



Sistem Monitoring Intensitas Cahaya, Kebisingan Kelembaban, dan Suhu Berbasis IoT pada Gedung Apartment Zahradni

Proposal Tugas Akhir

**Oleh:
Nur Khairi Al'Mashir (4242011013)**

**Program Studi Teknologi Rekayasa Elektronika
Jurusan Teknik Elektro
Politeknik Negeri Batam
2025**

Pernyataan Keaslian Tugas Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya yang berjudul : “Sistem Monitoring Intensitas Cahaya, Kebisingan Kelembaban, dan Suhu Berbasis IoT pada Gedung Apartment Zahradni” adalah **hasil karya sendiri, diselesaikan tanpa menggunakan bahan-bahanyang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.** Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, 24 Januari 2025



Nur Khairi Al'Mashir
NIM: 4242011013

Lembar Pengesahan

Proposal Tugas Akhir disusun untuk digunakan sebagai rencana kerja pada pelaksanaan Tugas Akhir

**Disusun oleh:
Nur Khairi Al'Mashir (4242011013)**

Tanggal Seminar: 09 01, 2025

Disetujui oleh :



**1. Nanta Fakhri Prebianto, S.ST.,
M.Sc
NIK:117183**



**1. Vivin Octowinandi, S.Tr.T., M.Sc
NIK: 120242**



**2. Illa Aryeni, S.ST., M.T
NIK: 122255**

Sistem Monitoring Intensitas Cahaya, Kebisingan, Kelembaban, dan Suhu Berbasis IoT pada Gedung Apartemen

Abstrak

Pembangunan apartemen sebagai salah satu pilihan hunian tinggi di perkotaan membutuhkan sistem pemantauan yang mampu memastikan kenyamanan penghuni dengan melacak faktor lingkungan dalam ruangan seperti tingkat cahaya, kebisingan, kelembaban, dan suhu. Tujuan dari penelitian ini adalah untuk membuat dan mengimplementasikan sistem pemantauan berbasis Internet of Things (IoT) yang memanfaatkan mikrokontroler ESP32 yang terhubung ke platform Thingsboard untuk pengamatan jarak jauh. Hasil dari pengujian menunjukkan bahwa sistem ini secara efektif mengidentifikasi dan menyajikan informasi waktu nyata tentang tingkat cahaya, kebisingan, kelembaban, dan suhu dengan akurasi yang cukup tinggi. Hasil pengujian menunjukkan bahwa sistem berhasil mendeteksi dan menampilkan data intensitas cahaya, kebisingan, kelembaban, dan suhu secara real-time dengan tingkat akurasi yang baik. Pengujian Sensor DHT22 menunjukkan error suhu rata-rata 4.47% dan kelembaban 2.28%, sensor cahaya memiliki error rata-rata 2.3%, serta sensor kebisingan menunjukkan error rata-rata 6.77%. Selain itu, sistem ini menunjukkan kinerja yang baik pada layar LCD dan melalui Thingsboard, di mana data yang dikumpulkan dapat diakses dengan cara yang mudah dan responsif. Secara keseluruhan, sistem ini efektif dalam monitoring kondisi lingkungan apartemen secara real-time, sehingga memungkinkan pengelola gedung untuk menerapkan tindakan yang tepat waktu dan sesuai untuk memastikan kenyamanan penghuni.

Kata Kunci: Apartemen, ESP32, Internet of Things (IoT), Resepsionis, Thingsboard.

IoT-Based Light Intensity, Noise, Humidity, and Temperature Monitoring System for Apartment Buildings

Abstract

Building apartments as a tall housing option in cities necessitates a monitoring system capable of ensuring residents' comfort by tracking indoor environmental factors like light levels, noise, humidity, and temperature. This study's objective is to create and implement a monitoring system based on the Internet of Things (IoT) that utilizes an ESP32 microcontroller linked to the Thingsboard platform for remote observation. The results from testing indicate that the system effectively identifies and presents real-time information about light levels, noise, humidity, and temperature with considerable accuracy. Evaluating the DHT22 sensor reveals a typical temperature deviation of 4.47% and humidity deviation of 2.28%, while the light sensor records an average inaccuracy of 2.3%, and the noise sensor reflects an average inaccuracy of 6.77%. Furthermore, the system demonstrates strong performance on the LCD display and via Thingsboard, where the data collected can be accessed in a straightforward and responsive manner. In summary, the system is proficient in overseeing the environmental conditions of apartment complexes in real-time, enabling building administrators to implement timely and suitable measures to ensure resident comfort.

Keywords: Apartment, ESP32, Internet of Things (IoT), Receptionist, Thingsboard. ESP32, Thingsboard.

Kata Pengantar

Pada kesempatan ini, penulis ingin menyampaikan penghargaan atas berbagai macam bantuan yang diberikan oleh individu-individu yang tercantum di bawah ini dalam menyelesaikan proyek akhir ini.

1. Bapak Vivin Octowinandi, S.Tr.T.,M.Sc sebagai dosen pembimbing yang telah memberikan waktu, arahan, bimbingan, serta motivasi dalam penyusunan tugas akhir ini.
2. Bapak Uuf Brajawidagda, S.T., M.T., Ph.D. selaku direktur dari Politeknik Negeri Batam.
3. Bapak Nanta Fakhri Prebianto, S.ST., M.Sc selaku kepala program studi Teknik Rekayasa Elektronika dan sebagai dosen penguji 1.
4. Bapak Illa Aryeni, S.ST., M.T selaku dosen pengampu mata kuliah Tugas Akhir dan sebagai dosen penguji 2 .
5. Kepada orang tua dan juga keluarga yang menjadi semangat, motivasi dan tentunya memberikan doa yang terbaik.
6. Seluruh dosen-dosen jurusan Teknik Elektro Politeknik Negeri Batam yang telah memberikan banyak ilmu dan pembelajaran kepada penulis.
7. Seluruh teman-teman seperjuangan yang telah membantu proses penyelesaian tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini memiliki keterbatasan, oleh karena itu penulis dengan tulus mencari umpan balik dan rekomendasi yang berguna dari berbagai sumber untuk mencapai kesempurnaan dalam Tugas Akhir ini. Pada akhirnya, penulis berharap Tugas Akhir ini dapat memberikan manfaat bagi semua pihak yang membutuhkan.

Republik Ceko, 5 Mei 2025



Nur Khairi Al Mashir

Daftar Isi

Pernyataan Keaslian Tugas Akhir ii

Lembar Pengesahan iv

Abstrak v

Abstract vi

Kata Pengantar vii

Daftar Isi viii

Daftar Gambar x

Daftar Tabel xi

Bab 1. Pendahuluan 1

 1.1. Latar Belakang 1

 1.2. Rumusan Masalah 2

 1.3. Tujuan 2

 1.4. Manfaat 3

 1.5. Batasan 3

Bab 2. Tinjauan Pustaka 4

 2.1. Apartemen 4

 2.2. Internet of Things 4

 2.3. DHT22 5

 2.4. Analog Sound Sensor 5

 2.5. Analog Ambient Light Sensor 6

2.6.	<i>ESP32</i>	7
2.7.	<i>LCD Display</i>	7
2.8.	<i>Arduino IDE</i>	8
2.9.	<i>Thingsboard</i>	9
Bab 3.	Metodologi Penelitian.....	10
3.1.	<i>Perancangan</i>	10
3.2.	<i>Diagram Blok Sistem</i>	11
3.3.	<i>Perancangan Perangkat Keras</i>	12
3.3.1	Perancangan Mekanikal	12
3.3.1	Perancangan Elektrikal	13
3.4.	<i>Perancangan Software</i>	15
3.5.	<i>Pengujian</i>	16
3.5.1.	Pengujian Unit	16
3.5.2.	Pengujian Sistem	19
3.6.	<i>Alat dan Bahan</i>	20
Bab 4.	Hasil dan Pembahasan.....	21
4.1.	<i>Hasil Pengujian Unit</i>	21
4.1.1.	Hasil Pengujian Adaptor	21
4.1.2.	Hasil Pengujian ESP32	22
4.1.3.	Hasil Pengujian LCD Display	24
4.1.4.	Hasil Pengujian DHT22	27
4.1.5.	Hasil Pengujian Analog Sound Sensor	30
4.1.6.	Hasil Pengujian Analog Ambient Light Sensor.....	32
4.2.	<i>Pengujian Sistem</i>	35
Bab 5.	Kesimpulan dan Saran	48
5.1.	Kesimpulan	48
5.2.	Saran.....	48
Daftar Pustaka		49

Daftar Gambar

Gambar 2. 1 Arsitektur Internet of Things	4
Gambar 2. 2 DHT22	5
Gambar 2. 3 Analog Sound Sensor	6
Gambar 2. 4 Analog Ambient Light Sensor V2	6
Gambar 2. 5 Datasheet ESP32	7
Gambar 2. 6 modul LCD karakter 20x4	8
Gambar 2. 7 Arduino IDE	8
Gambar 2. 8 Logo Thingsboard	9
Gambar 3. 1 Flowchart Perancangan Penelitian	10
Gambar 3. 2 Diagram Blok Sistem	11
Gambar 3. 3 Desain Mekanik	12
Gambar 3. 4 Desain Elektrikal	13
Gambar 3. 5 Flowchart Sistem	15
Gambar 4. 1 Pengujian Adaptor	22
Gambar 4. 2 Hasil Pengujian ESP32	24
Gambar 4. 3 Pengujian LCD Display	26
Gambar 4. 4 Pengukuran DHT 22	30
Gambar 4. 5 Analog Sound Sensor	32
Gambar 4. 6 Pengujian Ambient Light Sensor	35
Gambar 4. 7 Pengujian Sistem	43
Gambar 4. 8 Riwayat Pembacaan Suhu	44
Gambar 4. 9 Riwayat Pembacaan Kelembaban	45
Gambar 4. 10 Riwayat Pembacaan Cahaya	46
Gambar 4. 11 Riwayat Pembacaan Kebisingan	47

Daftar Tabel

Tabel 3. 1 Konfigurasi Setiap Pin	14
Tabel 3. 2 Format Pengujian Adaptor.....	16
Tabel 3. 3 Format Pengujian ESP32	17
Tabel 3. 4 Format Pengujian LCD Display	17
Tabel 3. 5 Format Pengujian DHT22	18
Tabel 3. 6 Format Pengujian Analog Sound Sensor	18
Tabel 3. 7 Format Pengujian Ambient Light Sensor	19
Tabel 3. 8 Format Pengujian Adaptor.....	19
Tabel 3. 9 Alat dan Bahan.....	20
Tabel 4. 1 Hasil Pengujian Adaptor.....	21
Tabel 4. 2 Hasil Pengujian ESP32	23
Tabel 4. 3 Hasil Pengujian LCD Display	25
Tabel 4. 4 Hasil Pengujian Sensor DHT22	28
Tabel 4. 5 Hasil Pengujian Analog Sound Sensor.....	31
Tabel 4. 6 Hasil Pengujian Ambient Light Sensor	34

Bab 1. Pendahuluan

1.1. Latar Belakang

Dalam publikasinya “Apartments & Dormitory,” James Hombeck menggambarkan apartemen sebagai jenis konstruksi yang terletak di lokasi yang dibagi menjadi beberapa area terpisah yang diatur baik secara vertikal maupun horizontal untuk alasan fungsional. Setiap bagian bertindak sebagai unit individu yang dapat beroperasi sendiri, terutama sebagai tempat tinggal yang berbagi tanah, peralatan, dan fasilitas umum. Ensiklopedia Nasional Indonesia mendefinisikan apartemen sebagai tempat tinggal vertikal kontemporer yang dirancang untuk mengatasi tantangan yang ditimbulkan oleh keterbatasan ruang kota. Apartemen yang ditujukan untuk tempat tinggal harus memberikan kenyamanan yang cukup bagi penghuninya untuk melakukan kegiatan sehari-hari berdasarkan kriteria yang telah ditetapkan.[1]

Bluyssen menyebutkan bahwa masalah utama yang mempengaruhi kenyamanan dalam ruangan di rumah-rumah di seluruh Eropa termasuk kebisingan, kenyamanan termal yang buruk, kelembapan, dan kualitas udara, terutama yang berasal dari ventilasi yang tidak memadai. Temuannya menunjukkan bahwa suhu dan kelembapan secara signifikan mempengaruhi iklim mikro dalam ruangan. Memperbarui jendela yang sudah ketinggalan zaman sering kali dapat menyebabkan penurunan kualitas udara karena ventilasi yang tidak memadai, tetapi jendela baru mungkin memiliki transmisi termal hingga tiga kali lebih sedikit.[2]

Kenyamanan visual resepsionis akan mempengaruhi penglihatan mereka, yang diatur oleh tata letak dan desain eksterior kompleks apartemen, menentukan seberapa banyak cahaya yang masuk ke dalam hunian. Faktor ini memengaruhi kapasitas resepsionis untuk menjalankan tugas dan mengidentifikasi ruang-ruang tertentu. [3] Prinsip yang sama berlaku untuk faktor-faktor seperti suhu, kelembapan, dan tingkat kebisingan. Lingkungan mempengaruhi setiap tindakan yang dilakukan seseorang. Untuk mengoptimalkan kinerja, penting untuk menciptakan ruang yang mendukung aktivitas manusia. Menurut pedoman kesehatan perumahan yang diuraikan dalam KepMenkes No.829/Menkes/SK/VII/1999, pencahayaan ruangan harus berada dalam kisaran 120 hingga 250 lux. Suhu yang disarankan harus dijaga antara 18°C dan 30°C, sementara tingkat kelembapan yang ideal harus berada di antara 40% dan 70%. Seperti yang dinyatakan dalam Keputusan Menteri Lingkungan Hidup No.48 tahun 1996, kebisingan di dalam ruangan tidak boleh melebihi 30 hingga 40 desibel.[4]

Setiap orang dapat memperoleh keuntungan dari kenyamanan teknologi modern yang berkembang pesat[5]. Dengan perspektif ini, teknologi yang berkaitan dengan internet of things telah diciptakan untuk memfasilitasi pengamatan lingkungan melalui koneksi internet. Istilah Internet of Things, atau

IoT, mengacu pada benda-benda yang dapat berbagi data atau informasi melalui jaringan secara mandiri, tanpa campur tangan manusia. Integrasi sistem mikroelektromekanis dan jaringan internet nirkabel menghasilkan teknologi IoT. Meskipun demikian, “A Things” dapat mencakup berbagai macam benda, termasuk ternak yang dilengkapi dengan chip transponder dan manusia yang dipasang implan jantung, di antara contoh-contoh lainnya. IoT terkait erat dengan komunikasi mesin-ke-mesin, yang sering disebut sebagai komunikasi cerdas, dan berlangsung tanpa memerlukan keterlibatan manusia atau komputer. [6]

Beberapa penelitian sebelumnya yang telah dilakukan antara lain; Mengamati Kualitas Udara dan Evaluasi Kebisingan di SMAN 1 Semarang, Kabupaten Klungkung. [5], Monitoring Cahaya, Suara, Suhu, dan Kelembaban pada Ruang Pembelajaran Berbasis Mikrokontroler [7], dan Sistem Monitoring pada Ruang Perpustakaan berbasis Internet of Things [8] Penelitian sebelumnya tidak dilakukan di dalam apartemen dan tidak dilacak melalui platform Thingsboard, sehingga menjadikan penyelidikan ini sebagai penelitian baru. Menanggapi masalah yang teridentifikasi, penulis tesis telah mengembangkan sistem pemantauan Intensitas Cahaya, Kebisingan, Kelembaban, dan Suhu yang dapat dilihat di situs web Thingsboard. Alat ini memungkinkan resepsionis untuk memantau kondisi secara real-time dari jarak jauh, sehingga mereka dapat dengan cepat dan akurat menanggapi masalah apa pun yang muncul di kompleks apartemen, memastikan mereka dapat memberi tahu individu yang tepat.

1.2. Rumusan Masalah

Rumusan masalah yang perlu diselesaikan dalam merancang sistem kontrol suhu, kelembaban, cahaya, dan keasaman berbasis IoT, antara lain:

1. Bagaimana cara merancang dan membangun *hardware* serta *software* yang mampu mendeteksi Intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT?
2. Bagaimana prinsip kerja dari alat *monitoring* Intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT?
3. Bagaimana kinerja dari alat monitoring suhu, intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT?

1.3. Tujuan

Tujuan dari penelitian ini yaitu sebagai berikut:

1. Mampu merancang *hardware* serta *software* yang dapat mendeteksi dan memantau intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT.
2. Mampu mengetahui prinsip kerja alat *monitoring* intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT.

3. Mampu mengetahui kinerja alat Mampu mengetahui prinsip kerja alat *monitoring* intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT.

1.4. Manfaat

Manfaat dari penelitian ini yaitu sebagai berikut:

1. Dengan adanya *monitoring* intensitas cahaya, kebisingan, kelembaban, dan suhu ini resepsionis dapat mengetahui kondisi apartmenya secara *real time*.
2. Resepsionis dapat melakukan evaluasi terhadap kondisi apartemen dari data yang didapatkan.

1.5. Batasan

1. Penelitian ini dilakukan di lantai 4 Apartment Zahradni, Plzen Republik Ceko.
2. Parameter yang di monitoring berupa Intensitas, Cahaya, Kebisingan, Kelembaban dan Suhu.
3. Mikrokontroler menggunakan ESP32.
4. Monitoring hanya dapat dilakukan menggunakan website thingsboard.

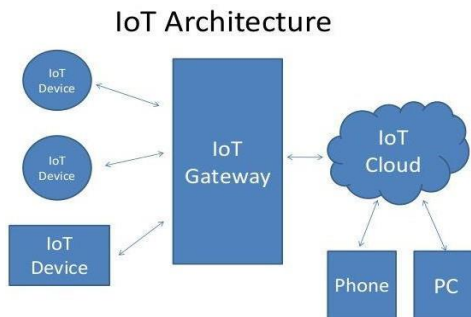
Bab 2. Tinjauan Pustaka

2.1. Apartemen

Apartemen merupakan unit hunian yang terbagi secara vertikal maupun horizontal namun masih merupakan bagian dari bangunan yang sama. Apartemen dilengkapi dengan berbagai fasilitas penunjang dan pelengkap yang dapat dimanfaatkan oleh masyarakat secara keseluruhan, khususnya para pebisnis, karyawan, dan individu yang mencari lingkungan yang cocok, aman, dan menyenangkan untuk tempat tinggal mereka. [9] Bangunan ini memiliki lebih dari dua lantai, dengan beberapa unit yang disusun berdasarkan harga dan kapasitas pada setiap lantainya. [10]

2.2. Internet of Things

Kumpulan perangkat yang saling terhubung melalui internet dan memiliki kemampuan untuk berinteraksi secara mandiri atau otomatis tanpa campur tangan manusia disebut sebagai “Internet of Things”. Ungkapan “Internet of Things” (IoT) merangkum konsep berbagi informasi melalui jaringan online yang menghubungkan berbagai gadget, termasuk sensor, aktuator, atau pengontrol IoT, yang dapat dikelola dari jarak jauh dan secara mandiri. Ditandai dengan penggunaan teknologi RFID, sensor, dan GPS untuk secara otomatis mengenali barang, informasi, dan tempat sesuai kebutuhan pengguna, Internet of Things diidentifikasi sebagai jaringan yang menghubungkan objek fisik ke internet. [11] Internet of Things membutuhkan sistem perlindungan untuk menjaga dari berbagai ancaman karena kemajuan teknologi yang terus menerus. Aspek-aspek utama dari Internet of Things meliputi perlindungan data, integritas operasional, dan keamanan fisik.[12] Gambar 2.1 merupakan Arsitektur Internet of Things.

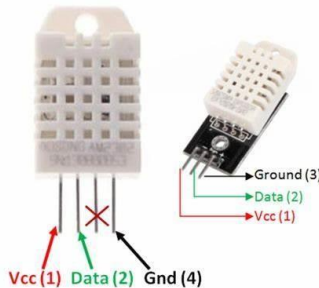


Gambar 2. 1 Arsitektur Internet of Things[13]

2.3. DHT22

Perangkat sensor digital yang disebut DHT22 digunakan untuk mengukur suhu sekitar dan kelembapan relatif. Perangkat ini bekerja dengan memperkirakan kondisi udara menggunakan termistor dan kapasitor, yang kemudian diubah menjadi sinyal yang dikirim melalui pin data. Manfaat sensor DHT22 ditonjolkan oleh kapasitasnya untuk memberikan pembacaan yang akurat. Sensor ini dapat dianggap sebagai sensor dengan mutu tertinggi karena responsnya, pembacaan data yang cepat, dan ketahanan terhadap gangguan. Perangkat ini memiliki desain yang kecil dan dapat mengirimkan sinyal hingga 20 meter.[14]

Dengan ketidakakuratan suhu $\pm 0,5^{\circ}\text{C}$, sensor ini dapat mendeteksi suhu antara -40°C dan 125°C . Selain itu, tingkat kesalahan untuk kelembapan berkisar antara $\pm 2\%$ hingga $\pm 5\%$. Sensor digital DHT22 memiliki konverter analog-ke-digital terintegrasi. Modul ini beroperasi antara 3 dan 5 volt. Gambar 2. Pin Sensor DHT22 Mirip dengan DHT11, sensor suhu dan kelembapan DHT22 memiliki empat pin, tetapi hanya tiga—VCC, GND, dan data—yang digunakan.[15] Gambar 2.2 merupakan gambar DHT 22.



Gambar 2. 2 DHT22 [26]

2.4. Analog Sound Sensor

Sensor suara analog adalah instrumen yang dapat mengubah getaran suara menjadi gelombang energi listrik yang sesuai. Alat ini beroperasi dengan membuat gelombang suara menghantam membran sensor, menyebabkannya bergerak dalam kumparan kecil yang diposisikan di bawahnya. Saat membran berosilasi, membran memotong medan magnet, dan gerakan ini terjadi karena struktur kumparan yang berongga. Laju pergerakan kumparan menentukan kekuatan sinyal listrik yang dihasilkan. Perangkat yang mengubah suara menjadi sinyal listrik disebut sebagai sensor suara. Komponen yang membentuk sensor suara termasuk mikrokontendor, umumnya dikenal sebagai mikrofon kondensor listrik.[16]

Modul pendeteksi suara yang menampilkan output analog yang menawarkan kompatibilitas yang sangat baik dengan mikrokontroler seperti Arduino atau

Raspberry PI disediakan dalam Modul Sensor Suara Analog V2.2 DFRobot Gravity. Spesifikasi Sensor Suara Analog V2.2 terdiri dari dua lubang pemasangan berdiameter 3 mm dengan jarak 5 cm, sinyal keluaran analog, dan suplai tegangan yang bervariasi antara 3,3 V dan 5,0 V. Kit ini juga termasuk kabel yang dilengkapi dengan ujung kerja standar pin emas.. [17] Gambar 2.3 merupakan gambar Analog Sound Sensor.



Gambar 2. 3 Analog Sound Sensor [17]

2.5. Analog Ambient Light Sensor

Sensor intensitas cahaya BH1750 adalah instrumen yang menghasilkan sinyal digital dan menghindari perhitungan yang rumit. Tidak seperti sensor konvensional seperti fotodioda dan resistor yang bergantung pada cahaya, yang menghasilkan sinyal analog dan memerlukan pemrosesan untuk menilai pembacaan intensitas cahaya, sensor BH1750 menawarkan akurasi yang lebih baik dan kesederhanaan penggunaan. Sensor ini memungkinkan pengukuran cahaya secara langsung dalam lux tanpa memerlukan perhitungan sebelumnya. Sensor ini dapat mengukur hingga maksimum 54612 lux, sedangkan level terendah yang dapat didaftarkan adalah 0 lux.[18] Gambar 2.4 merupakan Analog Ambiaent Light Sensor.



Gambar 2. 4 Analog Ambient Light Sensor V2 [17]

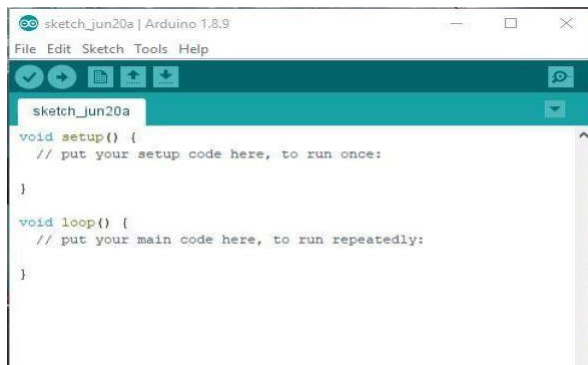
This module includes a microcontroller purpose-built to manage the LCD operations. The LCD comprises 16 pins, each serving distinct functions.[21] Gambar 2.6 merupakan Modul LCD karakter 4x20.



Gambar 2. 6 modul LCD karakter 20x4 [21]

2.8. Arduino IDE

Arduino IDE, atau Lingkungan Pengembangan Terpadu, berfungsi sebagai editor teks yang cocok untuk membuat kode, menulis pesan, dan terlibat dengan konsol teks. Program diunggah melalui koneksi antara Arduino IDE dan perangkat keras papan Arduino. Sebagai editor teks, Arduino IDE memungkinkan penulisan, pengeditan, dan validasi kode komputer. Kode yang dibuat untuk proyek Arduino, ditandai dengan ekstensi file .ino, disebut sebagai “sketsa” Arduino atau kode sumber Arduino.[23] Karena kode komputer merupakan komponen terpenting



Gambar 2. 7 Arduino IDE

dalam pembuatan alat, sistem tidak dapat berfungsi tanpanya [24] Gambar 2.7 Merupakan gambar Arduino IDE.

2.9. Thingsboard

Thingsboard adalah platform Internet of Things sumber terbuka untuk mengelola perangkat, mengumpulkan, memproses, dan memvisualisasikan data. ThingsBoard memungkinkan kita untuk mengumpulkan dan menampilkan data dari aset dan perangkat. Merekam kejadian yang kompleks, menganalisis telemetry yang masuk, dan menyalakan alarm. Memanfaatkan panggilan prosedur jarak jauh (RPC) untuk mengelola perangkat Anda. Membuat proses berdasarkan peristiwa REST API, permintaan RPC, peristiwa siklus hidup perangkat, dll. Membuat dasbor yang dinamis dan adaptif yang memberi pengguna akses ke telemetry dan wawasan aset atau perangkat [21] Thingsboard adalah platform sumber bebas yang dirancang untuk menangani perangkat Internet of Things, memungkinkan pengumpulan, pemrosesan, dan visualisasi data. Platform ini memungkinkan pengumpulan dan menampilkan informasi dari berbagai perangkat dan aset. Melacak kejadian yang rumit, mengevaluasi telemetry waktu nyata, dan memicu peringatan. Menggunakan panggilan prosedur jarak jauh untuk menangani perangkat Anda secara efektif. Kembangkan alur kerja yang dipengaruhi oleh peristiwa REST API, panggilan RPC, pembaruan siklus hidup perangkat, dan banyak lagi. Rancang dasbor yang fleksibel dan responsif yang memberikan wawasan dan telemetry kepada pengguna dari perangkat atau aset mereka.[25] Gambar 2.8 Merupakan Logo Thingsboard.

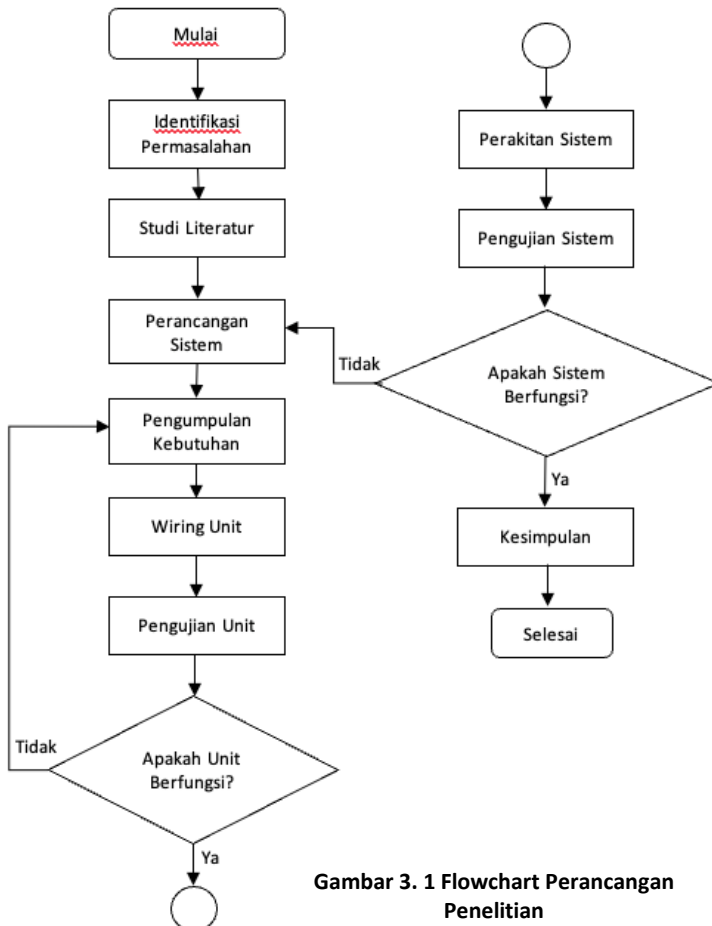


Gambar 2. 8 Logo Thingsboard [23].

Bab 3. Metodologi Penelitian

3.1. Perancangan

Dalam merancang sebuah sistem, diperlukan suatu metode pelaksanaan agar rancangan penelitian yang dibuat terstruktur dan jelas. Tahapan-tahapan dalam pelaksanaan ini tentunya harus efektif dan efisien, sehingga tidak membuang banyak biaya dan waktu secara percuma. Gambar 3.1 merupakan Flowchart Perencanaan Penelitian

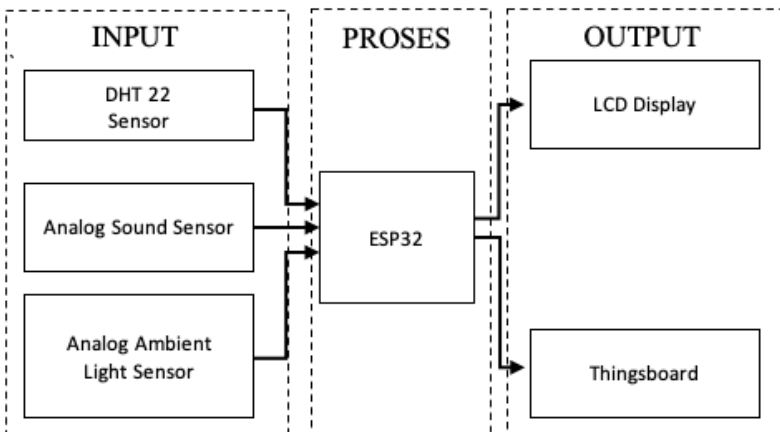


Gambar 3. 1 Flowchart Perancangan Penelitian

Dalam pembuatan sistem, beberapa fase upaya diilustrasikan pada Gambar 3.1. tahapan ini perlu dijalankan untuk mencapai hasil yang diinginkan.. Pada penelitian dan pembuatan alat ini terbagi menjadi beberapa tahapan yaitu identifikasi permasalahan, studi literatur, perancangan sistem, pengumpulan kebutuhan, wiring unit, perakitan sistem, pengujian sistem, dan kesimpulan.

3.2. Diagram Blok Sistem

Strategi teknis awal untuk pendekatan penelitian ini melibatkan pembuatan diagram blok. Tujuan dari diagram blok berfungsi sebagai panduan untuk menyusun operasi sistem perangkat keras. Pemilihan diagram blok yang sesuai akan mempengaruhi realisasi konsep yang diinginkan dalam menyelesaikan tugas akhir.



Gambar 3. 2 Diagram Blok Sistem

Pada Gambar 3.2 merupakan blok diagram dari sistem yang dibuat. Adapun bagian dari blok diagram dibagi menjadi 3 yaitu:

1. *Input*

Pada bagian input, terdapat sensor DHT22, Analog Sound Sensor, dan Analog Ambient Sensor.

2. *Proses*

Pada bagian proses terdapat ESP32 yang berfungsi untuk menerima data-data sensor, memproses, dan menganalisis data sensor untuk meneruskan data sensor ke thingsboard dan LCD Display.

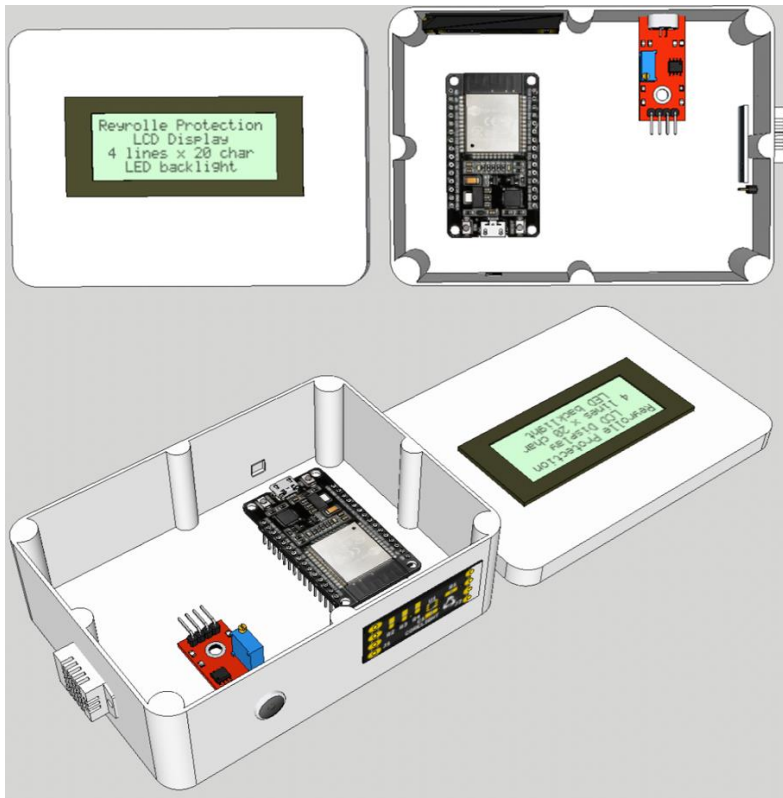
3. *Output*

Didalam output terdapat Dashboard thingsboard dan LCD Display yang berfungsi untuk menampilkan data sensor, Dashboard Thingsboard berfungsi untuk menampilkan data sensor secara online, LCD Display berfungsi untuk menampilkan data sensor secara offline.

3.3. Perancangan Perangkat Keras

Perancangan sistem perangkat keras dilakukan dengan membuat desain perancangan mekanikal serta desain perancangan elektrikal sebagai berikut.

3.3.1 Perancangan Mekanikal



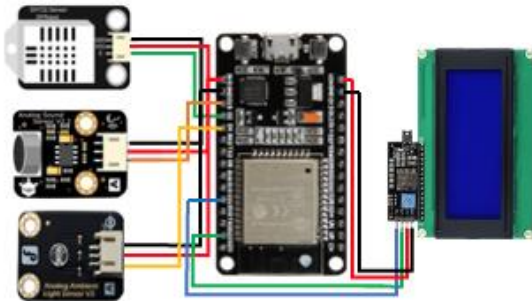
Gambar 3. 3 Desain Mekanik

Pada gambar 3.3 merupakan desain mekanik dari penelitian yang akan dilakukan, yang mana didalamnya terdapat beberapa komponen seperti ESP32, DHT 22, Analog Sound Sensor, Analog Ambient Light Sensor, dan LCD Display 20x4.

Fungsi komponen:

- Sensor DHT22 berfungsi untuk membaca suhu dan kelembaban ruangan apartemen
- Analog Sound Sensor berfungsi untuk membaca tingkat kebisingan ruangan apartemen
- Analog Ambient Light Sensor berfungsi untuk membaca intensitas cahaya pada ruangan apartemen
- ESP32 berfungsi untuk memproses data sensor dan mengirimkan pada LCD Display dan Thingsbord
- LCD Display berfungsi untuk menampilkan data-data sensor yang telah di proses oleh ESP32.

3.3.1 Perancangan Elektrikal



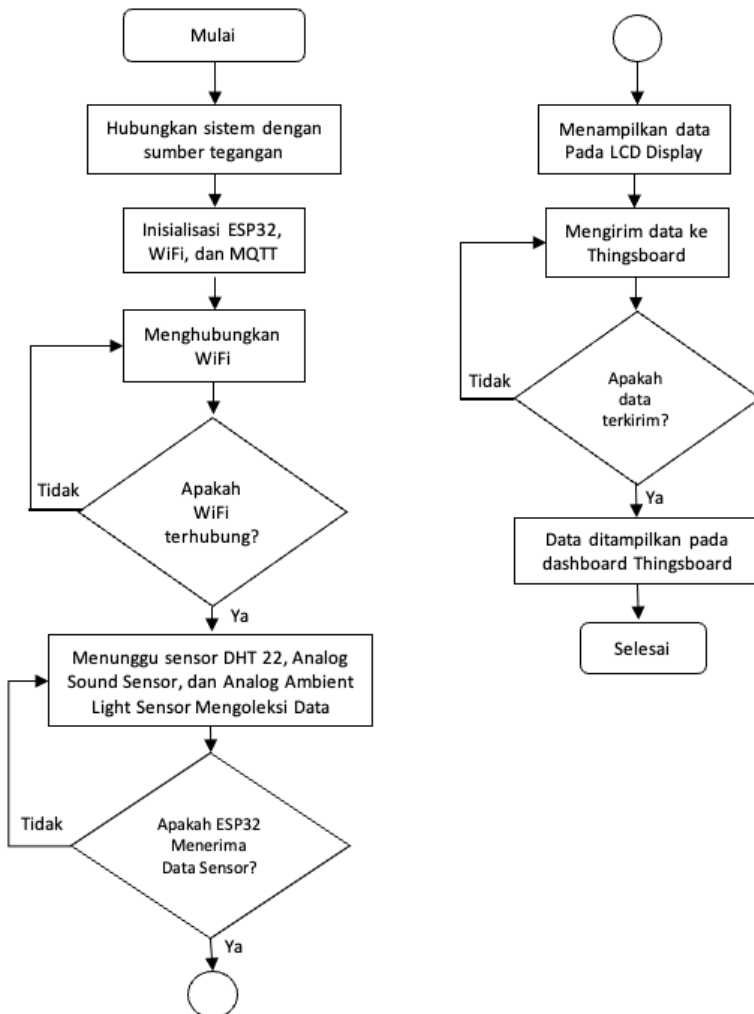
Gambar 3. 4 Desain Elektrikal

Gambar 3.4 merupakan desain elektrik pada penelitian ini yang mana dirancang dengan beberapa komponen, Seperti DHT22, Analog Sound Sensor, Analog Ambient Light Sensor, dan LCD Display 20x4 yang saling terhubung menjadi sebuah sistem. Tabel 3.1 merupakan Konfigurasi Setiap Pin.

Tabel 3. 1 Konfigurasi Setiap Pin

No	ESP32	Sensor DHT22
1	Pin 3V	Pin VCC
2	Pin GND	Pin GND
3	Pin D2	Pin Signal
No	ESP32	Analog Sound Sensor
1	Pin 3V	Pin VCC
2	Pin GND	Pin GND
3	Pin D15	Pin Signal
No	ESP32	Analog Ambient Light Sensor
1	Pin 3V	Pin VCC
2	Pin GND	Pin GND
3	Pin D4	Pin Signal
No	ESP32	LCD Display
1	Pin VIN	Pin VCC
2	Pin GND	Pin GND
3	Pin D21	Pin SCL
4	Pin D22	Pin SDA

3.4. Perancangan Software



Gambar 3. 5 Flowchart Sistem

Gambar 3.5 menjelaskan mengenai cara kerja sistem. Ketika sistem dihubungkan dengan sumber tegangan, maka akan menginisialisasi

mikrokontroler dan WiFi . Sistem akan menghubungkan ke WiFi apabila WiFi tidak terhubung akan mencoba untuk dapat terhubung apabila terhubung maka bisa langsung ke tahap selanjutnya yaitu menerima data sensor. Sehingga data sensor dapat dikirim ke Thingsboard dan LCD Display.

3.5. Pengujian

3.5.1. Pengujian Unit

Pada tahap ini, setiap bagian yang dimaksudkan untuk digunakan, menjalani pengujian awal berdasarkan penelitian teoretis yang dilakukan untuk menentukan fungsionalitasnya. Evaluasi ini terdiri dari:

1. Pengujian Adaptor

Pengujian adaptor dilakukan selama 5 kali percobaan yang mana output tegangan dari adaptor akan diukur menggunakan multimeter, Pengujian ini dilaksanakan untuk mengetahui apakah adaptor berfungsi sesuai yang diinginkan. Adaptor sangat penting karena menyalakan semua komponen elektronik yang membutuhkan tegangan. Berikut merupakan rumus perhitungan presentasi *error* tegangan keluaran pada adaptor.

selisih nilai pembacaan

$$\text{Error} = \frac{\text{selisih nilai pembacaan}}{\text{Tegangan Teori}} \times 100\%$$

Tegangan

Adapun format tabel pengujian adaptor dalam bentuk keluaran tegangan dan nilai *error* terdapat pada tabel 3.2.

Tabel 3. 2 Format Pengujian Adaptor

Pengujian ke-	Tegangan Ukur (VDC)	Tegangan Teori (VDC)	Error %
1			
2			
3			
4			
5			
Rata-Rata			

2. Pengujian ESP32

Pengujian ESP32 dilakukan agar peneliti mengetahui apakah ESP32 sudah terhubung ke jaringan WiFi atau tidak. Dengan dilakukan pengisian nama SSID beserta password dari WiFi yang digunakan. Kemudian dilakukan pemeriksaan pada serial monitor menunjukkan “terhubung” maka ESP32 sudah terhubung ke jaringan WiFi.

Adapun format tabel pengujian ESP32 terdapat pada tabel 3.3.

Tabel 3. 3 Format Pengujian ESP32

Pengujian ke-	Nama WiFi	Terkoneksi	
		Ya	Tidak
1			
2			
3			
4			
5			

3. Pengujian LCD Display

Memverifikasi pengaturan LCD 20x4 dimaksudkan untuk melihat apakah ESP32 dapat mengatur LCD untuk menampilkan teks. Sebelum melakukan verifikasi, pengaturan LCD harus terhubung dengan ESP32. Proses verifikasi melibatkan menampilkan karakter pada LCD berdasarkan kode yang ditulis dalam Arduino IDE.

Adapun format tabel pengujian LCD Display dalam menampilkan karakter terdapat pada tabel 3.4.

Tabel 3. 4 Format Pengujian LCD Display

Pengujian ke-	Perintah Tampilan	Tampilan LCD	Keterangan
1			
2			
3			
4			
5			

4. Pengujian Sensor DHT 22

Pengujian sensor suhu dilakukan dengan membandingkan pembacaan dari sensor DHT22 dengan termometer digital. Pengujian dilakukan dengan merekam suhu dan tingkat kelembapan dalam tiga skenario yang berbeda: panas, normal, dan dingin, dengan lima kali pengulangan untuk setiap kondisi. Percobaan ini menentukan nilai kesalahan untuk mengevaluasi pengukuran terhadap pengukuran yang dilakukan dengan termometer digital. Perhitungan nilai kesalahan untuk pembacaan menggunakan rumus langsung yang disajikan dalam persamaan.

$$Error = \frac{NS-NA}{NA} \times 100\%$$

Keterangan:

Error = Presentase perbedaan Nilai Pembacaan(%)

NS = Nilai Pembacaan Sensor

NA = Nilai Pembacaan Alat Ukur

Adapun format tabel pengujian sensor DHT22 dan nilai *error* terdapat pada tabel 3.5.

Tabel 3. 5 Format Pengujian DHT22

Pengujian Kondisi	DHT 22		Thermometer Digital		Selisih (%)	
	Suhu (°C)	Kelembaban (%)	Suhu (°C)	Kelembaban (%)	Suhu	Kelembaban
Dingin						
Normal						
Panas						

5. Pengujian Analog Sound Sensor*

Dalam mengevaluasi sensor suhu, hasil yang diperoleh dari Sensor Suara Analog disandingkan dengan hasil dari Pengukur Desibel. Penilaian dilakukan dalam tiga skenario yang berbeda, termasuk kondisi standar dan lingkungan yang bising, dengan masing-masing skenario diuji lima kali untuk menghitung nilai kesalahan untuk dibandingkan dengan pengukuran Decibel Meter. Penghitungan Nilai Error Pembacaan menggunakan rumus sederhana yang ditunjukkan pada rumus.

$$Error = \frac{NS-NA}{NA} \times 100\%$$

Keterangan:

Error = Presentase perbedaan Nilai Pembacaan(%)

NS = Nilai Pembacaan Sensor

NA = Nilai Pembacaan Alat Ukur

Adapun format tabel pengujian Analog Sound Sensor dan *error* terdapat pada tabel 3.6.

Tabel 3. 6 Format Pengujian Analog Sound Sensor

Pengujian Kondisi	Analog Sound Sensor (dB)	Sound Level Meter (dB)	Selisih (%)
Normal			
Bising			

6. Pengujian Analog Ambient Light Sensor

Pemeriksaan sensor suhu melibatkan pembandingan pembacaan yang diperoleh dari Sensor Cahaya Sekitar Analog dengan pembacaan dari Lux

Meter. Pengujian dilakukan dengan mengukur di 3 kondisi yaitu terang, normal, gelap masing-masing dilakukan 5 kali percobaannya dihitung nilai error nya untuk membandingkan hasil pengukuran dengan Lux Meter. Penghitungan Nilai Error Pembacaan menggunakan rumus sederhana yang ditunjukkan pada rumus.

$$Error = \frac{NS - NA}{NA} \times 100\%$$

Keterangan:

Error = Presentase perbedaan Nilai Pembacaan(%)

NS = Nilai Pembacaan Sensor

NA = Nilai Pembacaan Alat Ukur

Adapun format tabel pengujian Ambient Light Sensor dan nilai *error* terdapat pada tabel 3.7.

Tabel 3. 7 Format Pengujian Ambient Light Sensor

Pengujian Kondisi	Analog Ambient Sensor (lx)	Lux Meter (lx)	Selisih (%)
Terang			
Normal			
Gelap			

3.5.2. Pengujian Sistem

Untuk menentukan kinerja alat secara keseluruhan, seluruh sistem harus diuji. Respons semua komponen terhadap tindakan masukan dan keluaran setiap alat diperiksa selama pengujian ini. Alat akan dipasang pada lantai 4 gedung apartemen Zahradni selama 1 hari, data yang masuk pada thingsboard akan dianalisis untuk melihat fungsi alat apakah sudah sesuai dengan yang diinginkan atau belum.

Adapun format tabel pengujian sistem dan keteranganya terdapat pada tabel 3.8.

Tabel 3. 8 Format Pengujian Adaptor

Jam ke-	Suhu	Kelembaban	Cahaya	Kebisingan	LCD Display	Thingsboard	Keterangan
1							
2							
3							

3.6. Alat dan Bahan

Dibutuhkan alat dan bahan yang digunakan dalam membuat penelitian ini, Tabel 3.9 merupakan alat dan bahan yang digunakan.

Tabel 3. 9 Alat dan Bahan

No.	Alat/bahan	Harga Satuan (Rp.)	Jumlah	Total (Rp.)	Keterangan
1.	ESP32	Rp. 125.000	1	Rp. 125.000	Dana Pribadi
2.	Analog Sound Sensor	Rp. 95.000	1	Rp. 95.000	Dana Pribadi
3.	DHT22	Rp. 200.000	1	Rp. 200.000	Dana Pribadi
4.	LCD Display 20X4	Rp. 145.000	1	Rp. 145.000	Dana Pribadi
5.	Analog Ambient Light Sensor	Rp. 150.000	1	Rp. 150.000	Dana Pribadi
6.	Box Sistem	Rp. 100.000	1	Rp. 100.000	Dana Pribadi
7.	PCB	Rp. 45.000	1	Rp. 45.000	Dana Pribadi
8.	JST	Rp. 4.000	5	Rp. 20.000	Dana Pribadi
9.	Kabel Jumper	Rp. 1.000	30	Rp. 30.000	Dana Pribadi
10.	Alumunium Solder Extractor	Rp. 50.000	1	Rp. 50.000	Dana Pribadi
11.	Pin Header	Rp. 2.500	2	Rp. 5.000	Dana Pribadi
14.	Solder	RP. 150.000	1	Rp. 150.000	Dana Pribadi
15.	Power Supply	Rp. 85.000	1	Rp. 85.000	Dana Pribadi
Jumlah				Rp. 1.200.000	

Bab 4. Hasil dan Pembahasan

4.1. Hasil Pengujian Unit

4.1.1. Hasil Pengujian Adaptor

Pengujian adaptor dilakukan sebanyak 5 kali percobaan untuk mengukur output tegangan yang dihasilkan menggunakan multimeter, dengan tujuan untuk memastikan apakah adaptor berfungsi sesuai dengan yang diharapkan. Adaptor memiliki peran yang sangat penting dalam menyuplai tegangan pada semua komponen elektronik yang memerlukannya. Hasil pengujian ini akan memberikan gambaran mengenai kinerja adaptor dalam menyediakan tegangan yang stabil dan sesuai standar yang dibutuhkan.

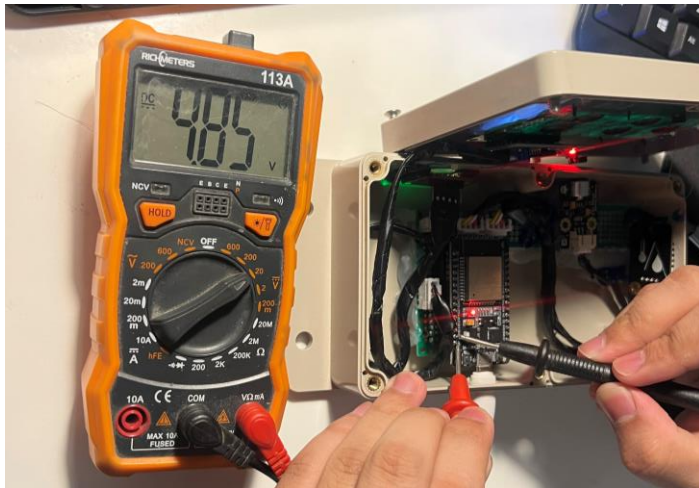
Adapun hasil pengujian adaptor dalam bentuk keluaran tegangan dan nilai *error* terdapat pada tabel 4.1.

Tabel 4. 1 Hasil Pengujian Adaptor

Pengujian ke-	Tegangan Ukur (VDC)	Tegangan Teori (VDC)	Error %
1	4.85 V	5 V	3.00%
2	4.78 V	5 V	4.40%
3	4.89 V	5 V	2.20%
4	4.85 V	5 V	3.00%
5	4.88 V	5 V	2.40%
Rata-Rata			3.00%

Dari tabel 4.1 hasil pengujian adaptor maka disimpulkan bahwa hasil tegangan output berkisar antara 4.78V hingga 4.89V sedangkan nilai teori adalah 5V. Persentase error tertinggi adalah 4.40% dan terendah 2.2%, dengan rata-rata error sebesar 3.00% .Sehingga nilai tersebut masih berada dalam batas toleransi umum untuk catudaya, sehingga adaptor dapat dikatakan berfungsi dengan baik dan layak digunakan untuk mensupli tegangan ke rangkaian elektronik.

Gambar 4.1 menunjukkan proses pengujian adaptor dengan menggunakan multimeter untuk mengukur output tegangan.



Gambar 4. 1 Pengujian Adaptor

4.1.2. Hasil Pengujian ESP32

Pengujian ESP32 dilakukan untuk memastikan apakah modul ESP32 berhasil terhubung ke jaringan WiFi. Proses ini dimulai dengan mengisi nama SSID dan password dari jaringan WiFi yang digunakan. Setelah itu, dilakukan pemeriksaan melalui serial monitor yang akan menampilkan status "connecting" dan informasi penting lainnya, seperti SSID, Gateway IP, Subnet Mask, RSSI, dan ESP32 IP. Jika informasi tersebut berhasil ditampilkan, maka dapat dipastikan bahwa ESP32 sudah terhubung ke jaringan WiFi. Berikut ini adalah potongan program yang digunakan untuk melakukan pengujian tersebut:

```
include <WiFi.h>

const char* ssid = "Test 1";
const char* password = "123456789";

void get_network_info(){
    if(WiFi.status() == WL_CONNECTED) {
```

```

Serial.print("[*] Network information for ");
Serial.println(ssid);

Serial.println("[+] BSSID : " + WiFi.BSSIDstr());
Serial.print("[+] Gateway IP : ");
Serial.println(WiFi.gatewayIP());
Serial.print("[+] Subnet Mask : ");
Serial.println(WiFi.subnetMask());
Serial.println((String)"[+] RSSI : " + WiFi.RSSI() + " dB");
Serial.print("[+] ESP32 IP : ");
Serial.println(WiFi.localIP());
}
}

void setup(){
  Serial.begin(115200);
  delay(1000);

  WiFi.begin(ssid, password);
  Serial.println("\nConnecting");

  while(WiFi.status() != WL_CONNECTED){
    Serial.print(".");
    delay(100);
  }
}

```

Adapun tabel hasil pengujian ESP32 terdapat pada tabel 4.2.

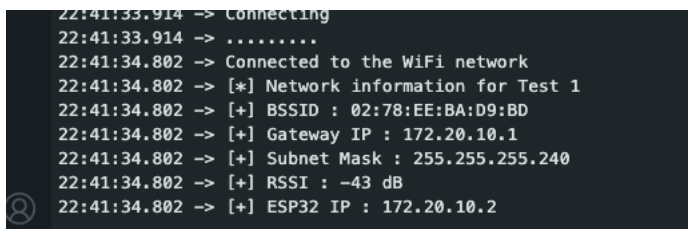
Tabel 4. 2 Hasil Pengujian ESP32

Pengujian ke-	Nama WiFi	Terkoneksi	
		Ya	Tidak
1	Test 1	✓	
2	Test 2	✓	
3	Test 3	✓	

4	Test 4	✓	
5	Test 5	✓	

Dari tabel 4.2 dapat dilihat bahwa ESP32 dapat terhubung ke 5 jaringan WiFi (Test 1, Test 2, Test 3, Test 4, dan Test 5). Hal ini menunjukkan bahwa ESP32 memiliki kemampuan yang baik dalam melakukan koneksi ke jaringan WiFi. Keberhasilan koneksi ESP32 ditampilkan melalui Serial Monitor pada Arduino IDE. Ketika berhasil terkoneksi, ESP32 akan mencetak informasi seperti alamat IP, BSSID, gateway, dan kekuatan sinyal (RSSI).

Gambar 4.2 menunjukkan hasil pengujian ESP32 yang tampil pada serial monitor Arduino IDE.



```

22:41:33.914 -> Connecting
22:41:33.914 -> .....
22:41:34.802 -> Connected to the WiFi network
22:41:34.802 -> [*] Network information for Test 1
22:41:34.802 -> [+] BSSID : 02:78:EE:BA:D9:BD
22:41:34.802 -> [+] Gateway IP : 172.20.10.1
22:41:34.802 -> [+] Subnet Mask : 255.255.255.240
22:41:34.802 -> [+] RSSI : -43 dB
22:41:34.802 -> [+] ESP32 IP : 172.20.10.2

```

Gambar 4. 2 Hasil Pengujian ESP32

4.1.3. Hasil Pengujian LCD Display

Pengujian LCD 20x4 sangat penting untuk menentukan apakah ESP32 dapat mengaturnya secara efektif untuk tampilan karakter yang akurat. Sebelum melakukan pengujian, koneksi antara rangkaian LCD dan ESP32 harus dibuat. Pengujian ini melibatkan menampilkan karakter pada LCD seperti yang ditentukan oleh kode yang dibuat dalam Arduino IDE. Di bawah ini adalah segmen kode yang digunakan untuk proses pengujian ini.

```

#include <LiquidCrystal_I2C.h>

// set the LCD number of columns and rows
int lcdColumns = 20;
int lcdRows = 4;

// set LCD address, number of columns and rows
// if you don't know your display address, run an I2C scanner sketch

```

```
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);
```

```
void setup() {  
  // initialize LCD  
  lcd.init();  
  // turn on LCD backlight  
  lcd.backlight();  
}  
  
void loop() {  
  // set cursor to first column, first row  
  lcd.setCursor(0, 0);  
  // print message  
  lcd.print("  Nur Khairi");  
  lcd.setCursor(0, 1);  
  lcd.print("  4242011013");  
  delay(2000);  
  lcd.clear();  
  
  lcd.setCursor(0, 2);  
  // print message  
  lcd.print("  Nur Khairi");  
  lcd.setCursor(0, 3);  
  lcd.print("  4242011013");  
  delay(2000);  
  lcd.clear();  
}
```

Adapun tabel hasil pengujian LCD Display dalam menampilkan karakter terdapat pada tabel 4.3.

Tabel 4. 3 Hasil Pengujian LCD Display

Pengujian ke-	Perintah Tampilan	Tampilan LCD	Keterangan
---------------	-------------------	--------------	------------

1	Nur Khairi 4242011013	Nur Khairi 4242011013	✓
2	Nur Khairi 4242011013	Nur Khairi 4242011013	✓
3	Nur Khairi 4242011013	Nur Khairi 4242011013	✓
4	Nur Khairi 4242011013	Nur Khairi 4242011013	✓
5	Nur Khairi 4242011013	Nur Khairi 4242011013	✓

Dari tabel 4.3 dapat dilihat bahwa setiap pengujian menunjukkan bahwa karakter berhasil ditampilkan dengan benar dan konsisten pada LCD. Hal ini ditunjukkan dengan tanda centang (✓) pada kolom keterangan di setiap pengujian. Dengan demikian, dapat disimpulkan bahwa LCD berfungsi dengan baik dan mampu menampilkan karakter sesuai dengan perintah yang diberikan.

Gambar 4.3 menunjukkan hasil pengujian LCD Display dengan menunjukkan karakter yang sesuai dengan yang diperintah



Gambar 4. 3 Pengujian LCD Display

4.1.4. Hasil Pengujian DHT22

Pengujian sensor DHT22 melibatkan perbandingan pembacaan yang diperoleh dari sensor DHT22 dengan pembacaan dari termometer digital. Proses ini mencakup pengukuran suhu dan kelembapan di tiga lingkungan yang berbeda: panas, normal, dan dingin, dengan setiap skenario diuji sebanyak lima kali. Untuk setiap uji coba, nilai kesalahan dihitung untuk menilai perbedaan antara pembacaan sensor DHT22 dan hasil termometer digital. Di bawah ini adalah sebagian kode yang digunakan untuk pengujian ini:

```
#include "DHTesp.h"
DHTesp dht;

void setup()
{
    Serial.begin(115200);
    Serial.println();
    Serial.println("Status\tHumidity (%)\tTemperature (C)\t(F)\tHeatIndex (C)\t(F)");
    String thisBoard= ARDUINO_BOARD;
    Serial.println(thisBoard);

    dht.setup(27, DHTesp::DHT22); // Connect DHT sensor to GPIO 17
}

void loop()
{
    delay(dht getMinimumSamplingPeriod());

    float humidity = dht getHumidity();
    float temperature = dht getTemperature();
    Serial.print("Suhu : ");
    Serial.print(temperature, 1);
    Serial.println(" °C");
    Serial.print("Kelembaban : ");
```

```

Serial print(humidity, 1);
Serial println(" %");
delay(500);
}

```

Adapun tabel hasil pengujian sensor DHT22 dan nilai *error* terdapat pada tabel 4.4.

Tabel 4. 4 Hasil Pengujian Sensor DHT22

Pengujian Kondisi	DHT 22		Thermometer Digital		Errpor (%)	
	Suhu (°C)	Kelembaban (%)	Suhu (°C)	Kelembaban (%)	Suhu	Kelembaban
Dingin	12.4	62.7	10.0	64.0	24.00%	2.02%
	12.6	62.2	12.0	61.0	5.00%	1.97%
	12.2	62.8	12.0	61.0	10.91%	2.95%
	13.0	63.4	12.0	62.0	8.33%	2.26%
	11.8	64.1	12.0	62.0	1.67%	3.29%
	Rata-Rata				9.98%	2.50%
Normal	27.6	38.5	27.0	40.0	2.22%	3.75%
	27.6	40.0	27.0	40.0	2.22%	1.00%
	28.2	40.2	28.0	40.0	0.71%	0.50%
	27.7	39.4	27.0	41.0	2.59%	3.90%
	26.4	40.0	27.0	41.0	2.22%	2.44%
	Rata Rata				1.99%	2.32%
Panas	32.3	76.3	32.0	74.0	0.94%	3.11%

	32.8	76.4	32.0	75.0	2.50%	1.87%
	33.0	77.3	33.0	75.4	0.00%	2.12%
	32.6	76.8	32.0	76.0	1.21%	1.05%
	32.8	76.4	32.0	75.0	2.50%	1.87%
	Rata-Rata				1.43%	2.01%
Total Rata-Rata					4.47%	2.28%

Dari tabel 4.4 dapat dilihat bahwa Sensor DHT22 menunjukkan kinerja yang relatif baik dalam mengukur suhu dan kelembaban, dengan error yang lebih kecil pada kelembaban (berkisar antara 1.72% hingga 3.11%) dibandingkan suhu (1.1% hingga 3.75%). Meskipun ada variasi pada error yang tercatat, sensor ini tetap memberikan hasil yang akurat, terutama dalam pengukuran kelembaban, yang memiliki konsistensi error lebih rendah, berkisar antara 0.50% hingga 2.50%. Di sisi lain, suhu cenderung menunjukkan variasi error yang lebih besar, dengan kisaran error antara 0.02% hingga 2.95% pada kondisi dingin dan 1.87% hingga 3.75% pada kondisi panas. Secara keseluruhan, meskipun ada sedikit perbedaan pada setiap pengujian, DHT22 memberikan hasil yang memadai dan dapat digunakan untuk aplikasi pengukuran suhu dan kelembaban dengan akurasi yang dapat diterima, baik dalam kondisi normal, dingin, maupun panas.

Gambar 4.4 menunjukkan hasil pengujian DHT 22 dengan menunjukan hasil pembacaan suhu dan kelembaban pada Thermometer digital dan serial monitor Arduino IDE.



Gambar 4. 4 Pengukuran DHT 22

4.1.5. Hasil Pengujian Analog Sound Sensor

Pengujian Analog Sound Sensor melibatkan evaluasi pembacaan sensor terhadap pembacaan Pengukur Desibel. Evaluasi ini dilakukan dalam dua skenario yang berbeda, khususnya lingkungan yang tenang dan bising, dengan masing-masing skenario dinilai pada lima kesempatan terpisah. Untuk setiap percobaan, margin kesalahan dihitung untuk membandingkan pembacaan dari Sensor Suara Analog dengan pembacaan dari Pengukur Desibel. Di bawah ini adalah bagian dari kode yang digunakan untuk evaluasi ini:

```
#include <WiFi.h>
#include <Wire.h>
#define SOUND_SENSOR_PIN 35
float soundMeter;
void setup() {
  Serial.begin(115200); // open serial port, set the baud rate to 9600 bps
}
void getDataSensor() {
  float voltageValue = analogRead(SOUND_SENSOR_PIN)/4096.0 * 3.3;
  soundMeter = voltageValue * 50.0;
  delay(250);
}
```

```

void loop() {
  getDataSensor();
  Serial.print("Sound Sensor : ");
  Serial.print(soundMeter, 1);
  Serial.println(" ");
  delay(100);
}

```

Adapun tabel hasil pengujian Analog Sound Sensor dan *error* terdapat pada tabel 4.5

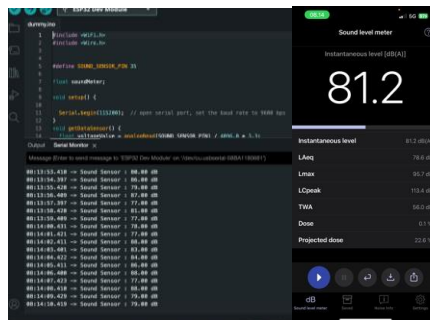
Tabel 4. 5 Hasil Pengujian Analog Sound Sensor

Pengujian Kondisi	Analog Sound Sensor (dB)	Sound Level Meter (dB)	Error (%)
Normal	48.00	44.0	9.09 %
	46.00	49.5	7.09 %
	49.00	46.6	5.16 %
	45.00	42.8	5.14 %
	49.00	42.7	14.74 %
	Rata-Rata		8.24 %
Bising	78.0	81.2	3.95 %
	77.0	81.5	5.52 %
	88.0	86.2	2.09 %
	83.0	82.9	0.12 %
	84.0	73.2	14.75 %
	Rata-Rata		5.29 %

Total Rata-Rata	6.77%
------------------------	--------------

Dari tabel 4.5 dapat dilihat bahwa Analog Sound Sensor menunjukkan kinerja yang bervariasi antara kondisi normal dan bising, dengan error pada kondisi normal yang berkisar antara 5.14% hingga 14.74% dan rata-rata error sebesar 8.24%, yang menunjukkan ketidaksesuaian signifikan antara pengukuran sensor suara analog dan sound level meter, sementara pada kondisi bising, error tercatat lebih rendah dan lebih konsisten, berkisar antara 0.12% hingga 5.52% dengan rata-rata error sebesar 5.29%, yang menunjukkan bahwa sensor suara analog memberikan hasil yang lebih akurat dan dapat diandalkan dalam pengukuran suara di lingkungan bising dibandingkan dengan kondisi normal.

Gambar 4.5 menunjukkan hasil pengujian Analog Sound Sensor dengan menunjukkan hasil pembacaan pada Sound Level Meter dan serial monitor Arduino IDE.



Gambar 4. 5 Analog Sound Sensor

4.1.6. Hasil Pengujian Analog Ambient Light Sensor

Pengujian Analog Ambient Light Sensor melibatkan penilaian pembacaannya terhadap pembacaan yang diperoleh dari Lux Meter. Penilaian ini dilakukan dalam tiga skenario yang berbeda: terang, biasa, dan redup, di mana setiap skenario dianalisis untuk menentukan margin kesalahan dan menyandingkan temuan dari Sensor Cahaya Sekitar Analog dengan pengukuran Lux Meter. Di bawah ini adalah kutipan dari program yang digunakan untuk evaluasi:

```
#include <Wire>
#include "Adafruit_VCNL4020.h"
Adafruit_VCNL4020 vcnl4020;
```

```

void setup() {
  Serial.begin(115200);
  while (!Serial) delay(10); // wait for serial port to start.
  Serial.println("Adafruit VCNL4020 Library Test Sketch");
  // Initialize sensor
  if (!vcnl4020.begin(&Wire)) {
    Serial.println("Failed to initialize VCNL4020!");
    while (1);
  }
  Serial.println("VCNL4020 initialized.");
  Serial.print("VCNL4020 Product ID & Revision (should be 0x21): ");
  Serial.println(vcnl4020.getProdRevision(), HEX);
  // To set up all the configuration, first disable everything
  vcnl4020.enable(false /* ALS Enable */, false /* Proximity Enable */, false
  vcnl4020.setOnDemand(false /* ALS on demand read */, false /* Prox on demand
read */);
  vcnl4020.setAmbientRate(AMBIENT_RATE_10_SPS);
  Serial.print("Ambient Rate: ");
void loop() {
  if (vcnl4020.isAmbientReady()) {
    Serial.print("Ambient: ");
    Serial.print(vcnl4020.readAmbient());
    Serial.println(" Lux");
    vcnl4020.clearInterrupts( // Clear Interrupts
  );
  }
}
}

```

Adapun tabel hasil pengujian Ambient Light Sensor dan nilai *error* terdapat pada tabel 4.6.

Tabel 4. 6 Hasil Pengujian Ambient Light Sensor

Pengujian Kondisi	Analog Ambient Sensor (lx)	Lux Meter (lx)	Error(%)
Terang	312	302	3.31%
	309	298	3.69%
	351	342	2.63%
	346	337	2.67%
	332	328	1.22%
	Rata-Rata		2.70%
Normal	156	153	1.96%
	163	161	1.24%
	161	164	1.83%
	155	154	0.65%
	151	157	3.82%
	Rata-Rata		1.90%
Total Rata-Rata			2.3%

Dari tabel 4.6 dapat dilihat bahwa Ambient Light Sensor menunjukkan kinerja terbaik dalam kondisi terang dengan error yang lebih kecil dan

konsisten, berkisar antara 1.22% hingga 3.69% dengan rata-rata error sebesar 2.70%, sementara pada kondisi normal, error tercatat lebih kecil lagi, antara 0.65% hingga 3.82%, dengan rata-rata error 1.90%, yang menunjukkan akurasi yang lebih baik dibandingkan dengan kondisi terang, sedangkan pada kondisi gelap, sensor ini menunjukkan error yang lebih besar dan kurang konsisten, dengan rentang antara 4.35% hingga 12.00%, serta rata-rata error sebesar 7.21%, yang mengindikasikan bahwa sensor cahaya analog kurang akurat dalam mengukur cahaya pada kondisi dengan intensitas cahaya rendah atau gelap.

Gambar 4.6 menunjukkan hasil pengujian Ambient Light Sensor dengan menunjukan hasil pembacaan pada Lux Meter dan serial monitor Arduino IDE.



Gambar 4. 6 Pengujian Ambient Light Sensor

4.2. Pengujian Sistem

Untuk menentukan kinerja alat secara keseluruhan, seluruh sistem perlu diuji secara menyeluruh. Respons semua komponen terhadap tindakan masukan dan keluaran setiap alat diperiksa selama pengujian ini. Alat akan dipasang pada lantai 4 gedung apartemen Zahradni selama 1 hari, dan data yang masuk ke ThingsBoard akan dianalisis untuk memastikan apakah fungsi alat sudah sesuai dengan yang diinginkan atau belum. Di bawah ini adalah sebagian kode yang digunakan untuk pengujian ini:

```

#include "DHTesp.h" // Click here to get the library:
http://librarymanager/All#DHTesp
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include "Adafruit_VCNL4020.h"
#include <WiFi.h>
#include <ThingsBoard.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"

WiFiClient wifiClient;
ThingsBoard tb(wifiClient);
String username = "test 1";
String password = "123456789";
constexpr char TOKEN[] = "hh18ai2ilRCaCzrYL3IY";
constexpr char THINGSBOARD_SERVER[] = "demo.thingsboard.io";
constexpr uint16_t THINGSBOARD_PORT = 1883U;
bool isNetworkAvailable = false;
bool isDeviceConnectedToServer = false;
unsigned long previousTime1 = 0;
unsigned long previousTime2 = 0;
unsigned long intervalData1 = 60000;
unsigned long intervalData2 = 10;

unsigned long previousSendData = 0;
unsigned long intervalSendData = 10;

void tbReconnect() {
  if (!tb.connected()) {
    Serial.println("Connecting to Thingsboard ... ");
    if (tb.connect(THINGSBOARD_SERVER, TOKEN, THINGSBOARD_PORT)) {
      isDeviceConnectedToServer = true;
      Serial.println("Thingsboard connected");
    } else {

```

```

        isDeviceConnectedToServer = false;
        Serial.println("Thingsboard connection failed");
    }
}

Adafruit_VCNL4020 vcnl4020;
int lcdColumns = 20;
int lcdRows = 4;
DHTesp dht;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);
void setup() {
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
    pinMode(2, OUTPUT);
    WiFi.begin(username_c_str(), password_c_str());

    Serial.begin(115200);
    Serial.println();
    Serial.println("Status\tHumidity (%)\tTemperature (C)\t(F)\tHeatIndex
(C)\t(F)");
    String thisBoard = ARDUINO_BOARD;
    Serial.println(thisBoard);
    dht.setup(27, DHTesp::DHT22); // Connect DHT sensor to GPIO 17
    lcd.init();

    lcd.backlight();
    Serial.println("Adafruit VCNL4020 Library Test Sketch");
    if (!vcnl4020.begin(&Wire)) {
        Serial.println("Failed to initialize VCNL4020!");
        while (1)
            ;
    }
    Serial.println("VCNL4020 initialized.");
}

```

```

// Verify product ID & revision
Serial.print("VCNL4020 Product ID & Revision (should be 0x21: ");
Serial.println(vcnl4020.getProdRevision(), HEX);

// To set up all the configuration, first disable everything
vcnl4020.enable(false /* ALS Enable */, false /* Proximity Enable */, false
/* Self-Timed Enable */);
vcnl4020.setOnDemand(false /* ALS on demand read */, false /* Prox on
demand read */);
vcnl4020.setAmbientRate(AMBIENT_RATE_10_SPS);
Serial.print("Ambient Rate: ");
switch (vcnl4020.getAmbientRate()) {
    case AMBIENT_RATE_1_SPS: Serial.println("1 samples/s"); break;
    case AMBIENT_RATE_2_SPS: Serial.println("2 samples/s"); break;
    case AMBIENT_RATE_3_SPS: Serial.println("3 samples/s"); break;
    case AMBIENT_RATE_4_SPS: Serial.println("4 samples/s"); break;
    case AMBIENT_RATE_5_SPS: Serial.println("5 samples/s"); break;
    case AMBIENT_RATE_6_SPS: Serial.println("6 samples/s"); break;
    case AMBIENT_RATE_8_SPS: Serial.println("8 samples/s"); break;
    case AMBIENT_RATE_10_SPS: Serial.println("10 samples/s"); break;
}

vcnl4020.setAmbientAveraging(AVG_1_SAMPLES);
Serial.print("Ambient Averaging: ");
switch (vcnl4020.getAmbientAveraging()) {
    case AVG_1_SAMPLES: Serial.println("1 sample"); break;
    case AVG_2_SAMPLES: Serial.println("2 samples"); break;
    case AVG_4_SAMPLES: Serial.println("4 samples"); break;
    case AVG_8_SAMPLES: Serial.println("8 samples"); break;
    case AVG_16_SAMPLES: Serial.println("16 samples"); break;
    case AVG_32_SAMPLES: Serial.println("32 samples"); break;
    case AVG_64_SAMPLES: Serial.println("64 samples"); break;
    case AVG_128_SAMPLES: Serial.println("128 samples"); break;
}

```

```

// Setup IRQ pin output when proximity data is ready
vcnl4020.setInterruptConfig(
    true /* Proximity Ready */,
    false /* ALS Ready */,
    false /* Threshold */,
    false /* true = Threshold ALS, false = Threshold Proximity */,
    INT_COUNT_1 /* how many values before the INT fires */
);
vcnl4020.enable(true /* ALS Enable */, true /* Proximity Enable */, true
/* Self-Timed Enable */);
// dont use on-demand, we will have continuous reads.
vcnl4020.setOnDemand(false /* ALS on demand read */, false /* Prox on
demand read */);
}
void loop() {
    delay(dht.getMinimumSamplingPeriod());
    float humidity = dht.getHumidity();
    float temperature = dht.getTemperature();
    float lux = vcnl4020.readAmbient();
    float val = analogRead(35);
    Serial.print("Suhu : ");
    Serial.print(temperature, 1);
    Serial.println(" °C");

    Serial.print("Kelembaban : ");
    Serial.print(humidity, 1);
    Serial.println(" %");

    if (vcnl4020.isAmbientReady()) {
        Serial.print("Ambient: ");
        Serial.print(vcnl4020.readAmbient());
        Serial.println(" Lux");
        vcnl4020.clearInterrupts( // Clear Interrupts

```

```

    false /* Proximity Ready */
    true /* ALS Ready */
    false /* Threshold Low */
    false /* Threshold High */
  );
}

//val = analogRead(35); //connect mic sensor to Analog 0
Serial.println(val, DEC); //print the sound value to serial

lcd.setCursor(0, 0);
lcd.print("Sound Meter:");
lcd.setCursor(13, 0);
lcd.print(val, DEC);
lcd.setCursor(16, 0);
lcd.print("dB");
lcd.setCursor(0, 1);
lcd.print("Suhu:");
lcd.setCursor(5, 1);
lcd.print(temperature, 1);
lcd.setCursor(9, 1);
lcd.print(" C");
lcd.setCursor(0, 2);
lcd.print("Kelembaban:");
lcd.setCursor(11, 2);
lcd.print(humidity, 1);
lcd.setCursor(15, 2);
lcd.print("%");

lcd.setCursor(0, 3);
lcd.print("Lux Meter:");
lcd.setCursor(10, 3);
lcd.print(vcnl4020.readAmbient(), 1);
lcd.setCursor(13, 3);
lcd.print("lx");

```

```

delay(2000);
lcd clear();

unsigned long startTime = millis();
if (startTime - previousTime1 >= intervalData1) {
    previousTime1 = startTime;
}

if (startTime - previousTime2 >= intervalData2 * 60000) {
    // randomTurbidity = random(450, 650);
    // DO = randomDO / 10.0;

    previousTime2 = startTime;
}

if (WiFi.status() == WL_CONNECTED) {
    isNetworkAvailable = true;
    digitalWrite(2, HIGH);
}
if (WiFi.status() != WL_CONNECTED) {
    isNetworkAvailable = false;
    digitalWrite(2, HIGH);
    delay(500);
    digitalWrite(2, LOW);
    delay(500);
}

if (!tb.connected()) {
    tb.reconnect();
}
tb.loop();
Serial.println();

```

```

delay(10);
if (startTime - previousSendData >= intervalSendData * 1000) {
    if (isNetworkAvailable && isDeviceConnectedToServer) {
        Serial.println("Sending data to Thingsboard");
        tb.sendTelemetryString("Temperature", String(temperature, 1).c_str());
        tb.sendTelemetryString("Humidity", String(humidity, 1).c_str());
        tb.sendTelemetryString("Lux Meter", String(lux, 1).c_str());
        tb.sendTelemetryString("Sound Meter", String(val, 1).c_str());
    }
    previousSendData = startTime;
}
}

```

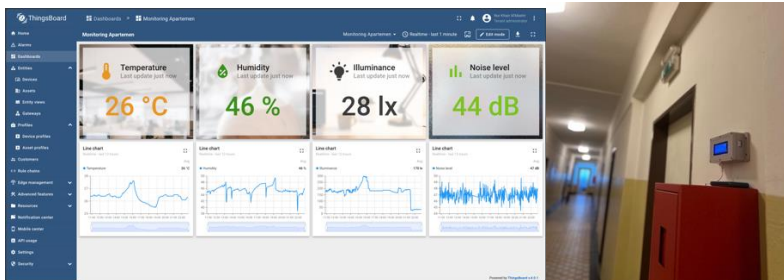
Adapun tabel hasil pengujian sistem terdapat pada tabel 4.7.

Tabel 4. 7 Hasil Pengujian Sistem

Pengujian jam ke-	Suhu (°C)	Kelembaban (%)	Cahaya (lx)	Kebisingan (dB)	LCD Display	Thingsboard	Keterangan
1	27	44	179	47	✓	✓	Baik
2	26	45	176	48	✓	✓	Baik
3	26	46	179	48	✓	✓	Baik
4	27	47	176	48	✓	✓	Baik
5	26	45	189	46	✓	✓	Baik
6	27	45	260	47	✓	✓	Baik
7	28	45	180	49	✓	✓	Baik
8	26	47	182	48	✓	✓	Baik
9	26	48	181	48	✓	✓	Baik
10	26	49	187	46	✓	✓	Baik
11	27	45	29	45	✓	✓	Baik
12	27	45	29	48	✓	✓	Baik

Dari tabel 4.6 dapat dilihat bahwa sistem dapat memberikan kinerja yang baik dengan memberikan pembacaan yang konsisten dan akurat terhadap intensitas cahaya (lux), suhu ($^{\circ}\text{C}$), kelembaban (%), serta tingkat kebisingan (dB) di setiap pengujian, di mana nilai-nilai tersebut tetap stabil dengan fluktuasi yang minim; tampilan LCD bekerja dengan baik dalam menampilkan hasil yang sesuai, dan evaluasi keseluruhan terhadap sistem, termasuk keandalan sensor dan komponen lainnya, menunjukkan kondisi "Baik" pada semua percobaan yang dilakukan. Selain itu, sistem ini terhubung dengan Thingsboard, yang memungkinkan pemantauan secara real-time dan visualisasi data sensor secara efektif. Melalui platform ini, data yang diperoleh dapat diakses dan dianalisis dengan mudah, memberikan kemudahan dalam pengawasan performa sensor dan mendeteksi potensi masalah atau penyimpangan lebih awal. Evaluasi pada Thingsboard menunjukkan bahwa data yang diterima dari sensor dapat ditampilkan dengan jelas dan responsive.

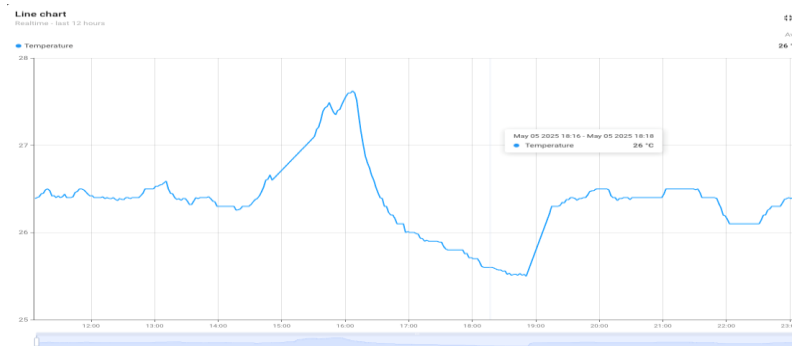
Gambar 4.7 menunjukkan hasil pengujian sistem dengan menunjukkan hasil pembacaan sensor pada Thingsboard dan penempatan alat untuk proses pengujian.



Gambar 4. 7 Pengujian Sistem

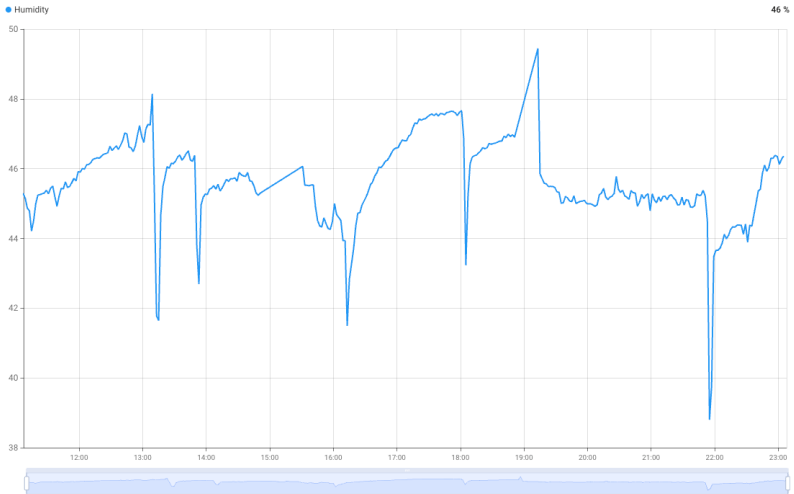
Gambar 4.8 menampilkan Riwayat pembacaan sensor suhu pada dashboard Thingsboard yang menunjukkan fluktuasi suhu selama 12 jam, di mana terlihat adanya kenaikan suhu tajam sekitar pukul 14:00, diikuti penurunan signifikan pada pukul 18:00, dan stabilisasi suhu yang lebih rata pada malam hari. Rata-rata suhu tercatat sekitar 26°C , yang menunjukkan kinerja sensor dalam mencatat perubahan suhu yang cukup stabil meskipun ada fluktuasi besar pada beberapa titik. Sistem sensor suhu berfungsi dengan baik, mampu mencatat perubahan

secara real-time dan responsif terhadap perubahan lingkungan, namun fluktuasi tajam yang terjadi memerlukan perhatian lebih lanjut untuk memastikan apakah itu disebabkan oleh faktor lingkungan atau masalah sensor. Dengan pengujian yang berlangsung selama 12 jam, sistem ini menunjukkan kemampuannya dalam memantau suhu secara efektif dalam rentang waktu panjang dan memungkinkan analisis yang akurat.



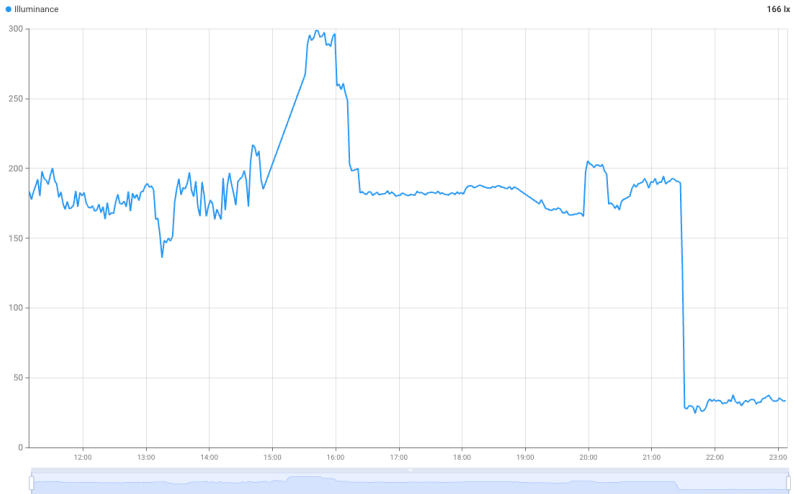
Gambar 4. 8 Riwayat Pembacaan Suhu

Gambar 4.9 menampilkan Riwayat pembacaan sensor kelembaban pada dashboard Thingsboard menunjukkan fluktuasi yang signifikan selama 12 jam, dengan nilai kelembaban yang bergerak antara 38% hingga 48%. Ada beberapa perubahan tajam yang terlihat, seperti penurunan drastis sekitar pukul 14:00 dan lonjakan mendadak pada pukul 18:00 serta 22:00, yang menunjukkan adanya perubahan cepat dalam kondisi lingkungan. Secara umum, kelembaban tercatat stabil pada kisaran 46%, namun ada fluktuasi tajam yang mungkin dipengaruhi oleh perubahan lingkungan yang mendadak, seperti perubahan suhu atau faktor eksternal lainnya. Meskipun begitu, sistem sensor kelembaban bekerja dengan baik dalam mendeteksi perubahan kelembaban secara real-time, memberikan data yang responsif terhadap perubahan tersebut. Fluktuasi besar yang terjadi pada beberapa titik, terutama pada pukul 14:00 dan 22:00, perlu diperhatikan lebih lanjut untuk memastikan bahwa itu bukan disebabkan oleh masalah teknis atau gangguan lain pada sensor. Secara keseluruhan, sistem ini terbukti efektif dalam memantau kelembaban lingkungan sepanjang hari.



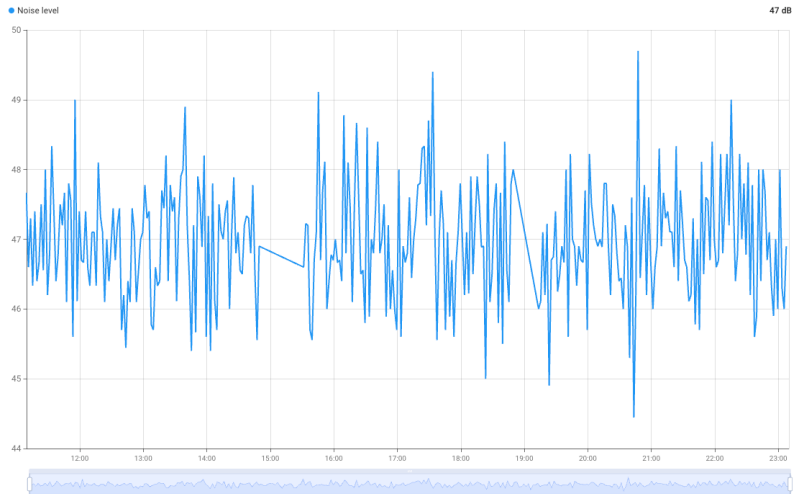
Gambar 4. 9 Riwayat Pembacaan Kelembaban

Gambar 4.10 menampilkan Riwayat pembacaan sensor cahaya pada dashboard Thingsboard menunjukkan fluktuasi signifikan dalam periode 12 jam, dengan lonjakan tajam mencapai hampir 300 lux sekitar pukul 14:00, yang kemungkinan disebabkan oleh peningkatan intensitas cahaya, seperti perubahan posisi matahari atau peningkatan sumber cahaya di sekitar sensor. Setelah itu, intensitas cahaya menunjukkan penurunan bertahap, namun pada pukul 20:00-21:00, terjadi penurunan drastis yang menyebabkan nilai iluminansi jatuh ke level rendah. Setelah penurunan tersebut, data kembali stabil pada nilai yang lebih rendah. Rata-rata iluminansi tercatat sekitar 166 lux, menunjukkan bahwa meskipun terjadi fluktuasi besar, pencahayaan rata-rata tetap dalam rentang moderat. Fluktuasi tajam ini mungkin mencerminkan pengaruh perubahan lingkungan atau aktivitas yang mempengaruhi sensor, seperti matahari terbenam atau perubahan pada sumber cahaya eksternal. Secara keseluruhan, sistem sensor iluminansi bekerja dengan baik, mencatat perubahan secara real-time dan memberikan data yang akurat, meskipun adanya penurunan tajam yang perlu diselidiki lebih lanjut untuk memastikan apakah itu disebabkan oleh faktor eksternal atau masalah teknis pada sensor.



Gambar 4. 10 Riwayat Pembacaan Cahaya

Gambar 4.11 menampilkan Riwayat pembacaan sensor kebisingan pada dashboard Thingsboard menunjukkan fluktuasi yang sangat tinggi dan cepat sepanjang 12 jam pengujian, dengan level kebisingan bergerak antara 44 dB hingga 49 dB. Fluktuasi tajam yang terjadi di grafik mengindikasikan bahwa tingkat kebisingan sangat dinamis, mungkin karena faktor lingkungan yang terus berubah, seperti aktivitas manusia atau peralatan yang beroperasi di sekitar sensor. Rata-rata tingkat kebisingan tercatat sekitar 47 dB, menunjukkan bahwa meskipun ada variasi yang signifikan, level kebisingan tetap berada dalam rentang yang moderat. Fluktuasi yang sering terjadi bisa disebabkan oleh sumber kebisingan yang bervariasi, seperti kendaraan, percakapan, atau aktivitas industri yang memengaruhi pengukuran. Sistem sensor kebisingan bekerja dengan baik dalam mendeteksi perubahan suara secara real-time dan memberikan data yang responsif, meskipun fluktuasi yang cukup tajam pada beberapa waktu tertentu perlu dipantau lebih lanjut untuk mengidentifikasi penyebab pasti dari variasi tersebut. Secara keseluruhan, sistem sensor ini efektif dalam memantau tingkat kebisingan sepanjang hari.



Gambar 4. 11 Riwayat Pembacaan Kebisingan

Bab 5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan hasil pengujian sistem monitoring Intensitas cahaya, kebisingan, kelembaban, dan suhu berbasis IoT pada gedung apartemen, Maka, dapat diambil beberapa kesimpulan:

1. Sistem monitoring berbasis IoT berhasil memantau kondisi lingkungan gedung apartemen secara real-time dengan kinerja yang stabil dan baik.
2. Sensor DHT22 menunjukkan error suhu rata-rata 4.47% dan kelembaban 2.28%, sensor cahaya memiliki error rata-rata 2.3%, serta sensor kebisingan menunjukkan error rata-rata 6.77%
3. LCD display menampilkan karakter dengan akurat, sesuai dengan data yang diperoleh dari sensor, dengan hasil pengujian menunjukkan bahwa semua karakter ditampilkan dengan benar.
4. Data sensor berhasil ditransmisikan ke Thingsboard secara akurat, dengan nilai yang stabil dan dapat diakses secara real-time oleh pengelola gedung.
5. Sistem memantau suhu, kelembaban, cahaya, dan kebisingan dengan konsisten, memberikan data yang dapat diandalkan dan membantu pengelola gedung dalam pengawasan lingkungan secara efektif.

5.2. Saran

Adapun saran ataupun pengembangan yang dapat dilakukan dari proyek ini, antara lain:

1. Untuk meningkatkan responsibilitas, sistem ini dapat diintegrasikan dengan perangkat otomatis yang dapat mengambil tindakan langsung berdasarkan data yang diterima, seperti menyesuaikan suhu atau intensitas cahaya di apartemen.
2. Walaupun Thingsboard sudah memberikan visualisasi yang efektif, mungkin akan lebih bermanfaat untuk menambahkan fitur-fitur seperti alarm atau pemberitahuan langsung kepada pengelola apartemen ketika salah satu parameter lingkungan melewati batas yang ditentukan.

Daftar Pustaka

- [1] K. Lintas, I. Bidang, dan R. Arsitektur, “Jurnal KaLIBRASI Perancangan Apartemen dengan Pendekatan Smart Building di Kelapa,” vol. 7, no. 1, hal. 60–68, 2024.
- [2] S. Raslanas, J. Alchimoviene, dan N. Banaitiene, “Residential Areas with Apartment Houses: Analysis of the Condition of Buildings, Planning Issues, Retrofit Strategies and Scenarios,” *Int. J. Strateg. Prop. Manag.*, vol. 15, no. 2, hal. 152–172, 2011, doi: 10.3846/1648715X.2011.586531.
- [3] N. A. Agustina, D. N. Gandarum, K. Lahji, dan A. Pendahuluan, “Kenyanaman Visual Pada Orientasi Bangunan Dan Desain Fasade Apartemen , Jakarta (Studi Kasus : Apartemen Season City , Jakarta Barat) Visual Comfort on Building Orientation and Façade of Apartment Design (Case Study : Season City Apartment , West Jakar,” *Semin. Intelekt. Muda* #2, no. September, hal. 263–267, 2019.
- [4] A. Z. Bagus Ari Prabowo¹, Adharul Muttaqien², “Monitoring Ruangan Berbasis IoT.” hal. 1, 2017.
- [5] S. Sintaro, A. Surahman, dan C. A. Pranata, “Sistem Pengontrol Cahaya Pada Lampu Tubular Daylight Berbasis Iot,” *J. Teknol. dan Sist. Tertanam*, vol. 2, no. 1, hal. 28, 2021, doi: 10.33365/jtst.v2i1.1034.
- [6] A. D. Limantara, Y. C. S. Purnomo, dan S. W. Mudjanarko, “Pemodelan Sistem Pelacakan Lot Parkir Kosong Berbasis Sensor Ultrasonic Dan Internet of Things (Iot) Pada Lahan Parkir Diluar Jalan,” *Semin. Nas. Sains dan Teknol.*, vol. 1, no. 2, hal. 1–10, 2017, [Daring]. Tersedia pada: jurnal.umj.ac.id/index.php/semnastek
- [7] I. W. D. dan D. Jumadi, “J urnal P endidikan F isika T adulako O nline BERBASIS KEARIFAN LOKAL UNTUK SEKOLAH MENENGAH PERTAMA For Junior High School,” vol. 9, no. December, hal. 84–88, 2021.
- [8] A. William dan A. Wagya, “Sistem Monitoring pada Ruang Perpustakaan berbasis Internet of Things,” *TESLA J. Tek. Elektro*, vol. 25, no. 2, hal. 126–138, 2023, doi: 10.24912/tesla.v25i2.26748.
- [9] R. G. Renaldy dan G. A. Susilo, “Apartemen Di Kota Malang Tema : Arsitektur Tropis,” *PENGILON J.*, vol. 5, no. 1, hal. 101–116, 2021.
- [10] I. Aldyanto, H. Izzati, dan A. Andiyani, “Low Rise Apartment Design,” *J. Arsit. Arch.*, 2022.
- [11] V. F. Aura *et al.*, “Jurnal informatika dan perancangan sistem (jips),” vol. 5, no. 1, hal. 37–44, 2023.
- [12] P. BANJIR Oleh, A. Kodir Al Bahar, T. Lanjar Hanafi, L. Aditya, dan A. Rinjani Harahap, “Aplikasai Node Mcu Esp 8266 Dan Sensor Ultrasonic Hc-Sr04 Sebagai,” vol. 12, no. 1, 2024.
- [13] F. Ciasaka, S. D. Panjaitan, dan B. W. Sanjaya, “Perancangan Sistem

- Kendali Supervisi Dan Akuisisi Data (Scada) Pada Panel Surya Berbasis Internet of Things,” *J. Electr. Eng. Energy, Inf. Technol.*, vol. 11, no. 1, 2023.
- [14] M. Luqman Ramadhani, A. Stefanie, dan R. Rahmadewi, “Implementasi Metode Fuzzy Logic Mamdani Pada Alat Pengereng Biji Kakao,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 3, hal. 2088–2093, 2023, doi: 10.36040/jati.v7i3.7132.
- [15] C. F. Hadi, R. M. Yasi, dan A. Prasetyo, “Model Decision Tree Forecasting Berbasis DHT22 pada Smart Hydroponic Microgreen,” *J. Telecommun. Electron. Control Eng.*, vol. 6, no. 1, hal. 29–38, 2024, doi: 10.20895/jtece.v6i1.1218.
- [16] Y. D. Siddik, A. Manaor, H. Pardede, dan H. Kahir, “Rancang Bangun Alat Deteksi Kebisingan Perpustakaan dengan Indikator Peringatan Berbasis Internet Of Things (IOT),” no. 4, 2024.
- [17] P. Kowol, P. Nowak, L. Di Nunzio, G. C. Cardarilli, G. Capizzi, dan G. Lo Sciuto, “Pipe Organ Design Including the Passive Haptic Feedback Technology and Measurement Analysis of Key Displacement, Pressure Force and Sound Organ Pipe,” *Appl. Syst. Innov.*, vol. 7, no. 3, hal. 1–16, 2024, doi: 10.3390/asi7030037.
- [18] F. D. Arachman, F. Antoni, K. K. Sel, G. Spreadsheet, A. Studio, dan P. S. Fotofoltaik, “MENGUNAKAN LOGIKA FUZZY DENGAN SENSOR LUX aman bagi lingkungan . Sel fotovoltai yang dipasang menghadap sisi yang sama tidak akan logika Fuzzy bertipe fuzzy sugeno dengan masukan dua buah nilai sensor TEMT6000 timur dan yang dibuat menggunakan software ,” 2018.
- [19] D. D. Artha Lesmana, I. M. Arsa Suyadnya, dan I. W. Shandyasa, “Rancang Bangun Perangkat Keras Sistem Smart Lampu Penerangan Jalan Umum Berbasis Internet of Things Guna Mendukung Implementasi Smart City,” *J. SPEKTRUM*, vol. 10, no. 3, hal. 21, 2023, doi: 10.24843/spektrum.2023.v10.i03.p3.
- [20] D. S. Estu, M. Yantidewi, B. M. Rusdi, M. B. Adikuasa, dan M. Khoiro, “Alat Monitoring Ketinggian Air Laut Berbasis IOT dengan Nodemcu ESP32 DAN HC-SR04 IOT-Based Sea Water Level Monitoring Tool with Nodemcu ESP32 and HC-SR04,” *Jurna Kolaboratif Sains*, vol. 6, no. 7, hal. 585–597, 2023, [Daring]. Tersedia pada: <https://jurnal.unismuhpalu.ac.id/index.php/JKS>
- [21] S. W. Abied Firmanda, I Nyoman Gede Widiarsa, Maskatim, “Alat Penguji Golongan Darah Berbasis Mikrokontroler Atmega 16 Dengan Tampilan LCD,” vol. 11, no. 0, hal. 1–23, 2024.
- [22] R. R. Hidayah, S. Nurcahyo, dan D. Dewatama, “Implementasi Pengaturan Suhu Menggunakan Mikrokontroler ESP32,” vol. 3, no. 3, hal. 106–115, 2024.

- [23] M. R. Saputra, N. Nofriadi, dan A. K. Syahputra, "Perangkap Hama Otomatis Bibit Sawit Metode Menggunakan Mikrokontroler Dengan Source Power Panel Surya," *JISTech (Journal Islam. Sci. Technol.*, vol. 8, no. 1, hal. 52, 2023, doi: 10.30829/jistech.v8i1.16646.
- [24] S. , N. K. H. Febri Agatha Bunga Rante, "Rancang Bangun Alat Bantu Pendeteksi Suara Tangisan BayiBagi Orang Tua Tunarungu Berbasis Internet of Things," *SIPARAPPE (Sistem Inf. - Pengabd. Masyarakat, Penerapan Penelitian)*, vol. Vol. 3, No, no. 2985–5063, 2024.
- [25] M. F. Imaduddin, D. P. Kartikasari, dan R. A. Siregar, "Implementasi Mekanisme Registrasi Dan Identifikasi Perangkat Node Sensor Di Gateway Berbasis Protokol Lora," vol. 1, no. 1, hal. 1–9, 2020.
- [26] M. Reski Ramadhan, "Rancang Bangun Box Pengering Sepatu Berbasis Mikrokontroler," *Electr. J. Rekayasa dan Teknol. Elektro*, vol. 17, no. 3, hal. 292–297, 2023, doi: 10.23960/elc.v17n3.2544.
- [27] M. Subito, T. Blestania, N. Amin, dan S. Dewi, "RANCANG BANGUN SISTEM MONITORING LEVEL PERMUKAAN AIR BERBASIS INTERNET OF THINGS (IoT)," *Foristek*, vol. 14, no. 1, 2023, doi: 10.54757/fs.v14i1.247.