasi, mengontrol peralatan berdaya besar, dan mendeteksi kebocoran arus untuk mengelola penggunaan listrik secara lebih efisien.

B. Meteran Listrik

Meteran listrik merupakan jembatan yang memiliki fungsi sebagai penghubung antara pengguna dengan pihak penyedia jasa, meteran listrik biasa digunakan sebagai pusat kendali energi listrik, kemudian untuk melihat total kWh tersisa dan juga biasa digunakan untuk mengisi ulang pulsa atau token yang akan dikonversikan menjadi saldo kWh, kemudian terdapat kata “Prabayar” merupakan istilah sistem pembayaran yang dilakukan sebelum menerima layanan dari penyedia jasa (Sari, M., & Prasetyo, D. E. 2002). Meteran listrik yang dihubungkan ke banyak rumah tentunya memiliki beberapa tipe yang berbeda, pemerintah menetapkan harga per-kWh yang berbeda berdasarkan kondisi ekonomi dari pengguna.

C. Internet of Things

Internet of Things (IoT) adalah konsep yang memperluas manfaat konektivitas internet dengan menghubungkan mesin, peralatan, dan benda fisik lainnya melalui sensor dan aktuator untuk mengumpulkan data dan mengelola kinerjanya sendiri. Kevin Ashton pertama kali mengusulkan ide ini pada tahun 1999. Saat ini, banyak perusahaan besar seperti Intel, Microsoft, dan Oracle mulai menerapkan teknologi IoT. IoT dianggap sebagai “*the next big thing*” dalam dunia teknologi informasi karena potensinya yang besar. Contoh implementasinya termasuk kulkas yang dapat memberitahu pemiliknya tentang persediaan makanan melalui SMS atau email.

D. Message Queuing Telemetry Transport (MQTT)

Message Queuing Telemetry Transport (MQTT) merupakan jenis protokol jaringan pada IoT (*Internet of Things*) yang dimana berfungsi sebagai komunikasi yang bersifat *machine to machine* atau M2M dapat bekerja pada bandwidth yang rendah sangat cocok untuk digunakan dalam *internet of things* (IoT) yang dapat bekerja dalam energi dan media penyimpanan yang minimum serta sifatnya yang ringan dalam pengiriman pesan. Pada protokol MQTT (*Message Queuing Telemetry Transport*) sendiri menggunakan metode komunikasi *publish/subscribe* yang dimana *publish* berfungsi sebagai pengirim pesan dan *subscribe* sebagai penerima pesan menggunakan protokol MQTT sendiri dimana dalam MQTT terdapat *broker* yang berfungsi sebagai topik yang dikirimkan oleh publisher untuk diteruskan ke *subscribe* berdasarkan permintaan dari pengguna

E. Monitoring

Dalam pengertian lain, *monitoring* merupakan pemantauan yang dapat dijelaskan sebagai kesadaran tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan kearah tujuan atau menjauh dari itu. *Monitoring* akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu (Zaida & Sunardi, 2019).

F. Android

Menurut (Warangkiran, Kaunang, Lumenta, & St, 2014), *Android* adalah sistem operasi untuk telepon seluler yang berbasis Linux. *Android* utamanya adalah produk Google, tetapi lebih tepatnya bagian dari *Open Handset Alliance*. *Open Handset Alliance* merupakan aliansi dari 30 organisasi yang berkomitmen untuk membawa sebuah perangkat seluler yang lebih baik dan terbuka untuk pasar. *Android* termasuk kernel berbasis Linux, aplikasi *end-user*, dan *framework* aplikasi. *User application* dibangun berbasiskan bahasa pemrograman Java. Bahkan aplikasi yang dibangun juga berbasiskan Java.

G. Flutter

Flutter adalah *framework open-source* yang dikembangkan oleh Google untuk membangun antarmuka pengguna (UI)

perangkat lunak yang responsif, indah, dan native secara lintas platform. Flutter menggunakan bahasa pemrograman Dart untuk pengembangan aplikasinya

H. Arduino IDE

Arduino IDE merupakan platform yang digunakan untuk pengolahan program atau wadah dari perintah yang diinginkan oleh pengguna, bahasa pemrograman yang digunakan untuk melakukan penulisan program ialah C/C++, program yang ditulis akan diunggah pada mikrokontroler sesuai dengan yang diinginkan oleh pengguna, Arduino sendiri memiliki cakupan yang luas dalam bidang pemrograman, banyaknya fitur dapat menyesuaikan bidang dari sistem atau alat yang akan dibangun.

I. Android SDK

Android SDK terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi, contoh kode program dan tutorial. Saat ini Android sudah dapat digunakan di platform x86 pada Linux (distribusi Linux apapun untuk desktop modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau lebih. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan *plugin Android Development Tools* (ADT), dengan ini pengembang dapat menggunakan IDE untuk mengedit dokumen Java dan XML serta menggunakan peralatan *command line* untuk menciptakan, membangun, melakukan *debug* aplikasi Android dan pengendalian perangkat Android (misalnya *reboot*, menginstal paket perangkat lunak).

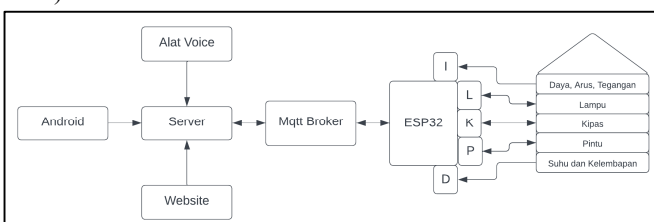
J. Agile Development

Agile development adalah sebuah konsep pengembangan yang memiliki arti cepat dilakukan, cepat merespon perubahan yang diminta oleh klien, dan melibatkan secara aktif dalam proses sehingga produk yang dihasilkan atau perangkat lunak yang dikembangkan merupakan hasil dari setiap pihak yang terlibat (Akbar, 2017). Metode *Agile Development* memiliki beberapa model pengembangan yang salah satunya adalah *scrum*.

II. METODE

A. Arsitektur Sistem

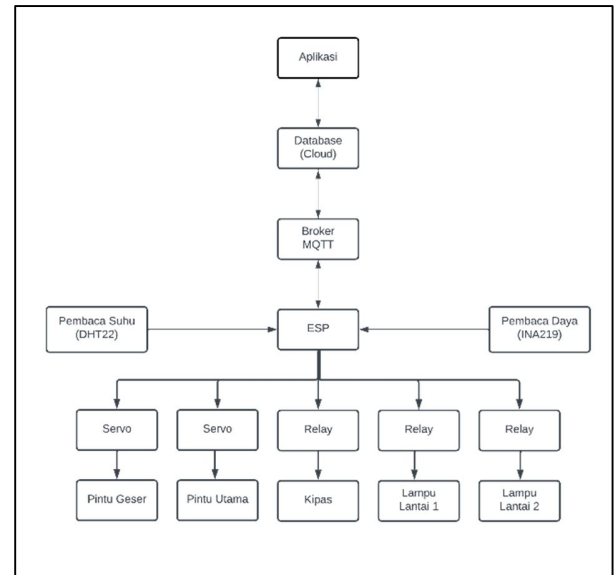
Untuk arsitektur sistem ini memiliki 3 perangkat yang saling berhubungan melalui server. Data yang dikeluarkan dari ketiga perangkat, alat voice, website dan android akan dimasukkan ke server untuk disimpan atau ketiga perangkat juga dapat mengambil nilai dari server seperti nilai lingkungan rumah (suhu dan kelembapan) dan energi (tegangan, arus, daya dan kwh).



Gambar 2.1 Arsitektur Rumah Pintar

B. Proses dan Kerja Sistem

Proses kerja dari aplikasi ini diterapkan pada prototipe rumah pintar yang memiliki 2 lantai pada lantai pertama terdapat dua lampu dan dua pintu dan pada lantai dua terdapat dua lampu, satu sensor suhu dan satu sensor untuk membaca arus, tegangan dan daya yang dapat dikontrol serta dipantau melalui aplikasi.



Gambar 2.2 Sistem kerja rumah pintar

Berdasarkan gambar 2.2, sistem kerjanya adalah:

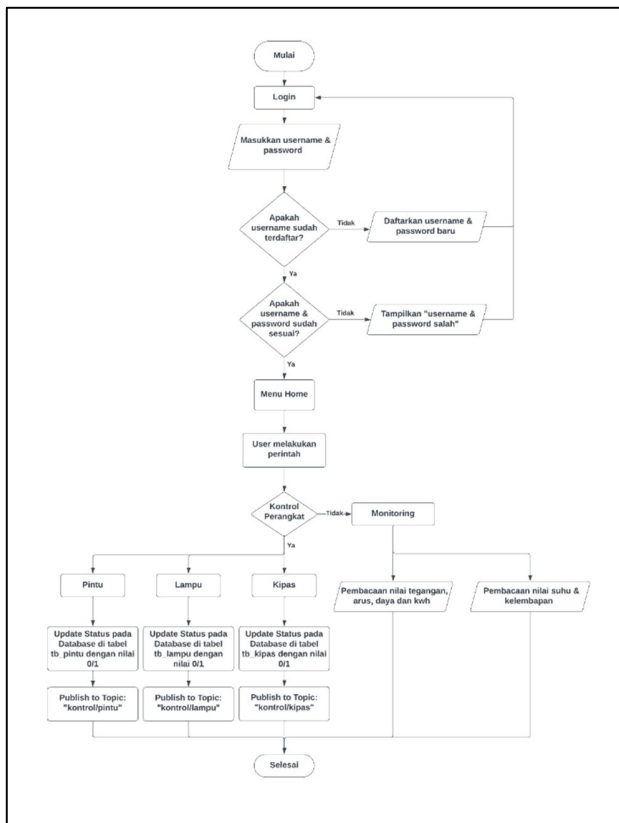
1. Aplikasi yang dibuat dapat mengontrol komponen yang terdapat pada ESP32 menggunakan *button* yang ada didalamnya untuk mematikan atau menyalakan device berupa lampu dan kipas sebagai pendingin ruangan serta membuka dan menutup pintu.
2. Database berfungsi untuk menyimpan nilai yang akan menjadi patokan untuk mengontrol device pada esp.
3. DHT22 (Pembaca Suhu) dan INA219 (Pembaca Daya) mengirim data ke ESP kemudian meneruskannya ke database yang nantinya akan ditampilkan pada aplikasi.
4. Aplikasi dapat menampilkan suhu dan juga kelembapan yang diperoleh dari pembacaan yang diperoleh dari ESP.
5. Aplikasi dapat memperlihatkan data dari arus, tegangan juga daya yang digunakan pada saat itu dan merata-ratakannya per hari.

C. Perancangan Perangkat Lunak

Aplikasi yang diprogram memiliki alur dari pengguna masuk kedalam aplikasi kemudian perlu untuk memasukkan username dan juga password agar tidak bisa sembarangan pengguna mengontrol rumah atau memantaunya dari aplikasi. Jika sudah memasukkan *username* dan juga *password*, pengguna akan masuk ke dalam aplikasi dimana mereka dapat mengontrol perangkat yang ada pada rumah pintardan juga melakukan monitoring dari suhu, kelembapan dan juga daya yang

digunakan pada rumah.

Hal pertama yang dibuat adalah *flowchart* untuk masuk ke dalam aplikasi untuk memastikan keamanan dari aplikasi tidak dapat masuk yang tidak memiliki username ataupun password. Pada gambar 2.3, penulis membuat *flowchart* untuk jalan sistem pada aplikasi.

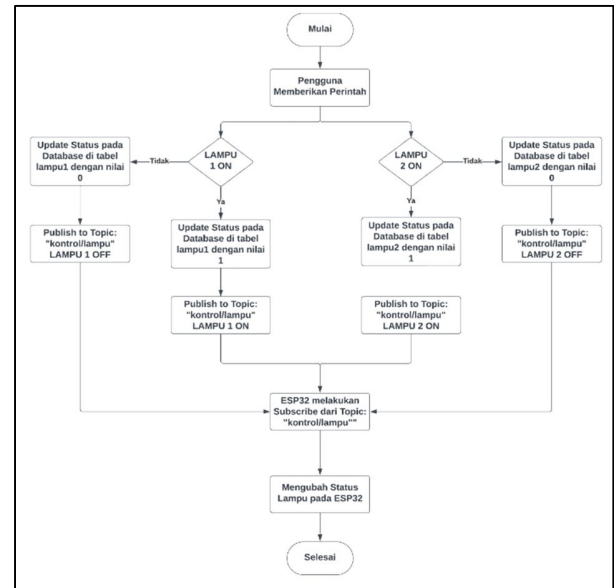


Gambar 2.3 Flowchart Aplikasi

Di gambar 2.3 dijelaskan alur dari proses pengguna menggunakan aplikasi. Pada saat pertama kali masuk ke dalam aplikasi, pengguna akan diarahkan untuk memasukkan username dan password. Jika sudah sistem akan memastikan apakah kedua data tersebut terdaftar pada database atau tidak. Jika tidak pengguna perlu mendaftarkan username dan password baru pada menu “*register*” dan jika data yang dimasukkan terdaftar sistem akan melanjutkan proses apakah username dan password yang dimasukkan sesuai. Apabila tidak sesuai sistem akan mengarahkan ulang ke bagian menu “*login*”. Jika sesuai pengguna akan diarahkan langsung ke dalam aplikasi untuk melakukan monitoring dan juga mengontrol device yang terdapat pada smart house.

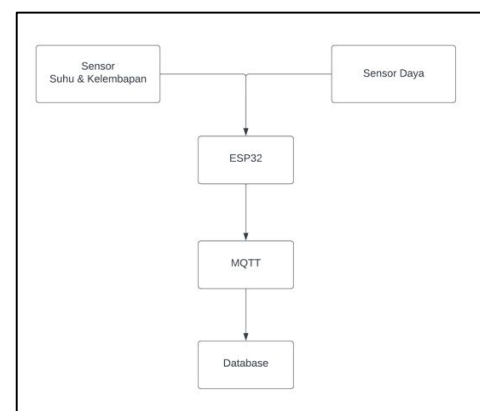
Selanjutnya penulis membuat *flowchart* untuk sistem kontrol dari perangkat. Setelah pengguna menyelesaikan proses *login*. Pengguna akan masuk ke dalam aplikasi. Jika pengguna mengaktifkan atau mematikan device maka aplikasi akan mengirim *string* yang telah ditentukan ke topik pada broker MQTT yang telah dibuat. Jika sudah berhasil mengirimkan string ke dalam MQTT, sistem akan mengkoneksikannya dengan database agar dapat mengontrol device dengan cara

mengirimkan data “0 dan 1” atau “*High*” dan “*Low*” dari aplikasi yang nantinya data pada database akan berubah yang akan dibaca oleh ESP32 untuk menyalakan atau mematikan device. Jika sinyal *High* atau 1 maka device akan menyala, sebaliknya jika data yang dibaca oleh ESP32 adalah 0 atau *Low* maka device akan mati.



Gambar 2.4 Flowchart Sistem Kontrol Device

Pada gambar 2.5 menunjukkan *flowchart* dari *monitoring* daya dan suhu. Pertama sensor suhu dan daya akan mengambil data dari energi listrik yang digunakan serta suhu dan kelembapan pada rumah. Kemudian akan meneruskannya ke ESP32 yang akan dikirimkan ke database agar dapat ditampilkan di aplikasi. Pada aplikasi, data daya akan di kalkulasikan menjadi kwh setiap pengiriman datanya ke database.



Gambar 2.5 Flowchart Monitoring

D.Desain Prototype Rumah Pintar

Desain prototype yang dibuat bertujuan untuk menyederhanakan tahap perakitan dari rumah pintar yang akan diimplementasikan pada platform IoT. Penjelasan rinci mengenai struktur ini dapat ditemukan dalam gambar yang disajikan di bawah ini.



Gambar 2.6 Tampak Depan Rumah



Gambar 2.7 Tampak Samping Rumah



Gambar 2.8 Tampak Belakang Rumah

E. Skenario Pengujian

Kita akan melakukan pengujian fungsional terhadap kontrol perangkat pada prototipe rumah pintar, yang meliputi lampu, kipas sebagai pendingin ruangan, dan pintu. Pengujian akan memeriksa apakah setiap perangkat dapat dikontrol melalui aplikasi yang telah dibuat, dengan fokus pada fungsionalitas pengendalian masing-masing perangkat melalui aplikasi tersebut.

Kemudian kita membuat skenario pengujian untuk pembacaan monitoring untuk pembaca. Penulis akan membuat skenario dimana membandingkan pembacaan suhu menggunakan thermometer dan pembacaan menggunakan sensor DHT22, data yang dihasilkan oleh sensor apakah akurat dari nilai yang diambil menggunakan thermo.

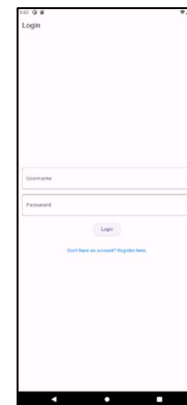
Dan juga kita membuat skenario untuk menguji pembacaan daya yang diperoleh menggunakan sensor. Beban yang digunakan adalah Kipas 12V dan juga LED Strip sebanyak dua buah.

III. HASIL DAN PEMBAHASAN

Hasil penelitian ini adalah sebuah aplikasi Android berbasis *Internet of Things* (IoT) yang bertujuan untuk mempermudah pengguna dalam mengontrol perangkat elektronik rumah secara otomatis atau jarak jauh. Aplikasi ini juga menyediakan prediksi pemakaian listrik, memungkinkan pengguna untuk mengoptimalkan konsumsi energi mereka. Dengan demikian, aplikasi ini diharapkan dapat membantu pengguna menghemat biaya tagihan listrik dan mengurangi dampak lingkungan dari produksi energi listrik yang berlebihan.

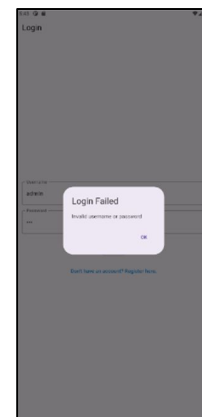
A. Pembuatan Aplikasi

Pertama, kita akan membuat halaman login untuk aplikasi, yang disebut *login_page.dart*. Di halaman login ini, pengguna akan diminta untuk memasukkan username dan password untuk mengakses aplikasi. Pengguna harus memasukkan data yang telah terdaftar dan disimpan di dalam database; jika belum terdaftar, mereka akan diminta untuk mendaftarkan akun terlebih dahulu.



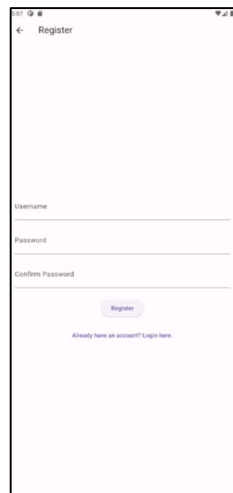
Gambar 3.1 Tampilan Login Aplikasi

Untuk memeriksa apakah data pengguna sudah ada, kita memerlukan file PHP yang memeriksa apakah data yang dimasukkan pengguna ke dalam aplikasi betul ataupun sudah ada di database. Jika betul atau sudah ada maka aplikasi akan meneruskannya ke halaman beranda, apabila salah atau belum ada maka pengguna akan diberikan notifikasi "*login failed*" dan pengguna perlu membuat akun terlebih dahulu di bagian registrasi.



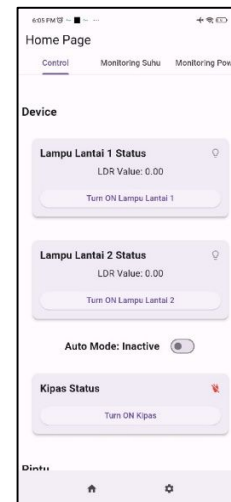
Gambar 3.2 Tampilan Aplikasi tidak Berhasil Login

Selanjutnya akan dibuat tampilan untuk pengguna melakukan registrasi apabila tidak memiliki akun. Aplikasi akan meminta pengguna untuk mengisi beberapa hal untuk mendaftarkan akun yaitu, username, password dan konfirmasi password untuk memastikan ulang password yang dimasukkan apakah sudah betul. Jika mengisi informasi yang diperlukan maka terdapat file php yang akan menyimpan informasi yang sudah dimasukkan tadi ke dalam tabel database.



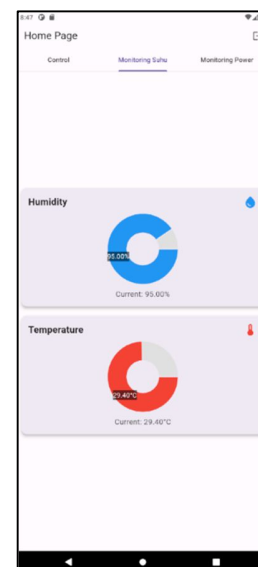
Gambar 3.3 Tampilan Registrasi Aplikasi

Selanjutnya, kita akan membuat tampilan untuk fitur kontrol dalam aplikasi. Pada tab pertama fitur kontrol, akan dibuat sesuai dengan jumlah lampu dan kipas yang ada, yaitu 2 lampu dan 1 kipas. Lampu akan diberi nama "Lampu Lantai 1" dan "Lampu Lantai 2", serta disertakan status apakah lampu tersebut menyala atau mati yang ditandai dengan ikon lampu. Kipas akan ditandai dengan ikon saklar untuk menunjukkan statusnya. Terdapat tombol ON dan OFF untuk mengontrol daya nyalakan/mati perangkat dari lampu dan kipas, yang akan mengirimkan perintah ke file `control_fanLamp.php`. Selain itu, terdapat juga tombol *elevated button* untuk mengaktifkan mode otomatis jika pengguna malas untuk menyalakan atau mematikan lampu secara manual. Mode otomatis meminta pengguna memasukkan waktu untuk menyalakan dan mematikan lampu, dan lampu akan berfungsi secara otomatis sesuai waktu yang ditentukan. Serta untuk menandakan lampu sudah menyala atau belum terdapat nilai LDR yang akan menjadi indikator lampu sudah berfungsi dengan baik. Kita juga akan membuat tampilan untuk mengontrol pintu dalam aplikasi. Pada tampilan tab pintu terdapat dua card yang akan ditampilkan yaitu Pintu Utama dan Pintu Samping beserta status dari pintu apakah Terbuka ditunjukkan dengan icon lock berwarna hijau atau Tertutup yang ditunjukkan dengan icon lock berwarna merah. Dan juga terdapat button Open dan Close untuk membuka dan menutup pintu yang akan mengarahkan perintah ke `control_pintu.php`.



Gambar 3.4 Tampilan Halaman Kontrol

Kemudian kita akan membuat tampilan untuk melakukan monitoring suhu dan kelembapan. Untuk data dari tampilan monitoring ini didapatkan dari sensor DHT11 yang akan dikirimkan ke database dengan nama file `baca_dht.php` yang berfungsi untuk membaca nilai suhu dan kelembapan yang dikirimkan dari ESP32 ke database.

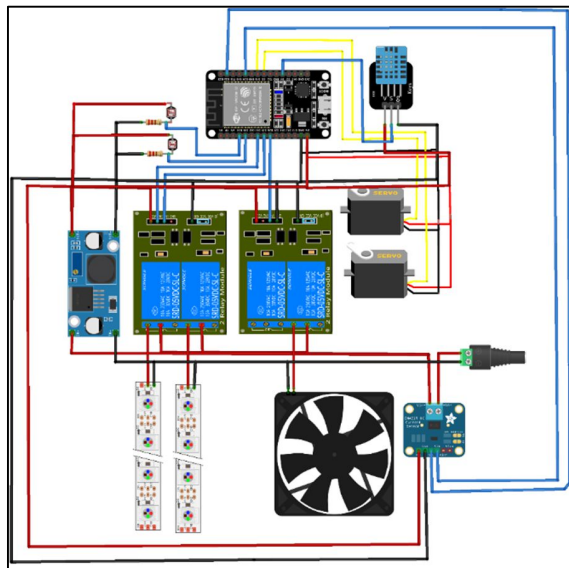


Gambar 3.5 Tampilan Monitoring Suhu

Yang terakhir kita akan membuat tampilan untuk melakukan monitoring tegangan, arus dan daya. Untuk data dari tampilan monitoring ini didapatkan dari sensor INA219 yang akan dikirimkan ke database dengan nama file `baca_ina219.php` yang berfungsi untuk membaca nilai tegangan, arus dan daya yang dikirimkan dari ESP32 ke database. Pada tampilan ini terdapat empat grafik yang dapat dipilih oleh pengguna, yaitu tegangan, arus, daya dan kwh yang ditampilkan sesuai dengan jam nilai tersebut dikirim oleh ESP32. Untuk data tegangan, arus dan daya akan diambil dari database sesuai nilai yang dikirimkan oleh ESP32 melalui sensor INA219.

Gambar 3.14 Tampilan Monitoring Daya

B. Rangkaian Komponen Smart House



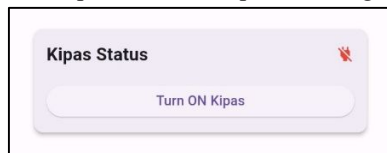
Gambar 3.6 Wiring untuk Rumah Pintar

Berikut ini merupakan rangkaian elektronika dari smart house. Pada smart house memiliki beberapa komponen yaitu, dua buah *LED Strip*, dua buah *relay dual channel*, satu *fan 12v*, satu *sensor INA219*, satu *DHT11*, satu regulator, satu buah *ESP32*, dua buah *LDR* dan *power supply 12v*.

C. Uji Coba Perangkat

Untuk pengujian yang pertama kita akan menguji fungsi control dari aplikasi ke perangkat yang berada di rumah pintar. Pengujian akan dilakukan pada prototipe rumah pintar yang akan dikendalikan pada aplikasi secara langsung menggunakan komunikasi MQTT. Dalam pengujian ini dari aplikasi (publisher) akan mengirimkan string atau pesan payload tertentu ke subscriber melalui topik MQTT yang sudah dibuat sebelumnya.

Untuk pengujian yang pertama kita akan membuat pengujian untuk mengontrol perangkat sebagai percobaan yaitu kipas pada prototipe rumah pintar melalui aplikasi sebagai publisher.



Gambar 3.7 Status Kipas pada Aplikasi

Jika user menekan button Turn ON Kipas pada aplikasi maka aplikasi akan mem-publish pesan atau mengirimkan payload berupa “KIPAS 1 ON” ke dalam topik kontrol/kipas agar dapat disubscribe oleh ESP32 yang dapat mengontrol kipas melalui relay yang tersambung dengan ESP32 agar dapat menyalakan kipasnya.



Gambar 3.8 Pesan pada Topik kontrol/kipas

Jika sudah berhasil mengirimkan pesan pada topik di MQTT, maka akan mengubah nilai pada database di table tb_kipas pada kolom “kipas” menjadi 1 yang berarti kipas sudah menyala.



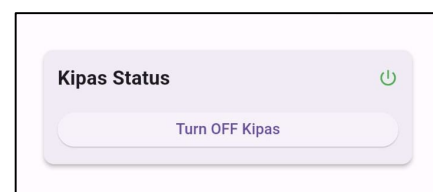
Gambar 3.9 Nilai pada Database Berubah

Kemudian kita akan memperlihatkan hasil pengujian menyalakan kipas pada prototipe rumah pintar melalui aplikasi sudah berhasil dilakukan.



Gambar 3.10 Kipas Berhasil Dinyalakan

Selanjutnya untuk mematikan kipasnya, seperti dengan menyalakannya yang harus dilakukan adalah menekan button pada aplikasi agar dapat mengontrolnya.



Gambar 3.11 Button Turn Off Kipas

Jika user menekan button Turn OFF Kipas pada aplikasi maka aplikasi akan mem-publish pesan atau mengirimkan payload berupa “KIPAS 1 OFF” ke dalam topik kontrol/kipas agar dapat disubscribe oleh ESP32 yang dapat mengontrol kipas melalui relay yang tersambung dengan ESP32 agar dapat mematikan kipasnya.



Gambar 3.12 Pesan Dikirimkan ke Topik

Jika sudah berhasil mengirimkan pesan pada topik di MQTT, maka akan mengubah nilai pada database di table `tb_kipas` pada kolom “kipas” menjadi 0 yang berarti kipas sudah dalam kondisi mati.



Gambar 3.13 Nilai pada Database Berubah Menjadi 0

Kemudian kita akan memperlihatkan hasil pengujian mematikan kipas pada prototipe rumah pintar melalui aplikasi sudah berhasil dilakukan. Sebagai indikasi kipas telah mati, tisu di depan kipas sudah tidak berterbangan yang menandakan kipas telah mati.



Gambar 3.14 Kipas pada Prototipe Rumah Pintar dalam Kondisi Mati

Untuk kedua perangkat lainnya dilakukan dengan metode yang sama. Berdasarkan percobaan yang telah dilakukan terhadap perangkat yang ada pada prototipe rumah pintar, berfungsi dengan baik. Pesan berupa string yang dikirimkan oleh pengguna melalui aplikasi ke dalam topik yang telah ditentukan berhasil dilakukan. Serta proses subscribe yang dilakukan oleh ESP32 untuk mengaktifkan atau menonaktifkan perangkat berfungsi dengan baik, apabila pesan atau string yang telah ditentukan di ESP32 berbeda dengan yang ada pada MQTT maka proses subscribe akan gagal dilakukan karena ESP32 tidak menemukan string yang sesuai. Walaupun jika topik tidak sesuai dengan yang telah dibuat pada aplikasi dan dikirimkan pada MQTT pada proses subscribe juga tidak dapat dilakukan. Oleh karena itu, untuk dapat mengendalikan perangkat yang berada pada prototipe rumah pintar kita harus memastikan topik yang digunakan sesuai dengan publisher dan subscribe agar pesan dapat dikirimkan dengan selaras serta mematikan pesan atau string yang dikirimkan oleh publisher dapat terbaca oleh subscriber yang telah kita tentukan sehingga pengujian untuk mengontrol perangkat dapat dilakukan dengan

baik. Hasil dari pengujian pengontrolan perangkat prototipe rumah pintar melalui aplikasi Android dapat dilihat pada tabel 3.1.

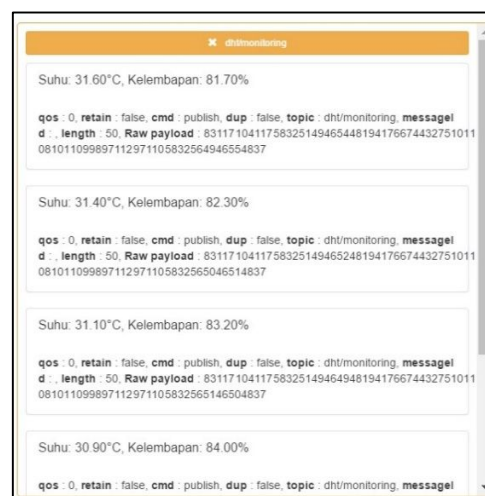
TABEL 3.1

RANCANGAN SKENARIO PENGUJIAN PENGONTROLAN PERANGKAT PADA APLIKASI

Kasus Pengujian	Hasil Pengujian
Menyalakan dan Mematikan Kipas melalui Aplikasi	Berhasil
Menyalakan dan Mematikan Lampu Dapur melalui Aplikasi	Berhasil
Menyalakan dan Mematikan Lampu Ruang Tengah melalui Aplikasi	Berhasil
Membuka dan Menutup Pintu Utama melalui Aplikasi	Berhasil
Membuka dan Menutup Pintu Geser melalui Aplikasi	Berhasil

Untuk tahap pengujian yang berikutnya, kita akan melakukan pengujian untuk sistem monitoring terhadap suhu dan kelembapan ruangan di prototipe rumah pintar, dimana kita akan memeriksa pembacaan suhu dan kelembapan menggunakan sensor DHT22 yang datanya nanti akan disimpan pada database dan ditampilkan pada aplikasi. Setiap satu menit data suhu dan kelembapan akan dikirimkan menggunakan protokol MQTT.

Yang pertama kita akan lakukan melihat pengiriman data suhu dan kelembapan oleh ESP32 yang dibaca menggunakan DHT22. Data yang dikirimkan akan muncul pada topik yang telah kita tentukan, dalam hal ini topiknya bernama “dht/monitoring” dengan broker MQTT yang digunakan yaitu “public.mqttserver.eu”. Untuk pengiriman data yang dilakukan ke MQTT dapat dilihat pada gambar dibawah ini.



Gambar 3.15 Pengiriman Data Suhu dan kelembapan pada Broker MQTT

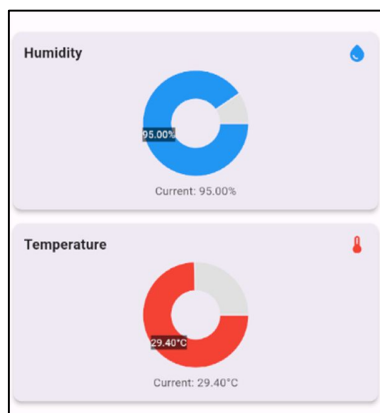
Jika data suhu dan kelembapan sudah terbaca pada broker MQTT maka datanya akan diteruskan ke database pada table

tb_monitoring dan disimpan sesuai dengan nama kolom suhu, kelembapan dan juga waktu datanya terkirim ke database. Data akan terus bertambah setiap satu menit sesuai dengan pengiriman data yang dilakukan oleh ESP32.

	id	suhu	kelembapan	tanggal
☐ Edit ☐ Copy ☐ Delete	192	31.70	81.00	2024-06-24 16:09:50
☐ Edit ☐ Copy ☐ Delete	191	31.60	81.70	2024-06-24 16:08:50
☐ Edit ☐ Copy ☐ Delete	190	31.40	82.30	2024-06-24 16:07:50
☐ Edit ☐ Copy ☐ Delete	189	31.10	83.20	2024-06-24 16:06:50
☐ Edit ☐ Copy ☐ Delete	188	30.90	84.00	2024-06-24 16:05:50
☐ Edit ☐ Copy ☐ Delete	187	30.70	84.50	2024-06-24 16:04:48

Gambar 3.16 Nilai Suhu dan Kelembapan pada Database

Data yang tersimpan pada database akan dimunculkan pada aplikasi dengan tampilan grafik lingkaran yang akan ter-update secara otomatis ketika ada nilai baru yang masuk pada database.



Gambar 3.17 Tampilan Suhu dan Kelembapan pada Aplikasi

Berdasarkan dari percobaan yang telah dilakukan, monitoring yang dilakukan berfungsi dengan baik. Proses pengiriman data dari ESP32 sampe ke database berjalan dengan baik dan juga dapat menampilkan nilainya pada aplikasi. Dari percobaan yang dilakukan, menulis melakukan pengujian dengan cara mengambil nilai suhu menggunakan 2 metode yaitu yang pertama nilai yang dikirimkan oleh ESP32 dan yang kedua menggunakan thermometer. Dengan dua metode yang dilakukan ini kita dapat mendapatkan perbandingan nilai yang didapatkan dari kedua perangkat dan mendapatkan nilai error yang diperoleh dari DHT22.

TABEL 3.2

RANCANGAN SKENARIO PENGUJIAN PEMBACAAN SUHU			
	Nilai Sensor	Nilai Thermo	Error
Suhu	30.7 °C	30.2 °C	1.65%
	30.9 °C	30.3 °C	1.98%
	31.1 °C	30.3 °C	2.64%
	31.4 °C	30.3 °C	3.63%
	31.6 °C	30.3 °C	4.29%
	Rata-Rata		2.83%

Untuk tahap pengujian yang berikutnya, kita akan melakukan pengujian untuk sistem monitoring terhadap daya yang terpakai pada prototipe rumah pintar, dimana kita akan memeriksa pembacaan tegangan, arus dan daya menggunakan sensor

INA219 yang datanya nanti akan disimpan pada database dan ditampilkan pada aplikasi. Setiap satu menit data tegangan, arus dan daya akan dikirimkan menggunakan protokol MQTT.

Yang pertama kita akan lakukan melihat pengiriman data tegangan, arus dan daya oleh ESP32 yang dibaca menggunakan INA219. Data yang dikirimkan akan muncul pada topik yang telah kita tentukan, dalam hal ini topiknya bernama “ina219/monitoring” dengan broker MQTT yang digunakan yaitu “public.mqttserver.eu”. Untuk pengiriman data yang dilakukan ke MQTT dapat dilihat pada gambar dibawah ini.



Gambar 3.18 Pengiriman Data INA219 pada Broker MQTT

Jika datanya sudah terbaca pada broker MQTT maka data tersebut akan diteruskan ke database pada table ina219_data dan disimpan sesuai dengan nama kolom *voltage*, *current*, *power*, *total_kwh* dan juga waktu datanya terkirim ke database. Data akan terus bertambah setiap satu menit sesuai dengan pengiriman data yang dilakukan oleh ESP32.

	id	voltage	current	power	total_kwh	timestamp
☐ Edit ☐ Copy ☐ Delete	134	11.68	0.44	5.09	0.002000000	2024-06-24 07:27:55
☐ Edit ☐ Copy ☐ Delete	133	11.7	0.45	5.1	0.002000000	2024-06-24 07:25:43
☐ Edit ☐ Copy ☐ Delete	132	11.72	0.44	5.25	0.002000000	2024-06-24 07:23:46
☐ Edit ☐ Copy ☐ Delete	131	11.72	0.44	5.13	0.001000000	2024-06-24 07:22:41
☐ Edit ☐ Copy ☐ Delete	130	11.7	0.45	5.09	0.001000000	2024-06-24 07:21:40
☐ Edit ☐ Copy ☐ Delete	129	11.68	0.44	5.11	0.001000000	2024-06-24 07:20:39
☐ Edit ☐ Copy ☐ Delete	128	11.69	0.44	5.09	0.001000000	2024-06-24 07:19:30
☐ Edit ☐ Copy ☐ Delete	127	11.69	0.45	5.25	0.001000000	2024-06-24 07:18:31
☐ Edit ☐ Copy ☐ Delete	126	11.71	0.44	5.24	0.001000000	2024-06-24 07:17:22
☐ Edit ☐ Copy ☐ Delete	125	11.71	0.44	5.14	0.001000000	2024-06-24 07:16:24
☐ Edit ☐ Copy ☐ Delete	124	11.69	0.44	5.13	0.001000000	2024-06-24 07:15:19

Gambar 3.19 Nilai Sensor INA219 yang Terkirim ke Database

Data yang tersimpan pada database akan dimunculkan pada aplikasi dengan tampilan grafik garis serta nilai dengan angka yang akan ter-update secara otomatis ketika ada nilai baru yang masuk pada database. Pengguna dapat memilih grafik mana yang ingin dilihat seperti tegangan, arus, daya ataupun total kwh.

Berdasarkan dari percobaan yang telah dilakukan, monitoring yang dilakukan berfungsi dengan baik. Proses pengiriman data dari ESP32 sampe ke database berjalan dengan baik dan juga dapat menampilkan nilainya pada aplikasi. Dari percobaan yang dilakukan, menulis melakukan pengujian

dengan menghidupkan tiga perangkat yaitu kipas 12v dan dua buah led strip untuk membandingkan nilai tetap yang dihitung secara manual dan nilai yang diambil oleh sensor INA219.

TABEL3.3

RANCANGAN SKENARIO PENGUJIAN PEMBACAAN DAYA		
Beban	Menit Ke-	Daya (W)
Kipas 12V dan 2 LED STRIP	1	5,14
	2	5,24
	3	5,25
	4	5,09
	5	5,11
	6	5,09
	7	5,13
	8	5,25
	9	5,10
	10	5,10

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Dalam pengembangan platform IoT untuk pengendali rumah pintar, berdasarkan pengujian yang dilakukan, sistem kontrol dan monitoring pada prototipe rumah pintar berhasil diuji secara fungsional melalui komunikasi MQTT. Pengujian fokus pada fitur kontrol seperti lampu, pintu, dan kipas, serta fitur monitoring untuk membaca suhu dan kelembapan ruangan. Kesimpulan yang dapat ditarik adalah sebagai berikut:

- 1) Fitur kontrol dapat beroperasi secara efisien dan responsif melalui aplikasi Android, memungkinkan pengguna untuk mengontrol perangkat elektronik rumah secara praktis dan intuitif.
- 2) Fitur monitoring menyediakan informasi yang relevan tentang kondisi suhu dan kelembapan ruangan melalui aplikasi Android, memberikan pemahaman yang lebih baik kepada pengguna mengenai lingkungan di dalam rumah mereka.
- 3) Potensi untuk melakukan prediksi pemakaian listrik melalui aplikasi Android dapat diungkapkan lebih lanjut dengan memanfaatkan data yang dikumpulkan dari pengendali alat elektronik rumah, memungkinkan pengguna untuk mengoptimalkan penggunaan energi dan meningkatkan efisiensi konsumsi listrik mereka.

B. Saran

Dalam penelitian yang telah dilakukan terdapat kekurangan dan beberapa hal yang perlu dikaji kembali agar aplikasi ini dapat dikembangkan menjadi lebih baik kedepannya. Maka dari itu terdapat beberapa saran untuk pengembangan penelitian lebih lanjut:

- 1) Mengembangkan algoritma yang lebih canggih dan tepat untuk memprediksi pemakaian listrik berdasarkan data historis dan pola penggunaan yang terkumpul dari berbagai perangkat elektronik di rumah. Hal ini dapat meningkatkan akurasi prediksi dan membantu pengguna untuk merencanakan penggunaan energi mereka dengan lebih efektif.

- 2) Memanfaatkan teknologi kecerdasan buatan untuk menganalisis data penggunaan energi secara lebih mendalam dan memberikan rekomendasi yang lebih personal untuk mengoptimalkan penggunaan listrik di rumah. Hal ini dapat mencakup saran tentang waktu terbaik untuk mengaktifkan atau mematikan perangkat, atau penyesuaian otomatis berdasarkan kebiasaan pengguna.

V. KUTIPAN

- [1] Pradana, Yopi Firdha & Kuswinardi, Wiwin (2020). "Rancang Bangun Aplikasi Android Penilaian Kinerja Guru dengan Metode Agile Development di MAN 1 Kabupaten Malang". *Jurnal Terapan Sains & Teknologi*, Vol. 2, No. 1, 2020
- [2] Efendi Yoyon (2018). "Internet of Things (IoT) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile". *Jurnal Ilmiah Ilmu Komputer*, Vol. 4, No. 1, April 2018
- [3] Kurniawan Iqbal (2021). "Desain Kontrol dan Monitoring Smart Home berbasis Android".
- [4] Kurniadi, Dede & Amelia, Lia (2018). "Sistem Kendali Perangkat Elektronik Rumah Berbasis Android dan Arduino". *Jurnal Algoritma*, vol. 15, no. 2, hlm. 37 - 42, Sep 2018.
- [5] S. Suraidi and S. Nathania, "Sistem Pengendali Smart-Kontak dengan Aplikasi Android dan Web", *TESLA*, vol. 21, no. 2, pp. 123–134, Jan. 2020.
- [6] Pakaya, Abd. Wahid (2020). "Aplikasi Smartphone Pengontrol Rumah Menggunakan Internet of Things", *Jurnal Nasional cosPhi*, Vol. 4 No. 1 Tahun 2020.
- [7] Nurfaizi, A (2022). "Sistem Monitoring Meteran Listrik Berbasis IoT untuk Listrik Prabayar" (Skripsi Sarjana, Universitas Islam Indonesia).
- [8] Arafat, M. K. (2016). "SISTEM PENGAMANAN PINTU RUMAH BERBASIS Internet Of Things (IoT) Dengan ESP8266", *Jurnal Ilmiah Fakultas Teknik "Technologia"*, 7(4), 262–268.
- [9] Prianti, M. (2021). "Simulasi Sistem Kontrol Exhaust Fan Menggunakan PID Pada Smoking Room Berbasis Arduino". (Tugas Akhir, Universitas Diponegoro).
- [10] Anonim. "BAB II Tinjauan Pustaka". [Online]. Available: https://repository.uksw.edu/bitstream/123456789/14079/2/T1_612_010050_BAB%20II.pdf. [Accessed: 20- Mar - 2024].
- [11] Putra, D (2022). "Rancang Bangun Smarthome Pengamanan Kelistrikan Berbasis Android". (Skripsi, Universitas Muhammadiyah Sumatra Barat).



Daniel Alexander Runtu, anak kedua dari dua bersaudara, lahir di Jakarta pada tanggal 14 Januari 2001. Penulis mulai menempuh pendidikan di TKK Ignatius Slamet Riyadi pada tahun 2006. Kemudian penulis menempuh pendidikan pada tahun 2007 sampai 2013 di SDK Ignatius Slamet Riyadi I. Pada tahun 2013 sampai 2016, penulis melanjutkan pendidikan di SMPK Ignatius Slamte Riyadi. Kemudian pada tahun 2016 sampai 2019 melanjutkan pendidikan di SMA Negeri 98 Jakarta. Pada tahun 2019, penulis melanjutkan studi di jenjang sarjana di Program Studi Teknik Informatika, Fakultas Teknik, Jurusan Teknik Elektro, Universitas Sam Ratulangi Manado.

Selama perkuliahan penulis tergabung dalam organisasi kemahasiswaan dan aktif berkegiatan dalam Himpunan Mahasiswa Elektro (HME) FT-UNSRAT dan Unit Kegiatan Mahasiswa Edukasi Robotika (EURO) Teknik Universitas Sam Ratulangi.