

Sistem Pendeteksi Kebocoran Gas Berbasis IoT Sebagai Alat Bantu Pada UMKM

¹Gaguk Suprianto, ²Alya Rizky Natasya, ³Arfi Indra Riskiawan

¹ Informatika, Universitas Hayam Wuruk Perbanas, Surabaya

² Informatika, Universitas Hayam Wuruk Perbanas, Surabaya

³ Informatika, Universitas Hayam Wuruk Perbanas, Surabaya

gaguk.suprianto@perbanas.ac.id, 202102011013@mhs.hayamwuruk.ac.id, 202102011001@mhs.hayamwuruk.ac.id

Abstract - One of the problems that still occurs a lot today is fire. This is caused by several things such as short circuits, cigarette nipples that are still burning and LPG leaks. Fires can occur in areas of office buildings, schools or homes. One of the causes of fires is LPG leakage caused by human negligence. So we need a system that can detect and monitor gas leaks in real time to minimize the occurrence of fires due to LPG gas leaks. Therefore, the system is designed in the form of hardware and software with the MQTT protocol. The protocol is lightweight, open and simple so that it can be used to transmit data quickly. This study uses the Waterfall method in which each stage can be passed if the previous stage is completed. The test result obtained are that the system can work properly when measuring sensitivity and delay sensor measurements. The author hopes that the research conducted will be able to provide insight and reference for other researchers and MSME or business actors in creating similar systems. In addition, subsequent research can use many sensors for the implementation of others with the MQTT protocol.

Keywords —LPG, MQTT, Waterfall

Abstrak—Salah satu permasalahan yang masih banyak terjadi hingga kini adalah kebakaran. Hal tersebut disebabkan oleh beberapa hal seperti hubungan arus pendek, putung rokok yang masih menyala dan kebocoran LPG. Kebakaran dapat terjadi pada area gedung kantor, sekolah maupun rumah. Salah satu penyebab kebakaran yakni kebocoran LPG yang disebabkan karena kelalaian manusia. Sehingga diperlukan sebuah sistem yang dapat melakukan deteksi dan monitoring kebocoran gas secara *real time* untuk meminimalisir terjadinya kejadian kebakaran akibat dari bocornya gas LPG. Oleh sebab itu peneliti merancang yang berupa perangkat keras dan perangkat lunak dengan protokol MQTT. Protokol tersebut ringan, terbuka dan sederhana sehingga dapat digunakan untuk mentransmisikan data dengan cepat. Penelitian ini menggunakan metode *Waterfall* yang dimana setiap tahapan dapat dilalui apabila tahapan sebelumnya selesai. Hasil pengujian diperoleh bahwa sistem bekerja dengan kinerja yang baik ketika dilakukan pengujian pengukuran sensitivitas sensor dan *delay*. Penulis berharap dari penelitian yang dilakukan mampu memberikan wawasan dan referensi untuk peneliti lain dan pelaku UMKM atau bisnis dalam membuat sistem yang serupa. Selain itu penelitian selanjutnya dapat menggunakan banyak sensor untuk implementasi yang lain dengan protokol MQTT.

Kata Kunci—LPG, MQTT, Waterfall

I. Pendahuluan

Saat ini banyak sekali jenis teknologi yang dikembangkan untuk memudahkan aktivitas manusia sehari-hari dan menyelesaikan masalah. Salah satu permasalahan yang masih sering terjadi hingga kini adalah kebakaran yang disebabkan oleh beberapa hal seperti hubungan arus pendek, putung rokok yang masih menyala dan kebocoran LPG. Kebakaran dapat terjadi pada area gedung kantor, sekolah, bangunan rumah dan tak terkecuali UMKM yang dalam kegiatan usahanya dapat berkontak secara langsung dengan gas. Menurut data yang dikeluarkan Dinas Pemadam Kebakaran Kota Surabaya bulan Desember tahun 2022, bahwasannya terdapat kejadian setidaknya 100 kebakaran bangunan di berbagai wilayah Kota Surabaya yang disebabkan oleh beberapa hal seperti gangguan listrik dan ledakan atau kebocoran gas [1]. Dari data yang dihimpun diatas membuktikan bahwasannya masih banyak kasus kebakaran yang diakibatkan karena berbagai hal yang berbeda salah satunya yaitu kebocoran gas. Gas merupakan salah satu energi yang diperlukan untuk memenuhi kebutuhan manusia dalam kegiatan rumah tangga atau kegiatan bisnis dalam skala besar dan kecil. Salah satu jenisnya yakni LPG yang merupakan bahan bakar gas yang berwujud gas propana atau butana atau campuran antara keduanya [2]. Kebocoran gas menjadi salah satu faktor terjadinya kebakaran yang diakibatkan karena kelalaian manusia. Sehingga diperlukan sebuah sistem yang dapat melakukan deteksi dan monitoring kebocoran gas secara *real time* untuk meminimalisir terjadinya kejadian kebakaran akibat dari bocornya gas LPG. Karena mencegah terjadinya kebakaran lebih penting daripada menanggulangi bencana tersebut [3].

Beberapa penelitian terkait alat pendeteksi kebocoran gas telah dilakukan. Pada penilitan [4] peneliti membuat alat pendeteksi gas dengan beberapa sensor gas yakni MQ2, MQ3 dan MQ5 berbasis Arduino UNO. Pengujian dilakukan dengan membandingkan kinerja beberapa sensor pada gas korek api. Sedangkan penelitian [5] peneliti membuat pendetesi kebocoran gas berbasis IoT. Kadar gas ditransmisikan dengan memanfaatkan jaringan internet dan ditampilkan via aplikasi blynk. Metode yang digunakan dalam penelitian tersebut yakni dengan mengkaji beberapa dokumen, perancangan alat dan pengujian alat. Penelitian [6] membuat pendeteksi kebocoran gas berbasis IoT namun untuk aplikasi notifikasi menggunakan telegram.

Sehingga berdasarkan penelitian-penelitian yang terdahulu, maka peneliti membuat alat pendeteksi dan monitoring gas LPG dengan protokol MQTT. Keuntungan menggunakan protokol MQTT yakni karena protokol tersebut merupakan suatu protokol sederhana dan ringan dalam proses komunikasinya. Selain itu protokol ini dalam konteks teknologi lebih baru daripada protokol HTTP [7]. Sehingga protokol tersebut cocok diimplementasikan dalam IoT. Penelitian ini menggunakan sebuah sensor bertipe MQ 2 yang dapat mendeteksi asap atau gas yang dapat digunakan sebagai alat pengamanan atau pencegahan pada kebakaran. Pada dasarnya sensor gas LPG mengeluarkan *output* tegangan analog serta mampu mengukur konsentrasi gas dari besaran 300 sampai 10.000 ppm [8]. Sensor ini dapat diatur tingkat sensitivitasnya melalui potensiometer dan keluaran analognya dapat dikonversi ke digital (ADC). Hasil deteksi sensor tersebut diproses oleh mikrokontroler NodeMCU ESP8266. Mikrokontroler tersebut mempunyai alokasi memori 4 MB dan beberapa fungsi seperti ADC dan WiFi [9] yang dapat dimanfaatkan untuk pengolahan sinyal analog dan transmisi data melalui jaringan nirkabel. Alat yang dirancang dipantau melalui layanan *Adafruit.IO* yang merupakan *software* yang dapat digunakan untuk mengontrol dan menyimpan data masukan dan keluaran [10]. Layanan ini merupakan penyedia layanan MQTT server, sehingga dapat diperuntukkan untuk melakukan remote dengan mekanisme *publish* dan *subscribe*.

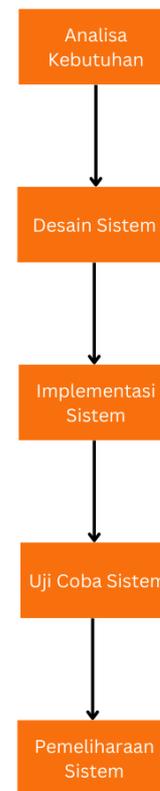
II. Metode Penelitian

A. Waktu dan Tempat Penelitian

Penulis melakukan penelitian yang dimulai pada bulan Oktober-Desember 2022. Adapun untuk objek penelitian merupakan UMKM kuliner yang dalam kegiatan sehari-harinya menggunakan gas LPG.

B. Metode

Penelitian dilaksanakan dengan menggunakan metode *Waterfall*. Penggunaan metode tersebut mempunyai keuntungan yakni tahapan demi tahapan harus dilalui setelah tahapan sebelumnya selesai [11]. Produk dapat berbentuk perangkat keras dan perangkat lunak.



Gambar 1. Metode *Waterfall*

1. Tahapan Analisis Kebutuhan
Pada tahapan pertama dilakukan analisa kebutuhan sistem mulai dari perangkat keras dan perangkat lunak. Kemudian dilakukan perencanaan perancangan sistem mulai dari pemilihan perangkat keras, komunikasi datanya dan penentuan perangkat lunak. Selain itu dalam tahapan yang pertama juga dikumpulkan data-data yang terkait dengan penelitian.
2. Tahapan Desain Sistem
Yang kedua merupakan tahapan desain sistem mulai dari desain rangkaian, desain sistem secara menyeluruh dan desain perangkat lunak. Hal ini dilakukan agar supaya sistem yang dihasilkan dapat bekerja dengan baik dan optimal.
3. Tahapan Implementasi sistem
Pada tahap ke tiga merupakan implementasi dari tahap desain yakni berupa perancangan perangkat keras, pemrograman perangkat keras atau pengkodean dan perancangan pemantauan hasil deteksi sensor.

4. Tahapan Uji Coba Sistem

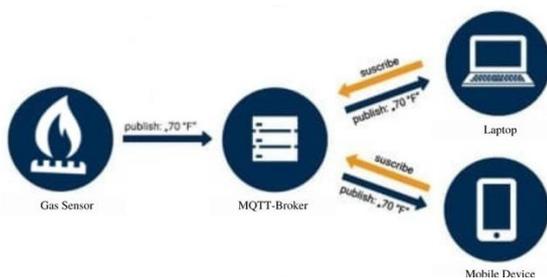
Setelah seluruh sistem selesai dirancang, maka dilakukan pengujian alat dan pemantauan hasil deteksi sensor yang bertujuan agar sistem yang telah dirancang diketahui kinerjanya, apabila masih terdapat hasil yang tidak sesuai, maka dilakukan perbaikan dan memastikan bahwa sistem berjalan dengan benar.

5. Tahapan Pemeliharaan Sistem

Tahapan yang terakhir yakni melakukan peninjauan ulang terhadap sistem yang telah dibuat baik dari perangkat keras dan perangkat lunaknya, apakah sudah sesuai dengan kebutuhan fungsional. Selain itu untuk mendukung sistem tersebut dilakukan pemeliharaan agar sistem dapat terus bekerja dengan benar, namun apabila terjadi error maka akan dapat langsung ditangani.

C. Protokol MQTT

Jenis protokol MQTT (*Message Queuing Telemetry Transport*) menerapkan mekanisme *Publish* dan *Subscribe*. Terdapat tiga bagian penting dalam protokol MQTT yakni blok *publisher*, blok *broker* dan blok *subscriber*. Blok *publisher* mempunyai tanggung jawab untuk mengirimkan informasi data sensor yang tersimpan dalam *topic* sedangkan *subscriber* memperoleh informasi berupa data sensor yang dikirim dari *publisher* dengan terlebih dahulu menjadi pelanggan pada *topic* yang tersimpan di *publisher* [12]. Dengan kata lain agar informasi diterima protokol MQTT *subscriber* harus mempunyai *topic* yang sama dengan *publisher*. Protokol ini bekerja diatas stack TCP/IP. Protokol MQTT merupakan protokol yang ringan, terbuka, sederhana dan *low bandwidth* sehingga membuat protokol tersebut dapat digunakan dalam banyak keadaan, termasuk ketika menggunakan sebagai kebutuhan *Internet of Things* (IoT) maupun komunikasi *machine-to-machine* (M2M) [13].



Gambar 2. Gambaran Sistem Perangkat

III. Hasil dan Pembahasan

Pada bagian dibawah ini merupakan penjelasan hasil dan pembahasan yang diuraikan berdasarkan tahapan-tahapan pada metode yang digunakan.

1. Hasil Analisis Kebutuhan

Hasil dari tahapan analisis kebutuhan yakni sistem yang dirancang dibagi menjadi 3 blok berupa *publisher*, *broker* dan *subscriber*.

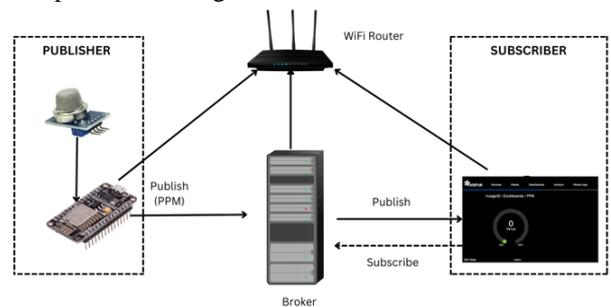


Gambar 3. Blok diagram perancangan sistem

Blok pertama yakni *publisher* terdiri dari perangkat *input* berupa sensor gas MQ2 dan mikrokontroler NodeMCU ESP8266 sebagai pengolah data sensor. Untuk pemrograman perangkat keras menggunakan *software open source Arduino IDE* dengan basis bahasa C yang disediakan di situs *arduino.cc* [14]. Blok berikutnya broker *Adafruit.IO* yang akan memfilter *topic* yang sesuai dengan *publisher* dan *subscriber*. Hasil deteksi sensor gas dipantau melalui layanan *Adafruit.IO*.

2. Hasil Desain Sistem

Berdasarkan tahapan sebelumnya, sistem yang dirancang terdiri dari *publisher*, *broker* dan *subscriber*. Blok Mikrokontroler NodeMCU ESP8266 bertugas sebagai *publisher* yang akan memberikan pesan dalam *topic* berupa data sensor gas. Data sensor tersebut ditransmisikan melalui broker yang bertugas sebagai perantara *publisher* dan *subscriber*. Broker menjadi layanan *server* yang sangat penting pada jaringan protokol MQTT agar perangkat dapat saling berkomunikasi satu sama lain [15]. Broker menerima seluruh pesan yang masuk, mengolah dan menyaring pesan agar pesan sampai pada *subscriber* dengan topik yang sesuai. Agar data sensor dapat dibaca oleh pengguna maka *subscriber* harus mempunyai *topic* berupa data sensor gas.



Gambar 4. Perancangan Sistem

3. Hasil Implementasi Sistem

Dibawah ini merupakan hasil perancangan perangkat keras berupa sensor gas MQ2, NodeMCU ESP8266 dan beberapa komponen *output* seperti lampu led dan buzzer sebagai indikator apabila sensor mendeteksi dengan batas maksimum.



Gambar 5. Perangkat Keras

Terdapat beberapa bagian penting dalam perprograman diantaranya menyiapkan kelas MQTT klien dengan mengirimkan perangkat yang menjadi *wifi client* dalam hal ini NodeMCU ESP8266, server MQTT *Adafruit.IO* dan detail *login*.

```
WiFiClient client;
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER,
AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);
Adafruit_MQTT_Publish Kadar_Gas =
Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME
"/feeds/Kadar_Gas");
```

Gambar 6. Setup klien MQTT

Dibawah ini merupakan program pembacaan sensor gas MQ2 ketika mendeteksi objek. Dengan menggunakan *compiler Arduino IDE*, untuk pembacaan dapat menambahkan *library MQ2*.

```
float*values= mq2.read(true); //jika
lpg_gas = mq2.readLPG();
```

Gambar 7. Program Pembacaan Sensor Gas

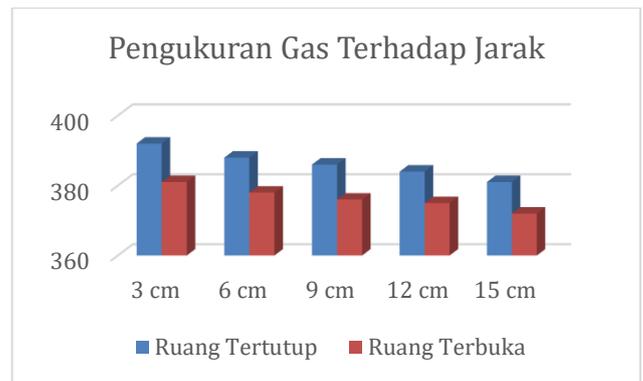
4. Hasil Uji Coba Sistem

Pengujian sistem dilakukan dengan dua cara untuk mengetahui kinerja sistem. Jenis pengujian yakni sensitivitas sensor dan delay pengiriman data ke *Adafruit* dengan protokol MQTT.

Pengujian sensor dilakukan terhadap jarak dan dua jenis kondisi ruangan (terbuka dan tertutup) dan dilakukan dalam durasi waktu masing-masing 15 menit. Alat diletakkan disekitar kompor gas untuk mengukur tingkat sensitivitasnya seperti pada gambar 8.



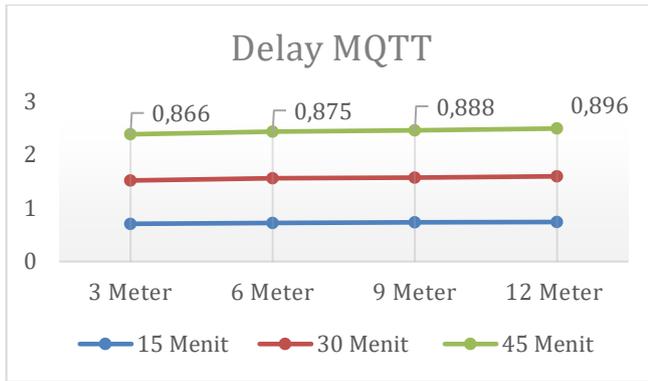
Gambar 8. Pengukuran Gas Terhadap Jarak
Hasil pengujian digambarkan dalam grafik perbandingan seperti gambar 9.



Gambar 9. Grafik Perbandingan Pengukuran Gas

Dapat dilihat dari gambar 9 bahwa kepekatan gas diudara dipengaruhi jarak dan kondisi ruangan. Pengujian pada ruang tertutup diperoleh nilai data sensor lebih tinggi daripada ruang terbuka. Hal ini disebabkan karena gas oleh benda padat seperti pintu dan tembok ruangan sedangkan pada ruang terbuka gas akan terurai keluar ruangan. Kondisi jarak juga dapat menentukan bahwa ketika situasi jarak sensor semakin jauh dengan objek maka nilai data sensor semakin kecil, yang artinya bahwa tingkat sensitivitas sensor juga dipengaruhi oleh jarak.

Berikutnya adalah hasil pengujian delay dengan protokol komunikasi MQTT. Tujuannya adalah untuk mengetahui waktu yang dibutuhkan slam pengiriman data dari pengirim ke penerima [16]. Pengiriman data diawali ketika perangkat keras tersambung dengan jaringan internet melalui *access point*. Setelah perangkat keras terhubung, maka lalu lintas data ditangkap menggunakan *tool wireshark*. Hasil pengamatan ditunjukkan dalam gambar 10.



Gambar 10. Grafik Delay MQTT

Berdasarkan pengamatan yang telah dilakukan diperoleh hasil rata-rata pengamatan delay bahwa ketika situasi jarak perangkat keras semakin jauh dengan perangkat klien maka waktu *delay* semakin tinggi. Hal ini berlaku sama dengan tiga kondisi waktu yang berbeda.

IV. Kesimpulan

Berdasarkan pengujian diperoleh kesimpulan bahwa sistem pendeteksi gas dengan protokol MQTT dapat digunakan dengan baik. Hal ini dibuktikan dengan pengujian sensitivitas sensor dan *delay*.

1. Untuk pengujian sensitivitas sensor menunjukkan bahwa tingkat sensitivitas sensor dapat dipengaruhi oleh jarak. Semakin jauh jarak dari objek yang dideteksi maka semakin rendah nilai data sensornya.
2. Pengujian *delay* menunjukkan bahwa semakin jauh jarak perangkat keras dengan perangkat klien, maka waktu pengiriman data sensor juga semakin lama.

Peneliti menyarankan untuk penelitian selanjutnya dengan penggunaan perangkat yang sama dapat melakukan pengukuran lebih banyak terkait parameter QoS (*Quality of Services*).

V. Daftar Pustaka

- [1] www.opendata.surabaya.go.id, "Jumlah kejadian kebakaran bangunan - Jumlah kejadian kebakaran bangunan - Satu Data Surabaya," 2022. https://opendata.surabaya.go.id/fa_IR/dataset/2500-5454-318/resource/969915c7-90d2-4711-bd44-9588ea0675f1 (accessed Mar. 23, 2023).
- [2] S. Sembiring, R. Panjaitan Leonardo, and A. Altway, "Pemanfaatan Gas Alam sebagai LPG (Liquified Petroleum Gas)," *JURNAL TEKNIK ITS*, vol. 8, 2019.
- [3] J. Tiara, P. Herman, ; Neneng, W. Isnawaty, and ; Candradewini, "IMPLEMENTASI KEBIJAKAN PENANGGULANGAN DAN PENCEGAHAN BAHAYA KEBAKARAN DI KOTA ADMINISTRASI JAKARTA SELATAN IMPLEMENTATION OF FIRE PREVENTION AND CONTROL POLICIES IN THE ADMINISTRATIVE CITY OF SOUTH JAKARTA," 2022.
- [4] N. Hidayat, S. Hidayat, N. A. Pramono, and U. Nadirah, "Sistem Deteksi Kebocoran Gas Sederhana Berbasis Arduino Uno," *Rekayasa*, vol. 13, no. 2, pp. 181–186, Aug. 2020, doi: 10.21107/rekayasa.v13i2.6737.
- [5] M. Gilang Ganesha, M. Ikhsan Sani, and M. Lisda Meisaroh, "(IoT GAS LEAKAGE DETECTOR BASED ON BLYNK)," 2020.
- [6] I. Istiyanto, R. Solehudin, Y. Nofarenzi, T. Setiyorini, and U. N. Mandiri, "Alat Pendeteksi Dini Kebocoran Gas LPG Dengan Sensor MQ2 Dan Sensor Api Berbasis IoT Menggunakan NodeMCU," 2022. [Online]. Available: <http://ejournal.bsi.ac.id/ejurnal/index.php/infortech>
- [7] N. P. Windryani, N. Bogi, and R. Mayasari, "ANALISA PERBANDINGAN PROTOKOL MQTT DENGAN HTTP PADA IOT PLATFORM PATRIOT COMPARISON ANALYSIS BETWEEN MQTT AND HTTP PROTOCOL IN PATRIOT IOT PLATFORM," 2019.
- [8] Sarmidi and R. Akhmad Fauzi, "JURNAL MANAJEMEN DAN TEKNIK INFORMATIKA PENDETEKSI KEBOCORAN GAS MENGGUNAKAN SENSOR MQ-2 BERBASIS ARDUINO UNO," *JUMANTAKA*, vol. 03, p. 1, 2019.
- [9] G. Suprianto and Wirawan, *2018 EECCIS : Electrical Power, Electronics, Communications, Controls and Informatics Seminar : Batu, East Java, Indonesia, October 9-11, 2018*. EECCIS, 2018.
- [10] U. Banten Jaya, S. Dwiyatno, R. Iskandar, E. Nuryani, and U. Serang Raya Jl Raya Serang Cilegon Drangong Taktakan Kota Serang Banten, "PENGENDALI LAMPU KANTOR MENGGUNAKAN GOOGLE ASSISTANT DAN ADAFRUIT. IO BERBASIS NODEMCU ESP8266," vol. 5, no. 1, 2021, [Online]. Available: <https://cf.shopee.co.id/file/c8a5271c5205c81eeb34dca2af1bdfcd>
- [11] N. Hidayati, "Penggunaan Metode Waterfall Dalam Rancang Bangun Sistem Informasi Penjualan," 2019.
- [12] M. Diono, A. Dwika Putri, H. Azwar, and dan Wahyuni Khabzli, "Sistem Monitoring Jaringan Sensor Node Berbasis Protokol MQTT," 2021. [Online]. Available: <https://jurnal.pcr.ac.id/index.php/elementer>
- [13] B. M. Susanto *et al.*, "IMPLEMENTASI MQTT PROTOCOL PADA SMART HOME SECURITY BERBASIS WEB," 2018.
- [14] P. Widya Kumara, "ROBOT LINE FOLLOWER BERBASIS MIKROKONTROLLER ARDUINO UNO ATMEGA328," *Jurnal Informanika*, vol. 5, no. 1, 2019.

-
- [15] A. Kurnianto, J. Dedy Irawan, and F. X. Ariwibisono, "PENERAPAN IOT (INTERNET OF THINGS) UNTUK CONTROLLING LAMPU MENGGUNAKAN PROTOKOL MQTT BERBASIS WEB," 2022.
- [16] L. Nurfiqin, Z. Sari, F. Dwi, and S. Sumadi, "Analisis Quality of Service (QoS) Protokol MQTT dan HTTP Pada Sistem Smart Metering Arus Listrik," *REPOSITOR*, vol. 3, no. 1, pp. 121–130, 2021.