

PROTOTYPE APLIKASI *SMART LIGHTING* UNTUK MENGONTROL LAMPU JALAN BERBASIS ANDROID MENGGUNAKAN ESP32

Youzy Natasya¹, Handri Santoso²

Universitas Pradita, Indonesia

Email: youzy.natasya@student.pradita.ac.id¹, handri.santoso@pradita.ac.id²

Abstract

Lighting is essential in life as it provides a sense of security, especially when it comes to street lighting at night. However, negligence often leads to streetlights not functioning in dark conditions. The development of technology, particularly in the field of Internet of Things (IoT), enables devices to interact and exchange data through the internet network. Therefore, IoT technology has been developed to control streetlights remotely using a mobile application. This mobile application consists of three features for controlling the lights. First is the manual control, where users can adjust the intensity of the light emitted by the lamp. Second is the scheduled-based control, allowing users to set specific times for the lamp to turn on and off. Lastly, automatic control, which adjusts the lamp based on the ambient light conditions in the area. The prototype is implemented by designing an IoT system using the WiFi module on the ESP32 microcontroller, LED lights, and an LDR sensor to measure light intensity. Additionally, a mobile application is developed to control the streetlights. Testing is conducted by controlling the lamps through the built application and gathering feedback through questionnaires to evaluate the usability of the application's interfaces. Based on the test results, the application effectively functions to control the streetlights according to the available features, and the application's interface is user-friendly.

Keywords: Lights, Internet of Things, LDR Sensor, control

Abstrak

Penerangan merupakan hal penting dalam kehidupan karena dapat memberikan rasa aman, terutama penerangan pada jalan di malam hari. Namun, sering terjadi kelalaian sehingga menyebabkan lampu jalan tidak menyala saat keadaan gelap. Perkembangan teknologi terutama di bidang *Internet of Things* (IoT) menjadikan perangkat dapat saling berinteraksi dan bertukar data melalui jaringan internet. Oleh karena itu, dibangun teknologi IoT yang dapat mengontrol lampu dari jarak jauh dengan menggunakan aplikasi mobile. Di mana pada aplikasi mobile ini terdapat tiga fitur untuk mengontrol lampu, yaitu secara manual dengan mengatur intensitas cahaya yang dipancarkan oleh lampu, berdasarkan jadwal dengan mengatur waktu menyala dan waktu mati, serta secara otomatis berdasarkan keadaan gelap-terang di lingkungan tersebut. Prototipe dilakukan dengan merancang sistem IoT dengan menggunakan modul WiFi pada mikrokontroler ESP32, LED, dan sensor LDR untuk mengukur intensitas cahaya, serta membangun aplikasi mobile untuk mengontrol lampu. Pengujian dilakukan dengan mengontrol lampu melalui aplikasi yang telah dibangun serta memberikan kuesioner untuk menguji antarmuka dari aplikasi. Berdasarkan dari hasil pengujian yang dilakukan menunjukkan bahwa aplikasi dapat berfungsi untuk mengontrol lampu sesuai dengan fitur yang tersedia serta antarmuka aplikasi yang cukup mudah digunakan.

Kata kunci: Lampu, *Internet of Things*, sensor LDR, mengontrol

PENDAHULUAN

Penerangan dengan memanfaatkan lampu di mana tidak ada matahari sebagai sumber cahaya alami merupakan hal yang sangat dibutuhkan. Misalnya penerangan pada jalanan memiliki tujuan untuk menjaga dan memberikan rasa aman bagi pengguna jalan di suatu wilayah. Tersedianya lampu jalan, terutama pada saat keadaan gelap dapat memberikan manfaat pada keamanan lingkungan sekitar serta meningkatkan keselamatan dan

kenyamanan pengendara maupun pejalan kaki yang melintas pada daerah tersebut (Ikhsan and Cahyo 2021).

Namun, pada penerapannya sering kali ditemukan lampu jalan yang tidak menyala ketika keadaan gelap. Padahal kurangnya penerangan ketika keadaan gelap dapat memberikan dampak negatif bagi pengguna jalan. Selain karena kemungkinan adanya kerusakan pada lampu itu sendiri, lampu yang tidak menyala ketika keadaan gelap bisa juga terjadi karena kelalaian operator dalam mengontrol penyalaan lampu. Penggunaan saklar untuk mengontrol lampu dinilai tidak efisien karena tidak menghemat waktu dan tenaga manusia (Abdullah 2019).

Pada saat ini, teknologi terus berkembang dengan pesat. Perkembangan teknologi ini dapat memberikan dampak positif di berbagai bidang. Salah satunya adalah untuk memberikan solusi dari permasalahan lampu jalan untuk memonitor dan mengontrol lampu melalui smartphone dari jarak jauh. Pemanfaatan teknologi ini juga dapat memberikan efisiensi anggaran biaya dari pemakaian lampu (Dwiparaswati 2023). Di mana pemakaian lampu umumnya sebesar 20% - 50% dari total penggunaan listrik, karena lampu dibiarkan menyala baik dalam keadaan sedang digunakan maupun tidak digunakan (Dwiandra et al. 2020). Konsep teknologi yang digunakan adalah dengan memanfaatkan teknologi Internet of Things (IoT), di mana perangkat dapat saling berinteraksi atau bertukar data melalui jaringan Wi-Fi (Yudhanto and Azis 2019).

Dalam penerapannya, IoT memerlukan sensor untuk mengumpulkan data yang kemudian data tersebut ditransmisi. Pada prototipe perancangan aplikasi Smart Lighting berbasis Wireless Network digunakan ESP32 sebagai data transmisi untuk menyalakan dan mematikan LED dengan mengatur intensitas cahaya dari LED, menyalakan dan mematikan LED berdasarkan jadwal yang telah diatur oleh pengguna aplikasi, dan menyalakan dan mematikan LED berdasarkan keadaan gelap dan terang dengan memanfaatkan sensor LDR, yaitu mengukur intensitas cahaya di lingkungan. Sehingga jika keadaan gelap, lampu yang terhubung ke sensor ini akan menyala (Suryo et al. 2020).

Implementasi aplikasi smart lighting berbasis wireless network dapat memberikan keuntungan dan efisiensi seperti menghemat energi karena pengguna dapat mengontrol dan mengatur intensitas cahaya dari lampu sesuai dengan kebutuhan. Selain itu, dengan adanya aplikasi smart lighting, operator dapat dengan mudah mengontrol dan memonitor kondisi lampu, termasuk mengatur waktu penyalaan dan mematikan lampu pada waktu yang tepat. Hal ini dapat meningkatkan keamanan dan keselamatan pengguna jalan, pengendara, dan pejalan kaki dengan memberikan penerangan yang memadai pada jalan yang diperlukan, terutama pada malam hari.

TINJAUAN PUSTAKA

Internet of Things (IoT)

Teknologi Internet of Things (IoT) adalah konsep di mana perangkat dapat saling terhubung dan berkomunikasi menggunakan internet tanpa interaksi manusia. Penerapan IoT dapat meningkatkan efisiensi di berbagai bidang. Salah satunya adalah dalam konteks pengendalian lampu, IoT memungkinkan lampu untuk terhubung ke jaringan dan

dikendalikan secara otomatis atau dari jarak jauh. Cara kerja dari IoT adalah dengan menggunakan sebuah logika pemrograman di mana setiap perintahnya menghasilkan interaksi antar mesin yang terhubung dan menggunakan internet sebagai penghubung (Moh Noor Al Azam 2022).

Penerapan Internet of Things (IoT) melibatkan beberapa komponen utama yang membentuk arsitektur penyusun, yaitu things, gateways, data lake, big data warehouse, data analytics, dan aplikasi pengguna. Things merupakan sensor yang berfungsi untuk mengumpulkan data, contohnya sensor LDR yang digunakan untuk mendeteksi intensitas cahaya. Gateway berperan sebagai penghubung antara sensor dengan server atau aplikasi pengendali. Data lake merupakan tempat untuk menyimpan data, ketika data akan diolah menjadi sesuatu yang bermakna akan diekstrak di big data warehouse, yang merupakan data yang telah difilter dan merupakan data yang terstruktur. Data tersebut kemudian dianalisa untuk meningkatkan sistem IoT. Aplikasi pengendali merupakan pusat untuk mengendalikan atau mengontrol perangkat IoT dari jarak jauh, contohnya aplikasi 7 smartphone. Dalam penerapannya, IoT terhubung ke dalam jaringan yang dapat menggunakan berbagai teknologi, seperti WiFi, untuk mengirim dan menerima data

Dalam menguji rangkaian dan sistem yang telah dibangun, untuk mencari permasalahan yang timbul sehingga mengakibatkan sistem jaringan tidak berfungsi dapat menggunakan model prosedur OSI Layer. Terdapat 7 lapisan pada OSI Layer, yaitu:

1. Physical Layer, merupakan lapisan pertama yang bertindak sebagai transmisi terhadap bit data.
2. Data Link Layer, merupakan lapisan kedua yang melakukan pemeriksaan apakah ada kesalahan dalam menyalurkan transmisi pada lapisan pertama.
3. Network Layer, merupakan lapisan ketiga yang bertindak dalam mengatur alamat IP agar setiap perangkat dapat terhubung satu sama lain dalam jaringan.
4. Transport Layer, merupakan lapisan keempat yang bertindak dalam menyalurkan bit.
5. Session Layer, merupakan lapisan kelima yang bertindak dalam mengontrol dialog dan mengatur koneksi komputer.
6. Presentation Layer, merupakan lapisan keenam yang bertindak dalam menentukan sintaks yang digunakan oleh server jaringan untuk berkomunikasi.
7. Application Layer, merupakan lapisan ketujuh yang merupakan pusat interaksi antara pengguna dengan aplikasi yang beroperasi menggunakan fungsionalitas jaringan.

Mikrokontroler ESP32

Dalam pengembangan aplikasi IoT, dibutuhkan mikrokontroler, yang merupakan komputer kecil untuk menjalankan suatu program dan menghubungkan komponen lain dalam sistem, seperti pengendalian, otomatisasi industri, akuisisi data, telekomunikasi, dan 8 lainnya (Yudhanto and Azis 2019). Ada berbagai jenis mikrokontroler di mana memiliki kelebihan dan kekurangan yang berbeda-beda, salah satunya adalah ESP32.

ESP32 merupakan mikrokontroler yang dikenalkan oleh Espressif System pada 2016. ESP32 dapat diprogram menggunakan Arduino IDE yang merupakan software open source, dengan melakukan instalasi pada hardware package. Salah satu kelebihan pada ESP32 ini

adalah adanya WiFi dan Bluetooth, sehingga memudahkan pembuatan sistem IoT yang memerlukan wireless network. Untuk mengkoneksikan ke jaringan WiFi, pada Arduino IDE diperlukan library yang menghubungkan ke jaringan WiFi. Setelah itu mendeklarasikan SSID dan password dari jaringan WiFi yang akan digunakan.

Firestore

Firestore didirikan oleh Andrew Lee dan James Tamplin pada tahun 2011 yang kemudian diakuisisi dan dikembangkan oleh Google pada tahun 2014. Firestore menawarkan berbagai fitur termasuk penyimpanan data, otentikasi pengguna, analitik, notifikasi, dan masih banyak lagi. Tersedianya berbagai library memungkinkan untuk diintegrasikan dengan Android, iOS, Java, dan lainnya. Firestore Realtime Database menyediakan layanan database yang dapat diakses secara realtime dan memiliki format penulisan yang berbeda dari database relasional, di mana dokumen dan kolom digunakan sebagai pengganti tabel dan field dalam database relasional (Kurniawan, Samsudin, and Triase 2021). Salah satu keunggulan dari Firestore Realtime Database adalah kemampuannya dalam menyinkronkan data secara otomatis antara klien dan server. Hal ini memungkinkan pembaruan data yang dilakukan di satu perangkat dapat dilihat secara langsung oleh perangkat lain yang terhubung.

Aplikasi Android

Aplikasi android merupakan perangkat lunak yang berjalan pada perangkat android, di mana aplikasi dapat diunduh melalui Google Play Store. Aplikasi android dibuat menggunakan Bahasa pemrograman Java. Salah satu software yang dapat digunakan untuk membuat aplikasi android adalah dengan menggunakan Android Studio.

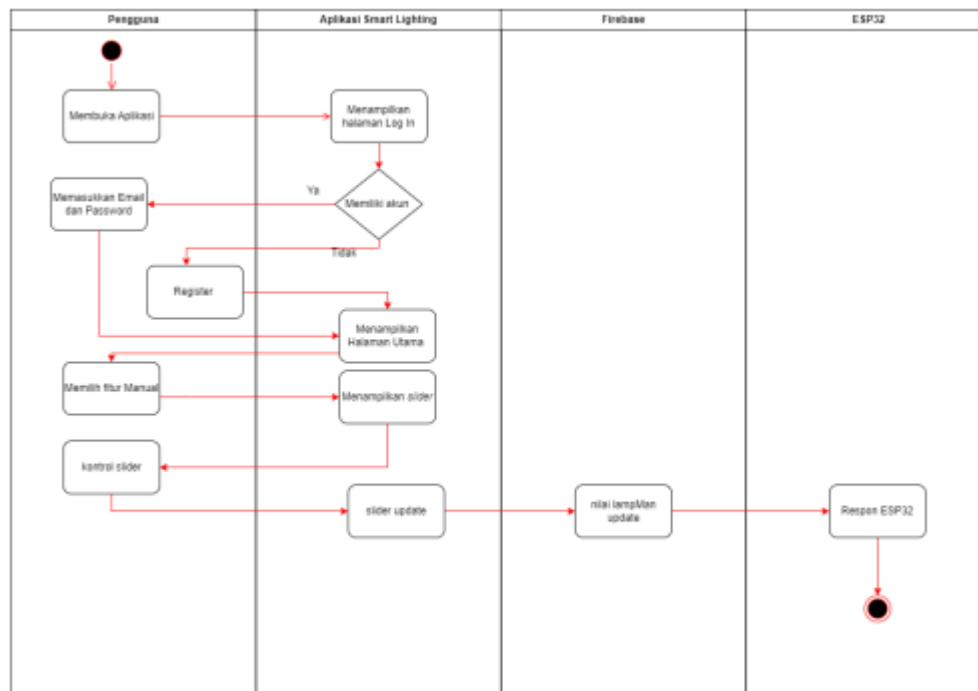
METODE

Metodologi penelitian yang digunakan dalam penelitian ini untuk membangun aplikasi smart lighting berbasis android adalah dengan menggunakan model SDLC (System Development Life Cycle) iterative, di mana tahapan pengembangan dengan model iterative dimulai dari tahap perencanaan, analisis, implementasi, testing, dan evaluasi.

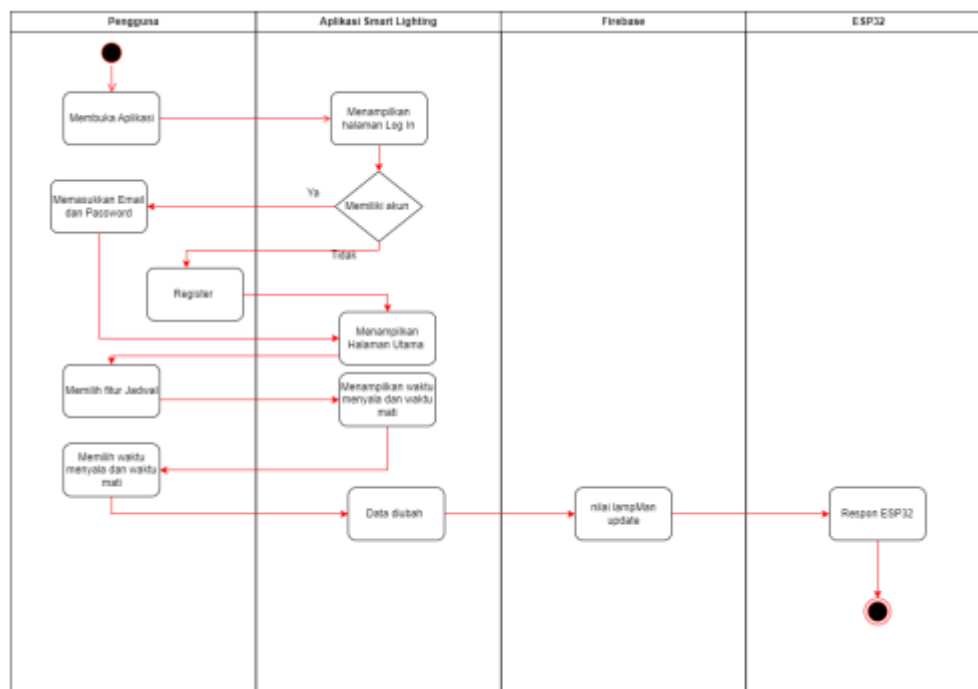


Gambar 1. Tahapan Pengerjaan

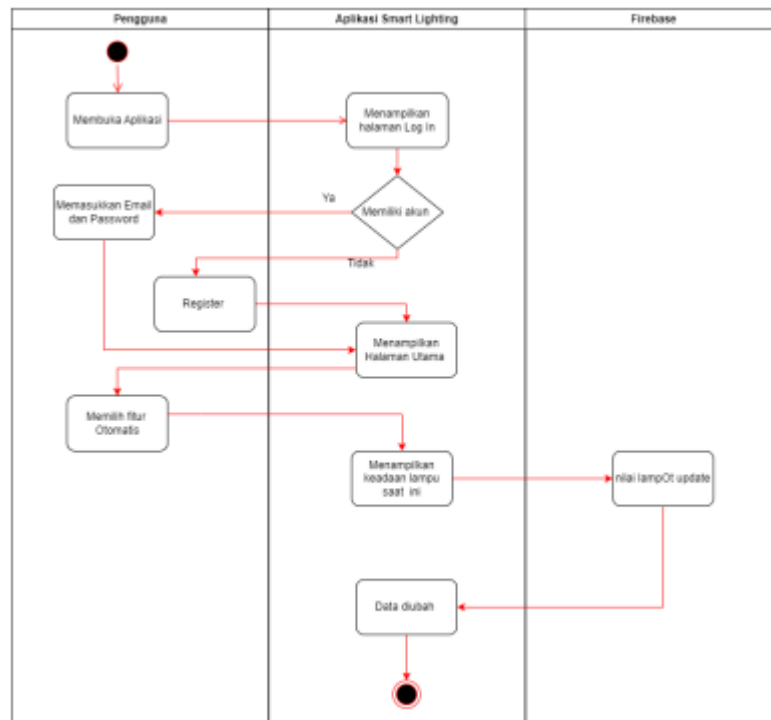
activity diagram kontrol lampu manual, Gambar 4 menunjukkan *activity diagram* kontrol lampu jadwal, Gambar 5 menunjukkan *activity diagram* kontrol lampu otomatis, dan Gambar 6 menunjukkan *activity diagram* menampilkan data sensor.



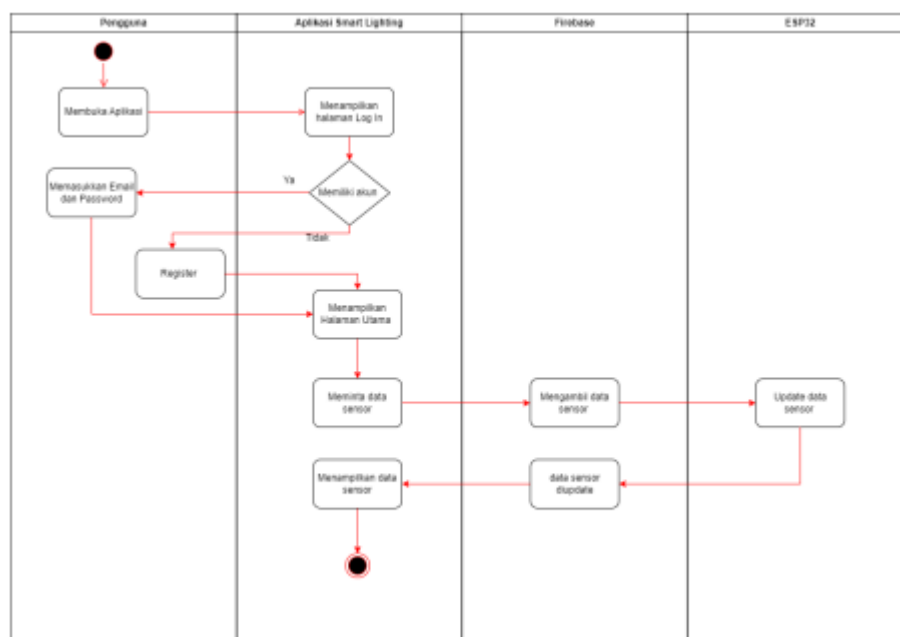
Gambar 3. Activity Diagram Kontrol Lampu Manual



Gambar 4. Activity Diagram Kontrol Lampu Jadwal

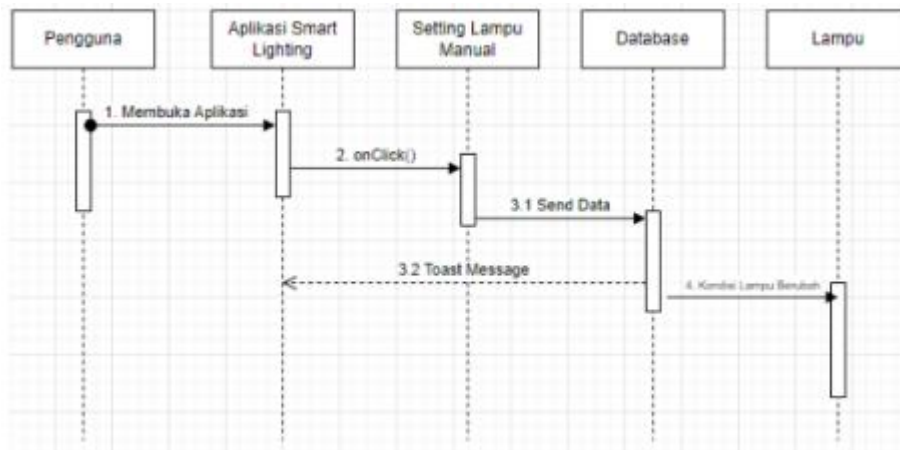


Gambar 5. Activity Diagram Kontrol Lampu Otomatis

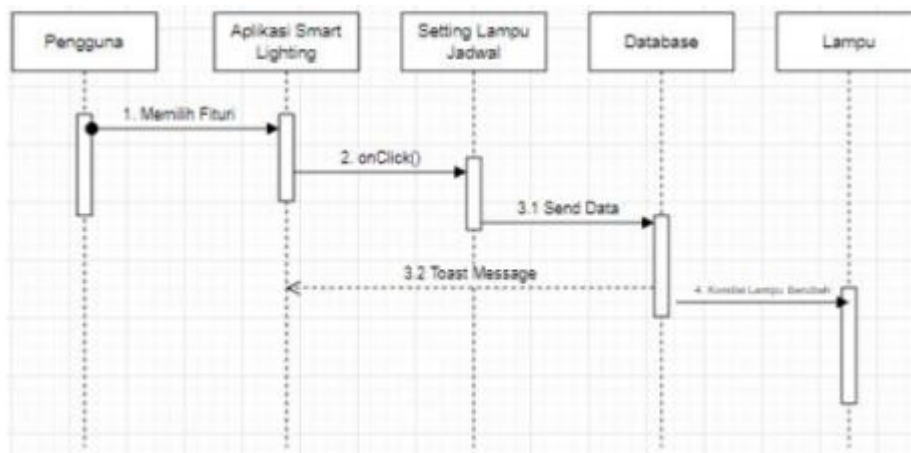


Gambar 6. Activity Diagram Menampilkan Data Sensor

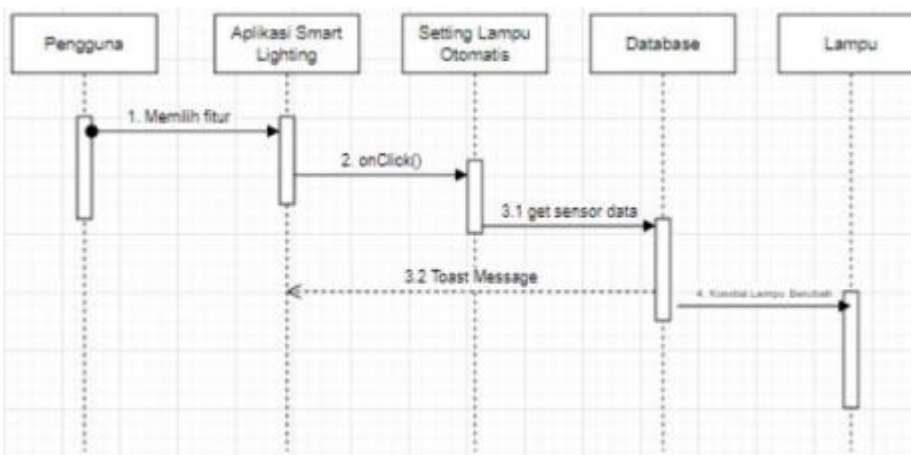
Sequence diagram adalah diagram yang menggambarkan interaksi antar objek dalam sistem. Diagram ini menggambarkan urutan pesan yang dikirim antara objek dalam urutan waktu dan memberikan gambaran tentang bagaimana objek berinteraksi satu sama lain. *Sequence diagram* dapat dilihat pada Gambar 7, 8, 9, dan 10.



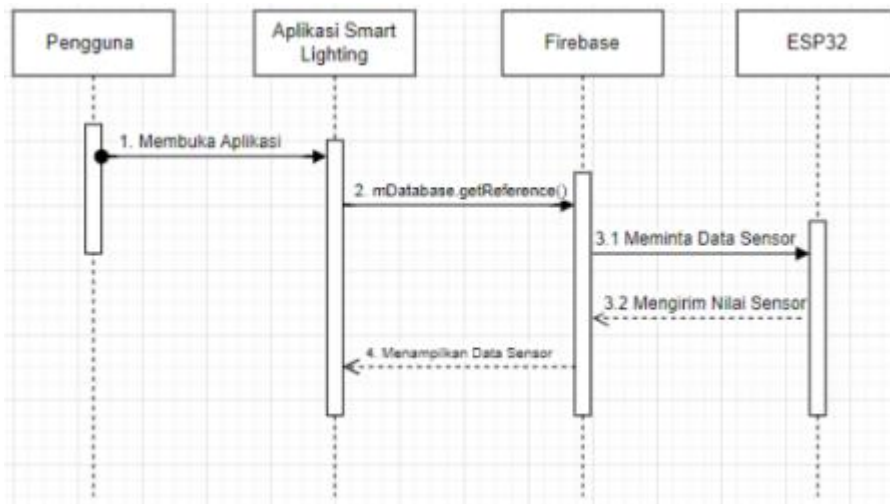
Gambar 7. *Sequence Diagram* Kontrol Lampu Manual



Gambar 8. *Sequence Diagram* Kontrol Lampu Jadwal

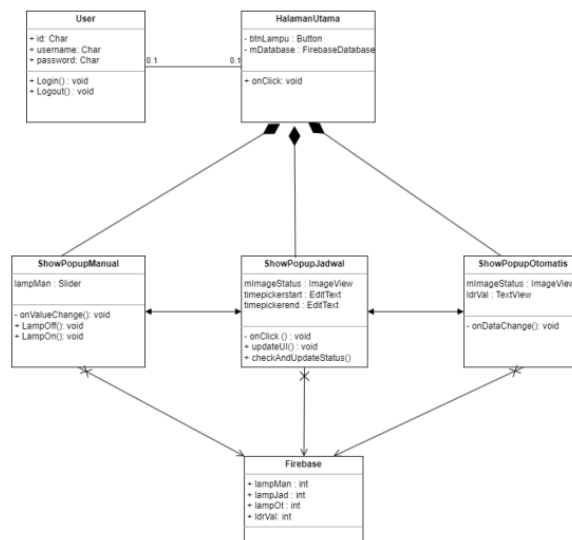


Gambar 9. *Sequence Diagram* Kontrol Lampu Otomatis



Gambar 10. Menampilkan Data Sensor

Class diagram merupakan diagram yang menggambarkan struktur kelas, hubungan antar kelas, atribut, dan metode dalam suatu sistem, di mana dapat dilihat pada Gambar 11.



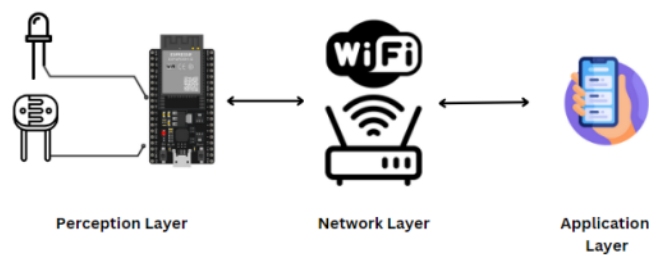
Gambar 11. Class Diagram

Pada tahapan ini juga dilakukan perancangan desain antarmuka aplikasi. Rancangan dari antarmuka aplikasi dapat dilihat pada Gambar 12.



Gambar 12. High Fidelity Prototype

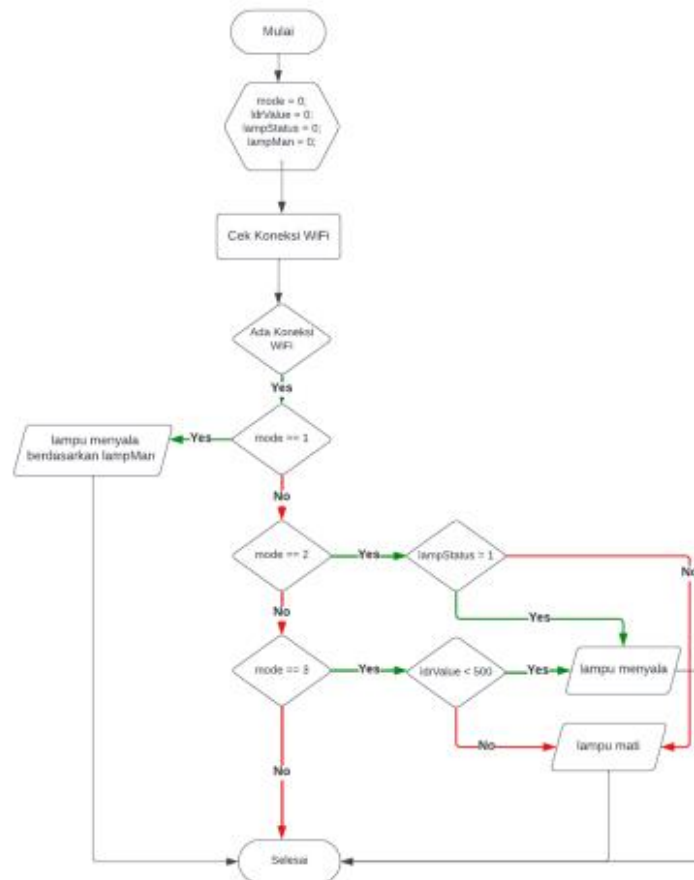
Arsitektur sistem IoT dibagi menjadi beberapa lapisan, di mana dapat dilihat pada Gambar 13. *Perception layer* merupakan sensor yang dapat mendeteksi keadaan lingkungan, di mana sensor pada sistem ini adalah sensor LDR untuk mengukur intensitas cahaya. Nilai sensor yang ditangkap akan disimpan ke database Firebase melalui jaringan WiFi. Kemudian aplikasi mobile akan menampilkan nilai sensor dan perubahan pada icon lampu berdasarkan kondisi LED.



Gambar 13. Arsitektur Sistem

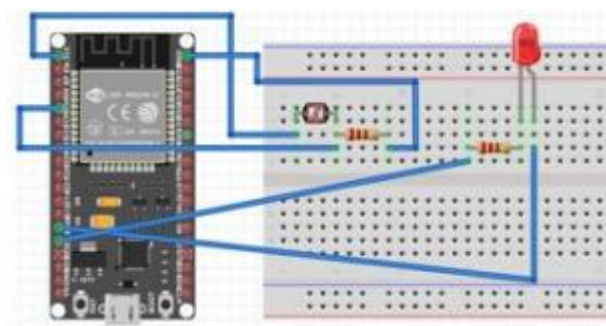
Tahap Implementasi

Pada tahap ini dilakukan pengcodingan pada Android Studio untuk membuat antarmuka aplikasi dan konfigurasi ke *firebase*, serta merangkai sistem IoT dengan mikrokontroler ESP32, LED, dan sensor LDR, dan melakukan *coding* pada Arduino IDE agar perangkat IoT yang telah dirangkai dapat terkoneksi ke WiFi dan menjalankan fungsinya. Flowchart dari aplikasi dapat dilihat pada Gambar 14.



Gambar 14. Flowchart Aplikasi

Rangkaian IoT dapat dilihat pada Gambar 15, di mana LED dihubungkan ke pin GND dan pin 13 pada ESP32 sebagai *output*. Sensor LDR dihubungkan ke pin 3V sebagai sumber daya positif, pin GND, dan pin analog 34 sebagai *input*.



Gambar 14. Rangkaian Sistem

Tahap Testing

Pada tahap ini, rangkaian *smart lighting* dihubungkan dengan aplikasi Arduino IDE serta menjalankan aplikasi Android. Pengujian dilakukan dengan mengontrol nyala dan mati lampu melalui aplikasi dengan menjalankan tiga fitur yang tersedia, yaitu manual dengan

mengatur intensitas cahaya yang akan dipancarkan oleh lampu, jadwal dengan mengatur waktu menyala dan waktu mati, serta otomatis dengan mengatur gelap-terang di keadaan sekitar sensor LDR. Pengujian juga dilakukan dengan menguji jaringan pada Arduino IDE apakah berhasil terhubung dengan WiFi atau tidak, serta apakah aplikasi berhasil mengirim nilai dan membaca nilai pada *firebase*.

Tahap Evaluasi

Pada tahap evaluasi, menganalisa kegagalan aplikasi yang telah diuji. Misalnya seperti lampu tidak berhasil menyala sesuai dengan fitur yang dipilih, lampu tidak menyala sesuai dengan jadwal maupun berdasarkan kondisi cuaca. Permasalahan – permasalahan 23 yang ditemukan kemudian dilakukan perbaikan dan dilakukan perkembangan supaya aplikasi dapat berjalan sebagaimana mestinya.

HASIL DAN PEMBAHASAN

Pengujian Aplikasi untuk Mengontrol Lampu secara Manual

Pengujian aplikasi dilakukan dengan mengaktifkan tombol “Manual”, sehingga akan menampilkan antarmuka untuk mengontrol lampu dengan menggunakan *slider*. Nilai dari *slider* akan tersimpan pada *firebase* lampMan yang kemudian pada Arduino akan menampilkan intensitas cahaya dari lampu berdasarkan nilai *firebase* lampMan. Pengujian aplikasi android untuk menyalakan lampu secara manual dapat dilihat pada tabel 1.

Tabel 1. Tabel Pengujian Lampu dengan Menggunakan Slider

No.	Nilai <i>Slider</i>	Kondisi Lampu
1.	0	Lampu Mati
2.	15	Lampu Menyala Sangat Redup
3.	43	Lampu Menyala Redup
4.	70	Lampu Menyala Agak Terang
5.	100	Lampu Menyala Terang

Pengujian Aplikasi untuk Mengontrol Lampu Berdasarkan Jadwal Waktu

Pengujian dilakukan dengan menekan tombol “Jadwal”, lalu mengatur waktu menyala dan waktu mati pada aplikasi. Hasil dari pengujian dapat dilihat pada Tabel 2.

Tabel 2. Tabel Pengujian Lampu dengan Mengatur Waktu Menyala dan Waktu Mati

No.	Waktu Saat Ini	Waktu Menyala	Waktu Mati	Keterangan
1.	10.00	09.50	09.55	Lampu mati. Ikon pada antarmuka aplikasi akan menunjukkan keterangan lampu mati

**PROTOTYPE APLIKASI SMART LIGHTING UNTUK
MENGONTROL LAMPU JALAN BERBASIS ANDROID
MENGUNAKAN ESP32**

Youzy Natasya¹, Handri Santoso²

DOI: <https://doi.org/10.54443/sibatik.v2i8.1298>

2.	10.05	09.56	10.10	Lampu menyala dan ikon pada aplikasi akan berupa keterangan lampu menyala. Lampu mati pada pukul 10.10 dan ikon kembali berubah menjadi keterangan lampu mati
3.	10.15	10.15	10.20	Lampu menyala dan ikon pada aplikasi akan berupa keterangan lampu menyala. Lampu mati pada pukul 10.20 dan ikon kembali berubah menjadi keterangan lampu mati
4.	10.20	10.25	10.30	Lampu mati dengan ikon pada antarmuka aplikasi lampu mati. Lampu menyala pada pukul 10.25 dan ikon berubah menjadi lampu menyala. Kemudian lampu mati lagi pada pukul 10.30 dan ikon kembali berubah menjadi lampu mati.
5.	10.35	11.00	11.10	Lampu mati dan ikon antarmuka pada aplikasi menunjukkan

				keterangan lampu mati.
--	--	--	--	------------------------

Pengujian Aplikasi Untuk Mengontrol Lampu secara Otomatis

Pengujian aplikasi dilakukan dengan mengaktifkan tombol “Otomatis”, sehingga akan menampilkan antarmuka kondisi lampu saat ini berupa ikon lampu menyala ataupun lampu mati. Perubahan ikon ini dilihat dari nilai *ldrVal* pada *firebase*. Nilai *ldrVal* didapatkan dari rangkaian *smart lighting* yang menggunakan sensor LDR. Jika nilai dari sensor LDR kurang dari 500, maka lampu akan menyala dan jika nilai sensor LDR lebih dari 500, lampu akan mati. Hasil dari pengujian dengan mengubah intensitas cahaya yang diterima sensor LDR dapat dilihat pada Tabel 3.

Tabel 3. Tabel Pengujian Lampu Berdasarkan Kondisi Cahaya

No.	Nilai <i>ldrVal</i>	Keterangan
1.	0	Lampu Menyala dan ikon menjadi keterangan lampu menyala.
2.	350	Lampu Menyala dan ikon menjadi keterangan lampu menyala.
3.	1256	Lampu Mati dan ikon menjadi keterangan lampu mati.

Pengujian Antarmuka dari Aplikasi *Smart Lighting*

Pengujian antarmuka aplikasi ini dilakukan dengan membagikan kuesioner ke 20 responden. Adapun pengujian ini dilakukan dengan menggunakan metode *End-User Computing Satisfaction* (EUCS), yaitu metode untuk mengevaluasi aplikasi berdasarkan pengalaman pengguna. Hasil yang didapatkan adalah sebagai berikut.

Tabel 4. Aplikasi android Smart Lighting menampilkan informasi yang jelas.

Parameter	Responden
Sangat Setuju	15
Setuju	5
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 5. Aplikasi android Smart Lighting menyediakan fitur pengontrolan lampu yang sesuai dengan kebutuhan pengguna

Parameter	Responden
Sangat Setuju	14
Setuju	5
Netral	1

Tidak Setuju	
Sangat Tidak Setuju	

Tabel 6. Aplikasi android Smart Lighting memberikan informasi yang berguna bagi pengguna dalam melakukan pengontrolan lampu

Parameter	Responden
Sangat Setuju	11
Setuju	9
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 7. Aplikasi Smart Lighting menampilkan informasi yang akurat

Parameter	Responden
Sangat Setuju	15
Setuju	5
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 8. Tombol pada aplikasi Smart Lighting berfungsi dengan baik

Parameter	Responden
Sangat Setuju	14
Setuju	6
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 9. Antarmuka dari aplikasi Smart Lighting memiliki tampilan yang menarik

Parameter	Responden
Sangat Setuju	14
Setuju	6
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 10. Kombinasi warna pada aplikasi Smart Lighting yang digunakan menarik

Parameter	Responden
Sangat Setuju	12
Setuju	8

Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 11. Antarmuka dari fitur kontrol lampu secara manual pada aplikasi Smart Lighting memiliki tampilan yang menarik

Parameter	Responden
Sangat Setuju	14
Setuju	5
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 12. Antarmuka dari fitur kontrol lampu berdasarkan jadwal pada aplikasi Smart Lighting memiliki tampilan yang menarik

Parameter	Responden
Sangat Setuju	14
Setuju	6
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 13. Antarmuka dari fitur kontrol lampu secara otomatis pada aplikasi Smart Lighting memiliki tampilan yang menarik

Parameter	Responden
Sangat Setuju	13
Setuju	7
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 14. Antarmuka dari fitur kontrol lampu pada aplikasi Smart Lighting memiliki tampilan yang mudah digunakan

Parameter	Responden
Sangat Setuju	14
Setuju	6
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 15. Aplikasi Smart Lighting tidak membutuhkan waktu untuk mempelajarinya

Parameter	Responden
Sangat Setuju	11
Setuju	8
Netral	1
Tidak Setuju	
Sangat Tidak Setuju	

Tabel 16. Aplikasi Smart Lighting sangat mudah digunakan

Parameter	Responden
Sangat Setuju	14
Setuju	6
Netral	
Tidak Setuju	
Sangat Tidak Setuju	

PENUTUP

Kesimpulan

Berdasarkan hasil analisis dari pengamatan pengujian aplikasi, maka penulis dapat menyimpulkan bahwa:

- Penggunaan ESP32 di mana menggunakan jaringan WiFi dan dikontrol menggunakan aplikasi Android, dapat membuat lampu menyala secara manual dengan mengatur intensitas cahaya dari lampu. Lampu juga dapat menyala berdasarkan jadwal yang telah ditentukan, serta menyala berdasarkan kondisi cuaca di lingkungan.
- Penggunaan ESP32 pada pengontrolan lampu memiliki jangkauan yang terbatas di mana hanya dapat digunakan pada jangkauan WiFi.

Saran

Berdasarkan hasil penelitian yang dilakukan, maka penulis memperoleh saran untuk dijadikan pengembangan lebih lanjut, yaitu:

- Mengimplementasikan pada lampu sesungguhnya.
- Meningkatkan konektivitas perangkat seperti mempertimbangkan penggunaan Bluetooth atau LoRa untuk memperluas cakupan jaringan dan mengurangi ketergantungan pada infrastruktur WiFi, di mana pada penelitian ini hanya menggunakan WiFi.
- Pengujian sistem IoT ini belum dilakukan secara menyeluruh khususnya pengujian layer network dan pengujian terhadap user experience. Oleh karena itu, untuk pengembangan selanjutnya perlu dilakukan pengujian tersebut.

DAFTAR PUSTAKA

- Abdullah, Muksin Hi. 2019. "Rancang Bangun Sistem Kontrol Lampu Listrik Menggunakan Remote Berbasis Mikrokontroler ATmega 8535" 2 (1): 40–47.
- Agustianti. 2022. *Metode Penelitian Kuantitatif Dan Kualitatif Tohar Media*.

- Dwiandra, Muh Ryan, Aldi Pryanto N, Muhammad Yusan Naim, and Muh Amin. 2020. "Rancang Bangun Sistem Kontrol Lampu Penerangan Menggunakan Aplikasi Android Berbasis Mikrokontroler" 3 (1): 10–13.
- Dwiparawati, Windy. 2023. "Simulasi Alat Pengendali Lampu Jarak Jauh Menggunakan Telegram," 81–89.
- Ikhsan, Jazaul, and Dwi Cahyo. 2021. "Penerangan Jalan Sebagai Keselamatan Pengguna Jalan Upaya Peningkatan," 2344–49. <https://doi.org/10.18196/ppm.47.704>.
- Kurniawan, Teguh, Samsudin, and Triase. 2021. "Implementasi Layanan Firebase Pada Pengembangan Aplikasi Sewa Sarana Olahraga Berbasis Android" 6 (1): 13–18.
- Moh Noor Al Azam, S.K.M.M.T. 2022. *Cara Cepat Belajar IoT: ESP32: Pengenalan Dan Instalasi Arduino IDE*. Cara Cepat Belajar IoT. Radnet Digital Indonesia. <https://books.google.co.id/books?id=y2JkEAAAQBAJ>.
- Suryo, Yoedo Ageng, Andi Rahmad Rahim, Heri Prastyono, Ahmad Kurnia Jaya, Dosen Program, Studi Teknik, Universitas Muhammadiyah Gresik, et al. 2020. "DedikasiMU (Journal of Community Service) Volume 2, Nomor 1, Maret 2020 PERANCANGAN SISTEM KONTROL OTOMATIS LAMPU BALAI DESA DAN JALAN BERBASIS RELAY TIMER DI DESA WOTANSARI BALONG PANGGANG" 2: 191–99.
- Yudhanto, Y, and A Azis. 2019. *Pengantar Teknologi Internet of Things (IoT)*. UNSPress. <https://books.google.co.id/books?id=IK33DwAAQBAJ>.