

**SISTEM KENDALI AIR CONDITIONER BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA
*APOSTOLICO GIANCARLO***

SKRIPSI

Oleh :
BALQIS SYAWLYA
NIM. 200605110089



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

**SISTEM KENDALI AIR CONDITIONER BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA
APOSTOLICO GIANCARLO**

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:
BALQIS SYAWLYA
NIM. 200605110089

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

HALAMAN PERSETUJUAN

SISTEM KENDALI *AIR CONDITIONER* BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA *APOSTOLICO GIANCARLO*

SKRIPSI

Oleh :
BALQIS SYAWLYA
NIM. 200605110089

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 23 Mei 2025

Pembimbing I,



Ajib Hanani, M.T
NIP. 19840731 202321 1 013

Pembimbing II,



Okta Qomaruddin Aziz, M.Kom
NIP. 19911019 201903 1 013

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Ir. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

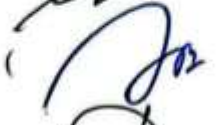
SISTEM KENDALI *AIR CONDITIONER* BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA *APOSTOLICO GIANCARLO*

SKRIPSI

Oleh:
BALQIS SYAWLYA
NIM. 200605110089

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 12 Juni 2025

Susunan Dewan Penguji

Ketua Penguji	: <u>Dr. Cahyo Crysdian, M.CS</u> NIP. 19740424 200901 1 008	()
Anggota Penguji I	: <u>Johan Ericka Wahyu P, M.Kom</u> NIP. 19831213 201903 1 004	()
Anggota Penguji II	: <u>Ajib Hanani, M.T</u> NIP. 19840731 202321 1 013	()
Anggota Penguji III	: <u>Okta Oomaruddin Aziz, M.Kom</u> NIP. 19911019 201903 1 013	()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Iqbal Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Balqis Syawlya

NIM : 200605110089

Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika

Judul Skripsi : Sistem Kendali *Air Conditioner* Berbasis *Internet of Things* Menggunakan Algoritma *Apostolico Giancarlo*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 12 Juni 2025

Yang membuat pernyataan,



Balqis Syawlya

NIM. 200605110089

MOTTO

“Libatkanlah Allah disegala hal, maka segala hal itu akan membuatmu mudah”

“Barangsiapa menempuh jalan untuk mencari ilmu, maka Allah akan memudahkan baginya jalan menuju Surga.”
(HR. Ibnu Majah)

“Yang penting bukan seberapa sering kita jatuh, tetapi seberapa sering kita bangkit setelah jatuh.”
(Vince Lombardi)

“Kunci untuk sukses adalah terus belajar, bertumbuh, dan beradaptasi.”
(Anonym)

“Keberhasilan tidak datang dari kekuatan, tetapi dari ketekunan.”
(Albert Einstein)

HALAMAN PERSEMBAHAN

“Puji Syukur kehadiran Allah Subhanahu wa Ta’ala, sholawat serta salam selalu dipanjatkan kepada baginda Nabiullah wa Rasulullah Muhammad Shallallahu Alaihi Wassalam.”

Dengan bangga dan syukur *Alhamdulillah*, skripsi ini saya persembahkan untuk kedua orang tua saya, ayah dan mama, walaupun ayah telah berbeda alam dengan saya namun dengan jerih payahnya dapat memberikan pendidikan Sarjana kepada saya dan mama saya atas doa – doanya yang beribu doa dipanjatkan untuk saya, yang selalu mendukung saya dalam perkuliahan saya untuk selalu dimudahkan, diberi keberkahan, dan kesuksesan dalam menggapai cita – cita atau impian – impian saya. Semoga Allah *Subhanahu wa Ta’ala* mengampuni kedua orang tua saya, menyayangi, mencintai dan mengasihi kedua orang tua saya sebagaimana kedua orang tua saya mengasihi dan mendidik saya sejak bayi. *Aamiin Allahumma Aamiin.*

Saya juga banyak berterima kasih kepada diri saya sendiri yang telah bersabar, kuat, dan terus berusaha dengan maksimal hingga tidak menyerah sampai saat ini dan seterusnya. Saya juga bangga dan bersyukur atas diri saya sendiri, kesehatan jasmani serta rohani yang Allah berikan kepada saya, hingga saya dapat menyelesaikan skripsi ini. Teruntuk diri saya, kelak bacalah kembali halaman persembahan skripsi yang telah dibuat pada tahun 2025 ini, ingat ya perjalananmu masih sangat panjang dan belum usai, teruslah berusaha secara maksimal. Teruslah berkarya, berinovasi, dan selalu menjadi pribadi yang bermanfaat untuk orang lain.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Puji Syukur *Alhamdulillah* penulis ucapkan kehadiran Allah *Subhanahu wa Ta'ala* yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan program studi Teknik Informatika di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang dengan menyelesaikan penelitian skripsi yang berjudul **“Sistem Kendali Air Conditioner Berbasis *Internet of Things* menggunakan Algoritma *Apostolico Giancarlo*.”**

Pada penulisan dan penelitian skripsi ini, penulis sadar bahwa banyak pihak yang terlibat dalam proses membimbing penulisan dan juga memberikan semangat serta dukungan moral atau materiil. Untuk itu, penulis menyampaikan terima kasih kepada:

1. Prof. Dr. H.M. Zainuddin, MA., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Sri Hariani, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT., IPU., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Ajib Hanani, M.T selaku Dosen Pembimbing I yang telah memberikan banyak dukungan dan bimbingan dalam penulisan skripsi ini.
5. Okta Qomaruddin Aziz, M.Kom selaku Dosen Pembimbing II yang telah

- memberikan banyak dukungan dan bimbingan dalam penulisan skripsi ini.
6. Dr. Cahyo Crysdiان selaku Ketua Penguji yang telah memberikan banyak saran dan juga dukungan untuk menyelesaikan skripsi ini.
 7. Johan Ericka Wahyu P, M.Kom selaku Dosen Penguji II yang telah menguji, memberikan saran dan masukan dalam penulisan skripsi ini.
 8. Seluruh Dosen dan jajaran Staff Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan banyak bantuan dalam skripsi ini.
 9. Seluruh staff admin dan laboran Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah bersedia membantu penulis dalam menyelesaikan permasalahan administrasi selama di perkuliahan.
 10. Kedua orang tua saya, ayah saya Almarhum Lukman Hidayat rahimahullah dan mama saya Erni Anggraini, yang memberikan banyak doa, materi, kasih sayang, dukungan dan semangat kepada penulis.
 11. Saudara saya kakak Ghifari Akbar Putra dan mbak ipar saya Divanita Kusuma yang telah memberikan bantuan kepada saya, baik saran maupun materi dan dukungan semangat kepada saya.
 12. Saudari saya adik Kalyca Chiara, yang telah membantu banyak juga dan sabar mendengarkan curahan hati saya dalam menjalani proses skripsi ini.
 13. Keponakan saya Husein dan Hafsa, yang telah memberikan semangat kembali, menjadikan *moodbooster* saya dikala sedang sedih menghadapi skripsi ini.

14. Seseorang spesial yang telah memberikan saran, semangat dan dukungan penuh kepada saya sejak masa perkuliahan hingga akhir perkuliahan.
15. Sahabat saya Dina Mariya dan Khairun Nisa, yang selalu bersedia dijadikan tempat untuk menghibur saya, *partner travelling* kemanapun termasuk ke alam, ketika saya lagi bersedih mengenai skripsi ini, dan selalu jadi *supporter* di kehidupan saya.
16. Sahabat saya dari balita Zanuar Ajeng Risma Rosita, yang memberikan semangat dan dukungan penuh kepada saya.
17. Teman baik saya sejak SMA, Shinta Abdillah, Alvina Maharani, Kurdaena Nurussafaah, yang selalu gas untuk ke *cafe* agar saya tidak selalu murung di kamar kost guna mengerjakan skripsi ini.
18. Teman baik kuliah saya, Eka Mulyaningrum dan Niken Sesanti, yang banyak membantu saya dalam memperoleh responden skripsi ini dan teman mengerjakan skripsi bersama.
19. Teman – teman Teknik Informatika angkatan 20 dan 21 yang telah memberikan banyak bantuan kepada saya.
20. Semua pihak yang tidak dapat disebutkan satu per satu, termasuk responden yang telah memberikan kontribusi, saran, dan dukungan dalam perjalanan penulisan skripsi ini.

Malang, 23 Mei 2025

Penulis

DAFTAR ISI

HALAMAN PENGANTAR	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
مستخلص البحث	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	6
1.3 Batasan Masalah	7
1.4 Tujuan Penelitian	7
1.5 Manfaat Penelitian	8
BAB II STUDI PUSTAKA	9
2.1 Landasan Teori	11
2.1.1 Algoritma <i>Apostolico Giancarlo</i>	11
2.1.2 <i>String Matching</i>	12
2.1.3 <i>Air Conditioner</i>	12
2.1.4 <i>Internet of Things</i>	13
2.1.5 Modul ESP32	14
2.1.6 Sensor IR Transmitter	14
2.1.7 IR Receiver	15
2.1.8 <i>Speech Recognizer</i>	16
2.1.9 <i>Stemming</i>	16
2.1.10 Aksesibilitas	16
2.2 Perbandingan Penelitian Terkait	18
BAB III DESAIN DAN IMPLEMENTASI SISTEM	21
3.1 Desain dan Implementasi Sistem	21
3.1.1 Proses pengambilan Data	22
3.1.2 Proses <i>Stemming</i>	22
3.1.3 Proses Algoritma <i>Apostolico Giancarlo</i>	23
3.1.4 Algoritma <i>Apostolico Giancarlo</i>	24
3.2 Perancangan Antarmuka (<i>Interface</i>)	27
3.3 Halaman Aplikasi	28
3.3 Rangkaian Komponen	29
3.4 Analisa Aksesibilitas	30
BAB IV UJI COBA DAN PEMBAHASAN	32
4.1 Uji Coba	32

4.1.1	Kendali menggunakan <i>Microphone</i>	32
4.1.2	Rangkaian <i>Hardware</i>	34
4.1.3	Rencana Uji Coba	35
4.2	Pembahasan	36
4.2.1	Pengujian Sistem.....	36
4.2.2	Proses <i>Table Bad Match</i>	36
4.2.3	Pengujian Perintah Suara.....	42
4.2.4	Aksesibilitas	49
4.3	Integrasi Islam.....	58
BAB V PENUTUP		61
5.1	Kesimpulan	61
5.2	Saran.....	62
DAFTAR PUSTAKA		
LAMPIRAN		

DAFTAR GAMBAR

Gambar 2.1 Modul ESP32	14
Gambar 2.2 IR Transmitter	15
Gambar 2.3 IR Receiver	15
Gambar 3.1 Desain Sistem.....	22
Gambar 3.2 <i>Flowchart Stemming</i>	23
Gambar 3.3 <i>Flowchart</i> Proses Algoritma <i>Apostolico Giancarlo</i>	24
Gambar 3.4 Rancangan Aplikasi	27
Gambar 3.5 Tampilan Utama dan Penggunaan <i>Speech Recognition</i>	28
Gambar 3.6 Halaman ketika AC Dihidupkan dan AC Dimatikan	29
Gambar 3.7 Rangkaian Komponen	30
Gambar 4.1 Logika <i>button_mic</i>	32
Gambar 4. 2 <i>Output</i> dari Logika <i>Button Mic</i>	33
Gambar 4. 3 Rangkaian Komponen <i>Hardware</i>	34
Gambar 4.4 Hasil Pertanyaan Pertama	50
Gambar 4.5 Hasil Pertanyaan Kedua	51
Gambar 4.6 Hasil Pertanyaan Ketiga	52
Gambar 4.7 Hasil Pertanyaan Keempat	53
Gambar 4.8 Hasil Pertanyaan Kelima	54
Gambar 4.9 Hasil Pertanyaan Keenam	55
Gambar 4. 10 Hasil Pertanyaan Ketujuh.....	56
Gambar 4. 11 Hasil Pertanyaan Kedelapan	57

DAFTAR TABEL

Tabel 2. 1 Tabel Perbandingan Penelitian Terkait	18
Tabel 3. 1 Hasil <i>pre-processing</i>	26
Tabel 3. 2 Pencocokan Pertama.....	26
Tabel 3. 3 Pencocokan Kedua	26
Tabel 3. 4 Pencocokan Ketiga	26
Tabel 4. 1 <i>Bad Match Pattern</i> Pertama.....	37
Tabel 4. 2 <i>Bad Match Pattern</i> Kedua.....	37
Tabel 4. 3 <i>Bad Match Pattern</i> Ketiga	38
Tabel 4. 4 <i>Bad Match Pattern</i> Keempat	38
Tabel 4. 5 <i>Bad Match Pattern</i> Kelima	39
Tabel 4. 6 <i>Bad Match Pattern</i> Keenam.....	39
Tabel 4. 7 <i>Bad Match Pattern</i> Ketujuh.....	40
Tabel 4. 8 <i>Bad Match Pattern</i> Kedelapan.....	40
Tabel 4. 9 <i>Bad Match Pattern</i> Kesembilan	41
Tabel 4. 10 <i>Bad Match Pattern</i> Kesepuluh	41
Tabel 4. 11 Hasil Pengujian Kalimat pada <i>Pattern</i> Pertama	42
Tabel 4. 12 Hasil Pengujian Kalimat pada <i>Pattern</i> Kedua	42
Tabel 4. 13 Hasil Pengujian Kalimat pada <i>Pattern</i> Ketiga.....	43
Tabel 4. 14 Hasil Pengujian Kalimat pada <i>Pattern</i> Keempat.....	43
Tabel 4. 15 Hasil Pengujian Kalimat pada <i>Pattern</i> Kelima.....	44
Tabel 4. 16 Hasil Pengujian Kalimat pada <i>Pattern</i> Keenam.....	44
Tabel 4. 17 Hasil Pengujian Kalimat pada <i>Pattern</i> Ketujuh	45
Tabel 4. 18 Hasil Pengujian Kalimat pada <i>Pattern</i> Kedelapan	45
Tabel 4. 19 Hasil Pengujian Kalimat pada <i>Pattern</i> Kesembilan.....	46
Tabel 4. 20 Hasil Pengujian Kalimat pada <i>Pattern</i> Kesepuluh	46

ABSTRAK

Syawlya, Balqis. 2025. **Sistem Kendali Air Conditioner Berbasis *Internet of Things* Menggunakan Algoritma *Apostolico Giancarlo***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Ajib Hanani, M.T (II) Okta Qomaruddin Aziz, M.Kom

Kata kunci: *Air Conditioner*, *Apostolico Giancarlo*, *Internet of Things*, Sistem Kendali

Penelitian ini dilakukan untuk mengendalikan *Air Conditioner* berbasis *Internet of Things* menggunakan algoritma *Apostolico Giancarlo*. Pengendalian sistem *Air Conditioner* berbasis *Internet of Things* ini ditujukan kepada orang yang mengalami disabilitas. Penerapan yang dilakukan dengan memasukkan perintah suara melalui aplikasi untuk kendali *Air Conditioner* menggunakan algoritma *Apostolico Giancarlo*. Terdapat 45 responden yang telah diuji coba dengan 8 pertanyaan dan pilihan skala linier dari 1 – 5 dari *Google Form* yang diberikan. Terdapat 10 *pattern* yang dimiliki di dalam algoritma *Apostolico Giancarlo* dengan melakukan 100 perintah suara yang dimasukan. Selanjutnya, menghitung standart *error* dengan hasil akhir 0,08. Kemudian, dilakukan pengujian aksesibilitas dengan tujuan bahwa sistem bisa digunakan oleh penyandang disabilitas.

ABSTRACT

Syawlya, Balqis. 2025. *Air Conditioner Control System Based on Internet of Things Using Apostolico Giancarlo Algorithm*. Thesis. Informatics Engineering Study Program, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Advisor: (I) Ajib Hanani, M.T (II) Okta Qomaruddin Aziz, M.Kom

This study was conducted to control the Internet of Things-based Air Conditioner using the Apostolico Giancarlo algorithm. The control of the Internet of Things-based Air Conditioner system is intended for people with disabilities. The application is carried out by entering voice commands through the application for Air Conditioner control using the Apostolico Giancarlo algorithm. There are 45 respondents who have been tested with 8 questions and a choice of a linear scale from 1 - 5 from the Google Form provided. There are 10 patterns in the Apostolico Giancarlo algorithm by performing 100 voice commands entered. Furthermore, calculating the standard error with a final result of 0.08. Then, accessibility testing is carried out with the aim that the system can be used by people with disabilities.

Key words: *Air Conditioner, Apostolico Giancarlo, Control System, Internet of Things,*

مستخلص البحث

شاوليا، بلقيس. ٢٠٢٥. نظام التحكم في مكيف الهواء المبني على إنترنت الأشياء باستخدام خوارزمية أبوسطوليكو جيانكارلو. رسالة تخرج. برنامج دراسة هندسة المعلوماتية، كلية العلوم والتكنولوجيا، الجامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج. المشرفان: (١) أجيب هاناني، مهندس ماجستير (٢) أكتا قمر الدين عزيز، ماجستير في الحوسبة.

الكلمات المفتاحية: أبوسطوليكو جيانكارلو، إنترنت الأشياء، مكيف هواء، نظام التحكم

أُجريت هذه الدراسة للتحكم في جهاز تكييف الهواء المعتمد على إنترنت الأشياء باستخدام خوارزمية أبوسطوليكو جيانكارلو. يُوجّه نظام التحكم هذا إلى الأشخاص ذوي الإعاقة، إذ يُنفَّذ عبر إدخال أوامر صوتية من خلال تطبيق يتحكم في المكيف وفق الخوارزمية المذكورة. شارك في التجربة ٤٥ مُستجيباً، وأُجريت لهم ثمانية أسئلة على مقياس خطي يتراوح بين ١ و ٥ باستخدام نموذج «جوجل فورم». تضمّ الخوارزمية عشرة أنماط، وقد أُدخلت مئة أمر صوتي. بعد ذلك حُسِب الخطأ المعياري فبلغ ٠,٠٨، وأُجري اختبار لإمكانية الوصول لضمان قابليّة استخدام النظام من قِبل ذوي الإعاقة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi memberikan banyak keuntungan bagi manusia, salah satunya adalah kemudahan dalam berkomunikasi dan mengakses informasi tanpa terhalang oleh batasan ruang, waktu, maupun jarak. Kemajuan ini tidak hanya mempermudah aktivitas kehidupan sehari-hari, tetapi juga menjadi salah satu bentuk sarana pengabdian kepada Allah *Subhanahu wa Ta'ala*. Segala kemudahan ini merupakan bagian dari nikmat yang diberikan oleh-Nya, berupa perpaduan antara karunia agama dan kemajuan teknologi yang saling melengkapi. Allah *Subhanahu wa Ta'ala* yang menciptakan dan mengendalikan alam semesta dan isinya, sehingga sebagai manusia dapat menikmati anugerah tersebut. Salah satu yang tersirat dari firman Allah *Subhanahu wa Ta'ala* dalam *Al-Qur'an* pada surah *Ar-Ra'd* ayat 02, berfirman:

اللَّهُ الَّذِي رَفَعَ السَّمَاوَاتِ بِغَيْرِ عَمَدٍ تَرَوْنَهَا ثُمَّ اسْتَوَىٰ عَلَى الْعَرْشِ وَسَحَّرَ الشَّمْسَ وَالْقَمَرَ كُلٌّ يَجْرِي لِأَجَلٍ مُّسَمًّى يُدَبِّرُ الْأَمْرَ يُفَصِّلُ الْآيَاتِ لَعَلَّكُمْ بِلِقَاءِ رَبِّكُمْ تُوقِنُونَ

“Allah yang meninggikan langit tanpa tiang (sebagaimana) yang kamu lihat, kemudian Dia bersemayam di atas ‘Arsy dan menundukkan matahari dan bulan. Masing – masing beredar hingga waktu yang ditentukan. Allah yang mengatur urusan (makhluk-Nya), menjelaskan tanda – tanda kebesaran-Nya, supaya kamu yakin akan pertemuanmu dengan Tuhanmu.” (QS. Ar-Ra'd : 02).

Allah menjelaskan keesaan-Nya melalui tanda-tanda dalam penciptaan, pengaturan alam semesta, serta kekuasaan-Nya yang mutlak. Hal ini menunjukkan bahwa hanya Dia-lah yang pantas disembah, dan segala bentuk ibadah seharusnya

ditujukan kepada-Nya semata. Dalam firman-Nya disebutkan bahwa Allah-lah yang meninggikan langit, yang luas dan megah itu, dengan kekuasaan-Nya yang luar biasa — tanpa adanya tiang yang terlihat menopangnya. Ini menegaskan bahwa langit berdiri tanpa penopang dari bawah; sebab bila ada, tentu manusia dapat melihatnya. Setelah penciptaan langit dan bumi, Allah bersemayam di atas ‘Arsy, makhluk ciptaan-Nya yang paling agung dan terletak di tempat tertinggi, sesuai dengan keagungan dan kesempurnaan-Nya. Dia juga menundukkan matahari dan bulan untuk mendukung keberlangsungan hidup manusia, hewan, serta tumbuh-tumbuhan. Keduanya bergerak dalam orbit yang teratur dan terkendali oleh kehendak Allah Yang Maha Perkasa dan Maha Mengetahui, hingga waktu yang telah ditentukan, yakni ketika alam semesta ini dilipat dan diganti dengan negeri akhirat sebagai tempat tinggal yang abadi. Pada saat itu, langit akan dilipat, bulan dan bumi akan diubah wujudnya. Allah mengatur segala urusan makhluk-Nya dan menampilkan tanda-tanda kekuasaan-Nya. Ini merupakan kombinasi dari kuasa penciptaan (al-Khalq) dan kehendak perintah-Nya (al-Amr). Artinya, Allah Yang Maha Agung memegang kendali penuh atas seluruh ciptaan-Nya, baik di langit maupun di bumi. Dia menciptakan, memberikan rezeki, meninggikan dan merendahkan derajat manusia, mengatur nasib mereka sesuai dengan ketetapan yang telah ditulis oleh pena-Nya (al-Qalam). Dengan banyaknya dalil dan penjelasan yang jelas, manusia diajak untuk membangun keyakinan terhadap aspek-aspek ketuhanan, terutama yang berkaitan dengan akidah dasar seperti Hari Kebangkitan, Pengumpulan, dan Keluarnya manusia dari kubur. Telah dipastikan bahwa Allah tidak menciptakan sesuatu

tanpa tujuan. Semua ciptaan-Nya memiliki makna dan hikmah, dan pada akhirnya, setiap amal perbuatan akan mendapatkan balasan: kebaikan dibalas dengan kebaikan, dan keburukan dibalas sesuai dengan perbuatannya. (as-Sa'di, 1421 H).

Pada penjelasan *Qur'an* Surah *Ar-Ra'd* ayat 02, dapat diartikan bahwa Allah menciptakan dan mengatur alam semesta dengan segala keteraturannya, manusia sebagai makhluk yang diberikan akal dan pengetahuan juga memiliki kemampuan untuk menciptakan dan mengendalikan teknologi di alam semesta. Teknologi yang diciptakan oleh manusia dalam bentuk mesin, komputer, atau alat – alat teknologi canggih lainnya adalah manifestasi dari pengetahuan dan kebijaksanaan yang Allah berikan kepada makhluk-Nya. Ayat tersebut mengajarkan bahwa untuk selalu mengingat dibalik kemampuan makhluk-Nya untuk menciptakan dan mengendalikan teknologi, ada kekuasaan Allah yang mengatur segalanya. Dengan demikian, tidak hanya menciptakan teknologi yang maju secara ilmiah, tetapi juga membawa kebaikan dan manfaat bagi alam semesta dan isinya.

Smartphone saat ini telah berkembang menjadi alat multifungsi yang menyatukan berbagai fungsi alat komunikasi dengan fitur – fitur canggih yang mendukung beragam kebutuhan manusia. Tidak hanya sebagai alat untuk telepon dan SMS, *smartphone* kini menjadi sarana pembelajaran, media hiburan, dan alat yang menyalurkan hobi seperti bermain *game*, *dance*, hingga menjadi *platform* untuk menjalankan bisnis. Kehadiran *smartphone* telah menjadi kebutuhan mendasar bagi setiap individu, baik dalam keadaan sehat maupun sakit.

Dalam aspek aksesibilitas, *smartphone* memberikan fitur – fitur yang dirancang untuk memastikan bahwa semua orang, termasuk yang memiliki keterbatasan fisik, sensorik, atau kognitif, dapat mengakses dan memanfaatkan teknologi ini. Contohnya, fitur *voice command* dan asisten virtual yang dapat digunakan *user* dengan keterbatasan penglihatan untuk mengoperasikan perangkat tanpa harus menyentuh layar. Fitur teks ke suara (*text-to-speech*) dan suara ke teks (*speech-to-text*) memudahkan komunikasi bagi mereka yang memiliki kesulitan dalam membaca atau menulis. Dengan pembaruan fitur yang terus – menerus, *smartphone* juga dapat diintegrasikan dengan sistem *smart home* yaitu mengendalikan perangkat rumah. Fitur ini tidak hanya menambah kenyamanan, tetapi juga memberikan aksesibilitas yang lebih baik bagi individu dengan mobilitas terbatas.

Dalam aktivitas sehari-hari di rumah, kita sering memanfaatkan berbagai perangkat elektronik untuk memenuhi kebutuhan, seperti menyalakan lampu sebagai sumber penerangan, menghidupkan AC saat cuaca panas, atau menyalakan televisi. Bagi orang yang sehat, tindakan-tindakan ini merupakan hal yang mudah dan rutin dilakukan. Namun, situasinya berbeda bagi individu yang mengalami gangguan mobilitas, seperti penderita kelumpuhan yang hanya dapat berada di tempat tidur atau menggunakan kursi roda. Keterbatasan fisik ini membuat mereka kesulitan dalam mengoperasikan perangkat rumah secara manual, yang biasanya hanya membutuhkan tekanan pada tombol. Dalam kondisi seperti ini, dibutuhkan solusi yang memungkinkan akses yang lebih mudah salah

satunya melalui teknologi berbasis *Internet of Things* (IoT), yang dapat memberikan kemudahan kontrol perangkat secara otomatis dan jarak jauh.

Di dalam penerapan sistem *smart home* yang telah banyak digunakan saat ini, teknologi dapat mengontrol semua peralatan rumah tangga hanya dengan menggunakan *smartphone*. Salah satu perangkat yang paling bermanfaat adalah AC (*Air Conditioner*) yang sangat penting untuk menjaga suhu dan kenyamanan dalam ruangan, terutama di cuaca yang panas. Meskipun banyak sistem kendali AC (*Air Conditioner*) yang saat ini menggunakan *remote* kontrol inframerah, sistem – sistem ini memerlukan interaksi manual yang intensif dan tidak selalu efisien. Misalnya, ketika *user* berada dalam jarak jauh dari perangkat, atau ketika *user* tidak dapat menggunakan tangan untuk mengoperasikan *remote*, bahkan saat *remote* tersebut hilang atau tidak ditemukan, atau apabila didalam suatu ruangan ramai, kendala – kendala ini dapat menimbulkan frustrasi.

Oleh karena itu, konsep *smart home* yang berintegrasi dengan IoT (*Internet of Things*) menjadi semakin relevan. IoT (*Internet of Things*) merupakan gagasan pada semua benda dalam kehidupan sehari-hari dapat saling berkomunikasi sebagai bagian dari satu kesatuan sistem menggunakan jaringan internet sebagai penghubungnya. Salah satu implementasi IoT (*Internet of Things*) yang paling menarik adalah penggunaan input suara. Dalam sistem ini, perintah suara yang diberikan oleh *user* diproses oleh perangkat *mobile* menggunakan bahasa alami, menjadikan perangkat *mobile* sebagai konsol sentral yang dapat mengendalikan berbagai peralatan rumah.

Perkembangan ini bukan hanya sebuah kemajuan teknologi, tetapi juga sebuah harapan di masa depan dengan kemudahan dan aksesibilitas sebagai prioritas dengan susahnyanya disabilitas saat ini atas keterbatasan yang dimilikinya. Salah satu faktor penting dalam sistem *Speech Recognition* adalah kemampuan untuk menguji suara terhadap pola kata per-kata dengan presisi. Pengujian pola seperti ini digunakan dalam berbagai program yang dapat ditemukan sehari – hari, seperti fitur pencarian pada editor teks, mesin pencari web seperti Google, analisis citra, hingga bioinformatika seperti pengujian rantai asam amino dan rantai DNA.

Teknologi ini diharapkan tidak hanya mampu memberikan kemudahan, tetapi juga mampu beradaptasi dengan berbagai situasi yang dihadapi *user*. Seiring dengan perkembangan ini, peneliti berharap bahwa teknologi tidak hanya menjawab kebutuhan teknis, tetapi juga menjadi lebih intuitif, empatik, dan dapat diakses oleh semua lapisan masyarakat.

Penelitian ini dirancang dengan mengintegrasikan elemen-elemen dari *Internet of Things* (IoT). Salah satu komponen penting yang digunakan adalah algoritma Apostolico-Giancarlo, yang merupakan penyempurnaan dari algoritma *Boyer-Moore*. Pada tahap pencocokan string, algoritma ini menggunakan struktur array untuk menyimpan informasi yang berfungsi sebagai penanda kecocokan antara bagian akhir pola dan teks. Fitur pengingat inilah yang menjadi salah satu nilai tambah algoritma ini, karena mampu meningkatkan efektivitas dalam proses pencarian.

1.2 Pernyataan Masalah

Adapun pernyataan masalah yang didapatkan yaitu:

- a. Bagaimana aksesibilitas sistem kendali AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) menggunakan algoritma *Apostolico Giancarlo*?
- b. Bagaimana performa sistem kendali AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) menggunakan algoritma *Apostolico Giancarlo*?

1.3 Batasan Masalah

Batasan masalah dibuat untuk mencegah melebarnya topik dalam penelitian, adapun batasan – batasan masalah pada penelitian ini yaitu:

- a. AC (*Air Conditioner*) yang digunakan jenis Sharp.
- b. *Device* yang digunakan sebagai *Speech Recognition* yaitu *smartphone android*.
- c. Menggunakan satu koneksi jaringan yang sama.
- d. Memasukan perintah suara dalam bahasa Indonesia.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu:

- a. Untuk mengukur aksesibilitas pada sistem kendali AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) menggunakan algoritma *Apostolico Giancarlo*.
- b. Untuk mengukur performa pada sistem kendali AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) menggunakan algoritma *Apostolico Giancarlo*.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu membantu para penyandang cacat fisik, orang yang sudah tua, memudahkan orang untuk memerintah kendali AC di dalam ruangan yang ramai dan memudahkan bagi orang yang sedang sakit dalam melakukan aktivitas sehingga memberikan kenyamanan dan kemudahan untuk mengendalikan AC (*Air Conditioner*).

BAB II

STUDI PUSTAKA

Bab ini menjelaskan mengenai penelitian terdahulu dan dasar teori yang digunakan untuk mendukung penelitian.

Penelitian dengan rancangan sebuah sistem perangkat keras berbasis IoT (*Internet of Things*) untuk manajemen penggunaan AC (*Air Conditioner*) secara otomatis yaitu mendeteksi suhu dan kelembaban udara, serta sensor PIR untuk mendeteksi keberadaan seseorang di dalam ruangan. Hasil dari penelitian rancangan ini menunjukkan bahwa sistem kontrol AC (*Air Conditioner*) otomatis berhasil diimplementasikan dengan baik. Berdasarkan hasil uji coba yang telah dilakukan, sistem ini mampu mengaktifkan dan menonaktifkan AC (*Air Conditioner*) secara otomatis dengan kondisi yang terdeteksi dan kinerja yang dihasilkan efektif (Pahrurrozi, 2020).

Penelitian yang mengembangkan sebuah sistem kendali jarak jauh berbasis IoT (*Internet of Things*) untuk mengendalikan AC (*Air Conditioner*) sebagai pengganti *remote* konvensional. Sistem ini menunjukkan kinerja yang baik dengan respon yang cepat antara *user* dan perangkat mikrokontroler. Namun, sistem ini memiliki keterbatasan yaitu apabila merk AC (*Air Conditioner*) diganti, maka kode program harus disesuaikan ulang dengan merk AC (*Air Conditioner*) yang baru (Pribadi, 2020).

Penelitian ini mengembangkan sistem pemantauan dan pengendalian suhu AC (*Air Conditioner*) otomatis berbasis IoT (*Internet of Things*) yang terintegrasi dengan aplikasi Blynk. Tujuan dari penelitian yang dilakukan untuk mempermudah *user* dalam mengatur suhu AC (*Air Conditioner*) sehingga pengguna tidak perlu sering melakukan pengaturan manual pada *remote* AC (*Air Conditioner*). Hasil dari uji coba menunjukkan bahwa sistem yang dikembangkan mampu menyesuaikan suhu AC (*Air Conditioner*) secara otomatis sesuai dengan suhu yang diinginkan oleh *user* (Widodo, 2021).

Sebuah sistem kontrol dan pemantauan AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) telah dikembangkan untuk memantau kondisi suhu ruangan dan mengendalikan suhu agar sesuai dengan preferensi *user*. Sistem ini memanfaatkan kontrol proposional dan teknologi IoT (*Internet of Things*) untuk pengoperasiannya. Hasil uji coba dari penelitian ini menunjukkan bahwa penguat proposional yang diterapkan terbukti efektif dalam mengendalikan sistem AC (*Air Conditioner*) (Lara, 2021).

Sebuah aplikasi *mobile* untuk pencarian istilah olahraga telah dirancang dengan menerapkan algoritma *Apostolico Giancarlo* memiliki kegunaan untuk mempermudah *user* dalam mencari informasi terkait istilah olahraga di dalam aplikasi. Aplikasi ini dikembangkan menggunakan *Eclipse Juno*, dengan fitur – fitur utama seperti pencarian, biodata, dan opsi penutupan aplikasi. Hasil implementasi yang telah dilakukan oleh peneliti menunjukkan bahwa program ini berhasil dan berfungsi sesuai dengan tujuang yang diharapkan (Silitonga, 2021).

Sebuah sistem mencari fasilitas umum di kawasan Aceh berbasis Android telah dikembangkan dengan menerapkan algoritma *Apostolico Giancarlo*. Algoritma ini bekerja dengan melakukan pencarian *string* melalui proses *pre-processing*. Implementasi algoritma yang dilakukan memiliki tujuan yaitu untuk mengatasi kesulitan dalam pencarian fasilitas umum. Hasil uji coba yang dilakukan oleh peneliti menunjukkan bahwa penggunaan algoritma *Apostolico Giancarlo* dapat menyederhanakan proses pencarian fasilitas umum di Aceh (Buulolo, 2021).

Sistem pencarian katalog mebel berbasis *mobile* telah dikembangkan dengan menerapkan algoritma *Apostolico Giancarlo* untuk meningkatkan efisiensi pencarian produk PT. Yasanda. Sistem ini dirancang untuk calon konsumen dan konsumen melakukan pencarian produk dengan cepat dan akurat, sesuai dengan preferensi yang diinginkan. Hasil uji coba yang dilakukan oleh peneliti menunjukkan bahwa algoritma ini terbukti menghasilkan jumlah pencarian yang lebih sedikit dan efisien, serta mampu meningkatkan kesesuaian dalam proses pencarian produk mebel (Barus, 2021).

2.1 Landasan Teori

2.1.1 Algoritma *Apostolico Giancarlo*

Algoritma *Apostolico Giancarlo* menggunakan ide dasar algoritma *Boyer Moore*, algoritma ini memiliki sebuah variable yang menyimpan akhiran yang sama, akhiran yang sama ini disimpan dalam sebuah *array* yang menyimpan semua karakter sama yang telah ditemukan. Pada prosesnya, tidak semua karakter yang akan dilakukan perbandingan. Karakter yang sama yang pernah

dibandingkan dan telah disimpan tidak akan diulangi lagi perbandingannya. *Apostolico Giancarlo* menyimpan *pattern* dari perbandingan yang sama tersebut agar perbandingan tersebut tidak terjadi lagi (Kementrian Kesehatan Republik Indonesia, 2018).

2.1.2 String Matching

Algoritma dalam pencarian string, atau yang dikenal juga dengan istilah *string matching*, merupakan metode yang digunakan untuk menemukan seluruh kemunculan dari sebuah string pendek yang disebut *pola* (pattern) dalam sebuah string yang lebih panjang, yaitu *teks* (text). Salah satu algoritma yang termasuk dalam kategori ini adalah algoritma Apostolico-Giancarlo. Teknik pencocokan string ini terbagi menjadi dua jenis, yaitu *exact matching* dan *heuristic* atau *statistical matching*. *Exact matching* digunakan untuk mencocokkan pola yang berasal dari satu teks secara tepat. Misalnya, dalam proses pencarian kata “pelajar” pada kalimat “saya seorang pelajar” dan “saya seorang siswa”, sistem akan mengidentifikasi bahwa kata “pelajar” hanya terdapat dalam kalimat pertama. Meskipun “pelajar” dan “siswa” memiliki arti yang serupa, sistem *exact matching* tidak menganggap keduanya sebagai padanan karena pencocokan hanya didasarkan pada bentuk kata yang persis sama (Irawan, 1977).

2.1.3 Air Conditioner

AC (*Air Conditioner*) atau HVAC (*Heating, Ventilating, Air Conditioning*) yaitu suatu sistem pengkondisi udara yang terdiri dari sistem motor, saluran pipa, sistem petukaran udara, sistem kontrol dan pertukaran panas. Tujuan

sistem ini untuk mengatur aliran udara serta memindahkan komponen udara yang tidak diinginkan dari suatu ruangan dengan tujuan untuk mendapatkan kondisi yang diinginkan oleh pengguna. Mengatur pergerakan udara untuk mengusir bau dan debu agar kenyamanan dan kesehatan tetap terjaga (Salaman Dian, 2019).

2.1.4 *Internet of Things*

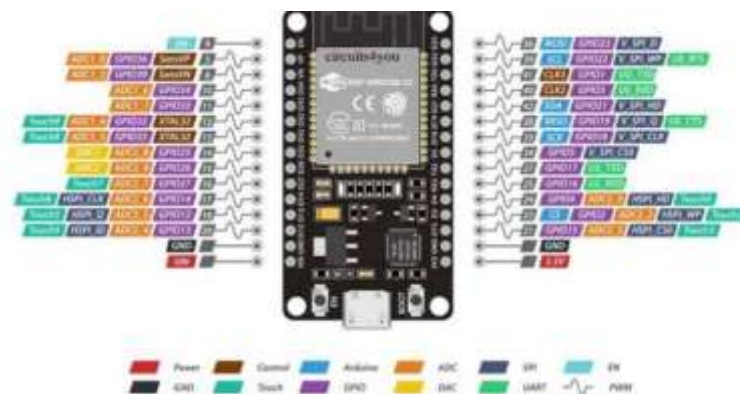
IoT (*Internet of Things*) merupakan suatu jaringan dengan menghubungkan berbagai objek yang memiliki identitas pengenalan seperti alamat IP, sehingga dapat saling bertukar informasi mengenai dirinya maupun lingkungan sekitarnya yang diindera. Objek dalam IoT (*Internet of Things*) dapat menggunakan maupun menghasilkan layanan-layanan dan saling bekerjasama untuk mencapai tujuan yang dituju. IoT (*Internet of Things*) menggeser definisi internet sebagai komputasi dimana saja, kapan saja, dan apapun yang terjadi. Tujuan IoT (*Internet of Things*) dapat membuat manusia berinteraksi dengan benda lebih mudah, bahkan dengan tujuan ini supaya benda saling berkomunikasi antar satu benda dengan benda yang lainnya. IoT (*Internet of Things*) juga dapat memulai sistem dengan cara membuat suatu koneksi antara mesin dengan mesin, sehingga mesin – mesin tersebut dapat bekerja dengan sendirinya sesuai data yang diperoleh. Sistem IoT (*Internet of Things*) menjadi lebih kompleks dengan sejumlah besar perangkat yang dapat saling berhubungan untuk menghasilkan layanan yang mendukung untuk pemrosesan (Dianti, 2017).

2.1.5 Modul ESP32

Modul ESP32 adalah sebuah *board* elektronik dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (Wi-Fi) dan *Bluetooth*. Memiliki beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi *monitoring* maupun *controlling* pada proyek IoT (*Internet of Things*). Modul ESP32 dapat diprogram dengan *compiler* Arduino, menggunakan Arduino IDE. Bentuk fisik Modul ESP32, terdapat port USB (mini USB) sehingga akan memudahkan dalam pemrogramannya (Nadiansyah, 2018).

Modul ESP32 merupakan modul turunan pengembangan dari modul platform IoT. Secara fungsi modul ini hampir menyerupai dengan platform modul Arduino, tetapi yang membedakan yaitu dikhususkan untuk “*Connected to Internet*”.

Bentuk fisik dari Modul ESP32 dapat dilihat Gambar 2.1



Gambar 2.1 Modul ESP32

2.1.6 Sensor IR Transmitter

Pemancar inframerah atau *Infrared Transmitter* merupakan jenis *Light Emitting Diode* (LED) yang menghasilkan radiasi dalam bentuk sinyal

inframerah, dikenal sebagai IR LED. Walaupun secara fisik menyerupai LED biasa, cahaya inframerah yang dihasilkan oleh IR LED tidak tampak oleh mata manusia. Oleh karena itu, untuk mendeteksinya digunakan alat seperti spektroskop cahaya, yang mampu menangkap pancaran radiasi tersebut pada spektrum elektromagnetik dengan panjang gelombang yang lebih besar dari cahaya merah. Gambar 2.2 dibawah ini menunjukkan gambar IR Transmitter (Pustaka, 2019).



Gambar 2.2 IR Transmitter

2.1.7 IR Receiver

IR Receiver disebut sebagai sensor inframerah karena dapat mendeteksi radiasi dari pemancar inframerah (IR Transmitter). IR receiver dapat terbuat dari photodiode maupun phototransistor. Photodiode pada IR receiver berbeda dengan diode pada umumnya, karena diode ini hanya mendeteksi inframerah. Gambar 2.3 dibawah ini menunjukkan gambar IR Receiver.



Gambar 2.3 IR Receiver

2.1.8 *Speech Recognizer*

Pada umumnya, *speech recognizer* memproses sinyal suara yang masuk dan menyimpannya dalam bentuk digital. Hasil proses digitalisasi tersebut kemudian dikonversi dalam bentuk spectrum suara yang akan dianalisa dengan membandingkan *template* suara pada *database* sistem. Cara kerja *speech recognizer* yang digunakan oleh peneliti saat ini yaitu *Google Speech Recognizer* memiliki cara kerja mengubah suara menjadi teks, kemudian dibandingkan dengan *pattern* pada algoritma *Apostolico Giancarlo*. Dalam penggunaan *Google Speech Recognizer* ini akan mempermudah untuk mengendalikan kontrol AC hidup maupun mati (Ii and Teori, 2015).

2.1.9 *Stemming*

Stemming adalah proses untuk memperoleh bentuk dasar atau kata akar dari sebuah kata yang mengalami penambahan imbuhan, dengan cara menghapus seluruh afiks yang melekat. Afiks tersebut bisa berupa awalan (*prefix*), akhiran (*suffix*), sisipan (*infix*), maupun gabungan antara awalan dan akhiran (*confix*) (Simarangkir, 2017).

2.1.10 *Aksesibilitas*

Aksesibilitas merupakan hak setiap individu untuk memperoleh kemudahan dalam melakukan perjalanan, yang merupakan bagian dari layanan kebutuhan dasar. Oleh karena itu, penyediaan aksesibilitas menjadi tanggung jawab pemerintah, tanpa mempersoalkan apakah sarana transportasi yang

disediakan benar-benar dimanfaatkan oleh masyarakat atau tidak (Rosselawaty Henny, 2023).

Adapun beberapa aspek standard komponen aksesibilitas pada penelitian ini memiliki 7 pedoman yang dapat diterapkan untuk meningkatkan aksesibilitas dan kegunaan aplikasi berbasis perintah suara (Murad, Candello, and Munteanu, 2023), yaitu:

1. Menggunakan bahasa yang sederhana dan jelas
2. Mendesain untuk kesalahan *user* apabila dimasukan perintah suaranya, maka akan meminta klarifikasi atau memberikan instruksi ulang
3. Menyediakan pertimbangan untuk teks *user* dengan penggunaan pendengaran
4. Menyederhanakan struktur perintah suara dengan menghindari perintah suara yang terlalu panjang atau kompleks
5. Mendesain untuk *user* yang memiliki gangguan kognitif dengan menyediakan fitur memperlambat perintah suara dan mengulangi instruksi jika diperlukan oleh *user*
6. Mendesain untuk *user* yang memiliki gangguan penglihatan dengan memastikan informasi *output* dan *input* dapat diakses melalui suara
7. Uji dengan *user* yang nyata, dengan melakukan pengujian kepada *user* yang memiliki disabilitas untuk memastikan sistem dapat digunakan

2.2 Perbandingan Penelitian Terkait

Tabel 2. 1 Tabel Perbandingan Penelitian Terkait

No.	Peneliti	Persamaan	Perbedaan
1.	(Pahrurrozi, 2020)	Mendeteksi suhu AC (<i>Air Conditioner</i>) secara otomatis	Menggunakan mikrokontroler D1, sensor suhu DHT22, sensor PIR, tidak menggunakan algoritma <i>Apostolico Giancarlo</i> dan tidak menggunakan <i>Speech Recognition</i>
2.	(Pribadi, 2020)	Sistem pengendali jarak jauh yang berbasis IoT (<i>Internet of Things</i>) untuk mengendalikan AC (<i>Air Conditioner</i>)	Menggunakan <i>monitoring</i> aplikasi Blynk, tidak menggunakan algoritma <i>Apostolico Giancarlo</i> dan tidak menggunakan <i>Speech Recognition</i>
3.	(Widodo, 2021)	Kontrol suhu AC (<i>Air Conditioner</i>) secara otomatis berbasis IoT (<i>Internet of Things</i>)	Menggunakan <i>monitoring</i> aplikasi Blynk, menggunakan komponen sensor suhu DHT11, Optocoupler, resistor 22 ohm, tidak menggunakan algoritma <i>Apostolico Giancarlo</i> dan tidak menggunakan <i>Speech Recognition</i>
4.	(Lara, 2021)	Sistem kendali AC (<i>Air Conditioner</i>) jarak jauh berbasis IoT (<i>Internet of Things</i>)	Alat yang digunakan yaitu kontrol tipe proposional, sensor DHT11, tidak menggunakan algoritma <i>Apostolico Giancarlo</i> dan tidak menggunakan <i>Speech Recognition</i>
5.	(Silitonga, 2021)	Metode algoritma <i>Apostolico Giancarlo</i>	Pencarian istilah olahraga pada sebuah aplikasi <i>mobile</i> dan tidak menggunakan <i>Speech Recognition</i>
6.	(Bulololo, 2021)	Metode algoritma <i>Apostolico Giancarlo</i>	Pencarian fasilitas umum di Aceh dan tidak menggunakan <i>Speech Recognition</i>
7.	(Barus, 2021)	Metode algoritma <i>Apostolico Giancarlo</i>	Pencarian katalog mebel berbasis <i>mobile</i> dan Metode algoritma <i>Apostolico Giancarlo</i>

Perbedaan pada penelitian – penelitian terkait yang sudah dipaparkan diatas dengan penelitian ini terletak pada fokus utama dari pengembangan sistem dan pendekatan yang digunakan. Beberapa penelitian sebelumnya, seperti yang dilakukan oleh (Pahrurrozi, 2020), (Pribadi, 2020), (Widodo, 2021), dan (Lara, 2021), berfokus pada pengembangan sistem kontrol AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) untuk manajemen suhu dan kelembaban udara secara otomatis. Pendekatan yang digunakan dalam penelitian – penelitian ini cenderung berfokus pada pengendalian suhu secara otomatis berdasarkan kondisi lingkungan atau melalui kendali jarak jauh berbasis aplikasi, namun tidak menggunakan algoritma spesifik untuk meningkatkan efisiensi pencarian atau pengendalian yang lebih presisi.

Penelitian ini juga berbeda dari penelitiann – penelitian yang menerapkan algoritma *Apostolico Giancarlo* seperti yang dilakukan oleh (Silitonga, 2021), (Buulolo, 2021), dan (Barus, 2021). Dalam penelitian – penelitian tersebut, algoritma *Apostolico Giancarlo* digunakan untuk mengoptimalkan pencarian istilah, fasilitas umum, dan katalog mebel. Implementasi algoritma dalam konteks tersebut bertujuan untuk mempercepat proses pencarian informasi yang spesifik, namun tidak mengaplikasikan algoritma ini dalam sistem kendali AC (*Air Conditioner*).

Penelitian ini berfokus pada teknologi IoT (*Internet of Things*) dan algoritma *Apostolico Giancarlo* dalam satu sistem kendali AC (*Air Conditioner*). Penggunaan algoritma ini dalam pengendalian AC (*Air Conditioner*) berbasis IoT (*Internet of Things*) merupakan inovasi baru yang belum diterapkan dalam

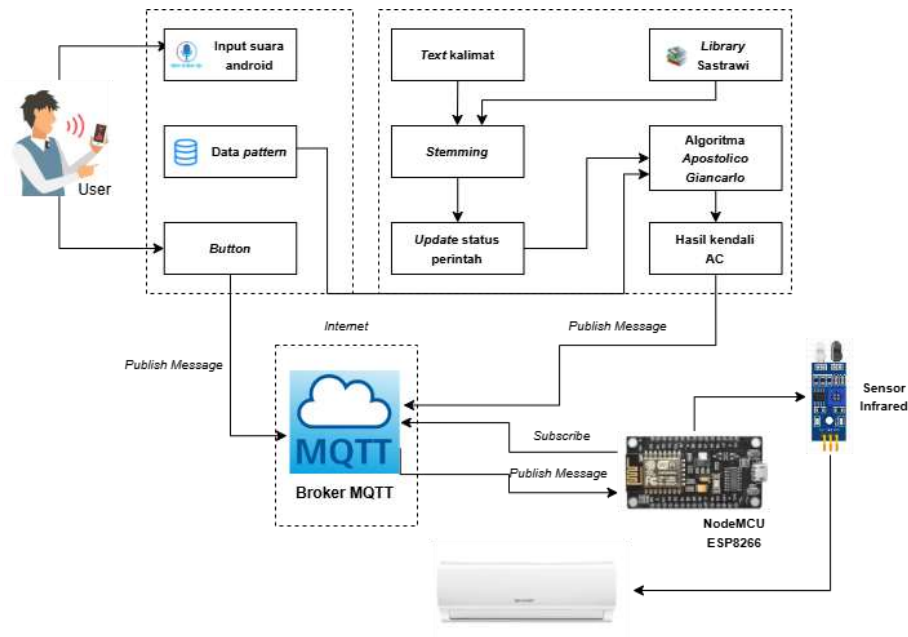
penelitian – penelitian sebelumnya. Algoritma *Apostolico Giancarlo* yang digunakan dalam penelitian ini diharapkan dapat meningkatkan efisiensi dan akurasi sistem kendali, baik dalam mendeteksi kondisi suara yang diperintahkan maupun dalam proses penyesuaian suhu sesuai preferensi *user*. Dengan demikian, penelitian ini mengisi perbedaan dalam penelitian – penelitian sebelumnya yaitu memperkenalkan penerapan algoritma *Apostolico Giancarlo* pencarian yang telah terbukti efisien dalam konteks pengendalian perintah berguna sebagai kontrol AC (*Air Conditioner*). Penggunaan algoritma ini dapat menjadi dasar bagian pengembangan lebih lanjut dalam pengendalian perangkat berbasis IoT (*Internet of Things*), yang tidak hanya fokus pada efisiensi kontrol, tetapi juga peningkatan kecepatan dalam menyesuaikan preferensi *user*.

BAB III

DESAIN DAN IMPLEMENTASI SISTEM

3.1 Desain dan Implementasi Sistem

Pada penelitian ini didalamnya terdiri dari *input*, proses dan *output* yang merupakan pengujian sistem dan perancangan. Pada tahapan *input* merupakan tahapan dalam pengambilan data, kemudian data tersebut dimasukkan dan diproses pada bagian proses, *user* dapat melakukan kontrol suara pada tombol. Apabila *user* menggunakan kontrol suara maka memilih fitur *Google Speech Recognition* yang akan diolah ke dalam bentuk teks dan diproses ke *Stemming* dengan menggunakan *Library Sastrawi* kemudian melakukan proses pencocokan *pattern* dengan metode algoritma *Apostolico Giancarlo*. Setelah *pattern* ditemukan, maka hasilnya akan dilanjutkan ke proses mengubah teks perintah menjadi data perintah kemudian membandingkan data tersebut dengan data yang tersimpan di *database*. Jika data perintah sama dengan data perintah di *database* maka proses dihentikan, tetapi ketika data tidak sama dengan data perintah di *database* maka akan dilanjutkan ke proses penyimpanan data ke *database* (*update*) lalu dikendalikan oleh *Modul ESP32* untuk menghidupkan dan mematikan AC (*Air Conditioner*). Pengujian sistem dilakukan dengan 2 cara, pertama penguji melakukan pengujian menggunakan algoritma *Apostolico Giancarlo* untuk mencari pencocokan *pattern* dan yang kedua menentukan akurasi sistem yang dibangun dengan menggunakan *Confusion Matrix*. Desain sistem dapat dilihat pada Gambar 3.1.



Gambar 3.1 Desain Sistem

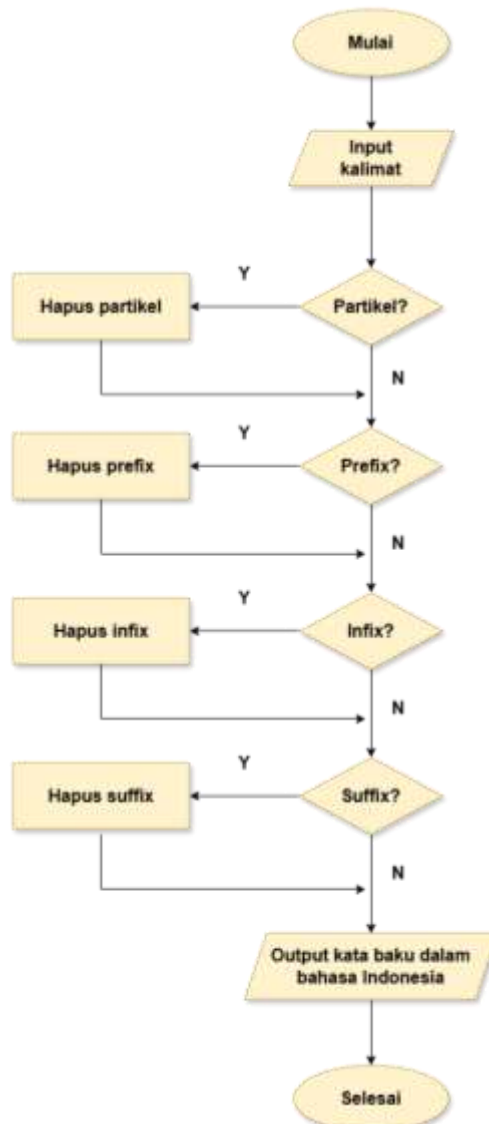
3.1.1 Proses pengambilan Data

Data yang digunakan pada penelitian ini memiliki data *primer* dan data sekunder dengan arti bahwa data *primer* merupakan data dari *input-an* suara *Speech Recognition* dengan menggunakan *Google Speech* dalam Bahasa Indonesia dan di konversi menjadi teks. Kemudian, pada data sekunder adalah data *pattern* yang dibuat oleh peneliti untuk kata kunci. Berikut data sekunder yang dibuat oleh peneliti : hidupkan AC, matikan AC, AC hidupkan dan AC matikan.

3.1.2 Proses Stemming

Stemming merupakan proses pengubahan kalimat atau kata yang berimbuhan menjadi kata baku. Dalam proses tersebut sistem akan mengolah

menjadi kata-kata dasar dalam Bahasa Indonesia. Proses *Stemming* dengan *Library* Sastrawi dalam pemrograman php dapat dilihat dari Gambar 3.2.

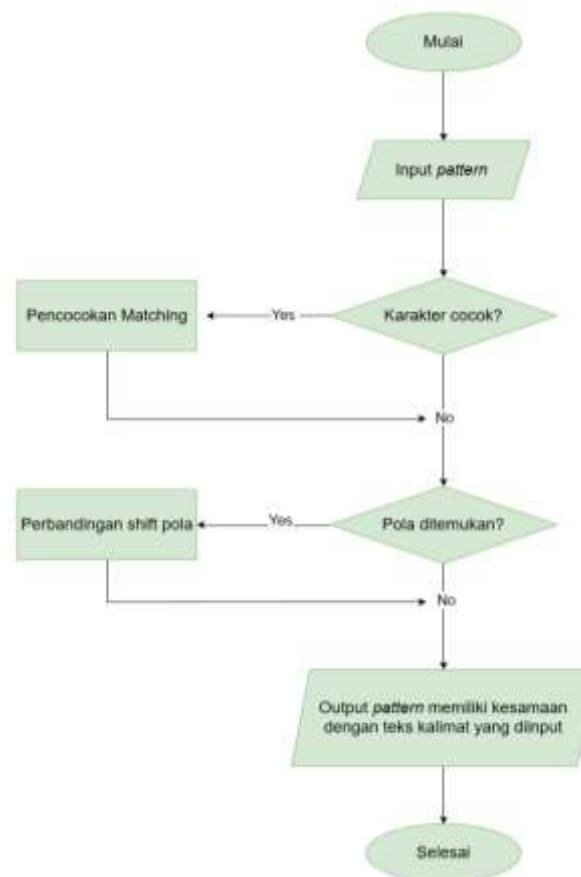


Gambar 3.2 Flowchart Stemming

3.1.3 Proses Algoritma Apostolico Giancarlo

Proses Algoritma Apostolico Giancarlo merupakan sebuah algoritma pencocokan dari *string* yang digunakan untuk mencari pola (*pattern*) dalam sebuah teks. Dalam proses tersebut, pola akan melakukan pencarian yang sudah

ditemukan selama proses pencocokan, sehingga dapat mengurangi waktu pencarian. Proses Algoritma *Apostolico Giancarlo* tersebut dapat dilihat pada Gambar 3.3



Gambar 3.3 Flowchart Proses Algoritma *Apostolico Giancarlo*

3.1.4 Algoritma *Apostolico Giancarlo*

Algoritma *Apostolico Giancarlo* merupakan algoritma pengembangan dari algoritma *BoyerMoore*. Pada proses pencocokan *string*, algoritma *Apostolico Giancarlo* menyimpan sebuah variabel dalam sebuah *array* yang menjadi faktor pengingat teks yang sama dengan akhiran dari *pattern* dari proses perbandingan terakhir. Penerapan algoritma *Apostolico Giancarlo* pada penelitian ini

mempermudah pencarian kata “AC HIDUPKAN” dan “AC MATIKAN” dengan efektif dan efisien.

Salah satu contoh implementasi algoritma *Apostolico Giancarlo* pada penelitian ini adalah melakukan pencocokan *pattern* “HIDUP” dengan teks “AC HIDUPKAN”. Adapun penyelesaian dalam pencocokan *pattern* “HIDUP” dengan teks “AC HIDUPKAN” dengan algoritma *Apostolico Giancarlo* adalah sebagai berikut:

1. Tahap pertama

Pada tahap pertama dalam proses *pre-processing* pada algoritma *Apostolico Giancarlo* adalah mengurutkan semua karakter teks yang diperiksa tanpa terjadinya perulangan karakter yang sama. Hasil pengurutan karakter teks “AC HIDUPKAN” adalah “ACDHIKNPU”

2. Tahap kedua

Pada tahap kedua peneliti melakukan proses pengecekan kemunculan setiap karakter teks yang diperiksa pada karakter *pattern* dimulai dari *index* karakter paling kanan ke arah karakter kiri. Nilai *index* diberikan nilai mulai angka 0 (nol), maka nilai *index pattern* “HIDUP” adalah “01234”. Karakter teks yang tidak memiliki kesamaan dengan *pattern* dinamakan dengan karakter buruk, maka diberi pengkodean dengan nilai $n + 1$, n adalah nilai maksimum dari *pattern*. Karakter teks yang memiliki kesamaan dengan *pattern* diberikan nilai sesuai dengan nilai *index* karakter tersebut. Dari proses pengecekan tersebut, maka dihasilkan *pre-processing* berikut ini, pada Tabel 3.1:

Tabel 3.1 Hasil *pre-processing*

Karakter teks	A	C	D	H	I	K	N	P	U	Spasi
Kode	5	5	2	0	1	5	5	4	3	5

3. Tahap ketiga

Pencocokan *pattern* dan teks dengan *shift* yang telah ditentukan pada kode di tabel hasil *pre-processing*. Adapun tahapan pencocokan *pattern* dengan teks pada penelitian ini dapat dilihat pada tabel 3.2 berikut:

Tabel 3.2 Pencocokan Pertama

Teks	A	C		H	I	D	U	P	K	A	N
<i>Pattern</i>	H	I	D	U	P						

Pada pencocokan tahap pertama di Tabel 3.2, terlihat bahwa P dengan I tidak sama, I merupakan karakter baik, maka dilakukan pergeseran *pattern* sebanyak 1 langkah.

Tabel 3.3 Pencocokan Kedua

Teks	A	C		H	I	D	U	P	K	A	N
<i>Pattern</i>		H	I	D	U	P					

Pada pencocokan tahap kedua di Tabel 3.3, terlihat bahwa P dengan D tidak sama, D merupakan karakter baik, maka pergeseran *pattern* sebanyak 2 langkah.

Tabel 3.4 Pencocokan Ketiga

Teks	A	C		H	I	D	U	P	K	A	N
<i>Pattern</i>				H	I	D	U	P			

Pada pencocokan tahap ketiga di Tabel 3.4 terlihat bahwa seluruh karakter *pattern* memiliki kesamaan dengan teks, maka proses *pattern* dihentikan.

3.2 Perancangan Antarmuka (*Interface*)

Proses perancangan antarmuka (*interface*) pada penelitian ini dirancang agar dapat memberikan kemudahan dalam membangun sistem sehingga dapat dipahami tujuannya dan fungsi dari setiap tampilan pada sistem tersebut. Pada Gambar 3.4 adalah tampilan awal pengguna.



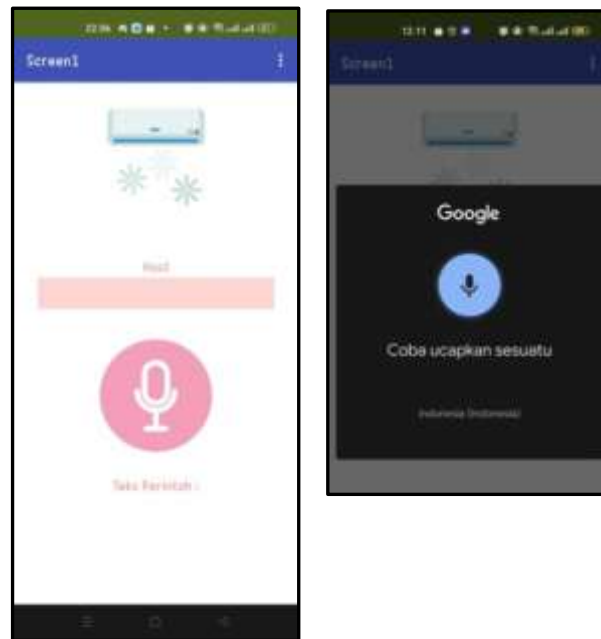
Gambar 3.4 Rancangan Aplikasi

Keterangan gambar:

1. *Button Google Speech Recognition* digunakan untuk kendali AC dengan menggunakan suara,
2. *Image View* untuk menampilkan status pada AC

3.3 Halaman Aplikasi

Tampilan awal halaman aplikasi dapat dilihat pada gambar 3.5 yang menampilkan *icon* gambar dan teks untuk menampilkan status AC (*Air Conditioner*). Kemudian, terdapat *button microphone* yang berfungsi untuk mengontrol AC (*Air Conditioner*). Selain itu, terdapat fitur *input text* untuk memasukkan *url ngrok* agar dapat mengakses server yang menyimpan logika *Apostolico Giancarlo* dan pengiriman pesan ke MQTT tanpa perlu *hosting*.



Gambar 3.5 Tampilan Utama dan Penggunaan *Speech Recognition*

Pada saat *button microphone* di tekan maka akan menampilkan fitur *speech* dari *Google* yang akan mengubah kalimat dari ucapan *user* menjadi teks, dapat dilihat pada Gambar 3.5. Ketika ucapan *user* menjadi teks, selanjutnya teks tersebut akan diproses oleh algoritma *Apostolico Giancarlo* untuk mencocokkan kalimat dengan *pattern* yang sudah ditentukan.

Pada tampilan Gambar 3.6 merupakan akhir proses algoritma *Apostolico Giancarlo* yang telah menemukan *pattern* pada kalimat ucapan dari *user* dan akan menghasilkan *output*, pada contoh kalimat “AC dihidupkan” dan “AC dimatikan”. Selanjutnya, dengan mengirimkan pesan dari teks *output* ke MQTT untuk diproses pada AC (*Air Conditioner*).



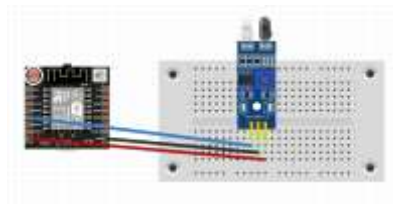
Gambar 3.6 Halaman ketika AC Dihidupkan dan AC Dimatikan

Pada Gambar 3.6 merupakan halaman yang berfungsi untuk tampilan ketika AC (*Air Conditioner*) menggunakan fitur *microphone* atau *button*.

3.3 Rangkaian Komponen

Rangkaian komponen elektronik ini yang digunakan oleh peneliti, antara lain: Modul ESP32, Sensor *Infrared Receiver & Transmit*, *Breadboard*.

Rangkaian komponen dapat dilihat pada Gambar 3.7.



Gambar 3.7 Rangkaian Komponen

Pada rangkaian komponen ini terdapat *Breadboard* berguna untuk mengalirkan arus listrik dan penghubung antara mikrokontroller dan sensor *Infrared Receiver & Transmitter*. Pada mikrokontroller Modul ESP32 disini berguna untuk mengendalikan berbagai peralatan komponen elektronik dan sangat cocok digunakan pada proyek yang berbasis *Internet of Things* karena didalamnya terdapat Wi-Fi dan Bluetooth. Dan juga, mikrokontroller Modul ESP32 ini dapat digunakan untuk membaca data dari sensor, sering digunakan dalam berbagai proyek seperti *smart home*. Kemudian, pada komponen elektronik sensor *Infrared Receiver & Transmitter* berguna untuk mengirim dan menerima sinyal inframerah pada AC.

3.4 Analisa Aksesibilitas

Pada analisa aksesibilitas disini menjabarkan bagaimana hasil dari 45 responden tersebut dapat dirata-ratakan pilihannya. Dengan skala yang digunakan 1-5 yaitu sangat tidak setuju sampai sangat setuju. Hal ini dianalisa melalui rumus rata – rata, yaitu dapat dilihat pada rumus dibawah ini.

$$\text{Rata – rata} = \frac{(\text{skala nilai} \times \text{pilihan responden}) + \dots + (\text{skala nilai} \times \text{pilihan responden})}{45} \quad (3.1)$$

Pada rumus diatas terdapat pencarian rata – rata yang dipilih oleh responden menunjukan nilai skala yang dipilih. Oleh karena itu, rumus didalamnya menunjukan skala nilai dikalikan dengan nilai pilihan dari responden dan ditambahkan dengan skala nilai selanjutnya yang dipilih oleh responden sampai skala nilai 5 atau terakhir yang dipilih oleh responden, kemudian dibagi dengan jumlah banyaknya responden yang memilih yaitu sebanyak 45 responden.

BAB IV

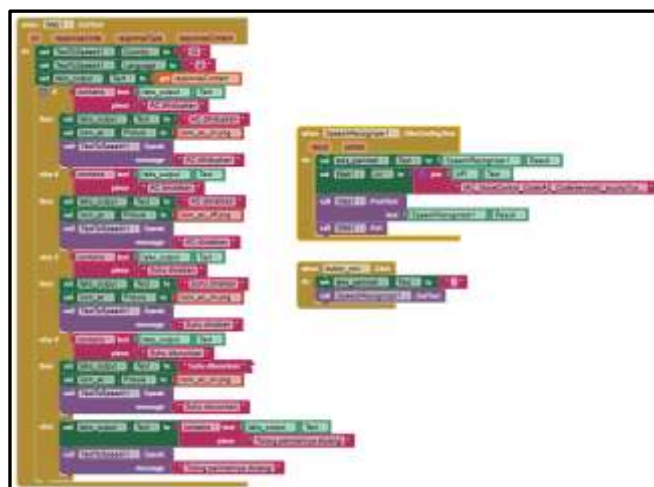
UJI COBA DAN PEMBAHASAN

4.1 Uji Coba

Pada tahapan uji coba ini merupakan hasil aplikasi dari perancangan yang telah dibuat. Melalui aplikasi sebagai wadah sistem uji yang siap digunakan. Untuk server aplikasi yang diuji agar terhubung dengan broker MQTT menggunakan *platform ngrok* yang dapat diakses secara gratis.

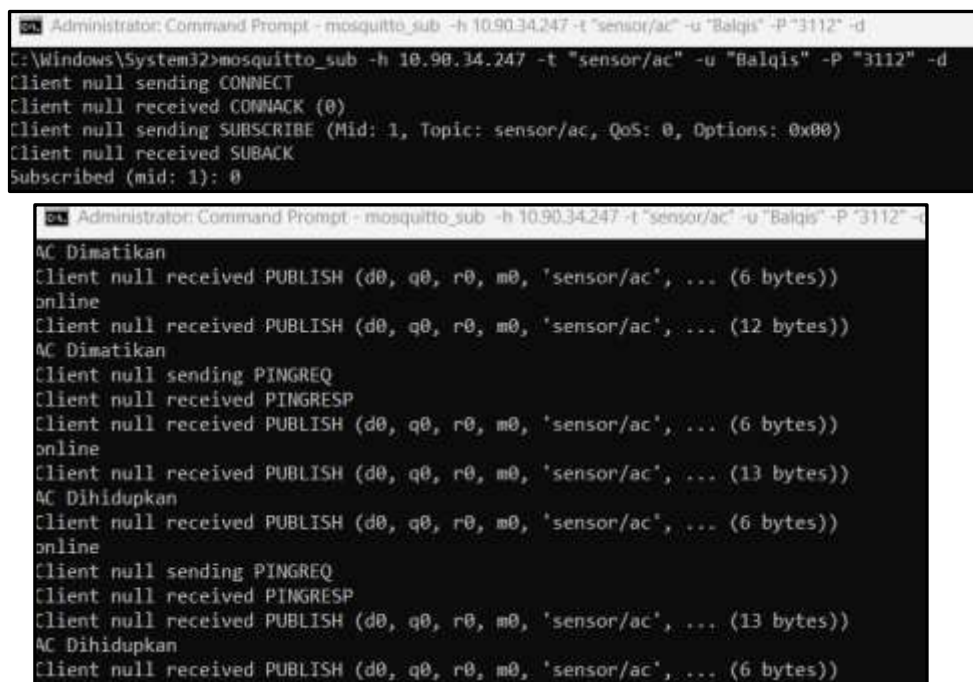
4.1.1 Kendali menggunakan *Microphone*

Pada perancangan kendali penggunaan fitur *Microphone* menggunakan aplikasi MIT App Inventor untuk memasukan perintah suara dari suara pengguna, didalamnya terdiri dari beberapa fungsi, yaitu *button_mic*, *Web1*, dan *SpeechRecognizer1*. Aplikasi ini juga berfungsi sebagai wadah aplikasi pembantu dan mempercepat pengerjaan untuk aplikasi *speech recognition*. Berikut Gambar 4.1 proses dari kendali *Microphone* pada MIT App Inventor.



Gambar 4.1 Logika *button_mic*

Urutan logika diawali dengan klik *button mic*, selanjutnya memanggil fitur *Google Speech* untuk mendeteksi kalimat yang diucapkan oleh *user*. Setelah itu, memanggil fungsi *text to speech* dari MIT App Inventor untuk mengubah kalimat yang diucapkan menjadi teks. Setelah mengeluarkan teks pada keterangan teks_perintah lalu mencari *pattern* dari *code Apostolico Giancarlo* melalui *url API* pada teks untuk *output AC (Air Conditioner)* apakah perintah yang diucapkan oleh *user* sesuai atau tidak sesuai.



```
Administrator: Command Prompt - mosquitto_sub -h 10.90.34.247 -t "sensor/ac" -u "Balqis" -P "3112" -d
C:\Windows\System32>mosquitto_sub -h 10.90.34.247 -t "sensor/ac" -u "Balqis" -P "3112" -d
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending SUBSCRIBE (Mid: 1, Topic: sensor/ac, QoS: 0, Options: 0x00)
Client null received SUBACK
Subscribed (mid: 1): 0

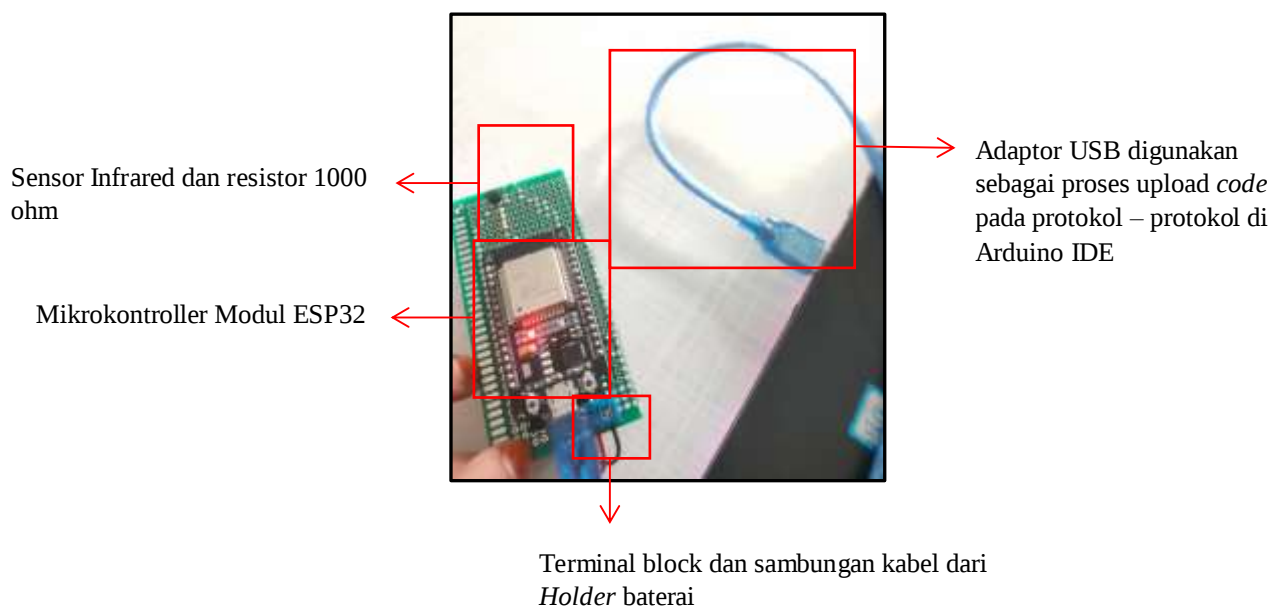
Administrator: Command Prompt - mosquitto_sub -h 10.90.34.247 -t "sensor/ac" -u "Balqis" -P "3112" -d
AC Dimatikan
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (6 bytes))
online
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (12 bytes))
AC Dimatikan
Client null sending PINGREQ
Client null received PINGRESP
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (6 bytes))
online
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (13 bytes))
AC Dihidupkan
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (6 bytes))
online
Client null sending PINGREQ
Client null received PINGRESP
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (13 bytes))
AC Dihidupkan
Client null received PUBLISH (d0, q0, r0, m0, 'sensor/ac', ... (6 bytes))
```

Gambar 4. 2 Output dari Logika *Button Mic*

Output Gambar 4.2 didalamnya memiliki informasi bahwa pada *button mic* dengan *id client button-mic* berhasil *publish* dengan pesan yang diucapkan oleh *user* ke *subscribe* dengan topik yang sesuai. Didalamnya juga berisi *username* dan *password* yang benar agar hanya pengguna yang hanya dapat menggunakan kendali AC (*Air Conditioner*).

4.1.2 Rangkaian *Hardware*

Rangkaian pada penelitian ini menggunakan Modul ESP32 sebagai mikrokontroller untuk menjalankan AC (*Air Conditioner*) dari perintah yang terkoneksi dengan sensor Infrared dan protokol – protokol yang terdapat pada Arduino IDE. Pada saat *user* menggunakan *button mic*, hasil suara yang masuk akan *diconvert* menjadi teks. Kemudian, teks tersebut akan dikirim ke server untuk mengetahui kesesuaian *pattern*. Apabila sudah sesuai maka akan dikirim perintahnya ke Modul ESP32 menggunakan MQTT dan mikrokontroller tersebut akan menjalankan perintah dari MQTT ke sinyal sensor Infrared. *Output* dapat dilihat dari AC (*Air Conditioner*) sesuai perintah yang tampil pada pesan MQTT. Berikut gambar rangkaian komponennya. Berikut Gambar 4.3 merupakan rangkaian komponennya.



Gambar 4. 3 Rangkaian Komponen *Hardware*

Pada rangkaian komponen diatas merupakan rangkaian dalam pemrosesan upload *code* di Arduino IDE. Setelah terupload dengan baik, maka kabel adaptor USB dapat dilepas dan terganti dengan *holder* baterai, hal ini digunakan agar jarak AC dengan sensor Infrared tidak jauh dan bisa diarahkan.

4.1.3 Rencana Uji Coba

Pengujian sistem dilakukan untuk memperoleh hasil pada sistem yang dibangun *error* atau tidak *error*, untuk mengukur algoritma *Apostolico Giancarlo* perlu digunakan rumus perhitungan *error*. Pada rumus *standard error* (Yusniyanti and Kurniati, 2017):

$$\text{Standard Error} = \frac{\sigma}{\sqrt{n}} \quad (4.2)$$

Keterangan:

σ adalah standard deviasi

n adalah jumlah uji coba

Untuk mengetahui standar deviasi maka digunakan rumus:

$$\mu = \frac{\sum x}{N} \quad (4.3)$$

Keterangan:

μ adalah mean (rata – rata)

$\sum x$ adalah jumlah dari data yang dihasilkan

N adalah banyaknya uji coba

Setelah itu menggunakan rumus standar deviasi:

$$\sigma = \sqrt{\frac{\sum (x - \mu)^2}{N}} \quad (4.4)$$

Keterangan:
 X adalah hasil nilai data

4.2 Pembahasan

Pada tahapan pembahasan ini akan menjelaskan pemrosesan *table bad*, pengujian eror algoritma *Apostolico Giancarlo* dan *usability* dari perintah yang diuji.

4.2.1 Pengujian Sistem

Pengujian sistem dilakukan menggunakan rumus perhitungan *error* pada rumus *standard error*. Tujuan dilakukan perhitungan eror dengan rumus *standard error* bisa digunakan untuk mengukur akurasi sistem dalam kendali kontrol perintah suara terhadap input suara yang diberikan.

4.2.2 Proses *Table Bad Match*

Pada proses *table bad match* berfungsi untuk mencari nilai dari setiap karakter pada *pattern*. Fungsi ini dapat diketahui dari jumlah pergeseran dari kiri ke kanan sesuai dengan nilai karakter *pattern*, sehingga dapat ditemukan pola pada suatu teks.

Untuk proses *table bad match* ini dimulai dari menentukan *index* atau nilai pada *pattern* dan menghitung panjang *pattern*. Kemudian, dihitung nilai karakter *bad match* menggunakan rumus. Berikut penjelasan mengenai proses *table bad match*.

Pattern: hidup ac

Panjang *pattern*: 8

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.1.

Tabel 4. 1 *Bad Match Pattern Pertama*

Karakter	Posisi terakhir dalam pola	Pergeseran
h	0	$8 - 1 - 0 = 7$
i	1	$8 - 1 - 1 = 6$
d	2	$8 - 1 - 2 = 5$
u	3	$8 - 1 - 3 = 4$
p	4	$8 - 1 - 4 = 3$
(spasi)	5	$8 - 1 - 5 = 2$
a	6	$8 - 1 - 6 = 1$
c	7	$8 - 1 - 7 = 0$

Pattern: ac hidup

Panjang *pattern*: 8

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.2.

Tabel 4. 2 *Bad Match Pattern Kedua*

Karakter	Posisi terakhir dalam pola	Pergeseran
a	0	$8 - 1 - 0 = 7$
c	1	$8 - 1 - 1 = 6$
(spasi)	2	$8 - 1 - 2 = 5$
h	3	$8 - 1 - 3 = 4$
i	4	$8 - 1 - 4 = 3$
d	5	$8 - 1 - 5 = 2$
u	6	$8 - 1 - 6 = 1$
p	7	$8 - 1 - 7 = 0$

Pattern: mati ac

Panjang pattern: 7

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.3.

Tabel 4. 3 *Bad Match Pattern* Ketiga

Karakter	Posisi terakhir dalam pola	Pergeseran
m	0	$7 - 1 - 0 = 6$
a	1	$7 - 1 - 1 = 5$
t	2	$7 - 1 - 2 = 4$
i	3	$7 - 1 - 3 = 3$
(spasi)	4	$7 - 1 - 4 = 2$
a	5	$7 - 1 - 5 = 1$
c	6	$7 - 1 - 6 = 0$

Pattern: ac mati

Panjang pattern: 7

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.4.

Tabel 4. 4 *Bad Match Pattern* Keempat

Karakter	Posisi terakhir dalam pola	Pergeseran
a	0	$7 - 1 - 0 = 6$
c	1	$7 - 1 - 1 = 5$
(spasi)	2	$7 - 1 - 2 = 4$
m	3	$7 - 1 - 3 = 3$
a	4	$7 - 1 - 4 = 2$
t	5	$7 - 1 - 5 = 1$
i	6	$7 - 1 - 6 = 0$

Pattern: ac nyala

Panjang pattern: 8

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.5.

Tabel 4. 5 *Bad Match Pattern* Kelima

Karakter	Posisi terakhir dalam pola	Pergeseran
a	0	$8 - 1 - 0 = 7$
c	1	$8 - 1 - 1 = 6$
(spasi)	2	$8 - 1 - 2 = 5$
n	3	$8 - 1 - 3 = 4$
y	4	$8 - 1 - 4 = 3$
a	5	$8 - 1 - 5 = 2$
l	6	$8 - 1 - 6 = 1$
a	7	$8 - 1 - 7 = 0$

Pattern:nyala ac

Panjang pattern: 8

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.6.

Tabel 4. 6 *Bad Match Pattern* Keenam

Karakter	Posisi terakhir dalam pola	Pergeseran
n	0	$8 - 1 - 0 = 7$
y	1	$8 - 1 - 1 = 6$
a	2	$8 - 1 - 2 = 5$
l	3	$8 - 1 - 3 = 4$
a	4	$8 - 1 - 4 = 3$
(spasi)	5	$8 - 1 - 5 = 2$
a	6	$8 - 1 - 6 = 1$
c	7	$8 - 1 - 7 = 0$

Pattern: naik suhu

Panjang pattern: 9

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.7.

Tabel 4. 7 *Bad Match Pattern* Ketujuh

Karakter	Posisi terakhir dalam pola	Pergeseran
n	0	$9 - 1 - 0 = 8$
a	1	$9 - 1 - 1 = 7$
i	2	$9 - 1 - 2 = 6$
k	3	$9 - 1 - 3 = 5$
(spasi)	4	$9 - 1 - 4 = 4$
s	5	$9 - 1 - 5 = 3$
u	6	$9 - 1 - 6 = 2$
h	7	$9 - 1 - 7 = 1$
u	8	$9 - 1 - 8 = 0$

Pattern: suhu naik

Panjang pattern: 9

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.8.

Tabel 4. 8 *Bad Match Pattern* Kedelapan

Karakter	Posisi terakhir dalam pola	Pergeseran
s	0	$9 - 1 - 0 = 8$
u	1	$9 - 1 - 1 = 7$
h	2	$9 - 1 - 2 = 6$
u	3	$9 - 1 - 3 = 5$
(spasi)	4	$9 - 1 - 4 = 4$
n	5	$9 - 1 - 5 = 3$
a	6	$9 - 1 - 6 = 2$
i	7	$9 - 1 - 7 = 1$
k	8	$9 - 1 - 8 = 0$

Pattern: turun suhu

Panjang pattern: 10

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.9.

Tabel 4. 9 *Bad Match Pattern* Kesembilan

Karakter	Posisi terakhir dalam pola	Pergeseran
t	0	$10 - 1 - 0 = 9$
u	1	$10 - 1 - 1 = 8$
r	2	$10 - 1 - 2 = 7$
u	3	$10 - 1 - 3 = 6$
n	4	$10 - 1 - 4 = 5$
(spasi)	5	$10 - 1 - 5 = 4$
s	6	$10 - 1 - 6 = 3$
u	7	$10 - 1 - 7 = 2$
h	8	$10 - 1 - 8 = 1$
u	9	$10 - 1 - 9 = 0$

Pattern: suhu turun

Panjang pattern: 10

Rumus: $m - 1 - i$

Dengan keterangan bahwa m adalah panjang *pattern* dan i adalah indeks karakter dalam pola. Tabel pergeseran *bad character* dari penjabaran diatas, dapat dilihat pada Tabel 4.10.

Tabel 4. 10 *Bad Match Pattern* Kesepuluh

Karakter	Posisi terakhir dalam pola	Pergeseran
s	0	$10 - 1 - 0 = 9$
u	1	$10 - 1 - 1 = 8$
h	2	$10 - 1 - 2 = 7$
u	3	$10 - 1 - 3 = 6$
(spasi)	4	$10 - 1 - 4 = 5$
t	5	$10 - 1 - 5 = 4$
u	6	$10 - 1 - 6 = 3$
r	7	$10 - 1 - 7 = 2$
u	8	$10 - 1 - 8 = 1$

Karakter	Posisi terakhir dalam pola	Pergeseran
n	9	$10 - 1 - 9 = 0$

4.2.3 Pengujian Perintah Suara

Algoritma *Apostolico Giancarlo* pada penelitian ini diimplementasikan pada pengujian kalimat perintah yaitu memiliki kata yang dicari sesuai pola *pattern* untuk dicocokkan. Pada pengujian penelitian ini menggunakan 100 kalimat perintah dengan 10 *pattern*. Berikut beberapa hasil pengujian dimulai dari kata “Tolong hidup AC-nya” pada Tabel 4.11.

Tabel 4. 11 Hasil Pengujian Kalimat pada *Pattern* Pertama

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong hidup AC-nya	hidup ac	1	0
Perintah 2	Bisakah kamu hidupkan AC	hidup ac	1	0
Perintah 3	Hidupkan AC-nya ya	hidup ac	1	0
Perintah 4	Bisa gak si hidupkan AC	hidup ac	1	0
Perintah 5	Ruangannya panas, hidupin AC	hidup ac	1	1
Perintah 6	Ini bisa nggak hidupkan AC	hidup ac	1	1
Perintah 7	Aku kepanasan hidupkan AC	hidup ac	1	0
Perintah 8	Tolong dong hidupkan AC-nya lagi sumuk	hidup ac	1	0
Perintah 9	Hidupin AC, gerah disini	hidup ac	1	1
Perintah 10	Mohon tolong hidup AC-nya yaa	hidup ac	1	0

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan hidup ac, memiliki jumlah *error* 3 dan perintah bisa dijalankan sebanyak 7 kali.

Tabel 4. 12 Hasil Pengujian Kalimat pada *Pattern* Kedua

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong dong AC hidupkan	ac hidup	2	0
Perintah 2	Bisakah kamu AC-nya hidup	ac hidup	2	0
Perintah 3	AC dihidupkan , bisa nggak	ac hidup	2	0
Perintah 4	AC-nya hidupin ya	ac hidup	2	1
Perintah 5	Aku sumuk, AC hidupkan	ac hidup	2	0

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 6	AC hidupkan yaa	ac hidup	2	0
Perintah 7	Mohon tolong AC dihidupkan	ac hidup	2	0
Perintah 8	Coba AC-nya hidupkan	ac hidup	2	1
Perintah 9	AC-nya tolong hidupin ya	ac hidup	2	1
Perintah 10	AC aku gerah, hidupkan ya	ac hidup	2	1

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac hidup, memiliki jumlah *error* 4 dan perintah bisa dijalankan sebanyak 6. Dapat dilihat pada Tabel 4.12.

Tabel 4. 13 Hasil Pengujian Kalimat pada *Pattern* Ketiga

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong dimatikan AC-nya ya	mati ac	3	0
Perintah 2	Bisakah matikan itu AC-nya, aku kedinginan	mati ac	3	1
Perintah 3	Aku dingin, tolong matikan AC ya	mati ac	3	0
Perintah 4	Matiin AC-nya	mati ac	3	1
Perintah 5	Mohon tolong apabisa matikan AC-nya	mati ac	3	0
Perintah 6	Apakah kamu bisa matikan AC-nya	mati ac	3	0
Perintah 7	Coba matikan AC-nya	mati ac	3	0
Perintah 8	Aku lagi kedinginan, tolong banget matikan AC-nya itu	mati ac	3	0
Perintah 9	Nggak bisa apa ya, dimatikan AC-nya	mati ac	3	1
Perintah 10	Dia lagi kedinginan, tolong ya dimatikan AC-nya	mati ac	3	1

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan mati ac, memiliki jumlah *error* 4 dan perintah bisa dijalankan sebanyak 6. Dapat dilihat pada Tabel 4.13.

Tabel 4. 14 Hasil Pengujian Kalimat pada *Pattern* Keempat

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong AC matikan ya	ac mati	4	0
Perintah 2	Bisakah AC-nya matikan	ac mati	4	0
Perintah 3	AC-nya dimatikan apabisa	ac mati	4	0
Perintah 4	AC matiin ya	ac mati	4	1
Perintah 5	AC-nya matikan, aku kedinginan	ac mati	4	0

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 6	Aku lagi kedinginan, AC-nya dimatikan ya	ac mati	4	0
Perintah 7	Tolong AC dimatikan	ac mati	4	0
Perintah 8	Mohon tolong AC dimatikan	ac mati	4	0
Perintah 9	AC matikan, dia lagi kedinginan	ac mati	4	
Perintah 10	Bisa nggak AC-nya dimatikan, ruangnya dingin banget	ac mati	4	1

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac mati, memiliki jumlah *error* 2 dan perintah bisa dijalankan sebanyak 8. Dapat dilihat pada Tabel 4.14.

Tabel 4. 15 Hasil Pengujian Kalimat pada *Pattern* Kelima

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	AC-nya tolong nyala ya	ac nyala	5	0
Perintah 2	AC-nya tolong nyalakan ya	ac nyala	5	0
Perintah 3	Mohon tolong AC-nya dinyalakan ya	ac nyala	5	0
Perintah 4	AC-nya nyalain, aku sumuk	ac nyala	5	1
Perintah 5	AC nyalakan yaa	ac nyala	5	0
Perintah 6	AC-nya dinyalakan, ini ruangnya panas	ac nyala	5	0
Perintah 7	AC-nya apabisa nyalakan ya	ac nyala	5	1
Perintah 8	Bisakaha AC nyalakan	ac nyala	5	0
Perintah 9	Apakah AC bisa dinyalakan	ac nyala	5	1
Perintah 10	AC nyalain	ac nyala	5	1

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 4 dan perintah bisa dijalankan sebanyak 6. Dapat dilihat pada Tabel 4.15.

Tabel 4. 16 Hasil Pengujian Kalimat pada *Pattern* Keenam

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong nyalakan AC-nya	nyala ac	6	0
Perintah 2	Nyalakan AC-nya yaa	nyala ac	6	0
Perintah 3	Aku lagi sumuk nih, tolong dinyalakan AC-nya	nyala ac	6	0
Perintah 4	Nyalakan AC-nya, dia gerah	nyala ac	6	0
Perintah 5	Mohon tolong nyalakan itu AC-	nyala ac	6	0

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
	nya			
Perintah 6	Nyalain tolong AC itu	nyala ac	6	1
Perintah 7	Nyalakan AC sekarang	nyala ac	6	0
Perintah 8	Aku mau nyala AC-nya	nyala ac	6	0
Perintah 9	Bisakah nyalakan AC buat saya	nyala ac	6	0
Perintah 10	Ruangan ini sangat gerah, tolong dinyalain AC-nya	nyala ac	6	1

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 2 dan perintah bisa dijalankan sebanyak 8. Dapat dilihat pada Tabel 4.16.

Tabel 4. 17 Hasil Pengujian Kalimat pada *Pattern* Ketujuh

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Tolong naik suhunya	naik suhu	7	0
Perintah 2	Tolong naik suhu AC	naik suhu	7	0
Perintah 3	Bisakah dinaikan suhu AC-nya	naik suhu	7	0
Perintah 4	Naikan suhu AC-nya	naik suhu	7	0
Perintah 5	Naik suhu AC-nya agar ruangan ini tidak terlalu dingin	naik suhu	7	0
Perintah 6	Coba dinaikan suhunya	naik suhu	7	1
Perintah 7	Apakah bisa naik suhu AC-nya	naik suhu	7	0
Perintah 8	Ruangan ini sangat dingin, tolong naik suhunya	naik suhu	7	0
Perintah 9	Aku butuh dinaikan suhunya, biar nggak dingin	naik suhu	7	0
Perintah 10	Naik suhunya segera	naik suhu	7	0

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 1 dan perintah bisa dijalankan sebanyak 9. Dapat dilihat pada Tabel 4.17.

Tabel 4. 18 Hasil Pengujian Kalimat pada *Pattern* Kedelapan

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Suhu naikan ya	suhu naik	8	0
Perintah 2	Suhu naik sekarang	suhu naik	8	0
Perintah 3	Bisakah suhu dinaikan	suhu naik	8	0
Perintah 4	Saya ingin suhu AC-nya dinaikan	suhu naik	8	1

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 5	Tolong suhu naikan	suhu naik	8	0
Perintah 6	Saya ingin suhunya dinaikan saat ini	suhu naik	8	0
Perintah 7	Aku ingin suhunya dinaikan agar ruangan tidak terlalu dingin	suhu naik	8	0
Perintah 8	Suhu dinaikan sekarang, jangan lama-lama	suhu naik	8	0
Perintah 9	Aku kedinginan mohon tolong suhunya dinaikan	suhu naik	8	0
Perintah 10	Tolong suhu dinaikan ya	suhu naik	8	0

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 1 dan perintah bisa dijalankan sebanyak 9. Dapat dilihat pada Tabel 4.18.

Tabel 4. 19 Hasil Pengujian Kalimat pada *Pattern* Kesembilan

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Turun suhunya	turun suhu	9	0
Perintah 2	Turun suhu sekarang	turun suhu	9	0
Perintah 3	Saya ingin AC-nya ini ya diturunkan suhunya	turun suhu	9	1
Perintah 4	Ruangannya gerah, tolong turunkan suhu AC-nya	turun suhu	9	0
Perintah 5	Coba turunkan suhunya ya	turun suhu	9	0
Perintah 6	Tolong turun suhu AC-nya supaya nggak gerah	turun suhu	9	0
Perintah 7	Aku mau diturunkan suhunya biar ruangannya lebih sejuk	turun suhu	9	0
Perintah 8	Turuh suhu sekarang, jangan lama – lama	turun suhu	9	0
Perintah 9	Bisakah kamu turunkan suhunya	turun suhu	9	0
Perintah 10	Mohon tolong turunkan suhu	turun suhu	9	0

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 1 dan perintah bisa dijalankan sebanyak 9. Dapat dilihat pada Tabel 4.19.

Tabel 4. 20 Hasil Pengujian Kalimat pada *Pattern* Kesepuluh

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 1	Mohon tolong suhu turunkan	suhu turun	10	0

No	Kalimat	Pattern	Pattern ditemukan pada iterasi	Error
Perintah 2	Suhu turunkan sekarang	suhu turun	10	0
Perintah 3	Bisakah suhunya diturunkan saat ini	suhu turun	10	0
Perintah 4	Saya ingin suhunya diturunkan	suhu turun	10	0
Perintah 5	Saya kegerahan tolong suhu AC-nya diturunkan sekarang	suhu turun	10	1
Perintah 6	Coba tolong suhunya diturunkan	suhu turun	10	0
Perintah 7	Saya ingin ruangan lebih dingin, tolong ya suhunya diturunkan	suhu turun	10	0
Perintah 8	Cuacanya lagi panas, tolong dong suhu AC-nya turunkan	suhu turun	10	1
Perintah 9	Suhunya mohon tolong diturunkan ya	suhu turun	10	0
Perintah 10	Sekarang ini suhunya turunkan ya	suhu turun	10	0

Berdasarkan uji coba kalimat sebanyak 10 kali hasilnya sesuai dengan perintah yaitu pada *pattern* yang ditentukan ac nyala, memiliki jumlah *error* 2 dan perintah bisa dijalankan sebanyak 8. Dapat dilihat pada Tabel 4.20.

Setelah melakukan uji coba kalimat perintah suara dari setiap *pattern*, maka langkah selanjutnya adalah menghitung *Standard Error*. Berikut perhitungan *Standard Error*:

Untuk mengetahui *Standard Error* dapat dilakukan perhitungan rata – rata terlebih dahulu,

$$\mu = \frac{\sum x}{N} = \frac{24}{100} = 0,24 \quad (4.4)$$

Pada persamaan 4.4 jumlah *error* sebanyak 24 dibagi dengan banyaknya uji coba perintah suara dari setiap *pattern* sebanyak 100 percobaan, dengan menghasilkan rata – rata nilai *error* uji coba 0,24.

$$\begin{aligned}
 \sigma &= \sqrt{\frac{\sum(x - \mu)^2}{N}} \quad (5) \\
 &= \sqrt{\frac{(2,76)^2 + (3,76)^2 + (3,76)^2 + (1,76)^2 + (3,76)^2 + (1,76)^2 + (0,76)^2 + (1,76)^2 + (0,76)^2 + (1,76)^2}{100}} \\
 &= \sqrt{\frac{7,6176 + 14,1376 + 14,1376 + 3,0976 + 14,1376 + 3,0976 + 0,5776 + 3,0976 + 0,5776 + 3,0976}{100}} \\
 &= \sqrt{\frac{7,6176 + 14,1376 + 14,1376 + 3,0976 + 14,1376 + 3,0976 + 0,5776 + 3,0976 + 0,5776 + 3,0976}{100}} \\
 &= \sqrt{\frac{63,576}{100}} = \sqrt{0,63576} = 0,8
 \end{aligned}$$

Kemudian setelah menghitung rata – rata dari uji coba seriap *pattern*-nya, maka langkah selanjutnya menghitung standard deviasi dengan memasukkan rumus akar sigma, nilai dari data dikurangi dengan nilai rata – rata yang dihasilkan yaitu 0,24 kemudian dikuadratkan dan dibagi dengan banyaknya uji coba yaitu 100. Dengan hasil bahwa standard deviasi memiliki nilai 0,8. Perhitungan dapat dilihat pada persamaan 4.5

Setelah menghitung standard deviasi, maka langkah terakhir yaitu melakukan perhitungan *standard error* dengan rumusnya yaitu standard deviasi dibagi dengan akar dari banyaknya uji coba. Yang artinya, 0,8 dibagi dengan akar dari 100, menghasilkan nilai *standard error* 0,08. Perhitungan rumus dapat dilihat pada persamaan 4.6.

$$Standard\ Error = \frac{\sigma}{\sqrt{n}} \quad (4.6)$$

$$\text{Standard Error} = \frac{0,8}{\sqrt{100}} = 0,08$$

Dari hasil *error* yang telah dihitung pada persamaan 4.6, menyimpulkan bahwa penelitian ini pada saat pengujian 100 kalimat dengan 10 *pattern* memiliki standard *error* yaitu 0,08.

4.2.4 Aksesibilitas

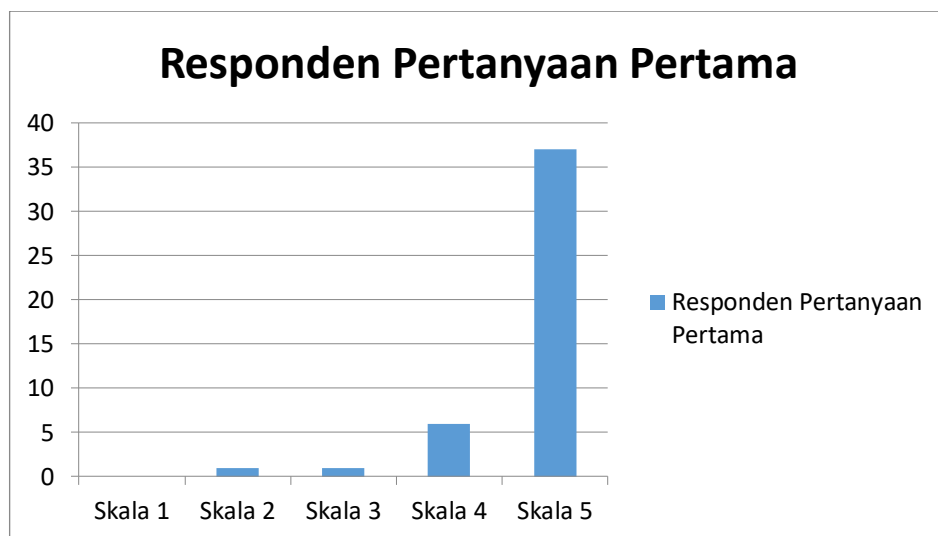
Setelah melakukan perhitungan standard *error* maka dilakukan analisa aksesibilitas sistem penelitian ini. Aksesibilitas merupakan salah satu aspek yang sangat penting dalam pengembangan sistem teknologi, khususnya untuk *user* dengan disabilitas. Dalam penelitian ini, peneliti menguji aksesibilitas sistem kendali AC berbasis perintah suara yang ditujukan untuk dapat dioperasikan oleh individu dengan keterbatasan penglihatan.

Sebagai bagian dari uji coba aksesibilitas, peneliti menguji 45 responden diminta untuk mengoperasikan penelitian ini untuk menguji apakah sistem dapat diakses dengan mudah tanpa menggunakan indera penglihatan dengan sepenuhnya. Berikut merupakan nama – nama responden yang berkontribusi dalam pengambilan uji coba aksesibilitas ini.

Responden akan menjawab 8 pertanyaan yang telah dicantumkan pada *Google Form* dengan skala linier dari 1 sampai 5 yaitu dari Sangat tidak setuju ke Sangat setuju. Berikut 8 pertanyaan yang diajukan kepada responden:

1. Apakah kamu bisa menyalakan atau mematikan AC tanpa melihat layar?

2. Apakah kamu bisa mengatur suhu AC tanpa melihat?
3. Apakah tombol-tombol pada sistem mudah ditemukan meskipun tanpa melihat?
4. Apakah kamu memahami respon atau umpan balik dari sistem?
5. Apakah kamu tetap bisa mengakses sistem meskipun tidak menyentuh layar secara langsung?
6. Apakah sistem ini tetap mudah digunakan meskipun kamu tidak mendapat penjelasan sebelumnya?
7. Apakah sistem bisa digunakan dengan nyaman oleh orang dengan keterbatasan atau disabilitas?
8. Apakah kamu merasa kesulitan saat menggunakan sistem dalam kondisi dengan mata yang tidak bisa melihat?



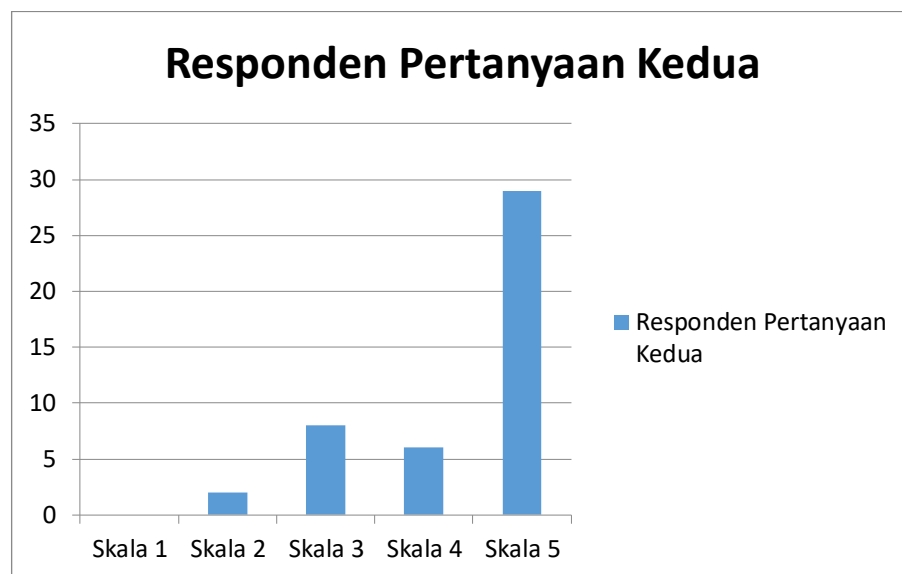
Gambar 4.4 Hasil Pertanyaan Pertama

Pada Gambar 4.4 yaitu hasil pertanyaan pertama, “Apakah kamu bisa menyalakan atau mematikan AC tanpa melihat layar?” , memiliki pilihan tidak

setuju dengan jumlah responden 1, pilihan netral dengan jumlah responden 1, pilihan setuju dengan jumlah responden 6 dan pilihan sangat setuju dengan jumlah responden 37. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(2 \times 1) + (3 \times 1) + (4 \times 6) + (5 \times 37)}{45} = 4,75$$

Pada perhitungan rata – rata menyatakan bahwa 4,75 lebih mendekati pilihan 5 (Sangat setuju), yang dapat disimpulkan responden banyak memilih sangat setuju, artinya bahwa sistem bisa menyalakan atau mematikan AC tanpa melihat layar.



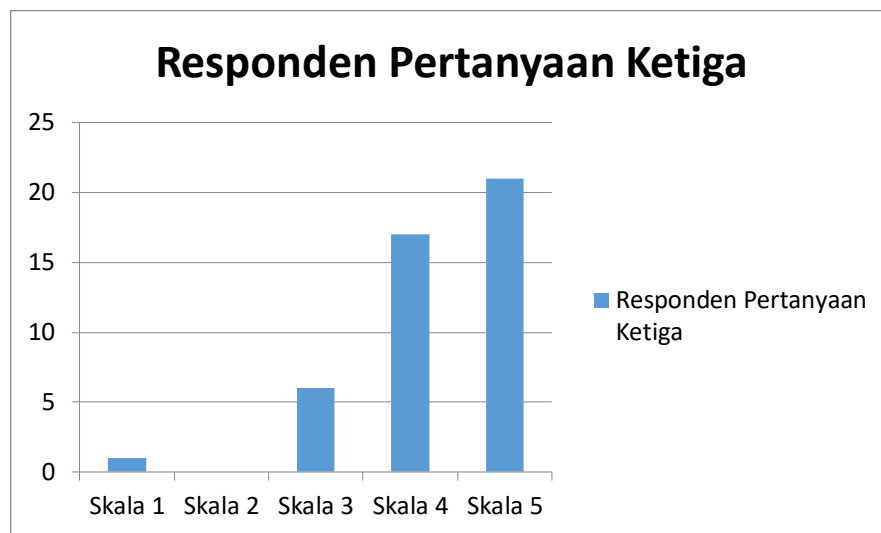
Gambar 4.5 Hasil Pertanyaan Kedua

Pada Gambar 4.5 yaitu hasil pertanyaan kedua, “Apakah kamu bisa mengatur suhu AC tanpa melihat?” , memiliki pilihan tidak setuju dengan jumlah responden 2, pilihan netral dengan jumlah responden 8, pilihan setuju dengan jumlah responden 6 dan pilihan sangat setuju dengan jumlah responden 29.

Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(2 \times 2) + (3 \times 8) + (4 \times 6) + (5 \times 29)}{45} = 4,37$$

Pada perhitungan rata – rata menyatakan bahwa 4,37 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa sistem bisa mengatur suhu AC tanpa melihat.

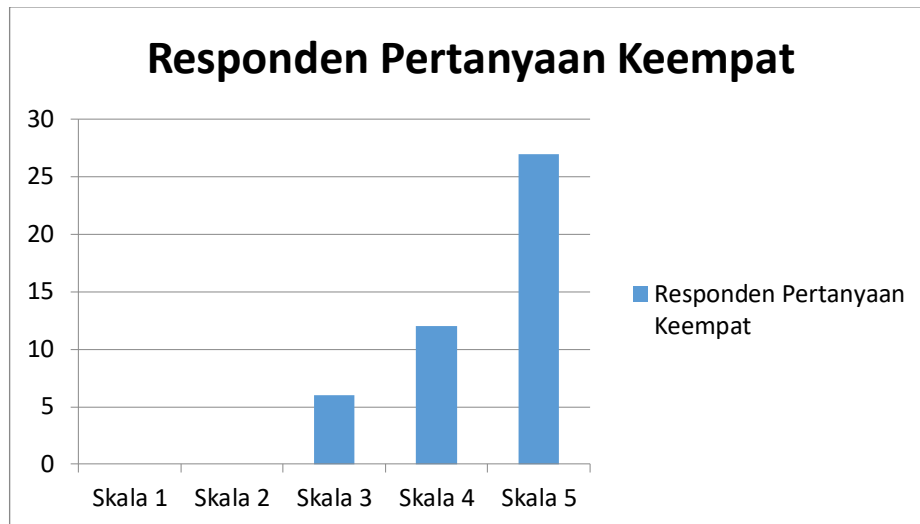


Gambar 4.6 Hasil Pertanyaan Ketiga

Pada Gambar 4.6 yaitu hasil pertanyaan ketiga, “Apakah tombol-tombol pada sistem mudah ditemukan meskipun tanpa melihat?” , memiliki pilihan sangat tidak setuju dengan jumlah responden 1, pilihan netral dengan jumlah responden 6, pilihan setuju dengan jumlah responden 17 dan pilihan sangat setuju dengan jumlah responden 21. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(1 \times 1) + (3 \times 6) + (4 \times 17) + (5 \times 21)}{45} = 4,26$$

Pada perhitungan rata – rata menyatakan bahwa 4,26 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa tombol - tombol sistem mudah ditemukan meskipun tanpa melihat.

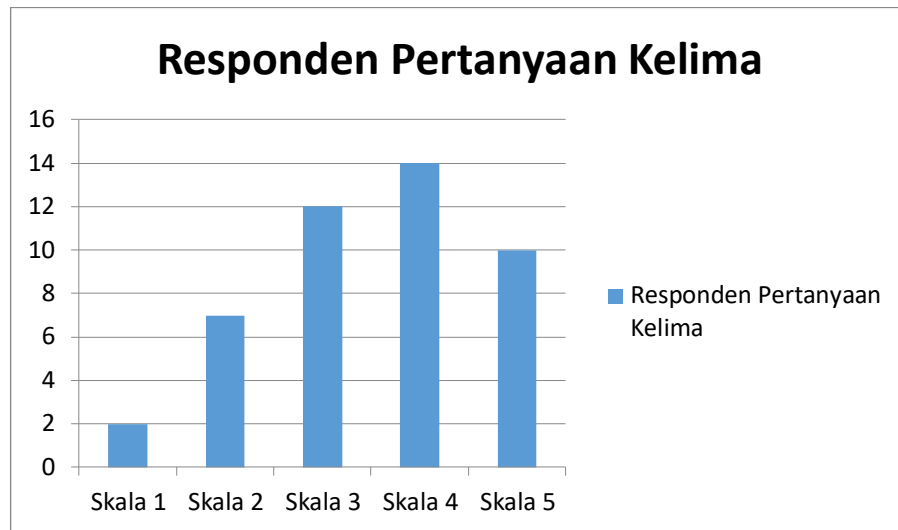


Gambar 4.7 Hasil Pertanyaan Keempat

Pada Gambar 4.7 yaitu hasil pertanyaan keempat, “Apakah kamu memahami respon atau umpan balik dari sistem?” , memiliki pilihan netral dengan jumlah responden 0, pilihan setuju dengan jumlah responden 12, pilihan sangat setuju dengan jumlah responden 27. Dengan hal ini bahwa, memiliki nilai rata – rata dalam dalam nomor rumor 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(3 \times 0) + (4 \times 12) + (5 \times 27)}{45} = 4,46$$

Pada perhitungan rata – rata menyatakan bahwa 4,46 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa sistem dapat dipahami oleh responden.

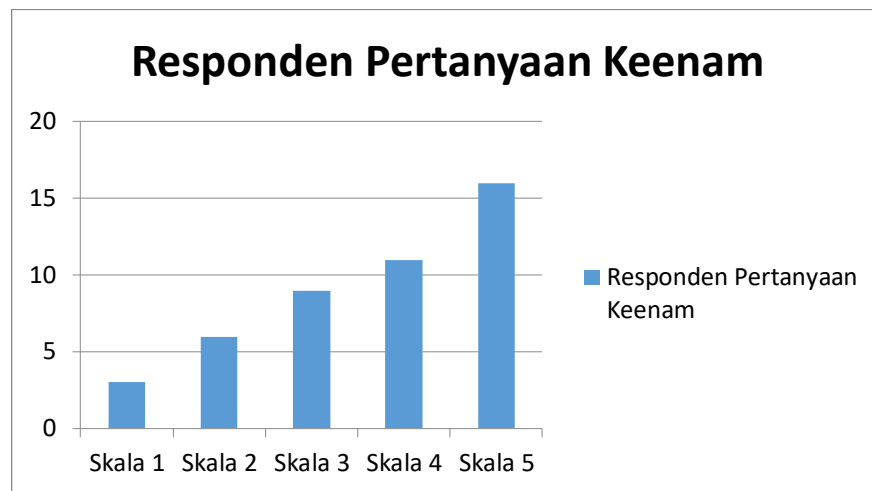


Gambar 4.8 Hasil Pertanyaan Kelima

Pada Gambar 4.8 yaitu pertanyaan kelima, “Apakah kamu tetap bisa mengakses sistem meskipun tidak menyentuh layar secara langsung?”, memiliki pilihan sangat tidak setuju dengan jumlah responden 2, pilihan tidak setuju dengan jumlah responden 7, pilihan netral dengan jumlah responden 12, pilihan setuju dengan jumlah responden 14 dan pilihan sangat setuju dengan jumlah responden 10. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumor 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(1 \times 2) + (2 \times 7) + (3 \times 12) + (4 \times 14) + (5 \times 10)}{45} = 3,51$$

Pada perhitungan rata – rata menyatakan bahwa 3,51 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa sistem tetap bisa mengakses meskipun tidak menyentuh layar secara langsung.

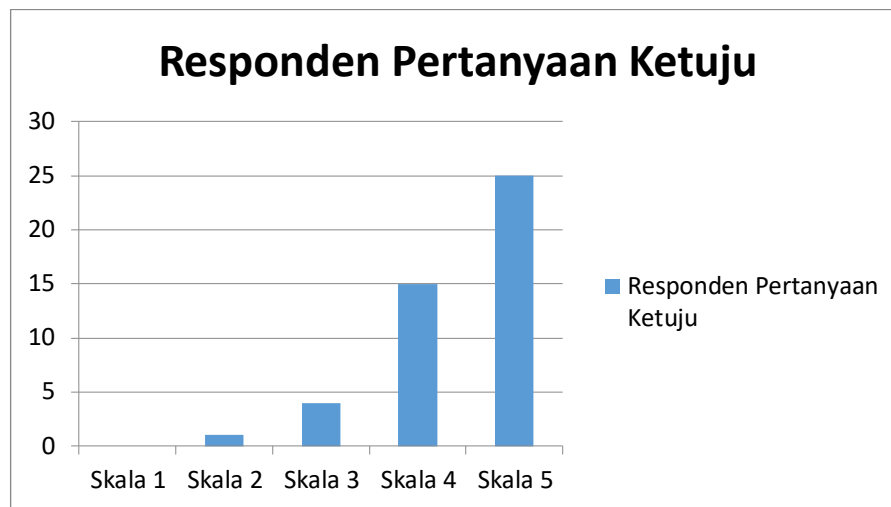


Gambar 4.9 Hasil Pertanyaan Keenam

Pada Gambar 4.9 yaitu hasil pertanyaan keenam, “Apakah sistem ini tetap mudah digunakan meskipun kamu tidak menyentuh layar secara langsung?”, memiliki pilihan sangat tidak setuju dengan jumlah responden 3, pilihan tidak setuju dengan jumlah responden 6, pilihan netral dengan jumlah responden 9, pilihan setuju dengan jumlah responden 11 dan pilihan sangat setuju dengan jumlah responden 16. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(1 \times 3) + (2 \times 6) + (3 \times 9) + (4 \times 11) + (5 \times 16)}{45} = 3,68$$

Pada perhitungan rata – rata menyatakan bahwa 3,68 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa sistem tetap mudah digunakan meskipun responden tidak menyentuh layar secara langsung.

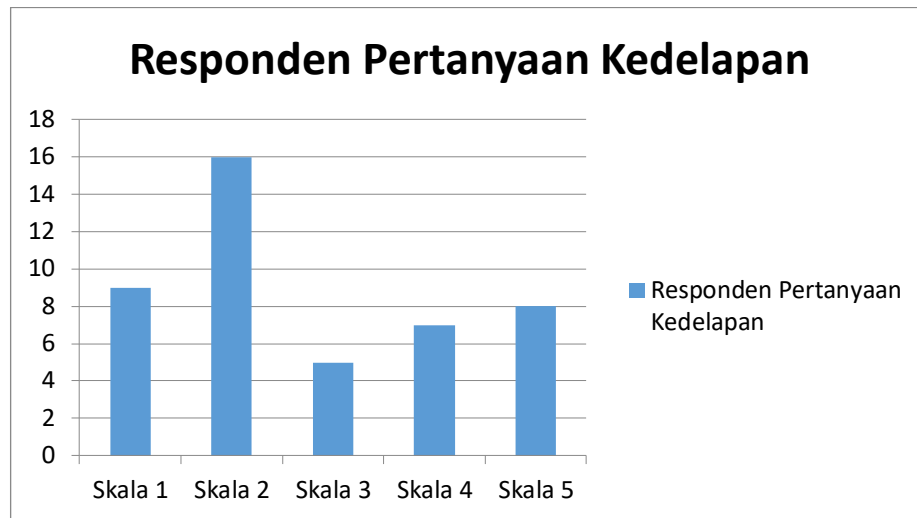


Gambar 4. 10 Hasil Pertanyaan Ketujuh

Pada Gambar 4.10 yaitu hasil pertanyaan ketujuh, “Apakah sistem bisa digunakan dengan nyaman oleh orang dengan keterbatasan atau disabilitas?”, memiliki pilihan tidak setuju dengan jumlah responden 1, pilihan netral dengan jumlah responden 4, pilihan setuju dengan jumlah responden 15, dan pilihan sangat setuju dengan jumlah responden 25. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(2 \times 1) + (3 \times 4) + (4 \times 15) + (5 \times 25)}{45} = 4,42$$

Pada perhitungan rata – rata menyatakan bahwa 4,42 lebih mendekati nilai skala 4 (Setuju), yang dapat disimpulkan responden banyak memilih setuju, artinya bahwa sistem bisa digunakan dengan nyaman oleh orang dengan keterbatasan atau disabilitas.



Gambar 4. 11 Hasil Pertanyaan Kedelapan

Pada Gambar 4.11 yaitu hasil pertanyaan kedelapan, “Apakah kamu merasa kesulitan saat menggunakan sistem dalam kondisi dengan mata yang tidak bisa melihat?”, memiliki pilihan sangat tidak setuju dengan jumlah responden 9, pilihan tidak setuju dengan jumlah responden 16, pilihan netral dengan jumlah responden 5, pilihan setuju dengan jumlah responden 7 dan pilihan sangat setuju dengan jumlah responden 8. Dengan hal ini bahwa, memiliki nilai rata – rata dalam nomor rumus 3.1 yang sudah dijelaskan pada BAB 3:

$$\text{Rata – rata} = \frac{(1 \times 9) + (2 \times 16) + (3 \times 5) + (4 \times 7) + (5 \times 8)}{45} = 2,75$$

Pada perhitungan rata – rata menyatakan bahwa 2,75 lebih mendekati nilai skala 3 (Netral), yang dapat disimpulkan responden banyak memilih netral, artinya bahwa responden ada yang merasa kesulitan dan ada yang merasa tidak kesulitan menggunakan sistem dalam kondisi dengan mata yang tidak bisa melihat.

Dengan hal ini, bahwa pada aksesibilitas yang dijawab oleh responden sebanyak 45 responden menghasilkan rata – rata pilihannya lebih menunjukan pada nilai skala yang keempat yaitu bahwa Setuju sistem tersebut dapat digunakan dengan memenuhi 7 pedoman standard komponen aksesibilitas yang sudah dijabarkan pada BAB 2.

4.3 Integrasi Islam

Keberhasilan sistem kendali *Air Conditioner* berbasis *Internet of Things* yang dapat dihubungkan dengan konsep kepemimpinan khalifah dalam Islam, yang mengutamakan pengelolaan dan pengendalian berbagai hal demi kepentingan umat. Seperti halnya Khalifah Umar bin Khattab yang berhasil mengendalikan dan mengelola sumber daya alam, seperti air dan irigasi, dengan bijaksana untuk memastikan kesejahteraan umat. Sistem kendali *Air Conditioner* yang telah dibuat juga berhasil mengendalikan suhu ruangan dengan efisien. Dengan bertujuan untuk memberikan kenyamanan, khususnya bagi penyandang disabilitas, agar lebih mudah mengakses dan mengendalikan teknologi tanpa kesulitan. Sebagaimana dalam Surah *Al – Baqarah* ayat 286 :

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

Ayat tersebut mengingatkan bahwa sebagai khalifah, dengan menciptakan sistem dan kondisi bertujuan untuk memudahkan umat, bukan membebani umat.

Allah *Subhanahu wa Ta'ala* memberikan manusia kemampuan berfikir dan akal kepada manusia sebagai ciptaan-Nya yang telah dijelaskan dalam Surah *Al - Baqarah* ayat 31 :

وَعَلَّمَ آدَمَ الْأَسْمَاءَ كُلَّهَا ثُمَّ عَرَضَهُمْ عَلَى الْمَلَائِكَةِ فَقَالَ أَنْبِئُونِي بِأَسْمَاءِ هَؤُلَاءِ إِنْ كُنْتُمْ صَادِقِينَ

“Dan Dia mengajarkan kepada Adam nama – nama (benda – benda) semuanya, kemudian Dia perlihatkan kepada para malaikat, seraya berfirman, “Sebutkan kepada-Ku nama semua (benda – benda) ini, jika kamu benar!”. (QS. *Al – Baqarah* : 31)

Pada surah *Al - Baqarah* ayat 31 mengandung keutamaan Adam atas malaikat karena Allah mengkhususkan baginya berupa ilmu tentang nama – nama segala sesuatu, sedangkan para malaikat diperintahkan untuk bersujud kepada Adam.

Ini adalah bagian ayat dimana Allah *Subhanahu wa Ta'ala* menyebutkan kelebihan nabi Adam atas para malaikat, keistimewaan ilmu tentang nama – nama dari segala sesuatu yang tidak diberikan kepada mereka (malaikat). Hal ini terjadi setelah mereka (malaikat) sujud kepadanya. Oleh karena itu, Allah menyebutkan ayat ini untuk menjelaskan kelebihan nabi Adam atas mereka dalam hal keilmuan. Allah *Subhanahu wa Ta'ala* berfirman, “Dan Dia mengajarkan kepada Adam nama – nama (benda – benda) seluruhnya.” (Tafsir Ibnu Katsir).

Oleh karena itu, dengan kemampuan nabi Adam dalam memberi nama sesuatu merupakan bukti bahwa manusia memiliki akal yang diciptakan oleh Allah *Subhanahu wa Ta'ala*. Seperti halnya pada sistem penelitian ini, salah satunya pada pencarian pola di dalam kalimat perintah *speech recognizer*

menggunakan algoritma *Apostolico Giancarlo* yang memerlukan akal agar sistem berjalan sesuai yang diinginkan. Oleh karena itu, dengan diberinya akal dapat menciptakan sesuatu yang bermanfaat dan memudahkan manusia. Sesuai dengan hadits riwayat Muslim tentang memudahkan orang lain yang berbunyi:

مَنْ نَفَّسَ عَنْ مُؤْمِنٍ كُرْبَةً مِنْ كُرْبِ الدُّنْيَا نَفَّسَ اللَّهُ عَنْهُ كُرْبَةً مِنْ كُرْبِ يَوْمِ الْقِيَامَةِ وَمَنْ يَسِّرْ عَلَى مُعْسِرٍ يَسِّرَ اللَّهُ عَلَيْهِ فِي الدُّنْيَا وَالْآخِرَةِ

“Barangsiapa yang menyelesaikan kesulitan seorang mukmin (orang lain) dari berbagai kesulitan – kesulitan dunia, niscaya Allah akan memudahkan kesulitan – kesulitannya pada hari kiamat. Barangsiapa yang memudahkan orang yang sedang kesulitan niscaya Allah mudahkan baginya di dunia dan akhirat”. (HR. Muslim : 2699)

Dengan sistem ini dapat memudahkan bagi para manusia apabila didalam ruangan tersebut ramai serta membantu disabilitas maupun non disabilitas sehingga proses kendali AC (*Air Conditioner*) dapat dilakukan secara otomatis.

Pada penelitian ini kontrol *Air Conditioner* dapat dilakukan menggunakan suara. Hal ini akan memudahkan manusia disabilitas yang mengalami kesulitan dalam mengoperasikan *Air Conditioner* menggunakan *remote*. Hal ini sesuai dengan HR. Muslim nomor 2699, di mana umat muslim diwajibkan membantu kesulitan orang lain, sehingga Allah *Subhanahu wa Ta'ala* akan membantunya dalam urusan dunia dan akhirat.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian kendali AC (*Air Conditioner*) menggunakan algoritma *Apostolico Giancarlo* dapat disimpulkan bahwa aksesibilitas pada sistem yang diuji coba oleh responden dengan jumlah 45, mendapatkan hasil diantaranya pada pertanyaan pertama memiliki nilai rata – rata 4,75 dengan arti bahwa nilai tersebut mendekati dengan pilihan yang kelima yaitu sangat setuju apabila sistem bisa menyalakan atau mematikan AC tanpa melihat layar, pada pertanyaan kedua memiliki nilai rata – rata 4,37 dengan arti bahwa nilai tersebut mendekati berada dipilihan keempat yaitu setuju apabila sistem bisa mengatur suhu AC tanpa melihat, pada pertanyaan ketiga memiliki nilai rata – rata 4,26 dengan arti bahwa nilai tersebut berada dipilihan keempat yaitu setuju apabila tombol - tombol sistem mudah ditemukan meskipun tanpa melihat, pada pertanyaan keempat memiliki nilai rata – rata 4,46 dengan arti bahwa nilai tersebut berada dipilihan keempat yaitu setuju apabila sistem dapat dipahami oleh responden, pada pertanyaan kelima memiliki nilai rata – rata 3,51 dengan arti bahwa nilai tersebut mendekati dengan pilihan yang keempat yaitu setuju apabila sistem tetap bisa mengakses meskipun tidak menyentuh layar secara langsung, pada pertanyaan keenam memiliki nilai rata – rata 3,68 dengan arti bahwa nilai tersebut mendekati dengan pilihan yang keempat yaitu setuju apabila sistem tetap mudah digunakan meskipun responden tidak menyentuh layar secara langsung, pada pertanyaan

ketujuh memiliki nilai rata – rata 4,42 dengan arti bahwa nilai tersebut berada dipilihan keempat yaitu setuju apabila sistem bisa digunakan dengan nyaman oleh orang dengan keterbatasan atau disabilitas, pada pertanyaan terakhir yaitu kedelapan memiliki nilai rata – rata 2,75 dengan arti bahwa nilai tersebut mendekati dengan pilihan yang ketiga yaitu netral apabila responden ada yang merasa kesulitan dan ada yang merasa tidak kesulitan menggunakan sistem dalam kondisi dengan mata yang tidak bisa melihat. Dari jawaban – jawaban responden yang telah diuji coba dengan pertanyaan – pertanyaan berdasarkan 7 pedoman komponen aksesibilitas juga dapat disimpulkan bahwa sistem pada penelitian ini memenuhi standard komponen aksesibilitas.

Kemudian dapat disimpulkan pula, bahwa performa hasil penelitian kendali AC (*Air Conditioner*) memiliki hasil *standard error* yang di mana telah diuji coba dengan 10 *pattern* menghasilkan 100 kalimat uji coba, dari kalimat tersebut menghasilkan *error* sebanyak 24 kalimat, hasil dari 24 kalimat *error* tersebut dimasukan ke dalam rumus *standard error* dengan perhitungan tersebut, menghasilkan nilai 0,08.

5.2 Saran

Sistem kendali AC (*Air Conditioner*) menggunakan algoritma *Apostolico Giancarlo* ini masih memiliki kekurangan, sehingga diperlukan pengembangan di masa depan agar dapat menjadi sistem yang lebih baik. Saran dari peneliti adalah sistem dapat dijadikan aplikasi sendiri yang langsung dapat di download di *Play Store* tanpa aplikasi bantuan, hal ini untuk lebih memudahkan aksesibilitas bagi penyandang disabilitas tanpa perlu membuka aplikasi bantuan yang di mana hal

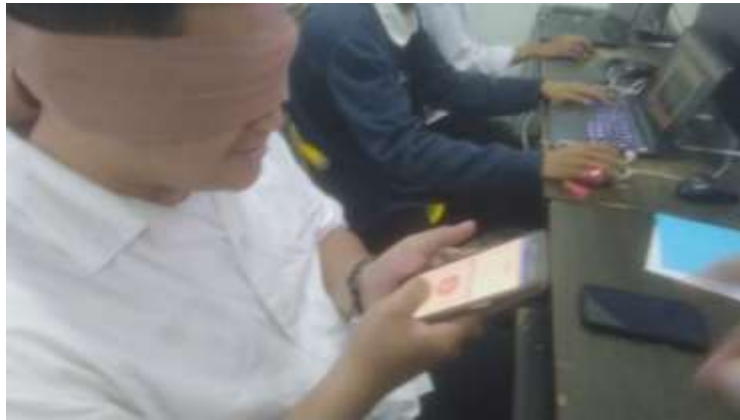
itu akan membingungkan penyandang disabilitas. Saran yang kedua dari peneliti adalah *pattern* yang dimasukan ke dalam algoritma bisa ditambah lebih banyak, hal ini untuk meminimkan terjadinya *error* dari kalimat yang dimasukan oleh pengguna.

DAFTAR PUSTAKA

- Afika, Nur.A, and Arif Widodo. 2021. "Smart AC Remote : Pengontrol Suhu Air Conditioner Otomatis Berbasis Internet of Thing Suhu Aktual Ruangan." *Jurnal Teknik Elektro* 10 Nomor 0: 681–88.
- Barus, Handayani Br. 2021. "Implementasi Algoritma Apostolico-Giancarlo Pada Pencarian Katalog Mebel PT.Yasanda Berbasis Mobile." *Kajian Ilmiah Informatika dan Komputer* 1(4): 143–47. <https://djournals.com/klik>.
- Dianti, Yira. 2017. "濟無No Title No Title No Title." *Angewandte Chemie International Edition*, 6(11), 951–952.: 5–24. [http://repo.iain-tulungagung.ac.id/5510/5/BAB 2.pdf](http://repo.iain-tulungagung.ac.id/5510/5/BAB%20.pdf).
- Henny Rosselawaty. 2023. "Pengaruh Aksesibilitas, Amenitas, Atraksi, Dan Pelayanan Tambahan Terhadap Minat Kunjungan Ke Destinasi Wisata Lon Malang." *Jurnal Masharif al- Syariah: Jurnal Ekonomi dan Perbankan Syariah* 8(4): 989–1002.
- Ii, B A B, and Landasan Teori. 2015. "BAB II LANDASAN TEORI 2.1 Speech Recognition." : 1–21.
- Irawan, Candra et al. 1977. "String Matching Adalah Suatu Algoritma Yang Digunakan Untuk Memecahkan Masalah Pencocokan Suatu Teks Terhadap Suatu Teks Lain . Ada Beberapa Algoritma Yang Dapat Digunakan Dalam Pencarian String Matching Contohnya Ialah Algoritma Boyer Moore Dan Brute For."
- Kementrian Kesehatan Republik Indonesia. 2018. "UNIVERSITAS SUMATERA UTARA Poliklinik UNIVERSITAS SUMATERA UTARA." *Jurnal Pembangunan Wilayah & Kota* 1(3): 82–91.
- Lara, S, and F F Muhammad. 2021. "Sistem Kontrol Dan Monitoring Air Conditioner Berbasis Internet of Things (Iot)." [http://repository.polman-babel.ac.id/id/eprint/326/1/Makalah Proyek Akhir Sistem Kontrol dan Monitoring Air Conditioner Berbasis IoT.pdf](http://repository.polman-babel.ac.id/id/eprint/326/1/Makalah%20Proyek%20Akhir%20Sistem%20Kontrol%20dan%20Monitoring%20Air%20Conditioner%20Berbasis%20IoT.pdf).
- Murad, Christine, Heloisa Candello, and Cosmin Munteanu. 2023. 1 Proceedings of the 5th International Conference on Conversational User Interfaces, CUI 2023 *What's the Talk on VUI Guidelines? A Meta-Analysis of Guidelines for Voice User Interface Design*. Association for Computing Machinery.
- Nadiansyah, Ricza Rahmad. 2018. "Sistem Pengendali Kipas Angin Berbasis Nodemcu Esp8226." *Universitas Teknologi Digital Indonesia*: 3–10. <http://eprints.akakom.ac.id/id/eprint/8173>.

- Pagan, Muasir, and Efori Buulolo. 2021. "Penerapan Algoritma Apostolico Giancarlo Pada Pencarian Fasilitas Umum Di Provinsi Aceh Berbasis Android." *Building of Informatics, Technology and Science (BITS)* 3(2): 95–100.
- Pahrurrozi, Muhammad. 2020. "Adaptive Classroom Berbasis IoT (Internet of Things), Manajemen Penggunaan Air Conditioner (AC) Secara Otomatis." *Adaptive Classrom Berbasis IOT (Internet Of Things), Manajemen Penggunaan Air Conditioner(AC) Secara Otomatis*.
- Pribadi, Octara. 2020. "Sistem Kendali Jarak Jauh Air Conditioner (AC) Berbasis IoT." *Jurnal TIMES* IX(1): 1–8. <https://ejournal.stmik-time.ac.id/index.php/jurnalTIMES/article/view/622>.
- Pustaka, B A B Tinjauan. 2019. "Sensor Infra Red." : 6–14.
- Salaman Dian. 2019. "C Dan Kelembaban 40-60 %. Tingkat Kenyamanan Suatu Ruang Juga Ditentukan Oleh Temperatur, Kelembapan, Sirkulasi Dan Tingkat Kebersihan Udara (." : 5–44.
- Silitonga, Frans Yus Mori. 2021. "Perancangan Aplikasi Istilah Olahraga Berbasis Mobile Dengan Penerapan Algoritma Apostolico-Giancarlo." *Journal of Computing and Informatics Research* 1(1): 14–18. <https://journal.fkpt.org/index.php/comforch/article/view/112>.
- Simarangkir, Manase Sahat H. 2017. "Studi Perbandingan Algoritma - Algoritma." *Jurnal Inkofar* 1(1): 40–46.
- Susila, Anak Agung Ngurah Hary, and Dewa Made Sri Arsa. 2023. "Analisis System Usability Scale (SUS) Dan Perancangan Sistem Self Service Pemesanan Menu Di Restoran Berbasis Web." *Majalah Ilmiah UNIKOM* 21(1): 3–8.
- Yusniyanti, Erna, and Kurniati Kurniati. 2017. "Analisa Puncak Banjir Dengan Metode MAF (Studi Kasus Sungai Krueng Keureuto)." *EINSTEIN e-JOURNAL* 5(1).

LAMPIRAN













Siti Muslihatul Jannah

Arya

Furkon

Rayhan Ramadhan

Annisa Ayu Permadani

Avicena Halim

Kautsar Quraisy Al Hamidy

Bu Gani

Nurul Aziza Kusuma Dewi

Eka M
Niken Dewi Sesanti
Kalyca Chiara
Roihan Farras Setyadi
Salma Chesha Putri Pramudya Wardani
Muhammad Zidane Putra Aditya
Bambang Supriyanto
Salmaa
Pak Gani

Naelul Khozy Syahidan Arsyadallah
Achmad Ramadan
Muhammad Annas Bintar Putra
Aisyah Dwi Permatasari
Al Ikhlasul Hakimi
Naura Desta Putri
Amirul
Hafidz Nurrois
Nabilasshabira

ummi
Achmad Raka
Erni
Dhamirah Nurul Afiqah
Doni
Dawud
Liyasya
Najah Muchsin Sanin
Akbar Fadhil

Radina
Rahma
Tazkia Ikhsanul
Zanuar Ajeng Risma
Salsabila Ramadanti Nuraini
Cantika
Zainatus Sholeha
Puja