

**SISTEM KONTROL SPEAKER MUROTTAL AL-QUR'AN
BERBASIS *INTERNET OF THINGS* MENGGUNAKAN
ALGORITMA AHO-CORASICK**

SKRIPSI

**Oleh :
HENY RIMADANA
NIM. 210605110009**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

**SISTEM KONTROL SPEAKER MUROTTAL AL-QUR'AN
BERBASIS *INTERNET OF THINGS* MENGGUNAKAN
ALGORITMA AHO-CORASICK**

SKRIPSI

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

Oleh :
HENY RIMADANA
NIM. 210605110009

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

HALAMAN PERSETUJUAN

SISTEM KONTROL SPEAKER MUROTTAL AL-QUR'AN BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA *AHO-CORASICK*

SKRIPSI

Oleh :
HENY RIMADANA
NIM. 210605110009

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 29 November 2024

Pembimbing I,



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

Pembimbing II,



Dr. Agung Teguh Wibowo Almais, M.T
NIP. 19860301 202321 1 016

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 KONTROL SPEAKER MUROTTAL AL-QUR'AN BERBASIS *INTERNET OF THINGS* MENGGUNAKAN ALGORITMA *AHO-CORASICK*

SKRIPSI

Oleh :
HENY RIMADANA
NIM. 200605110009

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 17 Desember 2024

Susunan Dewan Penguji

Ketua Penguji	: <u>Prof. Dr. Suhartono, M.Kom</u> NIP. 19680519 200312 1 001
Anggota Penguji I	: <u>Shoffin Nahwa Utama, M.T</u> NIP. 19860703 202012 1 003
Anggota Penguji II	: <u>Ajib Hanani, M.T</u> NIP. 19840731 202321 1 013
Anggota Penguji III	: <u>Dr. Agung Teguh Wibowo Almais, M.T</u> NIP. 19860301 202321 1 016

()
()
()
()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Malang




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

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Heny Rimadana
NIM : 210605110009
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Sistem Kontrol Speaker Murottal Al-Qur'an
Berbasis *Internet of Things* Menggunakan
Algoritma *Aho-corasick*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil 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, 19 Desember 2024

Yang membuat pernyataan,



Heny Rimadana

NIM. 210605110009

MOTTO

“LOVE WHAT YOU DO, DO WHAT YOU LOVE”

“Hidup bukan saling mendahului, bermimpilah sendiri-sendiri.”

“Skripsi yang baik adalah skripsi yang selesai karena orang yang menjalani bangku perkuliahan dengan baik itu bukan mereka yang pinter banget akademiknya, melainkan mereka yang punya keseimbangan antara kecerdasan emosional, mental, dan daya juangnya.”

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur, penulis memanjatkan terima kasih yang sebesar-besarnya kepada Allah SWT atas segala anugerah dan rahmat-Nya. Tak lupa, shalawat dan salam selalu tercurah kepada Nabi Muhammad SAW. Pencapaian dalam bentuk skripsi ini penulis persembahkan dengan hati yang penuh kebahagiaan kepada:

1. Mama dan Abah yang senantiasa memberikan kasih sayang, cinta, dan doa tanpa henti. Mereka adalah sumber kekuatan dan pilar penyangga yang selalu ada dalam setiap langkah yang penulis ambil.
2. Dosen Pembimbing, Bapak Ajib Hanani, M.T., dan Bapak Dr. Agung Teguh Wibowo Almais, M.T., yang dengan sabar memberikan bimbingan, dukungan, dan arahan yang tiada henti selama proses penelitian dan penulisan skripsi ini. Ilmu yang telah diberikan menjadi inspirasi dan pembelajaran yang sangat berharga. Tak lupa pula kepada Dosen Penguji, Bapak Prof. Dr. Suhartono, M.Kom dan Bapak Shoffin Nahwa Utama, M.T., yang memberikan evaluasi, kritik, serta saran konstruktif yang sangat membantu dalam memperbaiki kualitas penelitian ini. Terima kasih atas perhatian, waktu, dan masukan yang sangat berharga.
3. Diri saya sendiri yang telah melewati segala tantangan, kegagalan, dan kegelisahan karena menjadi pemeran utama dalam perjalanan ini yang tetap bertahan dan terus berjuang meskipun jalan yang dilalui tak selalu dimudahkan.

Terima kasih kepada semua pihak yang telah memberikan dukungan selama proses penyelesaian skripsi ini. Bagi teman-teman yang masih berjuang, teruslah melangkah maju lewati rintangannya karena *skripsi yang baik adalah skripsi yang selesai*.

KATA PENGANTAR

Assalamualaikum Wr. Wb.

Alhamdulillah Wasyukurillah segala puji bagi Allah SWT. atas segala rahmat dan karunia-Nya sehingga penulis dapat diberikan kesehatan dan kelancaran dalam menyelesaikan skripsi yang berjudul “Sistem Kontrol Speaker Murottal Al-Qur’an Berbasis *Internet of Things* Menggunakan Algoritma *Aho-Corasick*”. Shalawat serta salam semoga selalu tercurahkan kepada Nabi Muhammad SAW. Semoga kita sekalian mendapatkan syafaatnya di hari akhir kelak, *Aamiin*.

Rasa syukur dan terima kasih yang sebesar-besarnya penulis ucapkan kepada semua pihak-pihak yang senantiasa memberikan dukungan kepada penulis untuk dapat berproses dalam pengerjaan skripsi ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Hariani, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, ST., M.MT., IPU, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Bapak Ajib Hanani, M.T selaku Dosen Pembimbing I dan Bapak Dr. Agung Teguh Wibowo Almais, M.T selaku Dosen Pembimbing II yang dengan segala arahan, saran serta masukan yang diberikan sangat berarti dalam setiap langkah dalam proses pengerjaan skripsi ini.

5. Bapak Prof. Dr. Suhartono, M.Kom selaku dosen Penguji I dan Bapak Shoffin Nahwa Utama, M.T dosen Penguji II yang telah memberikan masukan dan saran yang membangun sehingga skripsi ini dapat dipertanggunghkan dengan lebih baik.
6. Segenap Dosen, Admin, Laboran dan jajaran pada Program Studi Teknik Informatika yang telah memberikan bimbingan dan bantuan selama studi.
7. PERTUNI (Persatuan Tunanetra Indonesia) Malang yang telah bersedia melakukan uji coba alat penelitian ini. Terima kasih atas kerjasama dan dukungannya yang sangat berarti dalam kelancaran penelitian ini. Semoga bapak/ibu selalu diberikan rezeki yang berlimpah.
8. Mama, Abah, dan seluruh keluarga besar yang selalu memberikan semangat untuk terus berusaha, serta doa-doa terbaik yang selalu dipanjatkan, sehingga skripsi ini dapat diselesaikan dengan lancar dan baik.
9. Seseorang yang tak kalah penting kehadirannya, Naufal Bakhtiar Ismail. Terimakasih telah memberikan semangat, doa, dukungan, motivasinya, dan segala kontribusi baiknya dalam perjalanan skripsi ini. Terimakasih telah membersamai penulis hingga saat ini, tetap semangat berjuang untuk perjalanan dan rintangan ke depannya.
10. Bestie-bestieku, Najah, Pian, Rafi, Irul yang rela ketilang polisi hanya untuk membuat lelucon di sela-sela mumet skripsian. Terlebih untuk Fuad atas pinjaman hp nya yang sangat berguna dalam penelitian ini. Radina, Eka, Julfa, Dila, Fitri yang sedang berjuang melawan kemalasan untuk menyelesaikan skripsi. Semangat terus ya, sehat-sehat kalian.

11. Teknik Informatika Angkatan 21 “Aster” yang sudah kebersamai 3,5 tahun saya di perantauan ini. Terimakasih atas kebersamaan, semangat, dan kenangan yang tak terlupakan. See u on top guys!
12. Komunitas ONTAKI yang telah menyediakan wadah bagi penulis untuk mengeksplorasi diri dalam dunia robotika. Terima kasih atas kesempatan dan pembelajaran berharga yang diberikan.
13. Seluruh elemen bumi yang tidak dapat saya sebutkan satu per satu, yang telah diciptakan-Nya untuk mengelilingi dan memberikan makna dalam hidup saya.

Dalam penyusunan skripsi ini masih tedapat kekurangan dan harapannya semoga skripsi ini dapat memberikan manfaat kepada pembaca.

Wassalamualaikum Wr. Wb.

Malang, 12 Desember 2024

Penulis

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PERSETUJUAN	iv
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL	xv
ABSTRAK	xvi
ABSTRACT	xvii
البحث مستخلص.....	xviii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Pernyataan Masalah.....	6
1.3 Batasan Masalah	6
1.4 Tujuan Penelitian.....	6
1.5 Manfaat Penelitian.....	7
BAB II STUDI PUSTAKA	8
2.1 Penelitian Terkait.....	8
2.2 Speaker Murottal Al-Qur'an.....	12
2.3 <i>Android</i>	13
2.4 <i>MIT App Inventor</i>	14
2.5 <i>Arduino IDE</i>	14
2.6 <i>ESP32</i>	14
2.7 <i>Internet of Things</i>	15
2.8 <i>String Matching</i>	16
2.9 Algoritma Aho-Corasick	17
2.10 <i>Speech Recognition</i>	18
BAB III METODE PENELITIAN	19
3.1 Pengumpulan Data.....	19
3.2 Desain dan Perancangan Sistem Kontrol.....	21
3.2.1 Tahapan Input.....	22
3.2.2 Tahapan Proses.....	22
3.2.2.1 <i>Case Folding</i>	22
3.2.2.2 <i>Tokenizing</i>	23
3.2.2.3 <i>Stopword Removal</i>	23
3.2.2.4 <i>Stemming</i>	24
3.2.2.5 Pencocokan Algoritma <i>Aho-Corasick</i>	24
3.2.3 Tahap Output.....	28
3.2.4 Desain Aplikasi	28

3.2.5 Perancangan Alat Sistem Kontrol	29
3.3 Implementasi Sistem.....	31
3.4 Pengujian Sistem	32
3.4.1 Perhitungan <i>Error</i> Sistem	32
3.4.2 <i>Sistem Usability Scale</i> (SUS).....	34
BAB IV UJI COBA DAN PEMBAHASAN	39
4.1 Hasil Implementasi Sistem Kontrol.....	39
4.1.1 Halaman <i>Software</i>	39
4.1.2 Implementasi Komunikasi IoT.....	41
4.1.3 Kontrol Speaker dengan <i>Speech Recognition</i>	43
4.1.4 Rangkaian <i>Hardware</i>	46
4.2 Hasil Pengujian Sistem	47
4.2.1 Proses Pencocokan Pola dengan <i>Aho-Corasick</i>	48
4.2.2 Pengujian Sistem Kontrol	52
4.2.3 Pengujian Error Sistem	53
4.2.4 Pengujian <i>System Usability Scale</i> (SUS)	59
4.3 Pembahasan	61
4.4 Integrasi Islam	62
4.4.1 Muamalah Ma'a Allah	62
4.4.2 Muamalah Ma'a An-Nas.....	63
4.4.3 Muamalah Ma'a Alam	65
BAB V KESIMPULAN DAN SARAN	66
5.1. Kesimpulan.....	66
5.2. Saran	66
DAFTAR PUSTAKA	
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2. 1 Desain Penelitian	15
Gambar 3. 1 Alur Penelitian.....	19
Gambar 3. 2 Desain Sistem	21
Gambar 3. 3 Cara Kerja Aho-Corasick	25
Gambar 3. 4 Tree Aho-Corasick	26
Gambar 3. 5 Failure Function	27
Gambar 3. 6 Output Aho-Corasick	28
Gambar 3. 7 Rancangan Tampilan Awal Aplikasi.....	29
Gambar 3. 8 Rancangan Tampilan Setelah Input Suara.....	29
Gambar 3. 9 Rancangan Komponen Tampak Samping	30
Gambar 3. 10 Rancangan Komponen Tampak Atas	30
Gambar 3. 11 SUS Score.....	36
Gambar 4. 1 Splash Screen	39
Gambar 4. 2 Halaman Utama Aplikasi	40
Gambar 4. 3 Fitur Speech Recognition	40
Gambar 4. 4 Ngrok di CMD.....	41
Gambar 4. 5 Ngrok mengakses localhost.....	42
Gambar 4. 6 Logo Maqiatto	42
Gambar 4. 7 Pesan Subscribe di Command Prompt	46
Gambar 4. 8 Rangkaian Hardware	46
Gambar 4. 9 Output Pattern Matching Aho-Corasick.....	50
Gambar 4. 10 Output Jika Perintah Tidak Ditemukan	51
Gambar 4. 11 Perintah Menyalakan Speaker	52
Gambar 4. 12 Perintah Mematikan Speaker.....	52
Gambar 4. 13 Perintah Memutar Surah Tertentu	53
Gambar 4. 14 Grafik Pengujian SUS	60

DAFTAR TABEL

Tabel 2. 1 Penelitian Terkait	10
Tabel 3. 1 Daftar Surah Juz 30	20
Tabel 3. 2 Case Folding.....	22
Tabel 3. 3 Tokenizing.....	23
Tabel 3. 4 Stopword Removal.....	23
Tabel 3. 5 Stemming	24
Tabel 3. 6 Tabel Transisi.....	26
Tabel 3. 7 Failure Function	27
Tabel 3. 8 Contoh Pengujian Error Sistem.....	33
Tabel 3. 9 Kuesioner SUS Penelitian	34
Tabel 3. 10 Skala Likert	34
Tabel 3. 11 Kategorisasi Hasil Pengujian SUS	36
Tabel 3. 12 Contoh Skor Jawaban Kuesioner	37
Tabel 3. 13 Contoh Perhitungan Skor SUS.....	37
Tabel 4. 1 Pola (Pattern) Penelitian.....	48
Tabel 4. 2 Hasil Pengujian Error Sistem oleh SR	54
Tabel 4. 3 Hasil Pengujian Error Sistem oleh NS	54
Tabel 4. 4 Hasil Pengujian Error Sistem oleh IT	54
Tabel 4. 5 Hasil Pengujian Error Sistem oleh W	55
Tabel 4. 6 Hasil Pengujian Error Sistem oleh WT	55
Tabel 4. 7 Hasil Pengujian Error Sistem oleh L.....	55
Tabel 4. 8 Hasil Pengujian Error Sistem oleh WN	56
Tabel 4. 9 Hasil Pengujian Error Sistem oleh SD	56
Tabel 4. 10 Hasil Pengujian Error Sistem oleh MM	56
Tabel 4. 11 Hasil Pengujian Error Sistem oleh TM	57
Tabel 4. 12 Pengujian dengan 5 kalimat	58
Tabel 4. 13 Skor Jawaban Pengujian SUS	59
Tabel 4. 14 Perhitungan Skor SUS.....	59
Tabel 4. 15 Kode Hexadecimal	62

ABSTRAK

Rimadana, Heny. 2024. **Sistem Kontrol Speaker Murottal Al-Qur'an Berbasis *Internet of Things* menggunakan Algoritma Aho-Corasick**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Ajib Hanani, M.T (II) Dr. Agung Teguh Wibowo Almais, M.T.

Kata kunci: *Sistem Kontrol, IoT, Speech Recognition, Algoritma Aho-Corasick*

Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem kontrol speaker Murottal Al-Qur'an berbasis IoT dengan perintah suara untuk penyandang disabilitas netra. Menggunakan speech recognition untuk menggantikan kesulitan penyandang disabilitas netra mengontrol perangkat dengan tombol atau *remote*. Dengan membangun aplikasi Android "*Qirabox*" yang terintegrasi dengan *Google Speech Recognition* dapat mengonversi perintah suara menjadi teks. Teks yang diterima kemudian diproses dengan algoritma *Aho-Corasick* untuk pencocokan pola dalam kalimat perintah. Selain itu, sistem ini menggunakan protokol MQTT untuk komunikasi IoT, ESP32 sebagai mikrokontroler, dan IR *Transmitter* untuk mengirimkan sinyal infrared ke speaker. Hasil pengujian sistem melibatkan 10 penyandang disabilitas netra dan 50 kali percobaan di PERTUNI Malang, menunjukkan nilai error sebesar 2%. Selain pengujian sistem, juga melakukan pengujian *System Usability Scale* (SUS) yang menghasilkan nilai 78.75 (*Good*). Mengartikan bahwa sistem tersebut sudah memenuhi kebutuhan pengguna. Berdasarkan hasil tersebut, penelitian ini dapat mengembangkan sistem kontrol speaker berbasis suara dan memberikan kontribusi signifikan dalam pengembangan teknologi asistif untuk memudahkan aksesibilitas bagi penyandang disabilitas netra.

ABSTRACT

Rimadana, Heny. 2024. *Controlling System for Al-Qur'an Murottal Speaker Based on Internet of Things Using Aho-Corasick Algorithm*. Undergraduate Thesis. Study Program of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang. Advisors: (I) Ajib Hanani, M.T (II) Dr. Agung Teguh Wibowo Almais, M.T.

Keywords: *Control System, IoT, Speech Recognition, and Aho-Corasick Algorithm*

This research aims to design and implement an IoT-based speaker control system for Murottal Al-Qur'an with voice commands, specifically for visually impaired individuals. It utilizes speech recognition technology to address the challenges faced by visually impaired users in controlling devices with buttons or remotes. An Android application, "Qirabox," integrated with Google Speech Recognition, converts voice commands into text. The received text is then processed using the Aho-Corasick algorithm for pattern matching in the command phrases. Additionally, the system employs the MQTT protocol for IoT communication, ESP32 as the microcontroller, and an IR Transmitter to send infrared signals to the speaker. The system was tested with 10 visually impaired participants through 50 trials conducted at PERTUNI Malang, resulting in an error rate of 2%. Furthermore, the System Usability Scale (SUS) evaluation yielded a score of 78.75 (Good), indicating that the system meets user needs. Based on these findings, this research contributes to the development of a voice-controlled speaker system and offers a significant contribution to the advancement of assistive technology, enhancing accessibility for visually impaired individuals.

البحث مستخلص

رمضان، هني. 2024. نظام التحكم في مكبر الصوت "مرتل القرآن" على أساس إنترنت الأشياء باستخدام خوارزمية *Aho-Corasick*. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: عجيب حنائي، الماجستير. المشرف الثاني: د. أغونج تيغوه ويو وو ألميس، الماجستير.

الكلمات الرئيسية: : نظام تحكم، إنترنت أشياء، تعرف على كلام، خوارزمية *Aho-Corasick*

يهدف هذا البحث إلى تصميم وتنفيذ نظام التحكم في مكبر الصوت "مرتل القرآن" على أساس إنترنت الأشياء مع أوامر صوتية للأشخاص ذوي الإعاقات البصرية. استخدام التعرف على الكلام لاستبدال صعوبة الأشخاص ذوي الإعاقات البصرية في التحكم في الجهاز باستخدام زر أو جهاز تحكم عن بعد. من خلال إنشاء تطبيق أندرويد "Qirabox" الذي تكامل مع جوجل للتعرف على الكلام، يمكنه تحويل الأوامر الصوتية إلى نص. ثم تتم معالجة النص المستلم باستخدام خوارزمية *Aho-Corasick* لمطابقة الأنماط في جمل الأوامر. بالإضافة إلى ذلك، استخدم النظام بروتوكول MQTT لاتصالات إنترنت الأشياء، و ESP32 كمتحكم دقيق، وجهاز إرسال الأشعة تحت الحمراء لنقل إشارات الأشعة تحت الحمراء إلى مكبر الصوت. أظهرت نتائج اختبار النظام الذي شمل 10 أشخاص من ذوي الإعاقات البصرية و 50 محاولة في PERTUNI مالانج قيمة خطأ بنسبة 2٪. بالإضافة إلى اختبار النظام، أجرى أيضا اختبار مقياس قابلية استخدام النظام (SUS) الذي نتج عنه درجة 78.75 (جيد). هذا يعني أن النظام قد لبي احتياجات المستخدمين. بناء على هذه النتائج، يمكن لهذا البحث تطوير نظام التحكم في مكبر الصوت على أساس الصوت وتقديم مساهمة كبيرة في تطوير التكنولوجيا المساعدة لتسهيل وصول الأشخاص ذوي الإعاقات البصرية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Komunikasi melalui suara adalah kemampuan dasar yang dimiliki oleh setiap individu yang berfungsi untuk menjalin interaksi dengan orang lain. Dalam keseharian, komunikasi ini biasanya dilakukan dengan berbicara secara langsung antar individu. Namun, seiring dengan perkembangan teknologi, komunikasi tidak hanya terbatas antar-manusia, tetapi juga melibatkan interaksi antara manusia dan komputer. Fenomena ini memunculkan konsep *Speech Recognition* atau pengenalan suara dalam ilmu komputer.

Speech Recognition merupakan teknologi yang memungkinkan sistem komputer mengenali dan memahami ucapan manusia (M. Siahaan et al., 2020). Teknologi ini telah banyak digunakan dalam berbagai aplikasi seperti alat bantu komunikasi bagi penerjemah, otomatisasi rumah, dan *assistive* penyandang disabilitas (Suhartono, 2018). Menilik dari sisi disabilitas yang menarik banyak perhatian membuat teknologi ini semakin berkembang. salah satu kelompok penyandang disabilitas yang sangat diuntungkan oleh teknologi *Speech Recognition* adalah tunanetra. Tunanetra adalah kondisi gangguan penglihatan yang dialami seseorang sehingga menyebabkan banyak kendala dalam aktivitas sehari-hari.

Menurut data yang disampaikan oleh Organisasi Kesehatan Dunia (WHO), dinyatakan bahwa penyandang gangguan penglihatan di seluruh dunia pada tahun 2023 mencapai 2,2 Miliar dan 50% diantaranya mengalami disabilitas netra total.

Dalam skala tersebut, Indonesia menempati peringkat ketiga penyandang disabilitas netra (Ansori, 2023). Adapun Data dari Kementerian Kesehatan Republik Indonesia mencatat bahwa jumlah penyandang tunanetra di Indonesia mencapai 1,5% dari total populasi. Dengan populasi Indonesia yang kini lebih dari 270 juta orang, jumlah individu dengan disabilitas tunanetra diperkirakan mencapai sekitar 4 juta orang (Imran, 2024). Diperkuat dengan data dari (Badan Pusat Statistik Provinsi Jawa Timur, 2019) tercatat bahwa 5987 jiwa penyandang tunanetra dari total 8307 jiwa penyandang disabilitas. Hal ini menjadikan banyaknya penelitian yang mengusung teknologi assistif yang dapat memberikan manfaat besar bagi penyandang disabilitas khususnya Tunanetra.

Dalam konteks keislaman, banyak tunanetra yang beraktifitas melibatkan Al-Qur'an seperti membaca, mendengarkan, hingga menghafal Al-Quran. Namun, tunanetra menghadapi tantangan besar karena tidak memiliki fungsi penglihatan. Mereka harus memaksimalkan indera lain untuk melakukan aktivitasnya. Untuk membantu tunanetra membaca Al-Qur'an, telah dipasarkan Al-Qur'an *braille* tetapi harganya cukup mahal sehingga tidak semua dari mereka dapat memilikinya (Kurnia & Khaerani, 2020).

Untuk mengatasi masalah ini, speaker murottal Al-Qur'an dirancang untuk memudahkan pendengaran ayat-ayat Al-Qur'an. Namun, tunanetra masih menghadapi kendala dalam pengoperasiannya, terutama saat harus meraba tombol atau remote untuk mengganti surah. Penggunaan remote juga terbilang tidak efektif karena kode numeriknya tidak dapat dilihat oleh tunanetra. Oleh karena itu, teknologi *speech recognition* dapat diimplementasikan pada speaker murottal Al-

Qur'an guna memfasilitasi tunanetra dalam berinteraksi dengan Al-Qur'an menjadi lebih mendalam dan efisien.

Teknologi ini memungkinkan kontrol speaker dengan perintah suara tanpa memerlukan pengoperasian manual melalui tombol atau remote. Implementasi teknologi ini sejalan dengan perhatian Islam terhadap kemudahan bagi individu yang memiliki keterbatasan fisik. Hal ini tercermin dalam firman Allah dalam Q.S. An-Nur 61:

لَيْسَ عَلَى الْأَعْمَى حَرْجٌ وَلَا عَلَى الْأَعْرَجِ حَرْجٌ وَلَا عَلَى الْمَرِيضِ حَرْجٌ

“Tidak ada dosa atas orang buta, atas orang pincang, dan atas orang sakit...”
(Q.S. An-Nur: 61).

Tafsir Kementerian Agama RI menegaskan bahwa ayat ini memberikan kemudahan bagi mereka dalam hal sosial dan interaksi dengan lingkungan sekitarnya. Keringanan ini juga menegaskan pentingnya menjaga inklusivitas dalam kehidupan bermasyarakat. Oleh karena itu, penerapan teknologi assistif seperti *speech recognition* pada speaker murottal Al-Qur'an diharapkan dapat meningkatkan aksesibilitas dan kenyamanan bagi tunanetra dalam menjalankan aktivitas keagamaan, sejalan dengan prinsip inklusivitas dan perhatian Islam terhadap individu yang memiliki kebutuhan khusus.

Selain memudahkan akses bagi mereka yang memiliki keterbatasan fisik, teknologi ini juga mendukung pentingnya belajar dan mengajarkan Al-Qur'an sebagaimana dianjurkan dalam ajaran Islam. Sebagaimana dalam hadits, Nabi Muhammad SAW bersabda:

خَيْرُكُمْ مَنْ تَعَلَّمَ الْقُرْآنَ وَعَلَّمَهُ

“Sebaik-baik kalian adalah orang yang belajar Al-Qur’an dan mengajarkannya.”
(HR. Bukhari).

Hadis ini mengajarkan bahwa salah satu perbuatan terbaik yang dapat dilakukan seorang Muslim adalah belajar dan mengajarkan Al-Qur’an. Dengan adanya teknologi ini, penyandang tunanetra tidak hanya dapat mendengarkan dan belajar Al-Qur’an, tetapi juga berpotensi mengajarkannya kepada orang lain melalui hafalan dan pemahaman mereka.

Terdapat beberapa penelitian lain yang menjadi referensi untuk penelitian ini, salah satunya penelitian yang dilakukan oleh (Akbar et al., 2019) bertujuan untuk pembuatan aplikasi dengan pemanfaatan *Google Speech* dalam untuk mengoreksi hafalan Al-Qur’an yang hanya termuat dari 3 surah, yakni Al Ikhlas, Al Kautsar, dan An Nas. Sedangkan, penelitian ini dilakukan pengembangan jumlah surah terkhusus surah-surah pada juz 30. Penelitian lain dilakukan (Salamah et al., 2020) berfokus pada pembuatan aplikasi Al-Qur’an audio untuk penyandang tunanetra dengan memberikan output berupa bunyian Al-Qur’an yang dihubungkan dengan *headset* dan *bluetooth*. Namun, penggunaan *bluetooth* memiliki keterbatasan jarak dan kestabilan koneksi yang dapat mempengaruhi pengalaman pengguna (Hanani & Hariyadi, 2020).

Seiring dengan kemajuan di bidang industri otomasi dan wireless, terciptalah konsep *Internet of Things*. IoT mengkoneksikan berbagai perangkat lewat jaringan internet sehingga dapat berkomunikasi tanpa batasan jarak antara pengguna dan perangkat. Dalam konteks religius, teknologi IoT memiliki potensi besar untuk

meningkatkan kenyamanan dan efisiensi dalam pelaksanaan kegiatan keagamaan, seperti penggunaan speaker murottal Al-Qur'an.

Selain penggunaan IoT, penelitian ini juga memanfaatkan *Google Speech* untuk memberikan perintah suara dari pengguna. Perintah suara diterima kemudian diubah menjadi teks. Dalam hal ini, diperlukan algoritma pencarian teks yang cepat dan akurat. Penelitian sebelumnya dilakukan oleh (Safwani et al., 2021) yang berfokus pada *patten mathing* untuk arti ayat Al-Qur'an menggunakan *Berry Ravindran Algorithm* dengan hasil akurasi 82,6% dengan inputan suara tetapi mengalami keterbatasan pencarian ketika ayat yang cukup panjang. Penelitian lain yang membahas *string matching* mengunggulkan algoritma *Aho-Corasick* untuk mencari banyak *pattern* dalam sebuah teks. Menurut (Sulaeman & Murnawan, 2023) *Aho-Corasick* memiliki waktu pemrosesan yang lebih efisien untuk pencocokan kata kunci dengan laporan dan informasi pada aplikasi *Lost and Found*.

Berdasarkan latar belakang di atas, dibuatlah sistem kontrol speaker murottal Al-Qur'an berbasis *Internet of Things* menggunakan algoritma pencarian teks *Aho-Corasick*. Sistem yang dibangun diharapkan dapat diimplementasikan untuk membantu penyandang disabilitas, terutama tunanetra atau orang lanjut usia yang memiliki keterbatasan dalam mengontrol speaker murottal Al-Qur'an. Dengan adanya sistem ini, diharapkan dapat meningkatkan kenyamanan dan efisiensi bagi tunanetra dalam menjalankan aktivitas keagamaannya. Teknologi ini juga diharapkan dapat memberikan kontribusi positif dalam pengembangan perangkat *assistive technology* yang lebih inklusif dan dapat diakses oleh banyak kalangan.

1.2 Pernyataan Masalah

Pernyataan masalah untuk penelitian ini adalah bagaimana merancang dan mengimplementasikan sistem kontrol speaker murottal al-qur'an berbasis *IoT* menggunakan algoritma *Aho-Corasick* dengan melakukan pengujian error sistem dan skala penilaian kegunaan sistem (*system usability scale*).

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini dibuat untuk memfokuskan pemaparan penelitian sebagai berikut.

1. Kontrol speaker menggunakan *speech recognition* dengan *smartphone* digunakan berbasis Android.
2. Surah-surah Al-Qur'an yang digunakan terbatas hanya surah pada juz 30.
3. Murottal yang digunakan adalah suara dari salah satu Qori yang bernama Syekh Misyari Rasyid
4. Kontrol speaker berupa menyalakan speaker, mematikan speaker, menaikkan volume, menurunkan volume, menghentikan dan memutar surah tertentu.

1.4 Tujuan Penelitian

Penelitian memiliki tujuan yakni merancang dan mengimplementasikan sistem kontrol speaker murottal al-qur'an berbasis *IoT* menggunakan algoritma *Aho-Corasick* dengan melakukan pengujian error sistem dan skala penilaian kegunaan sistem (*system usability scale*).

1.5 Manfaat Penelitian

Berdasarkan tujuan, Penelitian ini diharapkan memberikan beberapa manfaat sebagai berikut:

1. Memberikan keterlibatan dalam pengembangan *assistive technology* yang inklusif dan dapat diakses oleh berbagai kalangan, khususnya bagi penyandang disabilitas tunanetra.
2. Mendorong inovasi di bidang teknologi IoT dan Speech Recognition yang dapat diadaptasi untuk berbagai aplikasi lain dalam kehidupan sehari-hari.
3. Menjadikan penelitian ini sebagai referensi untuk penelitian lebih lanjut.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Peneliti-peneliti sebelumnya telah menjabarkan beberapa penelitian untuk bahan perbandingan dan referensi. Di samping itu, penelitian ini berupaya untuk menghindari adanya kemiripan dengan studi yang telah dilakukan sebelumnya. Pada bagian kajian pustaka, peneliti menyajikan berbagai hasil penelitian terdahulu sebagai berikut.

Terdapat beberapa penelitian sebelumnya yang menggunakan algoritma *Aho-Corasick* dalam pengaplikasian *string matching*. Salah satunya adalah penelitian oleh (Sulaeman & Murnawan, 2023) yang mengimplementasikan algoritma *Aho-Corasick* untuk fitur pencarian pada aplikasi yang dibuat yakni *Lost and Found*. Penelitian tersebut memberikan hasil yang membuktikan bahwa algoritma *Aho-Corasick* memiliki waktu pemrosesan yang lebih cepat dalam pencocokan kata kunci.

Penelitian lain memberikan hasil bahwa *Aho-Corasick* unggul dalam hal kecepatan dan efisiensi pencocokan pola, membuatnya lebih cocok untuk situasi di mana waktu pemrosesan adalah faktor utama. Algoritma *Aho-Corasick* mampu mengidentifikasi dan mencocokkan pola dengan cepat dan efektif, sehingga sangat berguna dalam aplikasi yang memerlukan pemulihan data yang cepat dan akurat. (Sahib et al., 2021).

Ditemukan dua penelitian lain yang membandingkan algoritma *Aho-Corasick* dengan algoritma *string matching* lainnya pada kurun waktu 5 tahun terakhir. Penelitian yang dilakukan oleh (B. P. Siahaan & Paradongan, 2020) membandingkan *performance* dua algoritma yakni *Aho-Corasick* dan *Knuth Morris*. Dinyatakan bahwa algoritma *Aho-Corasick* dinilai memiliki keunggulan yang signifikan dalam hal kecepatan pencocokan dan pencarian string dalam dokumen. Algoritma *Aho-Corasick* dapat memproses pencarian beberapa pola secara simultan dengan efisiensi tinggi, sementara *Knuth-Morris-Pratt* lebih fokus pada pencarian satu pola dalam satu waktu. Hasil penelitian menunjukkan bahwa *Aho-Corasick* lebih unggul dalam memfasilitasi pengguna untuk mencari informasi secara cepat dan efektif dalam dokumen yang besar.

Selain itu penelitian oleh (Kini et al., 2022) yang mengusulkan sistem pengamanan *website* dengan algoritma *Hashing* dan *Aho-Corasick*. Penelitian tersebut mengemukakan bahwa algoritma *Aho-Corasick* memiliki efisiensi yang lebih tinggi dalam hal waktu ketika digunakan untuk proses pencocokan string. Efisiensi ini terutama terlihat pada kemampuan *Aho-Corasick* dalam menangani beberapa pola pencarian sekaligus dengan cara yang optimal, di mana algoritma ini membangun sebuah automaton yang memungkinkan pencocokan simultan dari berbagai pola dengan kompleksitas waktu yang lebih rendah dibandingkan algoritma pencocokan string konvensional. Oleh karena itu, algoritma *Aho-Corasick* dapat mengurangi beban komputasi dan meningkatkan kinerja sistem pengamanan *website* secara keseluruhan.

Berdasarkan beberapa penelitian yang membahas algoritma *Aho-Corasick* di atas, belum ada yang meneliti terkait implementasinya dalam pencarian surah-surah Al-Qur'an. Sedangkan, penelitian yang dilakukan oleh (Kamel et al., 2023) yang membuat sistem *Qur'an Smart Speaker* tidak menggunakan algoritma apapun dalam pencarian *string matching*, sehingga terdapat perbedaan pada penelitian ini yang menggunakan algoritma *Aho-Corasick*. Selain itu, penelitian tersebut menggunakan *Raspberry Pi* sebagai otak dari sistem yang dibuat dan masih menggunakan kabel untuk terhubung ke speaker. Berbeda dengan penelitian ini yang menggunakan integrasi IoT dengan penggunaan ESP32 dan IR *Transmitter* yang ditembakkan pada speaker sehingga tidak memerlukan kabel dan dapat di kontrol secara jarak jauh.

Di sisi lain, terdapat penelitian yang membahas tentang pembuatan aplikasi Al-Qur'an untuk penyandang tunanetra yang dilakukan oleh (Salamah et al., 2020). Penelitian tersebut mirip dengan penelitian ini dalam hal pembuatan teknologi *assistive* yang membantu penyandang tunanetra berinteraksi dengan Al-Qur'an dengan input suara. Namun, penelitian tersebut tidak terfokus pada penggunaan algoritma tertentu dalam pencocokan *string matching*. Selain itu terdapat perbedaan yang sangat mencolok terkait jenis *hardware* atau alat yang dibuat yakni berupa *headset*.

Tabel 2. 1 Penelitian Terkait

No	Peneliti	Judul	Metode	Hasil
1.	Naufal Ferdy Sulaeman, Murnawan	Implementasi Algoritma Aho-Corasick pada Pencarian di Aplikasi Lost and Found	Aho-Corasick	Algoritma Aho-Corasick menghasilkan waktu 145.68 ms dibandingkan metode pencocokan kata kunci pattern yang tersedia di dalam database yang menghasilkan 459.93 ms. Hal ini terbukti bahwa pemrosesan dengan

No	Peneliti	Judul	Metode	Hasil
				algoritma Aho-Corasick lebih optimal.
2.	Nurul Hidayah Ab Rahman, Hussein Ismael Sahib, Ali Kazem Al-Qaysi, Mothana L. Attiah	Comparison Of Data Recovery Techniques On Master File Table Between Aho-Corasick and Logical Data Recovery Based On Efficiency	Aho-Corasick	Aho-Corasick unggul dalam hal kecepatan dan efisiensi pencocokan pola sehingga sangat berguna dalam aplikasi yang memerlukan pemulihan data yang cepat dan akurat
3.	Budi Prasetyo Siahaan, Paradongan	Implementasi Algoritma Aho-Corasick untuk Pencarian Document Full Teks Tipe Pdf	Aho-Corasick dan Knuth Morris.	Algoritma Aho-Corasick dinilai memiliki keunggulan yang signifikan dalam hal kecepatan pencocokan dan pencarian string dalam dokumen. Algoritma Aho-Corasick dapat memproses pencarian beberapa pola secara simultan dengan efisiensi tinggi, sementara Knuth-Morris-Pratt lebih fokus pada pencarian satu pola dalam satu waktu. Hasil penelitian menunjukkan bahwa Aho-Corasick lebih unggul dalam memfasilitasi pengguna untuk mencari informasi secara cepat dan efektif dalam dokumen yang besar.
4.	Shreya Kini, Anushka Parvate Patil, Kini M, Adithya Balasubramanyam	SQL Injection Detection and Prevention using Aho-Corasick Pattern Matching Algorithm	Hashing dan Aho-Corasick	Algoritma Aho-Corasick memiliki efisiensi yang lebih tinggi dalam hal waktu ketika digunakan untuk proses pencocokan string. Efisiensi ini terutama terlihat pada kemampuan Aho-Corasick dalam menangani beberapa pola pencarian sekaligus dengan cara yang optimal. Oleh karena itu, algoritma <i>Aho-Corasick</i> dapat mengurangi beban komputasi dan meningkatkan kinerja sistem pengamanan website secara keseluruhan.
5.	Ahmad Kamel, Elang Dardian Marindani, Herry Sujaini	Qur'an Smart Speaker with Voice Control Using	-	hasil pengujian menunjukkan rata-rata waktu deteksi kata bangun sekitar 0,86 detik, sedangkan waktu pengenalan

No	Peneliti	Judul	Metode	Hasil
		Raspberry Pi and Google Assistant Library		perintah sekitar 6,74 detik. Ketiga, dari beberapa jenis mikrofon yang diuji, mikrofon dengan amplifier terbukti menjadi pilihan terbaik. Meski memiliki beberapa kekurangan, seperti perlunya daya tambahan dan kepekaan terhadap noise, mikrofon ini cocok untuk kontrol pada jarak hingga 1 meter.
6.	Irma Salamah, Ade Silvia Handayani, Albertia Youlanda.	Aplikasi Untuk Al-Qur'an Audio Juz 30 Bagi Penyandang Tunanetra Menggunakan Voice Recognition Berbasis Android	-	Aplikasi Al-Qur'an khusus juz 30 menggunakan Arduino UNO dengan alat berupa headset

2.2 Speaker Murottal Al-Qur'an

Speaker atau pengeras suara merupakan alat yang menggetarkan udara melalui komponen berbentuk membran untuk mengubah sinyal elektrik menjadi frekuensi suara sehingga gelombang suara yang mencapai gendang telinga dapat terdengar sebagai bunyi. (Siliwangi et al., 2023). Salah satu modifikasi dalam pengembangan speaker adalah speaker yang dimuat surah-surah Al-Qur'an yakni Speaker Murottal Al-Qur'an.

Speaker murottal Al-Qur'an merupakan perangkat yang digunakan untuk memutar rekaman Al-Qur'an. Alat ini sangat berguna bagi orang yang ingin mendengarkan bacaan Al-Qur'an, baik untuk mempelajari tajwid, memperbaiki makhraj, maupun untuk tujuan spiritual lainnya. Speaker ini biasanya dilengkapi dengan berbagai fitur seperti pilihan surah dan ayat. Dalam konteks teknologi assistif, speaker murottal menjadi inovasi bagi penyandang tunanetra yang

kesulitan membaca Al-Qur'an secara langsung karena keterbatasan dalam navigasi atau pemilihan surah dan ayat (Kamel et al., 2023)

2.3 *Android*

Android merupakan sistem operasi mobile berbasis Linux yang dikembangkan oleh Google. OS ini dirancang khusus untuk perangkat seluler dengan layar sentuh, seperti smartphone dan tablet. *Android* sangat populer dan digunakan secara luas di seluruh dunia, terutama karena sifatnya yang *open-source* yang memungkinkan pengembang bebas mengakses dan memodifikasi kode sumbernya. Hal ini memungkinkan pengembangan aplikasi yang lebih beragam dan dapat diakses tanpa biaya tambahan, membuat smartphone berbasis *Android* umumnya lebih ekonomis dibandingkan dengan yang menggunakan sistem operasi berbayar. Dengan berbagai pilihan yang ditawarkan, *Android* mempermudah akses teknologi bagi lebih banyak orang dan memperluas kemampuan mereka untuk berkomunikasi secara luas (Marliana et al., 2023).

Pada tahun 2007 *Android* pertama kali diperkenalkan dengan versi *Beta*. Berkembang menjadi *Android* 1.0 (Astro) pada 23 September 2008 dan berkembang hingga versi terbaru yaitu *Android* 7.0 (Nougat) (Ulum, 2022). Dalam pengembangan aplikasi, *Android* menyediakan berbagai *Application Programming Interfaces* (API) dan alat bantu seperti *Android Studio* yang memudahkan pengembang untuk membuat aplikasi dengan fitur yang kaya dan antarmuka yang ramah pengguna. Aplikasi berbasis *Android* juga dapat memanfaatkan perangkat keras seperti mikrofon, kamera, dan sensor lainnya, memungkinkan integrasi teknologi canggih seperti pengenalan suara.

2.4 MIT App Inventor

Menurut (Satria et al., 2020) MIT App Inventor adalah platform yang memungkinkan pengembangan aplikasi secara visual, diciptakan oleh *Massachusetts Institute of Technology* bekerja sama dengan Google. Platform ini bertujuan untuk mengenalkan pengguna untuk membuat aplikasi *Android* dengan menggunakan blok-blok logika yang mudah dipahami. MIT App Inventor mendukung integrasi dengan berbagai komponen *hardware* dan API, termasuk sensor, kamera, dan fitur-fitur lainnya pada perangkat *Android*. Dalam penelitian ini, MIT App Inventor digunakan untuk mengembangkan aplikasi kontrol yang memungkinkan pengguna untuk memberikan perintah suara yang kemudian diterjemahkan menjadi aksi pada speaker murottal Al-Qur'an.

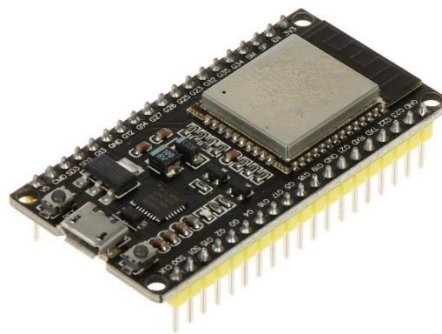
2.5 Arduino IDE

Arduino IDE merupakan perangkat lunak yang berfungsi untuk memprogram serta mengunggah kode ke papan mikrokontroler Arduino. Kode yang ditulis untuk Arduino sering disebut sebagai sketsa. Integrated Development Environment (IDE) ini merujuk pada alat pengembangan perangkat lunak terpadu yang menawarkan berbagai fitur pemrograman melalui antarmuka berbasis menu (Ulum, 2022).

2.6 ESP32

ESP32 merupakan mikrokontroler *System on Chip (SoC)* yang memiliki berbagai fitur, termasuk *WiFi*, *Bluetooth* versi 4.2, serta beragam *periferal* lainnya. Mikrokontroler ini unggul karena menyediakan komponen lengkap seperti prosesor, penyimpanan, dan akses ke *General Purpose Input Output (GPIO)*.

ESP32 juga dapat berperan sebagai alternatif pengganti Arduino, dengan kelebihan tambahan berupa dukungan koneksi langsung ke *WiFi* (Nizam et al., 2022). Mikrokontroler ESP32 tersedia dalam dua versi, yaitu dengan 30 pin GPIO dan 36 pin GPIO. ESP32 juga dilengkapi dengan interface USB to UART yang dapat dihubungkan dengan *software* Arduino IDE (Wagyana, 2019). Berikut gambar yang menunjukkan salah satu bentuk ESP32 pada gambar 2.1.



Gambar 2. 1 ESP32

2.7 *Internet of Things*

Menurut (Susanto et al., 2022) *Internet of Things* (IoT) adalah teknologi yang menghubungkan berbagai objek di sekitar kita melalui internet, sehingga aktivitas harian menjadi lebih praktis dan efisien. IoT memungkinkan benda-benda untuk saling berkomunikasi melalui jaringan internet. Penerapan IoT semakin meluas dalam berbagai aspek kehidupan, menunjukkan peran pentingnya. Teknologi ini juga membuka peluang dan gagasan baru dalam perkembangan di berbagai bidang, baik dalam skala kecil maupun besar.

Internet of Things bekerja dengan mengumpulkan dan mengirimkan informasi digital yang diperoleh dari beragam perangkat sensor. Berbagai sensor ini melibatkan teknologi seperti *Radio Frequency Identification* (RFID), sensor

inframerah (IR), dan *Global Positioning System* (GPS). Data yang dikumpulkan dari sensor-sensor tersebut diproses lebih lanjut guna memungkinkan interaksi dan kontrol yang lebih efektif terhadap objek atau sistem yang saling terhubung (Megawati & Lawi, 2021).

2.8 *String Matching*

String merupakan susunan karakter yang dapat terdiri dari huruf, angka, dan simbol lainnya. Implementasi string mencakup penggunaan dalam kata, frasa, dan kalimat. *String Matching* atau biasa dikenal pencocokan string sangat penting dalam proses pencarian karakter dalam teks, dokumen, dan berbagai sumber informasi lainnya (Anisa, 2024). *String matching* merupakan proses untuk menemukan sebuah pola atau biasa disebut *pattern*, (Sari & Ginting, 2021).

Dilihat dari garis besarnya teknik *string matching* dibedakan menjadi dua macam yakni, *Exact String Matching* dan *Inexact String Matching*. *Exact String Matching* merupakan proses yang memastikan bahwa susunan karakter dalam *string* yang dicari memiliki jumlah dan urutan yang identik atau sama dengan *string* target. Bagian algoritma ini dapat diimplementasikan ketika pengguna perlu menemukan *string* dalam dokumen yang benar-benar sesuai dengan *string* masukan. Adapun beberapa contoh algoritmanya seperti, *Brute Force*, *Knuth-Morris-Pratt*, *Boyer Moore* dan sebagainya. Sedangkan, *Inexact String Matching* lebih cocok untuk *string matching* secara samar atau keduanya hanya memiliki kemiripan tekstual maupun ucapan. Teknik ini bertujuan untuk menemukan nilai dari beberapa string yang hampir sama, bukan hanya menghasilkan kecocokan atau ketidakcocokan (Sari & Ginting, 2021).

2.9 Algoritma Aho-Corasick

Menurut (Pandey & Taruna, 2021) algoritma *Aho-Corasick* menjadi salah satu *Exact String Matching* yang dapat digunakan untuk menemukan beberapa pola dalam teks secara *real-time*. Algoritma ini menggunakan automata pengembangan dari *Knuth-Morris-Pratt Algorithm* yang membuatnya sangat efektif dalam mengidentifikasi kemunculan pola dalam waktu yang singkat. Proses pembentukan automata dalam algoritma *Aho-Corasick* dimulai dengan membuat sebuah pohon dengan akar yang kosong. Setiap kali sebuah pola cocok, *state* atau kondisi baru dibuat dalam pohon tersebut. Struktur pohon ini memungkinkan algoritma untuk dengan cepat dan efisien mencocokkan beberapa pola dalam teks. Selama segmen pencarian, mesin state dilalui dalam antisipasi mencapai state pencocokan yang sempurna. Ketika mesin state berada dalam state pencocokan yang sempurna, berarti pola yang dicari telah ditemukan dalam teks.

Algoritma *Aho-Corasick* memiliki kompleksitas waktu yang sangat efisien. Untuk tahap *preprocessing*, kompleksitas waktunya adalah $O(nk + m)$, di mana n adalah panjang teks, k adalah jumlah pola, dan m adalah total panjang semua pola. Untuk segmen pencarian, kompleksitas waktunya adalah $O(n + m + z)$, di mana z adalah jumlah total kemunculan semua pola dalam teks (Pandey & Taruna, 2021). Pemilihan algoritma *Aho-Corasick* dalam penelitian ini didasarkan pada beberapa alasan kuat yang mendukung efisiensi dan efektivitas dalam menemukan pola-pola dalam teks. Beberapa penelitian terdahulu dilakukan oleh (Trivedi, 2020), (Sulaeman & Murnawan, 2023) dan (Azarudeen et al., 2021) memberikan referensi

penggunaan algoritma string matching yang efisien dalam kasus pencarian ayat-ayat Al-Qur'an.

2.10 *Speech Recognition*

Speech Recognition merupakan proses mengenali suara berdasarkan kata-kata yang diucapkan, dengan cara mengubah sinyal akustik yang diterima oleh perangkat audio (Rabbani, 2021). Dalam penelitian ini digunakan *Google Speech Recognition* untuk *convert* suara menjadi teks. Penggunaan *Google Speech Recognition* mempermudah pengguna dalam mengontrol speaker murottal Al-Qur'an dengan mengucapkan perintah.

Pada penelitian yang dilakukan oleh (Wisnu, 2024) menjelaskan bahwa *Speech Recognition* memiliki dua komponen utama yakni *front-end* dan *back-end*. Bagian *Front-end* bertujuan untuk mengubah suara menjadi sinyal digital. Sedangkan, bagian *back-end* memiliki kegunaan masing-masing sebagai berikut.

a. *Acoustic Model*

Proses ini mencocokkan suara kata-kata yang diucapkan dengan suku kata yang serupa.

b. *Language Model*

Mengubah suku kata yang dikenali oleh acoustic model menjadi kata-kata yang sesuai dengan ucapan pengguna.

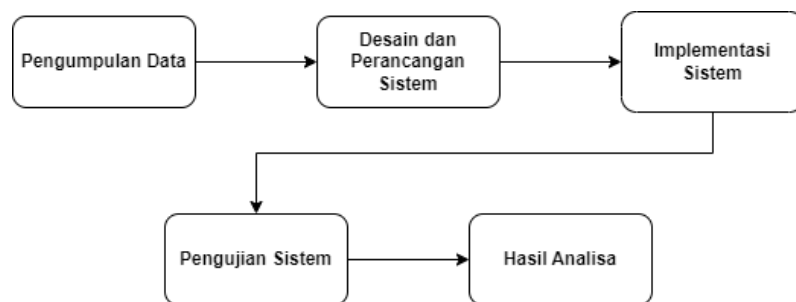
c. *Lexicon*

Berfungsi sebagai kamus yang menyediakan kata-kata beserta cara pengucapannya.

BAB III

METODE PENELITIAN

Pada bab ini dibahas secara rinci alur penelitian dari sistem kontrol speaker murottal Al-Quran yang dimulai dengan pengumpulan data sebagai fondasi utama, dilanjutkan dengan desain dan perancangan sistem untuk menentukan arsitektur dan fungsionalitas yang diperlukan. Tahap berikutnya adalah implementasi sistem dari desain yang telah dirancang diubah menjadi bentuk nyata. Selanjutnya, dilakukan pengujian sistem untuk memastikan kinerjanya, diakhiri dengan hasil analisa untuk mengevaluasi efektivitas dan efisiensi sistem yang telah dibuat. Penjelasan setiap tahap ini bertujuan untuk memberikan gambaran komprehensif mengenai proses penelitian yang dilakukan. Tahapan-tahapan ini disusun secara sistematis untuk memastikan bahwa penelitian berjalan dengan terstruktur dan efektif. Adapun alur penelitian dalam pembuatan sistem kontrol ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Alur Penelitian

3.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini menggunakan dua jenis yakni primer dan sekunder. Data primer diperoleh melalui tahap uji coba alat berupa

pengujian error sistem dan pengujian SUS kepada penyandang tunanetra di PERTUNI (Persatuan Tunanetra Indonesia) yang berada di Kota Malang. Adapun data sekunder diperoleh dari studi literatur dari jurnal ilmiah, penelitian sebelumnya terkait teknologi IoT dan algoritma Aho-Corasick, dan murottal Al-Qur'an Juz 30 yang sudah tersimpan di dalam *SD card*. Data murottal ini digunakan untuk memutar surah tertentu sebagai respons terhadap perintah suara yang diberikan oleh pengguna. Surah-surah Al-Qur'an yang dimasukkan dalam *pattern* pada penelitian ini terdapat pada Tabel 3.1.

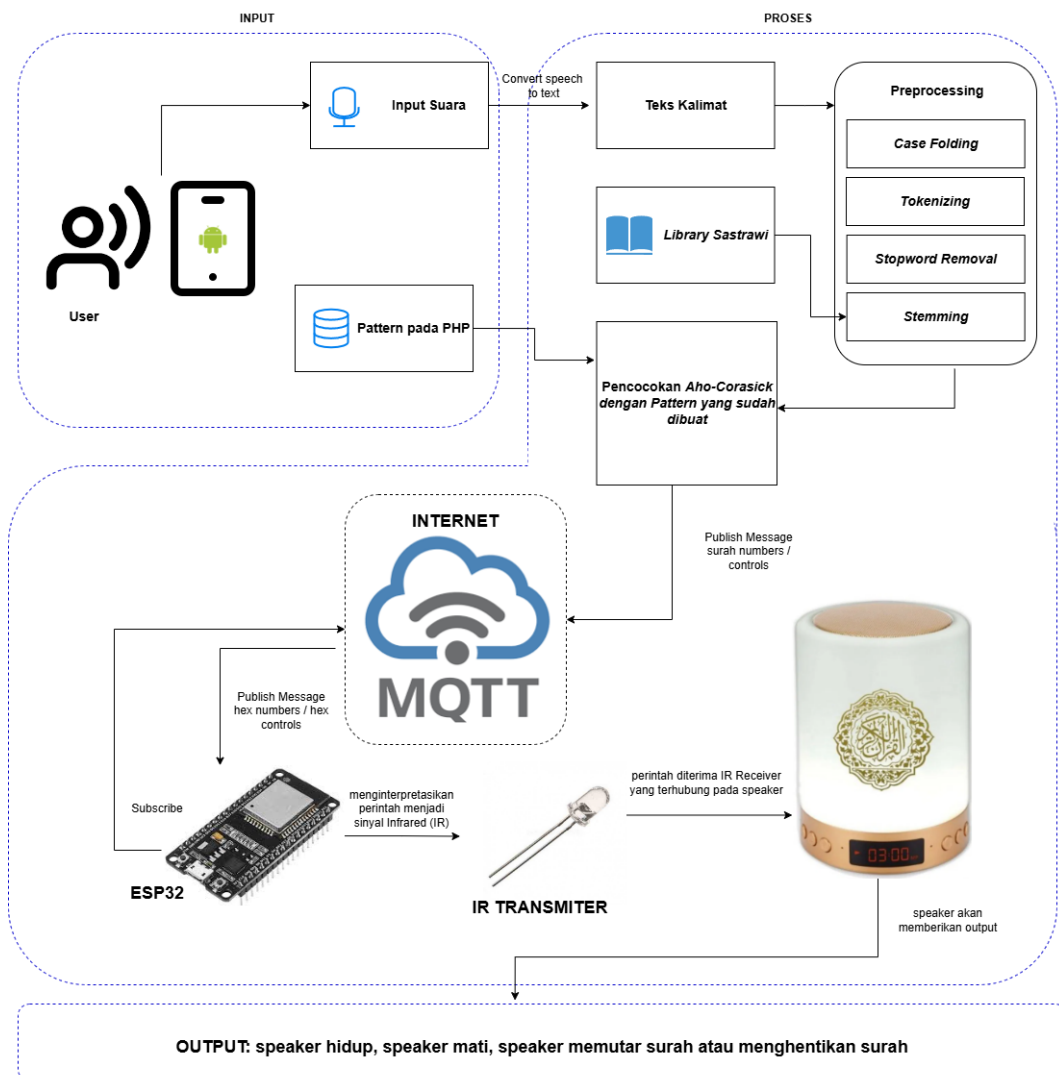
Tabel 3. 1 Daftar Surah Juz 30

No	Nama Surah	Nomor Surah
1.	An-Naba	Surah ke-78
2.	An-Nazi'at	Surah ke-79
3.	Abasa	Surah ke-80
4.	At-Takwir	Surah ke-81
5.	Al-Infitar	Surah ke-82
6.	Al-Mutaffifin	Surah ke-83
7.	Al-Insyiqaq	Surah ke-84
8.	Al-Buruj	Surah ke-85
9.	At-Tariq	Surah ke-86
10.	Al-A'la	Surah ke-87
11.	Al-Ghashiyah	Surah ke-88
12.	Al-Fajr	Surah ke-89
13.	Al-Balad	Surah ke-90
14.	Ash-Shams	Surah ke-91
15.	Al-Lail	Surah ke-92
16.	Ad-Duha	Surah ke-93
17.	Al-Inshirah	Surah ke-94
18.	At-Tin	Surah ke-95
19.	Al-Alaq	Surah ke-96
20.	Al-Qadr	Surah ke-97
21.	Al-Bayyinah	Surah ke-98
22.	Az-Zalzalah	Surah ke-99
23.	Al-Adiyat	Surah ke-100
24.	Al-Qari'ah	Surah ke-101
25.	At-Takathur	Surah ke-102
26.	Al-Asr	Surah ke-103
27.	Al-Humazah	Surah ke-104
28.	Al-Fil	Surah ke-105
29.	Quraishy	Surah ke-106
30.	Al-Ma'un	Surah ke-107
31.	Al-Kausar	Surah ke-108
32.	Al-Kafirun	Surah ke-109

No	Nama Surah	Nomor Surah
33.	An-Nasr	Surah ke-110
34.	Al-Lahab	Surah ke-111
35.	Al-Ikhlâs	Surah ke-112
36.	Al-Falaq	Surah ke-113
37.	An-Nas	Surah ke-114

3.2 Desain dan Perancangan Sistem Kontrol

Dalam penelitian ini, sistem kontrol terdiri dari tiga tahapan utama yakni input, proses, dan output. Adapun desain sistem dalam penelitian ini dapat dilihat pada gambar 3.2.



Gambar 3. 2 Desain Sistem

3.2.1 Tahapan Input

Dalam tahapan input, pengguna dapat memberikan perintah suara dengan memanfaatkan fitur *Google Speech Recognition* pada aplikasi yang di buat. Adapun aplikasinya berbasis *android*. Perintah suara yang diinputkan akan diubah menjadi teks dengan bantuan fitur *Speech Recognizer* di *MIT APP Inventor*.

3.2.2 Tahapan Proses

Tahapan proses dalam perancangan sistem kontrol yang dibuat adalah *preprocessing text* yang terdiri dari *case folding*, *tokenizing*, *stopword removal*, dan *stemming*. Setelah teks di *preprocessing* maka teks sudah siap untuk dilakukan pencocokan pola menggunakan algoritma *Aho-Corasick*. Pencocokan pola dilakukan dengan pattern yang sudah ada di dalam *code PHP*. Berikut penjabaran terkait tahapan proses dalam perancangan sistem kontrol.

3.2.2.1 Case Folding

Case Folding merupakan tahapan *preprocessing text* yang bertujuan untuk menyamaratakan semua kata dalam teks menjadi huruf kecil(Darmawan et al., 2023). Tahap ini bermanfaat untuk menghilangkan redunansi data sehingga data lebih optimal pada proses pengujian. Dalam penelitian ini tahap *case folding* digunakan pada inputan yang sudah di *convert* dari suara ke bentuk text. Berikut contoh *case folding* yang dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Case Folding

Sebelum Case Folding	Sesudah Case Folding
Nyalakan Speaker ya	nyalakan speaker ya
Matikan Speaker dong	matikan speaker dong
Tolong Nyalakan Speaker	tolong nyalakan speaker
Memberhentikan Surah Al-Ikhlas	memberhentikan surah al-ikhlas

Sebelum Case Folding	Sesudah Case Folding
Putarkan Surah Al-Ikhlas	putarkan surah al-ikhlas

3.2.2.2 Tokenizing

Tokenizing merupakan *preprocessing text* yang dilakukan dengan pemisahan kalimat menjadi per-kata (Darmawan et al., 2023). Berikut contoh tokenizing pada penelitian ini.

Tabel 3. 3 Tokenizing

Sebelum Tokenizing	Sesudah Tokenizing
nyalakan speaker ya	'nyalakan', 'speaker', 'ya'
matikan speaker dong	'matikan', 'speaker', 'dong'
tolong nyalakan speaker	'tolong', 'nyalakan', 'speaker'
memberhentikan surah al-ikhlas	'memberhentikan' 'surah', 'al-ikhlas'
putarkan surah al-ikhlas	'putarkan', 'surah', 'al-ikhlas'

3.2.2.3 Stopword Removal

Pada proses ini dilakukan eliminasi kata-kata *stoplist* yang sering muncul namun tidak terlalu penting atau tidak terkait dengan tema spesifik (Wisnu, 2024). Kata yang dieliminasi akan dibandingkan dengan daftar kata stoplist yang telah ada pada library sastraswi. Berikut contoh penggunaan *stopword removal* pada penelitian ini dapat dilihat pada 3.4.

Tabel 3. 4 Stopword Removal

Sebelum Stopword Removal	Sesudah Stopword Removal
'nyalakan', 'speaker', 'ya'	nyalakan, speaker
'matikan', 'speaker', 'dong'	matikan speaker
'tolong', 'nyalakan', 'speaker'	tolong nyalakan speaker
'memberhentikan' 'surah', 'al-ikhlas'	memberhentikan surah al-ikhlas
'putarkan', 'surah', 'al-ikhlas'	putarkan surah al-ikhlas

3.2.2.4 Stemming

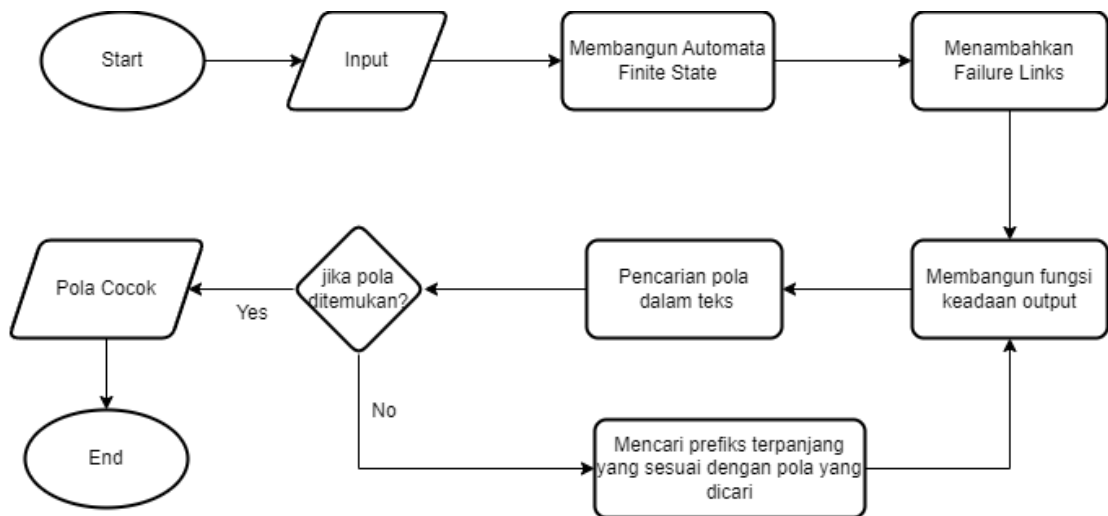
Pada proses ini dilakukan pengubahan kata yang memiliki imbuhan menjadi kata dasar. Sebagai contoh, kata nyalakan, menyalakan, menyala, akan diubah ke bentuk kata dasarnya “nyala”. Adapun proses pemogramannya digunakan *library sastrawi*. *Stemming* ini bertujuan untuk mencocokkan variasi kata agar tidak muncul variasi kata yang terlalu banyak. Berikut contoh proses stemming pada penelitian ini yang ditampilkan pada tabel 3.5.

Tabel 3. 5 Stemming

Teks Sebelum <i>Stemming</i>	Text Sesudah <i>Stemming</i>
nyalakan, speaker	nyala speaker
matikan, spreaker	mati spreaker
tolong, nyalakan, speaker	tolong nyala speaker
memberhentikan, surah, al-ikhlas	henti surah
putarkan, surah, al-ikhlas	putar surah al-ikhlas

3.2.2.5 Pencocokan Algoritma Aho-Corasick

Algoritma Aho-Corasick memiliki urutan tahapan-tahapan sistematis dalam penyelesaian masalah. Algoritma ini memiliki tahap *preprocessing* yang terdiri dari *go*, *failure*, dan *output*. *Go* adalah buka fungsi utama dan setel node dan root utama. *Failure* adalah ketika kata kunci tidak cocok maka gagal dihitung. *Output* adalah hasil setelah kondisi ketersediaan cocok (Azarudeen et al., 2021). Berikut cara kerja Aho-corasick yang dapat dilihat pada gambar 3.3.



Gambar 3. 3 Cara Kerja Aho-Corasick

Adapun contoh penggunaan *string matching* menggunakan algoritma *Aho-Corasick* dapat dilihat dari beberapa tahapan berikut.

a. Tahap awal

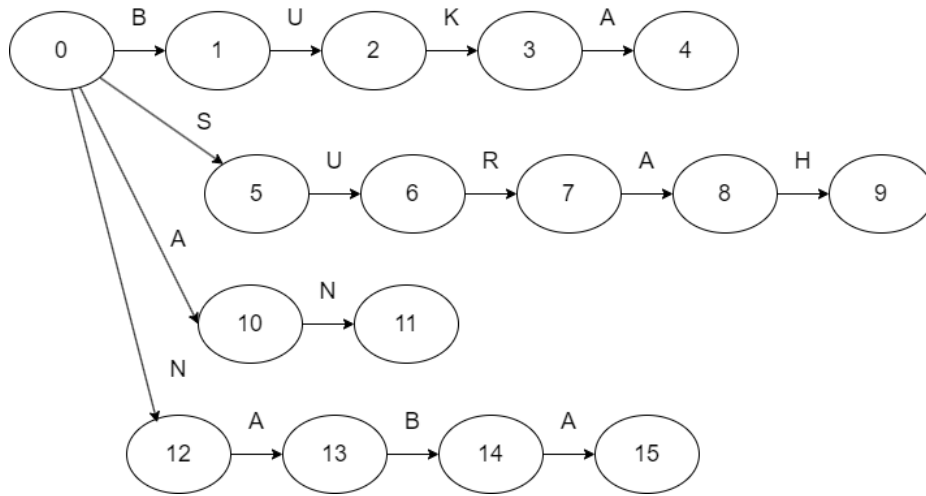
Tahap awal dilakukan proses pembentukan *trie* dilakukan dengan menggunakan pola (*pattern*) yang telah diberikan. Pola ini berfungsi sebagai panduan dalam membangun struktur *trie*, memastikan setiap *node* dan cabang diatur sesuai dengan urutan karakter dalam pola.

Text:

BUKA SURAH AN NABA

Pattern:

BUKA, SURAH, AN, NABA



Gambar 3. 4 Tree Aho-Corasick

b. Tahap *Failure*

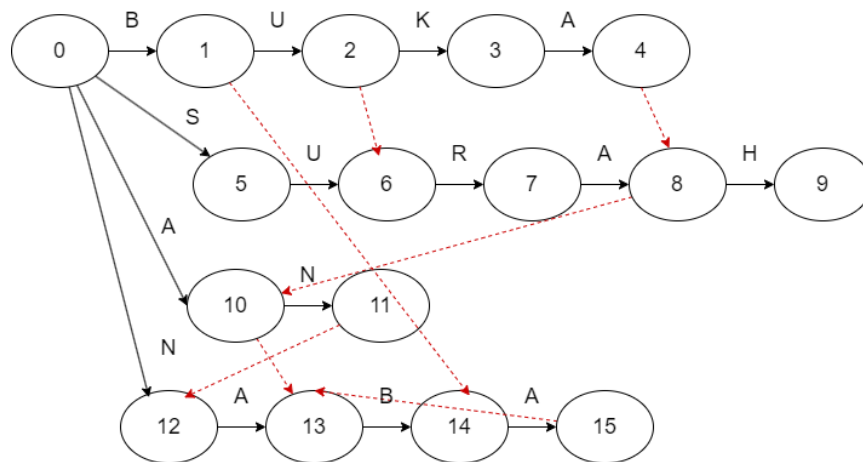
Pada tahap ini, proses dilakukan dengan pembentukan tabel transisi dari automata (*multiple pattern tree*), pemeriksaan trie dan penyusunan tabel *failure function* untuk memastikan setiap kata kunci terlampir dengan benar. Tabel *failure function* yang dihasilkan akan mempermudah identifikasi dan pencocokan kata kunci dalam proses pencarian string.

Tabel 3. 6 Tabel Transisi

State	Input							
	B	U	K	A	S	R	H	N
0	1	-	-	-	-	-	-	-
1	-	2	-	-	-	-	-	-
2	-	-	3	-	-	-	-	-
3	-	-	-	4	-	-	-	-
4	-	-	-	-	5	-	-	-
5	-	6	-	-	-	-	-	-
6	-	-	-	-	-	7	-	-
7	-	-	-	8	-	-	-	-
8	-	-	-	-	-	-	9	-
9	-	-	-	10	-	-	-	-
10	-	-	-	-	-	-	-	11
11	-	-	-	-	-	-	-	12
12	-	-	-	13	-	-	-	-

State	Input							
13	14	-	-	-	-	-	-	-
14	-	-	-	15	-	-	-	-

Setelah tabel transisi dari automata dibuat, langkah selanjutnya adalah memeriksa *trie* dan menyusun tabel fungsi kegagalan (*failure function*) yang dapat dilihat pada gambar 3.5 dan tabel 3.7.



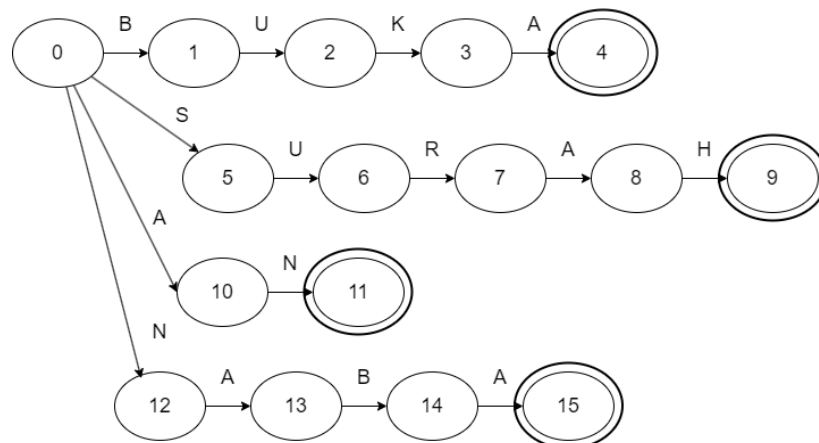
Gambar 3. 5 Failure Function

Tabel 3. 7 Failure Function

Node	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Failure	0	14	6	0	8	0	0	0	10	0	13	12	0	0	0	13

c. Tahap Output Aho-Corasick

Setelah teks diproses, cari semua simpul akhir dalam automaton yang berisi pola yang cocok. Simpul akhir adalah tempat di mana pencocokan pola berakhir. Identifikasi dan kumpulkan pola-pola yang terhubung ke simpul akhir tersebut. Outputkan hasil pencocokan untuk semua pola yang ditemukan dalam teks.



Gambar 3. 6 Output Aho-Corasick

3.2.3 Tahap Output

Jika kondisi pola dikenali, sistem akan mempersiapkan pesan untuk dikirim ke broker MQTT (*publish*) berupa kode angka atau kode kontrol. Setelah itu, ESP32 yang terhubung ke broker MQTT dan *subscribe* pada topik yang sama. ESP32 akan mengubah perintah yang diterima menjadi sinyal yang dapat dipahami oleh IR *Transmitter*. Ini dilakukan dengan mengeksekusi kode hex yang mengatur pengiriman sinyal IR. ESP32 akan mengendalikan pin GPIO yang terhubung ke IR *Transmitter*. Ketika perintah diterima, pin ini akan dinyalakan dan dimatikan sesuai dengan pola sinyal IR yang diinginkan. Sinyal dikirimkan oleh IR *Transmitter* yang ditangkap oleh IR *Receiver* yang terhubung dengan speaker murottal Al-Qur'an yang akan menginterpretasikan sinyal tersebut dan melakukan sesuai perintah.

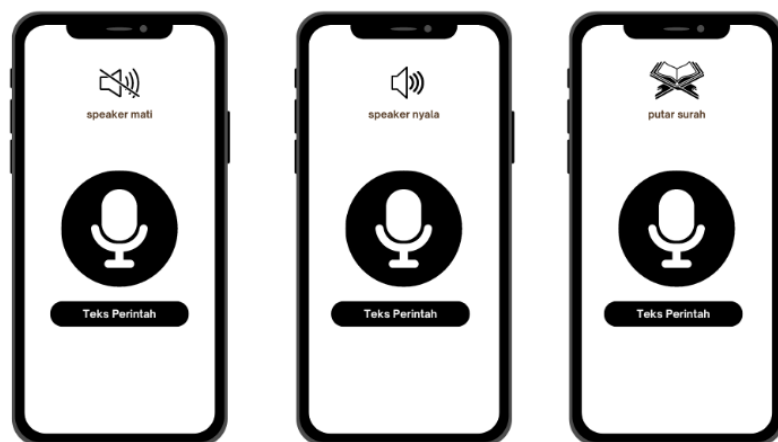
3.2.4 Desain Aplikasi

Pada tahap ini dilakukan perancangan desain antarmuka (*interface*) aplikasi untuk mempermudah proses pembuatan sistem dan memastikan bahwa tujuan serta fungsi setiap tampilan dapat dipahami dengan jelas. Desain antarmuka berfokus

pada kemudahan penggunaan dan efektivitas dalam mencapai tujuan penelitian. Berikut adalah rancangan tampilan aplikasi yang dirancang untuk penelitian ini yang dapat dilihat pada gambar 3.7 dan 3.8.



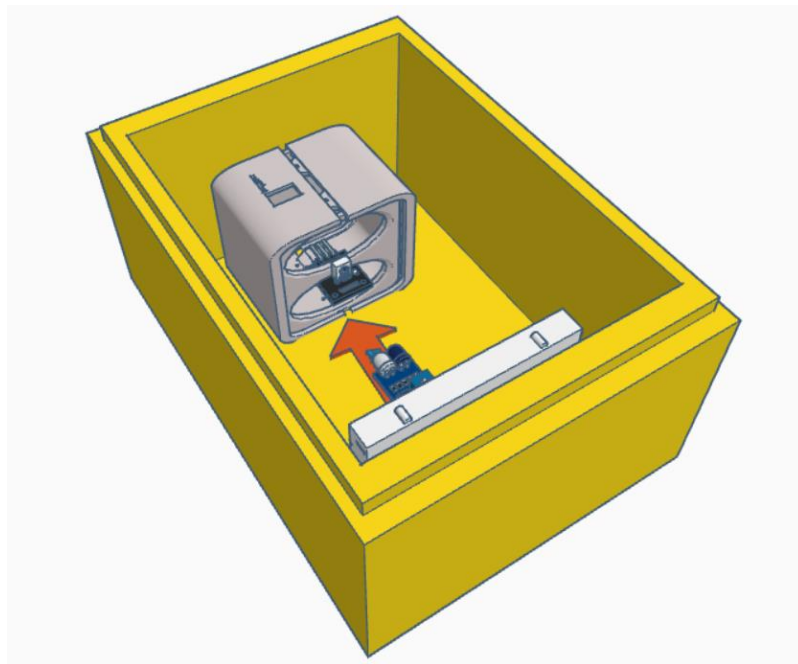
Gambar 3. 7 Rancangan Tampilan Awal Aplikasi



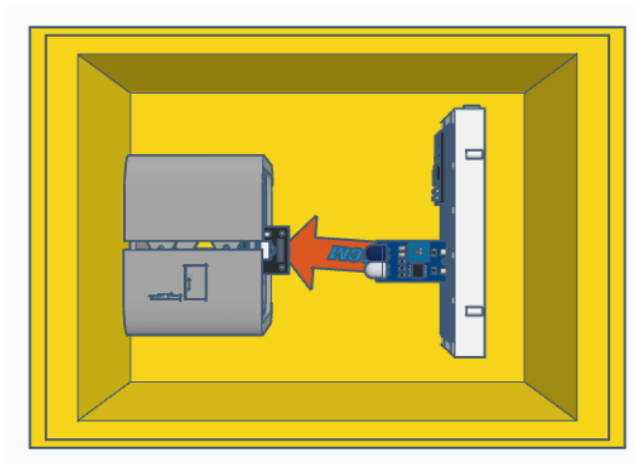
Gambar 3. 8 Rancangan Tampilan Setelah Input Suara

3.2.5 Perancangan Alat Sistem Kontrol

Adapun rancangan komponen yang digunakan dalam penelitian ini terdiri dari ESP32, *Infrared Transmitter*, *Breadboard*, dan Speaker Murottal Al-Qur'an yang dapat dilihat pada gambar 3.9. dan 3.10.



Gambar 3. 9 Rancangan Komponen Tampak Samping



Gambar 3. 10 Rancangan Komponen Tampak Atas

Pada gambar 3.9 dan 3.10 menunjukkan sebuah rancangan komponen yang terdiri dari ESP32 dan sensor *infrared* (IR) yang ditempatkan pada breadboard dalam sebuah kotak (*box*). Sensor IR ini berfungsi untuk mengambil kode *hexadecimal* (hex) dari setiap surah yang ada pada speaker murottal yang terletak 5 cm dari sensor. ESP32 bertindak sebagai pengontrol utama yang menerima data

dari sensor IR dan memproses kode hex yang diterima. Rancangan komponen ini dibuat agar sistem dapat mendeteksi perintah dari pengguna sehingga speaker dapat melakukan respon sesuai perintah seperti mematikan speaker, menghidupkan speaker, memutar surah atau menghentikan surah.

3.3 Implementasi Sistem

Dalam implementasinya, sistem kontrol ini terdapat 2 perangkat yang saling terhubung yakni *software* dan *hardware*. *Software* yang dibuat berbasis *android* akan mengirimkan data perintah yang diinputkan, kemudian diproses oleh sistem menggunakan algoritma Aho-Corasick. Pemanfaatan algoritma Aho-Corasick dipilih karena kemampuannya dalam mencari beberapa kata kunci sekaligus dalam satu proses pencarian yang efektif untuk mengenali perintah-perintah suara yang berbeda yang dimasukkan oleh pengguna. Setelah itu, sistem akan mengirimkan kode surah dan kontrol ke MQTT yang menjadikan sistem kontrol ini terhubung dengan *Internet of Things*. Penerapan *Internet of Things* (IoT) pada sistem ini memungkinkan kontrol jarak jauh dan interaksi antara perangkat yang terhubung melalui jaringan internet.

Adapun *hardware* atau alat yang dibuat pada penelitian ini digunakan untuk menerima perintah dari *software*. Ketika perintah diterima dari aplikasi *android*, mikrokontroler akan memproses data tersebut, mengidentifikasi perintah berupa mematikan, menyalakan, menaikkan volume, menurunkan volume, menghentikan surah atau memutar surah yang diminta, dan mengirimkan kode hex melalui *IR Transmitter* ke *IR Receiver* yang terhubung ke speaker. Sehingga speaker dapat memberikan output sesuai perintah.

3.4 Pengujian Sistem

Pengujian sistem dilakukan dengan dua macam. Pengujian pertama dilakukan untuk mengetahui *error* sistem. Pengujian kedua dilakukan dengan *Sistem Usability Scale* (SUS) berdasarkan skala *likert* yang ditujukan kepada penyandang disabilitas tunanetra.

3.4.1 Perhitungan *Error* Sistem

Pada tahap ini, dilakukan pengujian terhadap sistem untuk mengukur tingkat kesalahan dalam pengenalan dan pemrosesan perintah suara. Pengujian ini bertujuan untuk mengevaluasi seberapa besar *error* sistem yang dihasilkan dalam menjalankan fungsi-fungsi yang diharapkan, seperti mengubah surah atau memutar ayat Al-Qur'an sesuai dengan perintah suara yang diberikan. Berikut rumus perhitungan rata-rata *error* dalam setiap pengujian sistem yang ditunjukkan pada persamaan 3.1.

$$\mu = \frac{\sum Ei}{n} \times 100\% \quad (3.1)$$

Keterangan:

μ : rata-rata error dalam bentuk persen (%)

n : jumlah percobaan

$\sum Ei$: jumlah kesalahan dari semua percobaan

Proses pengujian pada penelitian ini dilakukan sebanyak 50 kali dengan merekap keberhasilan atau kegagalan di setiap percobaan. Dalam setiap uji coba, sistem dianggap berhasil jika perintah suara dikenali dan dijalankan dengan benar. Dianggap gagal jika terjadi kesalahan dalam pengenalan atau pelaksanaan perintah.

Dikarenakan pengujian terfokus untuk mencari rata-rata *error* sistem maka, setiap pengujian berhasil diberikan angka 0 dan pengujian gagal diberikan angka 1.

Hasil pengujian ini kemudian digunakan untuk menghitung rata-rata *error* sistem. Rata-rata kesalahan dihitung dengan membagi jumlah kesalahan yang terjadi dengan total jumlah pengujian yang dilakukan, kemudian dikalikan dengan 100% untuk mendapatkan persentase rata-rata *error*. Data ini penting untuk memahami tingkat keandalan sistem yang memerlukan perbaikan lebih lanjut. Berikut contoh pengujian rata-rata *error* sistem dalam 15 kali percobaan.

Tabel 3. 8 Contoh Pengujian Error Sistem

Percobaan	Teks Input	Hasil Pengujian	Jumlah Kesalahan (Error)
1.	Nyalakan speaker	Benar	0
2.	Matikan speaker	Benar	0
3.	Tolong nyalakan speaker	Benar	0
4.	Stop surah Al-ikhlas	Benar	0
5.	Putarkan surah Al-ikhlas	Benar	0
6.	Tolong nyalain speaker dong	Salah	1
7.	Matiin speaker dong	Benar	0
8.	Ganti surah ke Al-kafirun	Benar	0
9.	Al-maun stop	Salah	1
10.	surah Al-ikhlas segera diputar	Salah	1
11.	Speaker menyala	Benar	0
12.	Speaker ayo mati	Salah	1
13.	Tolong ubah surah jadi An-naba	Benar	0
14.	Hentikan surah Al-ikhlas	Benar	0
15.	Putarkan surah An-Naba	Benar	0
ΣEi (kesalahan dari semua percobaan)			4

Pada tabel di atas terdapat beberapa kesalahan yang dilakukan sistem. Adapun rata-rata *error* dari contoh percobaan di atas adalah sebesar 26,66% yang ditunjukkan pada persamaan 3.2.

$$\mu = \frac{\Sigma Ei}{n} \times 100\% \quad (3.2)$$

$$\mu = \frac{4}{15} \times 100\%$$

$$\mu = 26,66\%$$

3.4.2 Sistem Usability Scale (SUS)

Pada penelitian ini, selain menggunakan perhitungan *error* sistem juga terdapat pengujian skala kegunaan sistem. Metode *System Usability Scale (SUS)* merupakan salah satu metode dalam evaluasi tingkat implementasi suatu sistem atau aplikasi dengan penggunaan kuesioner (Ramadhan et al., 2019). Kuesioner yang digunakan pada penelitian ini terdiri dari 10 pernyataan yang mencakup aspek-aspek penting dari termasuk kemudahan penggunaan, efisiensi, dan kepuasan dari sistem yang dibuat. Pengujian *System Usability Scale (SUS)* diharapkan memberikan gambaran tentang sejauh mana sistem dapat memenuhi harapan pengguna.

Tabel 3. 9 Kuesioner SUS Penelitian

No	Pernyataan
1.	Saya merasa sistem ini akan sering digunakan tunanetra
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.
3.	Saya merasa nyaman menggunakan sistem ini
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.
10.	Saya merasa sistem kontrol speaker ini membingungkan

Penelitian ini juga penilaian oleh responden menggunakan skala *likert* dengan 5 skala poin yang dapat dilihat pada tabel 3.10.

Tabel 3. 10 Skala Likert

Skala Poin	Keterangan
1.	Sangat Tidak Setuju

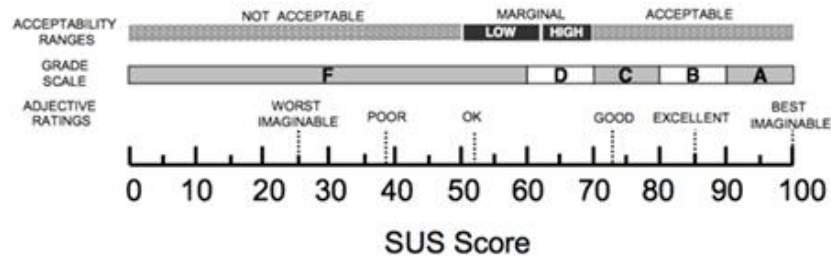
2.	Tidak Setuju
3.	Netral
4.	Setuju
5.	Sangat Setuju

Pemilihan skala *likert* 5 poin didasarkan pada kemudahan dan kejelasan dalam pengisian kuesioner oleh responden (Nur Kholifah et al., 2023). Skala ini memungkinkan responden untuk mengekspresikan tingkat persetujuan atau ketidaksetujuan mereka dengan lebih mudah dan jelas. Skala 5 poin juga memberikan variasi yang cukup dalam pilihan tanpa membingungkan responden dengan terlalu banyak opsi sehingga dapat menghasilkan data yang lebih baik untuk diolah.

Pada penelitian yang dilakukan oleh (Ramadhan et al., 2019) menguraikan bahwa terdapat sejumlah ketentuan atau aturan penting dalam perhitungan nilai SUS. Ketentuan-ketentuan ini harus diperhatikan saat menghitung skor pada kuesioner:

1. Skor pengguna akan dikurangi 1 poin untuk setiap pertanyaan yang memiliki nilai positif. Dalam penelitian ini, pertanyaan dengan nilai positif adalah pertanyaan dengan nomor ganjil, yaitu nomor 1, 3, 5, 7, dan 9.
2. Setiap pertanyaan dengan nilai negatif akan dihitung dengan mengurangi skor pengguna dari angka 5. Dalam penelitian ini, pertanyaan yang bernilai negatif adalah pertanyaan bernomor genap, yaitu 2, 4, 6, 8, dan 10.
3. Skor yang didapatkan pada hasil penjumlahan skor dari poin 1 dan 2 di atas akan dikalikan dengan 2,5.

Ketentuan ini diterapkan untuk setiap responden yang berpartisipasi. Adapun hasil dari perhitungannya akan bernilai dengan rentang antara 0 hingga 100.



Gambar 3. 11 SUS Score

Hal ini dijelaskan dalam penelitian yang dilakukan oleh (Nuryasin & Ferina, 2024) dimana hasil pengujian akan dikategorikan berdasarkan penilaian tertentu yang dapat dilihat pada tabel 3.11.

Tabel 3. 11 Kategorisasi Hasil Pengujian SUS

Jenis Kategori	Rentang Nilai	Keterangan
Kategori A	> 80,3	Excellent
Kategori B	68-80,3	Good
Kategori C	68	OK
Kategori D	51-67	Poor
Kategori F	< 51	Awful

Kemudian, skor untuk setiap jawaban responden akan dihitung dengan menjumlahkan seluruh skor, lalu mencari rata-ratanya dengan membagi total skor tersebut dengan jumlah responden. Aturan perhitungan skor SUS yang telah ditentukan dapat dilihat pada persamaan 3.3.

$$SUS\ Score = ((Q1 - 1) + (5 - Q2) + (Q3 - 1) + \dots + (5 - Q10) * 2,5) \quad (3.3)$$

Keterangan:

SUS Score = nilai *System Usability Scale*

Q =: pertanyaan

Adapun rumus rata-rata yang digunakan setelah didapatkan skor SUS dapat dilihat pada persamaan 3.4.

$$\mu SUS = \frac{\Sigma SUS Score}{nR} \quad (3.4)$$

Keterangan:

μSUS = nilai rata-rata *System Usability Scale*

$\Sigma SUS Score$ = jumlah nilai *System Usability Scale*

nR = banyaknya responden

Berikut contoh pengujian SUS dengan skala *likert* pada penelitian ini dengan 5 responden disajikan dalam bentuk tabel sebagai berikut.

Tabel 3. 12 Contoh Skor Jawaban Kuesioner

Responden	Contoh Skor Jawaban Kuesioner									
	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇	Q ₈	Q ₉	Q ₁₀
Responden 1	4	1	5	1	3	3	3	1	3	5
Responden 2	5	1	5	2	3	3	4	1	3	5
Responden 3	5	1	5	1	5	3	4	1	4	5
Responden 4	5	1	5	2	5	2	3	2	5	5
Responden 5	3	1	5	3	5	1	3	1	5	4

Setelah memperoleh skor asli dari jawaban kuesioner, perhitungan dilakukan sesuai aturan dan ketentuan yang telah ditetapkan sebelumnya untuk mendapatkan nilai SUS. Hasil perhitungan ini disajikan dalam tabel 3.13.

Tabel 3. 13 Contoh Perhitungan Skor SUS

Responden	Contoh Perhitungan Skor SUS										Jumlah	Total skor (*2,5)
	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇	Q ₈	Q ₉	Q ₁₀		
Responden 1	3	4	4	4	2	2	2	4	2	0	27	67.5
Responden 2	4	4	4	3	2	2	3	4	2	0	28	70
Responden 3	4	4	4	4	4	2	3	4	3	0	32	80
Responden 4	4	4	4	3	4	3	2	3	4	0	31	77.5
Responden 5	2	4	4	2	4	4	2	4	4	1	31	77.5
Jumlah											149	372.5
Rata-rata SUS											74.5	

Berdasarkan tabel 3.13, didapatkan hasil akhir dari perhitungan SUS adalah 74,5. Hal ini dapat diartikan bahwa sistem kontrol speaker murottal Al-Qur'an yang dibuat termasuk ke dalam kategori B dengan penilaian "*Good*" yang menunjukkan bahwa sistem ini memenuhi kebutuhan pengguna dalam hal *usability*.

3.5. Hasil Analisa Sistem

Setelah melakukan serangkaian pengujian terhadap sistem kontrol speaker murottal Al-Qur'an berbasis *Internet of Things* menggunakan algoritma *Aho-Corasick*, peneliti akan mengidentifikasi beberapa faktor yang mempengaruhi performa sistem. Hasil pengujian nantinya dijadikan dasar untuk pemahaman yang lebih mendalam tentang kelebihan dan kekurangan sistem, serta potensi perbaikan di penelitian selanjutnya.

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Hasil Implementasi Sistem Kontrol

Sistem kontrol yang dibuat diimplementasikan sesuai dengan perancangan yang telah dijelaskan pada bab sebelumnya. Sistem kontrol ini terdiri dari dua komponen utama, yaitu *software* yang mencakup aplikasi berbasis *Android* yang dilengkapi dengan fitur *speech recognition* untuk menerima perintah suara dari pengguna serta *hardware* yang mencakup speaker dan beberapa komponen robotika. Berikut dipaparkan secara rinci sistem diimplementasikan dari segi aplikasi, konektivitas, serta rangkaian *hardware* yang digunakan.

4.1.1 Halaman *Software*

Halaman awal aplikasi menampilkan logo aplikasi yakni *QiraBox*. Pada laman ini, pengguna diperkenalkan dengan identitas visual aplikasi yang minimalis. Halaman awal ini berfungsi sebagai *splash screen* yang hanya menampilkan logo dan muncul dalam beberapa detik sebelum pengguna diarahkan ke halaman utama.



Gambar 4. 1 Splash Screen

Pada gambar 4.1 terdapat *splash screen* yang hanya muncul dalam beberapa detik kemudian pengguna diarahkan ke halaman utama kontrol speaker. Halaman ini merupakan antarmuka utama untuk interaksi pengguna. Pada bagian atas layar, terdapat kolom teks yang diberi label “*Host*” di mana pengguna diharuskan memasukkan API dari server MQTT yang terhubung dengan sistem.



Gambar 4. 2 Halaman Utama Aplikasi

Pada gambar 4.2, Di bagian tengah halaman, terdapat *button icon* mikrofon yang berfungsi sebagai tombol untuk memulai proses *speech recognition*. Pengguna cukup menekan tombol ini untuk memberikan perintah suara yang kemudian akan diproses oleh sistem. Setelah perintah suara diberikan, teks hasil *speech recognition* akan ditampilkan di bawah label “Teks Perintah”



Gambar 4. 3 Fitur Speech Recognition

Pada gambar 4.3, terdapat fitur speech recognition yang terhubung dengan *google speech*. Adapun *User interface* aplikasi ini dirancang dengan mempertimbangkan kemudahan akses terutama bagi pengguna dengan kebutuhan khusus seperti penyandang tunanetra. Penggunaan elemen-elemen visual yang besar dan sederhana, serta interaksi berbasis suara. Desain halaman aplikasi ini tidak hanya memprioritaskan fungsi teknis tetapi juga memastikan bahwa interaksi pengguna dengan sistem berjalan dengan lancar dan mudah diakses.

4.1.2 Implementasi Komunikasi IoT

Dalam sistem ini, komunikasi antara *hardware* dan *software* dilakukan melalui jaringan internet dengan memanfaatkan platform *Ngrok* dan *broker MQTT Maqiatto*. Implementasi komunikasi IoT ini memungkinkan interaksi antara aplikasi yang berjalan di perangkat Android dengan ESP32 sebagai pengendali utama speaker murottal yang terhubung ke jaringan.

```

Session Status      online
Account             210605110009@student.uin-malang.ac.id (Plan: Free)
Version             3.16.1
Region              Asia Pacific (ap)
Latency             91ms
Web Interface       http://127.0.0.1:4040
Forwarding           https://67b6-180-248-16-20.ngrok-free.app -> http://localhost:80

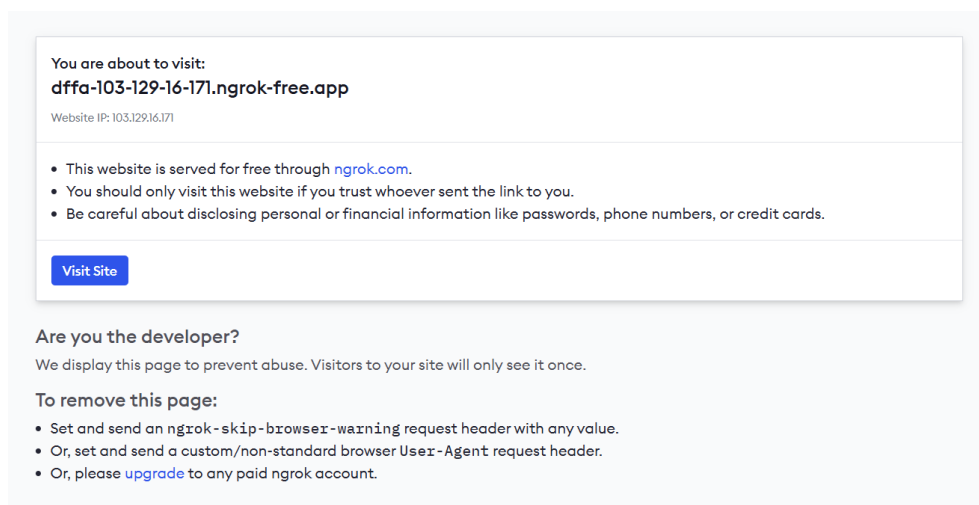
Connections         ttl    opn    rt1    rt5    p50    p90
                   4      0      0.00   0.00   3.67   9.44

HTTP Requests
-----
18:56:59.299 +07 GET /qirabox_code/service3.php 200 OK
18:56:57.599 +07 GET /icons/unknown.gif          200 OK
18:56:57.599 +07 GET /icons/back.gif             200 OK
18:56:57.443 +07 GET /qirabox_code/              200 OK
18:56:56.194 +07 GET /favicon.ico                200 OK
18:56:55.603 +07 GET /icons/blank.gif            200 OK
18:56:55.606 +07 GET /icons/folder.gif           200 OK
18:56:55.605 +07 GET /icons/text.gif             200 OK
18:56:55.605 +07 GET /icons/image2.gif           200 OK
18:56:54.883 +07 GET /                           200 OK

```

Gambar 4. 4 Ngrok di CMD

Ngrok digunakan untuk menyediakan akses jarak jauh ke *localhost* yang dijalankan pada perangkat yang terhubung ke ESP32. *Ngrok* bertindak sebagai jembatan antara *localhost* dan internet tanpa memerlukan konfigurasi jaringan yang rumit. Dengan *Ngrok*, alamat URL yang aman disediakan untuk menghubungkan aplikasi Android dengan server lokal di mana ESP32 berada, sehingga perintah dari pengguna dapat diteruskan dengan baik.



Gambar 4. 5 Ngrok mengakses localhost

Adapun *Maqiatto* digunakan sebagai broker MQTT (*Message Queuing Telemetry Transport*) yang bertugas mengelola komunikasi antara aplikasi dan ESP32 melalui protokol MQTT. *Maqiatto* adalah broker MQTT berbasis *cloud* yang dapat melakukan pertukaran pesan yang cepat dan andal antara perangkat IoT dan aplikasi.



Gambar 4. 6 Logo Maqiatto

Dalam sistem ini, ESP32 berfungsi sebagai klien MQTT yang berlangganan (*subscribe*) ke topik-topik tertentu yang digunakan untuk mengontrol speaker murottal. Aplikasi *Qirabox* di *Android* bertindak sebagai klien yang menerbitkan (*publish*) pesan ke topik tersebut, misalnya perintah untuk memutar surah. *Maqiatto* memastikan pesan-pesan ini dikirim dengan cepat dan efisien antara aplikasi dan ESP32, memungkinkan kontrol speaker secara real-time.

Dengan adanya *Ngrok* dan *Maqiatto*, komunikasi dalam sistem ini dapat diakses dengan baik dalam lingkungan lokal maupun di luar jaringan local dan sistem dapat dikontrol secara jarak jauh dengan stabilitas dan keamanan yang terjamin, tanpa harus bergantung pada konfigurasi jaringan yang kompleks.

4.1.3 Kontrol Speaker dengan *Speech Recognition*

Kontrol speaker dengan *speech recognition* yang dibuat pada *MIT App Inventor* dilakukan untuk memastikan bahwa sistem dapat menerima dan mengeksekusi perintah suara dengan baik. Tahapan kontrol speaker dengan *speech recognition* dapat dilihat *script* didokumentasikan dengan *pseudocode* di bawah ini.

```
// Langkah 1: Mendengarkan perintah suara
Jika tombol mic == "click"
Panggil SpeechRecognizer1()

// Langkah 2: Mengambil teks perintah
teks_perintah = SpeechRecognizer1.Result

// Langkah 3: Mengirim perintah ke server via API
URL = "http://qirabox/service3.php?perintah=" +
teks_perintah
Web1.PostText(URL)
```

```
// Langkah 4: Terima dan proses respons dari server

respons_server = Web1.Response

    SpeechRecognizer1.Result = respons_server
```

Setelah perintah yang dikirim sudah diproses oleh server, maka aplikasi akan menerima respons dari server, yang kemudian disimpan kembali dalam variabel *SpeechRecognizer1.Result*. Hasil ini akan menunjukkan apakah perintah yang diucapkan sesuai dengan pola yang diharapkan, dan sistem akan menindaklanjuti dengan tindakan yang sesuai berdasarkan hasil yang diterima dari server.

Selanjutnya, dalam menangani respons yang diterima dari server setelah perintah kontrol speaker dikirim. Ketika *Web1.GotText* dipanggil, respons dari server diambil dan disimpan dalam variabel *teks_output*. Berdasarkan isi respons, aplikasi akan memeriksa apakah perintah yang diterima adalah “speaker nyala”, “speaker mati”, “volume naik”, atau “volume turun”. Apabila ditemukan list surah yang ada pada server, maka sistem akan mengatur teks output untuk memutar surah yang diminta dan menampilkan icon *PutarSurah.png*. Jika perintah tidak ditemukan maka dioutputkan “Perintah tidak dikenali”. Sebagaimana *script* yang didokumentasikan dengan *pseudocode* di bawah ini.

```
// Langkah 5: Respons yang diterima dari server disimpan dalam
variabel

    If Web1.GotText == "called"

        teks_output = Web1.Response

// Langkah 6: Evaluasi respons untuk menentukan tindakan
if

    If teks_output == "speaker nyala"
```

```

        NyalakanSpeaker()
    Else if teks_output == "speaker mati"
        MatikanSpeaker()
    Else if teks_output == "volume naik"
        NaikkanVolume()
    Else if teks_output == "volume turun"
        TurunkanVolume()
    Else if teks_output == "henti speaker"
        HentikanSpeaker()
    Else if teks_output == "mulai speaker"
        MulaiSpeaker()
    Else if teks_output == "putar surah"
        PutarSurah()
    Else
        ("Perintah tidak dikenali")

```

Adapun Output yang ditampilkan di *command prompt* merupakan hasil dari proses *subscribe* ke broker MQTT *Maqiatto*. Pesan yang diterima berisi beberapa perintah yang berkaitan dengan kontrol speaker atau nomor surah. Setelah pesan diterima oleh sistem melalui MQTT, pesan tersebut diproses lebih lanjut. Sistem akan memeriksa instruksi yang diterima, seperti perintah untuk mengubah volume atau menghidupkan perangkat, dan kemudian mengirimkan perintah yang sesuai ke *hardware* yang terhubung. Berikut output *command prompt* di dapat dilihat pada gambar 4.7.

```

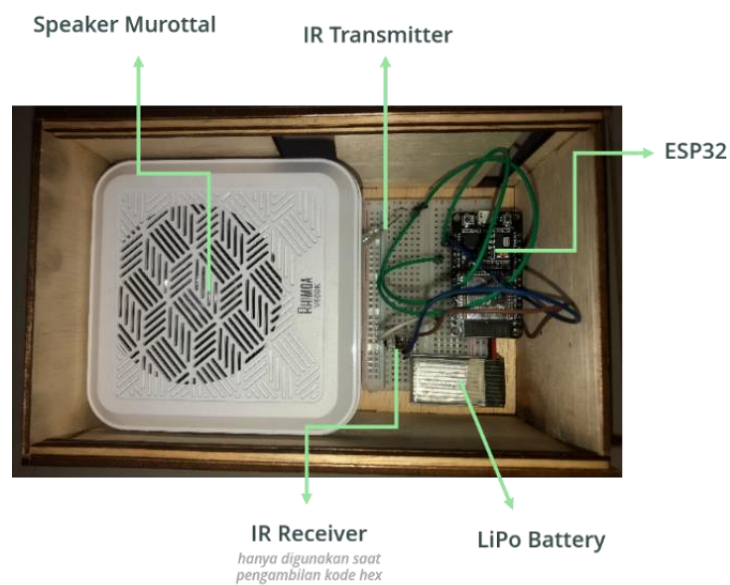
C:\Program Files\mosquitto>mosquitto_sub -h maqiatto.com -p 1883 -u "henyrimadanaa31@gmail.com"
-P "qirabox" -t "henyrimadanaa31@gmail.com/speaker"
offline
80
78
power
power
volume_up
volume_down

```

Gambar 4. 7 Pesan Subscribe di Command Prompt

4.1.4 Rangkaian *Hardware*

Rangkaian *hardware* yang digunakan dalam sistem kontrol speaker murottal Al-Qur'an berbasis *Internet of Things* (IoT) ini terdiri dari beberapa komponen utama yang saling terhubung untuk menjalankan fungsi kontrol speaker.



Gambar 4. 8 Rangkaian Hardware

Berdasarkan gambar 4.8, Speaker berfungsi sebagai output utama untuk memutar ayat-ayat Al-Qur'an berdasarkan perintah suara yang diberikan pengguna melalui aplikasi *android*. Komponen ESP32 bertindak sebagai pusat pengontrol sistem, di mana mikrokontroler ini memproses perintah yang diterima dan mengirim sinyal melalui *IR Transmitter* untuk mengontrol fungsi speaker, seperti

memutar, menghentikan, atau mengganti surah. *IR Transmitter* mengirim sinyal inframerah ke speaker untuk mengeksekusi perintah berdasarkan kode *hexadecimal* yang telah direkam sebelumnya menggunakan *IR Receiver*. Meskipun *IR Receiver* tidak digunakan dalam operasi, komponen ini berperan penting dalam pengaturan awal untuk mengambil kode-kode hex dari *remote* kontrol speaker. Sistem ini didukung oleh baterai LiPo 400 mAh sebagai sumber daya portabel yang memungkinkan operasi berkelanjutan tanpa memerlukan koneksi listrik langsung. Sistem ini dirancang agar tunanetra dapat mengontrol speaker secara efisien menggunakan perintah suara, tanpa harus mengoperasikan tombol fisik atau remote, sehingga meningkatkan aksesibilitas dan kenyamanan penggunaan.

4.2 Hasil Pengujian Sistem

Pengujian sistem dilakukan untuk mengevaluasi kinerja sistem sehingga memastikan bahwa seluruh fungsi sistem, mulai dari kontrol speaker melalui perintah suara hingga performa algoritma *Aho-Corasick* berjalan sesuai dengan yang diharapkan. Selain itu, pengujian dilakukan untuk mengidentifikasi potensi *error* sistem serta mengevaluasi tingkat kegunaan sistem berdasarkan *feedback* pengguna melalui *System Usability Scale* (SUS).

Pengujian ini mencakup beberapa aspek penting, proses pencocokan dan performa algoritma *Aho-Corasick* dalam mencocokkan *pattern*, pengujian hardware, pengujian *error* sistem, dan penilaian pengalaman pengguna melalui metode SUS.

4.2.1 Proses Pencocokan Pola dengan *Aho-Corasick*

Pada penelitian ini, pola-pola yang dimasukkan ke dalam proses pencocokan dengan algoritma *Aho-Corasick* meliputi beberapa pola yang dapat dilihat pada tabel 4.1.

Tabel 4. 1 Pola (Pattern) Penelitian

No	Pattern
1.	nyala speaker
2.	mati speaker
3.	jangan nyala speaker
4.	jangan mati speaker
5.	naik volume
6.	volume naik
7.	turun volume
8.	volume turun
9.	henti speaker
10.	stop speaker
11.	mulai speaker
12.	mulai lagi
13.	putar surah an naba
14.	putar surah an aziat
15.	putar surah abasa
16.	putar surah at takwir
17.	putar surah al infitar
18.	putar surah al mutaffifin
19.	putar surah al insyiqaq
20.	putar surah al buruj
21.	putar surah at tariq
22.	putar surah al ala
23.	putar surah al ghashiyah
24.	putar surah al fajr
25.	putar surah al balad
26.	putar surah asy syams
27.	putar surah al lail
28.	putar surah ad dhuha
29.	putar surah asy syarh
30.	putar surah at tin
31.	putar surah al alaq
32.	putar surah al qadr
33.	putar surah al bayyinah

No	Pattern
34.	putar surah az zalzalah
35.	putar surah al adiyat
36.	putar surah al qariah
37.	putar surah at takathur
38.	putar surah al asr
39.	putar surah al humazah
40.	putar surah al fil
41.	putar surah quraisy
42.	putar surah al maun
43.	putar surah al kautsar
44.	putar surah al kafirun
45.	putar surah an nasr
46.	putar surah al lahab
47.	putar surah al ikhlas
48.	putar surah al falaq
49.	putar surah an nas

Pola-pola di atas adalah pola yang sudah di lakukan *preprocessing text* dan sudah siap untuk di lakukan pencocokan dengan algoritma *Aho-Corasick*. Pola-pola tersebut diolah oleh sistem untuk dikenali sebagai perintah yang akan dieksekusi oleh perangkat. Setiap pola ini kemudian dimasukkan ke dalam struktur data *Trie* yang efisien, yang memungkinkan pencarian cepat.

Pembangunan *Trie* dimulai dengan setiap pola dimasukkan sebagai deretan *node* yang merepresentasikan setiap karakter dari pola tersebut. Misalnya, dalam pola “nyala speaker”, karakter pertama “n” dimasukkan ke dalam *Trie* pada node awal, diikuti oleh karakter “y” , “a” dan seterusnya hingga seluruh pola terbentuk dalam *Trie*. Jika ada pola lain yang memiliki awalan yang sama, seperti “nyala”, maka pola tersebut dapat berbagi jalur di dalam *Trie* hingga karakter yang berbeda muncul.

Pattern	:	nyala speaker
Panjang Pattern	:	13
Tahap Go (Transisi)	:	Tahap Go - Karakter saat ini: 'n' Tahap Go - Transisi ditemukan untuk 'n', pindah ke state: 1 Tahap Go - Karakter saat ini: 'y' Tahap Go - Transisi ditemukan untuk 'y', pindah ke state: 2 Tahap Go - Karakter saat ini: 'a' Tahap Go - Transisi ditemukan untuk 'a', pindah ke state: 3 Tahap Go - Karakter saat ini: 'l' Tahap Go - Transisi ditemukan untuk 'l', pindah ke state: 4 Tahap Go - Karakter saat ini: 'a' Tahap Go - Transisi ditemukan untuk 'a', pindah ke state: 5 Tahap Go - Karakter saat ini: '' Tahap Go - Transisi ditemukan untuk '', pindah ke state: 6 Tahap Go - Karakter saat ini: 's' Tahap Go - Transisi ditemukan untuk 's', pindah ke state: 7 Tahap Go - Karakter saat ini: 'p' Tahap Go - Transisi ditemukan untuk 'p', pindah ke state: 8 Tahap Go - Karakter saat ini: 'e' Tahap Go - Transisi ditemukan untuk 'e', pindah ke state: 9 Tahap Go - Karakter saat ini: 'a' Tahap Go - Transisi ditemukan untuk 'a', pindah ke state: 10 Tahap Go - Karakter saat ini: 'k' Tahap Go - Transisi ditemukan untuk 'k', pindah ke state: 11 Tahap Go - Karakter saat ini: 'e' Tahap Go - Transisi ditemukan untuk 'e', pindah ke state: 12 Tahap Go - Karakter saat ini: 'r' Tahap Go - Transisi ditemukan untuk 'r', pindah ke state: 13 Tahap Output - Pola cocok ditemukan: 'nyala speaker'
Tahap Failure (Tabel Fail)	:	
Perintah ditemukan	:	nyala speaker

Gambar 4. 9 Output Pattern Matching Aho-Corasick

Pada gambar 4.9, Tahapan pertama dalam proses pencocokan disebut Tahap Go, di mana algoritma mulai mencocokkan input teks dengan *Trie*. Setiap karakter dari input akan diperiksa apakah sesuai dengan transisi yang ada dalam *Trie*. Sebagai contoh, ketika input “nyala speaker”, dimasukkan, algoritma akan mulai dari node awal dan mencari transisi yang cocok untuk karakter “n”, kemudian “y” hingga seluruh kata “nyala speaker” cocok dengan pola yang ada di dalam *Trie*. Jika semua karakter cocok, algoritma akan melanjutkan ke simpul akhir dan menganggap pola telah ditemukan.

Pattern	:	nyali speaker
Panjang Pattern	:	13
Tahap Go (Transisi)	:	Tahap Go - Karakter saat ini: 'n' Tahap Go - Transisi ditemukan untuk 'n', pindah ke state: 1 Tahap Go - Karakter saat ini: 'y' Tahap Go - Transisi ditemukan untuk 'y', pindah ke state: 2 Tahap Go - Karakter saat ini: 'a' Tahap Go - Transisi ditemukan untuk 'a', pindah ke state: 3 Tahap Go - Karakter saat ini: 'l' Tahap Go - Transisi ditemukan untuk 'l', pindah ke state: 4 Tahap Go - Karakter saat ini: 'i' Tahap Go - Tidak ada transisi, kembali ke state awal (0) Tahap Go - Karakter saat ini: '' Tahap Go - Tidak ada transisi, kembali ke state awal (0) Tahap Go - Karakter saat ini: 's' Tahap Go - Transisi ditemukan untuk 's', pindah ke state: 14 Tahap Go - Karakter saat ini: 'p' Tahap Go - Transisi ditemukan untuk 'p', pindah ke state: 15 Tahap Go - Karakter saat ini: 'e' Tahap Go - Transisi ditemukan untuk 'e', pindah ke state: 16 Tahap Go - Karakter saat ini: 'a' Tahap Go - Transisi ditemukan untuk 'a', pindah ke state: 17 Tahap Go - Karakter saat ini: 'k' Tahap Go - Transisi ditemukan untuk 'k', pindah ke state: 18 Tahap Go - Karakter saat ini: 'e' Tahap Go - Transisi ditemukan untuk 'e', pindah ke state: 19 Tahap Go - Karakter saat ini: 'r' Tahap Go - Transisi ditemukan untuk 'r', pindah ke state: 20
Tahap Failure (Tabel Fail)	:	Tahap Failure - Tidak ada transisi untuk 'i', kembali ke state fail: 0
Perintah tidak ditemukan		

Gambar 4. 10 Output Jika Perintah Tidak Ditemukan

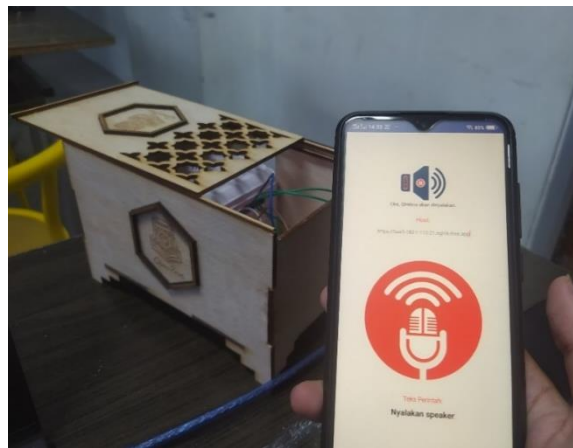
Jika tidak ada transisi yang cocok pada satu titik tertentu dalam proses pencocokan, maka algoritma akan masuk ke tahap *Failure* seperti pada gambar 4.10. Pada tahap ini, algoritma menggunakan transisi kegagalan untuk kembali ke simpul yang relevan dan melanjutkan pencocokan dari titik tersebut. Misalnya, jika setelah mencocokkan sebagian pola "nyala", tidak ada transisi yang sesuai untuk karakter berikutnya, algoritma akan mengikuti transisi kegagalan untuk menemukan node yang dapat melanjutkan proses pencocokan tanpa harus memulai dari awal.

Setelah algoritma berhasil mencocokkan seluruh pola, akan dihasilkan Output Pencocokan. Misalnya, jika input "nyala speaker" cocok dengan salah satu pola yang dimasukkan ke dalam *Trie*, sistem akan memberikan output bahwa perintah tersebut telah dikenali dan akan dilanjutkan untuk dieksekusi, seperti menyalakan

speaker. Namun, jika tidak ada pola yang cocok, output akan menunjukkan bahwa perintah tidak dikenali oleh sistem.

4.2.2 Pengujian Sistem Kontrol

Pada penelitian ini, proses pengujian ini meliputi penggunaan *hardware* yang dihubungkan ke jaringan *hostpot* lainnya dan telah diintegrasikan dengan server MQTT dan dikontrol melalui aplikasi “Qirabox” yang dibuat oleh peneliti.



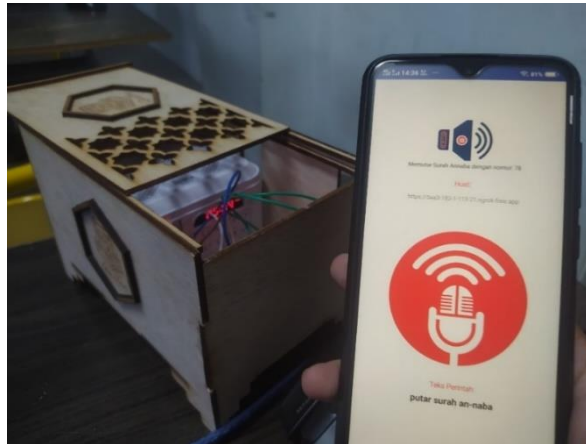
Gambar 4. 11 Perintah Menyalakan Speaker

Pada gambar 4.11 merupakan tampilan speaker saat dinyalakan dengan perintah “nyalakan speaker”.



Gambar 4. 12 Perintah Mematikan Speaker

Pada gambar 4.12 merupakan tampilan speaker saat dimatikan dengan perintah “matikan speaker”.



Gambar 4. 13 Perintah Memutar Surah Tertentu

Pada gambar 4.13 merupakan tampilan speaker saat pemutaran surah dengan perintah “putar surah an-naba”.

4.2.3 Pengujian Error Sistem

Pengujian error sistem dilakukan untuk mengukur sejauh mana tingkat kesalahan yang terjadi dalam sistem. dalam mengenali perintah suara yang diberikan oleh pengguna. Pengujian ini melibatkan 10 penyandang tunanetra yang masing-masing diminta untuk memberikan 5 kali input perintah suara untuk sistem yang dibuat. Setiap input perintah suara dicatat dan dievaluasi untuk melihat sejauh mana sistem dapat mengenali perintah secara tepat atau terjadi error dalam pemrosesan suara tersebut. Berikut hasil pengujian error sistem yang peneliti tuangkan dalam beberapa tabel berbeda setiap partisipannya.

Tabel 4. 2 Hasil Pengujian Error Sistem oleh SR

HASIL PENGUJIAN ERROR SISTEM				
BAPAK SURATNO				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	nyalakan speaker	nyala speaker	✓	
Perintah 2	naikkan volume	naik volume	✓	
Perintah 3	turunkan volume	turun volume	✓	
Perintah 4	surah al-ikhlas	surah al ikhlas	✓	
Perintah 5	putar surah al-qadr	putar surah al qadr	✓	
Total Error Sistem			0	

Tabel 4. 3 Hasil Pengujian Error Sistem oleh NS

HASIL PENGUJIAN ERROR SISTEM				
IBU NANIK SETYANI				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	hentikan speaker	henti speaker	✓	
Perintah 2	nyalakan speakernya	nyala speaker	✓	
Perintah 3	mulai speaker ini	mulai speaker	✓	
Perintah 4	putar surat al ash'r	-	✓	
Perintah 5	putar surah al ash'r	putar surah al ash'r	✓	
Total Error Sistem			0	

Tabel 4. 4 Hasil Pengujian Error Sistem oleh IT

HASIL PENGUJIAN ERROR SISTEM				
BAPAK IMAM TAUFIK				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	hentikan speaker	henti speaker	✓	
Perintah 2	nyalakan speaker	nyala speaker	✓	
Perintah 3	putar surah an-naba	putar surah annaba	✓	
Perintah 4	putar surah al-adiyat	putar surah aladiyat	✓	
Perintah 5	matikan speaker	matikan speaker	✓	
Total Error Sistem			0	

Tabel 4. 5 Hasil Pengujian Error Sistem oleh W

HASIL PENGUJIAN ERROR SISTEM				
BAPAK WANDI				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	nyalakan speaker	nyala speaker	✓	
Perintah 2	putar surah an-nas	putar surah annas	✓	
Perintah 3	putar surah ad dhuha	putar surah addhuha	✓	
Perintah 4	stop speaker	stop speaker	✓	
Perintah 5	mulai lagi	mulai lagi	✓	
Total Error Sistem			0	

Tabel 4. 6 Hasil Pengujian Error Sistem oleh WT

HASIL PENGUJIAN ERROR SISTEM				
IBU WATI				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	mulai speaker ini	mulai speaker	✓	
Perintah 2	putar surah an nas	putar surah annas	✓	
Perintah 3	naikkan lah volumenya	naik volume	✓	
Perintah 4	tambah naikkan volume	naik volume	✓	
Perintah 5	matikan speaker	mati speaker	✓	
Total Error Sistem			0	

Tabel 4. 7 Hasil Pengujian Error Sistem oleh L

HASIL PENGUJIAN ERROR SISTEM				
BAPAK LASIMAN				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	naikkan volume	naikkan volume	✓	
Perintah 2	surat al an-nas	-	✓	
Perintah 3	surah al an-nas	surah al an-nas	✓	
Perintah 4	surah ad dhuha	surah ad dhuha	✓	
Perintah 5	stop speaker ini	stop speaker ini	✓	
Total Error Sistem			0	

Tabel 4. 8 Hasil Pengujian Error Sistem oleh WN

HASIL PENGUJIAN ERROR SISTEM				
IBU WIWIK NUR'AINI				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	nyalakan speaker	nyala speaker	✓	
Perintah 2	matikan speaker	mati speaker	✓	
Perintah 3	tolong nyalakan speaker	nyala speaker	✓	
Perintah 4	putar surat an-naba	-	✓	
Perintah 5	putar surah an-naba	putar surah an-naba	✓	
Total Error Sistem			0	

Tabel 4. 9 Hasil Pengujian Error Sistem oleh SD

HASIL PENGUJIAN ERROR SISTEM				
BAPAK SARDIANTO				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	nyalakan speaker	nyala speaker	✓	
Perintah 2	nyala speaker	nyala speaker	✓	
Perintah 3	putarkan surah an-naba	putar surah annaba	✓	
Perintah 4	matikan speaker	mati speaker	✓	
Perintah 5	nyalakan speaker ini	nyala speaker	✓	
Total Error Sistem			0	

Tabel 4. 10 Hasil Pengujian Error Sistem oleh MM

HASIL PENGUJIAN ERROR SISTEM				
BAPAK M. MURYONO				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	tolong nyalakan speaker	nyala speaker		✓
Perintah 2	nyalakan speaker	nyala speaker	✓	
Perintah 3	matikan speaker	mati speaker	✓	
Perintah 4	tolong nyalakan speaker	nyala speaker	✓	
Perintah 5	Putarkan surah an naba	putar surah an naba	✓	
Total Error Sistem			1	

Tabel 4. 11 Hasil Pengujian Error Sistem oleh TM

HASIL PENGUJIAN ERROR SISTEM				
BAPAK TRI MARTA				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	turunkan volumenya	turun volume	✓	
Perintah 2	turunkan volume lagi	turun volume	✓	
Perintah 3	putar surah al-qariah	putar surah alqariah	✓	
Perintah 4	putarkan surah al lail	putar surah allail	✓	
Perintah 5	matikan speakernya	mati speaker	✓	
Total Error Sistem			0	

Berdasarkan hasil pengujian yang dilakukan sebanyak 50 kali terhadap 10 penyandang tunanetra, ditemukan bahwa error sistem terjadi satu kali saat sistem tidak merespons perintah untuk menyalakan speaker yang ditunjukkan pada persamaan 4.1.

$$\mu = \frac{\sum Ei}{n} \times 100\%$$

$$\mu = \frac{1}{50} \times 100\%$$

$$\mu = 2\%$$
(4.1)

Pada pengujian error, sistem ini terdapat error sebesar 0,02 atau 2%. Setelah dilakukan penelusuran lebih lanjut, diketahui bahwa penyebab kegagalan tersebut adalah ketidaksesuaian status speaker di database yang mana status speaker tercatat sebagai “ON”, sementara kondisi aktual speaker masih dalam keadaan mati. Hal ini terjadi karena sebelumnya speaker dimatikan secara manual menggunakan remote sehingga status dalam database tidak diperbarui. Oleh karena itu, dapat disimpulkan bahwa kegagalan ini tidak sepenuhnya kesalahan sistem, melainkan disebabkan oleh ketidaksesuaian status oleh intervensi manual.

Selain itu, terdapat beberapa kasus kegagalan yang terjadi akibat kesalahan pada input yang diberikan oleh pengguna, seperti penggunaan kata atau format yang tidak sesuai dengan pola yang telah ditetapkan di dalam kode sistem. Kesalahan ini berasal dari variasi input pengguna yang tidak sesuai, misalnya pengucapan “surat” yang seharusnya adalah “surah”. Hal ini menyebabkan sistem tidak dapat mengenali perintah tersebut. Namun, sistem tetap memberikan *feedback* “perintah tidak sesuai kontrol” atau “perintah tidak dikenali”.

Selain itu, peneliti juga melakukan uji coba sistem dengan perintah yang panjang 5 kalimat. Dapat dilihat pada tabel 4.12 di bawah ini.

Tabel 4. 12 Pengujian dengan 5 kalimat

HASIL PENGUJIAN ERROR SISTEM				
DENGAN 5 KALIMAT				
No. Perintah	Perintah	Pattern	Respon sistem	
			Merespon	Tidak merespon
Perintah 1	Halo coba bantu turunkan volume	turun volume	✓	
Perintah 2	Coba naikan volume speaker ini	naik volume	✓	
Perintah 3	Sekarang putarkan surah al-qariah dong	putar surah alqariah	✓	
Perintah 4	Aku mau putar surah an-naba ya	putar surah annaba	✓	
Perintah 5	Oke yasudah matikan speaker ini	mati speaker	✓	
Perintah 6	Qirabox ayo menyalakan speaker ini	nyala speaker	✓	
Perintah 7	Tolong bantu putar surah al-qori'ah	putar surah alqoriah	✓	
Perintah 8	Tolong putarkan surah az-zalzalah dong	putar surah azzalzalalah	✓	
Perintah 9	Saya mau hentikan speaker ini	henti speaker	✓	
Perintah 10	Tolong mulai lagi speaker ini	mulai speaker	✓	
Total Error Sistem			0	

Hasil dari pengujian menggunakan 5 kalimat menunjukkan bahwa tidak terdapat error pada sistem ini. Dapat diartikan, apabila perintah sudah sesuai dengan pattern yang ada maka, speaker akan memberikan output sesuai perintah.

4.2.4 Pengujian *System Usability Scale* (SUS)

Pengujian ini dilakukan untuk mengukur tingkat kegunaan sistem kontrol speaker murottal Al-Qur'an yang dirancang khusus bagi tunanetra. Pengujian ini melibatkan 10 penyandang tunanetra sebagai responden yang diminta untuk menanggapi 10 pernyataan kuesioner SUS. Mengingat keterbatasan penglihatan para responden, metode pengisian kuesioner dilakukan melalui wawancara langsung, sehingga setiap pernyataan dapat dijelaskan dengan detail dan mudah dipahami oleh para responden. Adapun skor jawaban asli dari pengujian SUS dapat dilihat pada tabel 4.13.

Tabel 4. 13 Skor Jawaban Pengujian SUS

Responden	Skor Jawaban Kuesioner									
	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇	Q ₈	Q ₉	Q ₁₀
Responden 1 (SR)	4	4	5	4	5	1	5	4	5	1
Responden (NS)	5	1	5	4	5	1	5	3	5	4
Responden 3 (LS)	4	2	5	4	5	2	5	4	4	3
Responden 4 (IT)	5	3	5	4	5	3	5	2	3	2
Responden 5 (WT)	3	3	4	3	5	5	5	3	5	3
Responden 6 (WD)	4	3	5	4	5	1	5	4	5	2
Responden 7 (SD)	5	1	5	2	5	1	5	2	4	3
Responden 8 (WN)	5	1	5	3	5	1	5	2	4	2
Responden 9 (TR)	5	1	5	1	5	1	5	1	5	1
Responden 10 (MY)	4	3	5	4	5	1	4	3	5	1

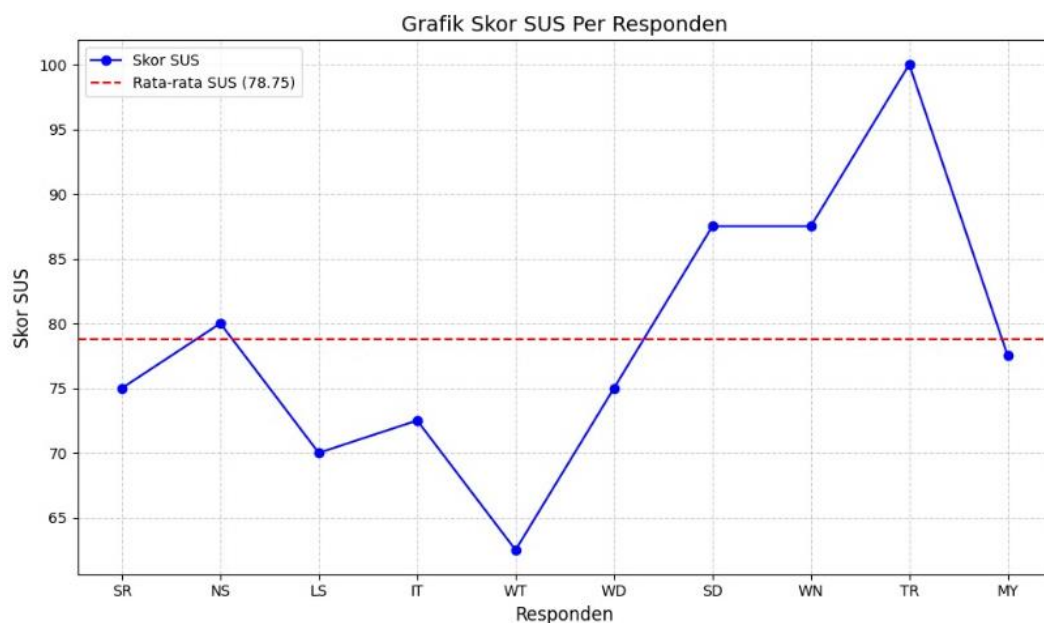
Setelah didapatkan skor jawaban asli dihitung SUS menggunakan aturan yang telah ditetapkan. Hasil perhitungan disajikan pada tabel 4.14.

Tabel 4. 14 Perhitungan Skor SUS

Responden	Perhitungan Skor SUS										Jumlah	Total skor (*2,5)
	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇	Q ₈	Q ₉	Q ₁₀		
Responden 1 (SR)	3	1	4	1	4	4	4	1	4	4	30	75
Responden (NS)	4	4	4	1	4	4	4	2	4	1	32	80
Responden 3 (LS)	3	3	4	1	4	3	4	1	3	2	28	70
Responden 4 (IT)	4	2	4	1	4	2	4	3	2	3	29	72.5

Responden	Perhitungan Skor SUS										Jumlah	Total skor (*2,5)
	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇	Q ₈	Q ₉	Q ₁₀		
Responden 5 (WT)	2	2	3	2	4	0	4	2	4	2	25	62.5
Responden 6 (WD)	3	2	4	1	4	4	4	1	4	3	30	75
Responden 7 (SD)	4	4	4	3	4	4	4	3	3	2	35	87.5
Responden 8 (WN)	4	4	4	2	4	4	4	3	3	3	35	87.5
Responden 9 (TR)	4	4	4	4	4	4	4	4	4	4	40	100
Responden 10 (MM)	3	2	4	1	4	4	3	2	4	4	31	77.5
Jumlah												787.5
Rata-rata SUS												78.75

Berdasarkan hasil pengujian menggunakan System Usability Scale (SUS), sistem ini memperoleh nilai SUS sebesar 78,75. Adapun visualisasi terkait pengujian SUS yang sudah dilakukan dapat dilihat pada gambar 4.14 di bawah ini.



Gambar 4. 14 Grafik Pengujian SUS

Grafik di atas menunjukkan skor *System Usability Scale* (SUS) dari 10 responden yang diwakili oleh kode identitas mereka, yaitu SR, NS, LS, IT, WT, WD, SD, WN, TR, dan MY. Skor SUS merupakan metrik standar yang digunakan untuk mengevaluasi tingkat kegunaan suatu sistem berdasarkan persepsi pengguna.

Dari grafik tersebut, didapatkan bahwa nilai rata-rata SUS adalah 78,75 yang ditunjukkan oleh garis horizontal berwarna merah. Nilai rata-rata SUS ini tergolong dalam kategori B dengan penilaian “*Good*”, yang menunjukkan bahwa sistem telah memenuhi kebutuhan pengguna dari segi kemudahan penggunaan (*usability*). Hal ini mengindikasikan bahwa perangkat ini berhasil memberikan pengalaman yang nyaman dan mudah dioperasikan bagi penyandang tunanetra.

4.3 Pembahasan

Sistem kontrol yang dikembangkan dalam penelitian ini mencakup dua komponen utama, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*) yang bekerja secara terintegrasi untuk mengendalikan speaker murottal Al-Quran. Pada sisi perangkat keras, sistem ini menggunakan speaker sebagai output utama, yang terhubung dengan mikrokontroler ESP32 untuk mengelola fungsi-fungsi dasar seperti menyalakan, mematikan, mengatur volume atau memutar ayat dalam Al-Quran. Adapun sisi perangkat lunak, aplikasi Android “*Qirabox*” berfungsi sebagai alat untuk mengirimkan perintah suara kepada sistem kontrol.

Setelah aplikasi Android “*Qirabox*” menerima perintah suara dari pengguna, aplikasi ini mengonversi suara menjadi kode kontrol dan kode surah yang sesuai. Kode-kode tersebut kemudian dikirim melalui protokol komunikasi MQTT (*Message Queuing Telemetry Transport*), yang berfungsi sebagai perantara untuk menghubungkan aplikasi dengan sistem kontrol *hardware*.

Setelah kode kontrol dan kode surah diterima oleh sistem kontrol melalui MQTT, kode-kode ini kemudian diubah menjadi format yang dapat dipahami oleh *hardware*, yaitu kode *hexadecimal*. Kode *hexadecimal* ini mewakili instruksi yang

akan diteruskan ke *infrared transmitter*, yang bertanggung jawab untuk mengirimkan sinyal ke speaker. Di sisi penerima, *infrared receiver* pada speaker menangkap sinyal yang dikirim oleh *transmitter* dan mengubahnya kembali menjadi perintah yang dapat diproses oleh mikrokontroler untuk mengeksekusi tindakan yang diminta. Adapun kode *hexadecimal* terdapat pada tabel berikut.

Tabel 4. 15 Kode Hexadecimal

No	Tombol Remote	Kode Hexadecimal
1.	Power (on/off)	0xFFA25D
2.	Stop (pause/play)	0xFF629D
3.	Volume Down (-)	0xFF6897
4.	Volume Up (+)	0xFF9867
5.	Angka 0	0xFFB04F
6.	Angka 1	0xFF30CF
7.	Angka 2	0xFF18E7
8.	Angka 3	0xFF7A85
9.	Angka 4	0xFF10EF
10.	Angka 5	0xFF38C7
11.	Angka 6	0xFF5AA5
12.	Angka 7	0xFF42BD
13.	Angka 8	0xFF4AB5
14.	Angka 9	0xFF52AD

4.4 Integrasi Islam

4.4.1 Muamalah Ma'a Allah

Muamalah Ma'a Allah adalah interaksi seorang hamba dengan Allah yang mencakup segala bentuk ibadah, termasuk mendengarkan dan merenungkan Al-Qur'an. Dalam konteks penelitian ini, mendengarkan Al-Qur'an adalah salah satu cara penyandang tunanetra untuk beribadah dan mendekatkan diri kepada Allah. Dalam keterbatasan penglihatan, mendengarkan lantunan ayat-ayat suci menjadi jembatan yang kuat antara hamba dan penciptanya. Allah menjanjikan rahmat bagi

mereka yang khusyuk mendengarkan Al-Qur'an, sebagaimana firman-Nya dalam QS. Al-A'raf ayat 204.

وَإِذَا قُرِئَ الْقُرْآنُ فَاسْتَمِعُوا لَهُ وَأَنْصِتُوا لَعَلَّكُمْ تُرْحَمُونَ

“Dan apabila Al-Qur'an dibacakan, maka dengarkanlah dan diamlah, agar kalian mendapat rahmat.” (Q.S. Al-A'raf: 204).

Berdasarkan tafsir yang diriwayatkan oleh Ibnu Abi Hatim dari Abu Hurairah bahwa ayat ini turun berkenaan dengan orang-orang yang membaca Al-Qur'an dengan nyaring saat shalat dan bermakmum dengan Nabi Muhammad SAW. Ayat ini menekankan bahwa disaat mendengarkan bacaan Al-Qur'an diharuskan untuk diam dengan seksama. Prinsip “mendengar” oleh Al-Mahalli dan As-Suyuthi yakni mendengar tanpa berbicara. Artinya, mendengarkan Al-Qur'an dengan penuh perhatian adalah suatu bentuk ibadah yang mendatangkan rahmat dari Allah.

Korelasi surah Al-A'raf ayat 204 dengan penelitian ini terletak pada peran teknologi yang memfasilitasi tunanetra untuk mendengarkan Al-Qur'an bukan hanya membantu mereka dalam menjalankan ibadah, tetapi juga menjadi perantara untuk mendapatkan rahmat dari Allah. Mendengarkan Al-Qur'an adalah ibadah yang dapat memberikan ketenangan, keberkahan, dan rahmat, sebagaimana dijanjikan dalam ayat ini.

4.4.2 Muamalah Ma'a An-Nas

Muamalah Ma'a An-Nas mencakup segala bentuk hubungan dan interaksi antar manusia dalam kehidupan sosial. Adapun sistem kontrol speaker murottal Al-Qur'an dengan perintah suara ini dibuat untuk menciptakan teknologi assistif, khususnya bagi penyandang tunanetra yang memiliki keterbatasan dalam membaca

teks Al-Qur'an secara langsung. Teknologi ini bertujuan untuk memfasilitasi penyandang tunanetra dalam berinteraksi dengan Al-Qur'an melalui pendengaran. Hal ini sejalan dengan ajaran Islam yang menganjurkan tolong-menolong dalam kebaikan dan mempermudah sesama. Sebagaimana firman Allah dalam Surat Al-Maidah ayat 2:

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ

“Dan tolong-menolonglah kamu dalam (mengerjakan) kebajikan dan takwa, dan jangan tolong-menolong dalam berbuat dosa dan permusuhan.” (Q.S. Al-Maidah: 2).

Menurut Tafsir Ibnu Katsir, menjelaskan bahwa perintah *“tolong-menolonglah kamu dalam kebajikan dan takwa”* merujuk pada anjuran untuk saling membantu dalam melakukan amal-amal kebaikan yang diperintahkan oleh Allah, seperti ibadah, zakat, sedekah, dan perbuatan baik lainnya. Sebaliknya, Ibnu Katsir menegaskan bahwa bagian dari ayat ini yang menyebutkan *“dan jangan tolong-menolong dalam berbuat dosa dan permusuhan”* merupakan larangan keras untuk tolong menolong dalam perbuatan dosa dan pelanggaran.

Adapun Korelasi antara surah Al-Maidah ayat 2 dengan penelitian ini terletak pada prinsip dasar tolong-menolong dalam kebaikan dan ketakwaan yang menjadi landasan pengembangan teknologi speaker murottal untuk tunanetra. Penelitian ini berfokus pada pembuatan alat yang memungkinkan tunanetra mendengarkan Al-Qur'an dengan lebih mudah melalui teknologi berbasis suara, yang sejalan dengan perintah Allah untuk saling membantu dalam menjalankan kebaikan dan ketakwaan.

4.4.3 Muamalah Ma'a Alam

Muamalah Ma'a Alam menggambarkan hubungan manusia dengan alam sebagai ciptaan Allah yang turut memuliakan firman-Nya. Dalam konteks penelitian ini, sistem kontrol speaker murottal yang dirancang untuk tunanetra tidak hanya berfungsi untuk memudahkan manusia dalam mendengarkan Al-Qur'an, tetapi juga membawa keberkahan bagi lingkungan sekitarnya.

Berdasarkan hadits riwayat Abu Nu'aim dari Abdullah bin Mas'ud dalam *Hilyat Aal-Auliya'* sebagai berikut.

عَنْ عَبْدِ اللَّهِ بْنِ مَسْعُودٍ، قَالَ: «إِنَّ الْقُرْآنَ يُرِيدُ أَنْ يُتْلَى، فَإِذَا تُتْلِي، أَتَتْ الْمَاءُ تَرْتَعِدُ

“Sesungguhnya Al-Qur'an ingin agar ia dibacakan. Ketika ia dibacakan, air pun bergetar dan merespons dengan gemetar.” (HR. Abu Nu'aim).

Hadits ini menunjukkan bahwa alam, termasuk air, merespons bacaan Al-Qur'an dengan penghormatan yang tinggi, menandakan betapa mulianya bacaan Al-Qur'an di mata alam semesta. Alam turut serta dalam memuliakan ayat-ayat Allah, dan ini mengajarkan bahwa setiap bacaan Al-Qur'an tidak hanya bermanfaat bagi pendengarnya, tetapi juga bagi lingkungan di sekitarnya.

Korelasi antara hadits ini dengan penelitian terletak pada aspek sistem kontrol speaker murottal ini yang tidak hanya memfasilitasi tunanetra untuk mendengarkan Al-Qur'an, tetapi juga secara tidak langsung membawa keberkahan bagi alam melalui bacaan Al-Qur'an. Oleh karena itu, penelitian ini tidak hanya bermanfaat bagi pengguna tunanetra, tetapi juga memberikan kontribusi spiritual yang lebih luas, di mana alam turut serta dalam memuliakan bacaan Al-Qur'an.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Penelitian ini merancang dan mengimplementasikan sistem kontrol speaker murottal Al-Qur'an berbasis *Internet of Things (IoT)* menggunakan algoritma *Aho-Corasick* dengan konsep *speech recognition* yang menjadi *assistive technology* khususnya bagi penyandang disabilitas netra. Sistem ini bekerja dengan memproses input suara melalui aplikasi *Android “Qirabox”* yang terhubung dengan *Google Speech Recognition*, inputan suara dikonversi menjadi teks, kemudian dilakukan *preprocessing*, dan pencocokan pola menggunakan algoritma *Aho-Corasick* sehingga kode surah atau kontrol dikirimkan ke MQTT. Selanjutnya, MQTT mengirimkan kode *hexadecimal* ke mikrokontroler ESP32 yang mengubahnya menjadi sinyal *infrared* melalui IR Transmitter untuk mengontrol speaker murottal. Hasil pengujian sistem menunjukkan tingkat error sebesar 2%, yang disebabkan oleh ketidaksesuaian status speaker di database “ON” dengan kondisi aktual speaker akibat pemutusan manual menggunakan *remote*. Adapun hasil pengujian *System Usability Scale (SUS)* menghasilkan skor 78,75 yang termasuk dalam kategori B dengan penilaian “Good”, menunjukkan bahwa sistem telah memenuhi kebutuhan pengguna dari segi kemudahan penggunaan (*usability*).

5.2. Saran

Penelitian terkait sistem kontrol ini masih terdapat banyak kekurangan yang perlu dikembangkan lagi oleh peneliti selanjutnya diantaranya:

1. Pengembangan terkait jumlah surah dapat dimasukkan 114 surah al-qur'an, sistem kontrol speaker murottal ini juga dapat dipecah menjadi pengulangan per ayat sehingga pengguna dapat murajaah sesuai ayat yang diinginkan.
 2. Implementasi *speech recognition* dapat digunakan untuk *assistive technology* lain khususnya untuk penyandang disabilitas seperti kursi roda, tongkat tunanetra, dan lain sebagainya.
 3. Penambahan fitur *talkback* khusus penelitian yang ditujukan untuk tunanetra.
- Dengan menerapkan saran-saran tersebut, diharapkan sistem yang dibuat selanjutnya dapat ditingkatkan fungsionalitasnya dan lebih bermanfaat untuk pengguna.

DAFTAR PUSTAKA

- Akbar, A., Husodo, A. Y., & Zubaidi, A. (2019). The Implementation of the Google Speech on Qur'an Recitation Correction Application Based on Android. *JTIKA*, 1(1). <http://jtika.if.unram.ac.id/index.php/JTIKA/>
- Anisa. (2024). *Sistem Kendali Jendela Berbasis Internet of Things Menggunakan Algoritma Horspool*. <http://etheses.uin-malang.ac.id/66043/>
- Ansori, A. N. A. (2023). *RI Duduki Peringkat Ketiga Dunia dalam Kasus Kebutaan*. <https://www.liputan6.com/disabilitas/read/5194116/ri-duduki-peringkat-ketiga-dunia-dalamkasus-kebutaan?page=4>
- Azarudeen, K., Chakkaravarthy, G. V., Murugiah, P., & Kharthikeyan, S. (2021). A Novel Approach for Pattern String Matching in Intrusion Detection System. *Journal of Physics: Conference Series*, 1916(1), 012247. <https://doi.org/10.1088/1742-6596/1916/1/012247>
- Badan Pusat Statistik Provinsi Jawa Timur. (2019). *Banyaknya Desa/Kelurahan Menurut Keberadaan Penyandang Cacat*. <https://jatim.bps.go.id/statictable/2019/10/04/1557/banyaknya-desa-kelurahan-menurut-keberadaan-penyandang-cacat-.html>
- Darmawan, G., Alam, S., Imam Sulisty, M., Studi Teknik Informatika, P., Tinggi Teknologi Wastukencana Purwakarta, S., & Artikel, R. (2023). ANALISIS SENTIMEN BERDASARKAN ULASAN PENGGUNA APLIKASI MYPERTAMINA PADA GOOGLE PLAYSTORE MENGGUNAKAN METODE NAÏVE BAYES INFO ARTIKEL ABSTRAK. *Jurnal Ilmiah Teknik Dan Ilmu Komputer*, 2(3), 100–108. <https://doi.org/10.55123>
- Hanani, A., & Hariyadi, M. A. (2020). Smart Home Berbasis IoT Menggunakan Suara Pada Google Assistant. *Jurnal Ilmiah Teknologi Informasi Asia*, 14.
- Imran, M. (2024). Peningkatan Pemberdayaan Penyandang Tunanetra melalui Perancangan Social Media Newsletter di Yayasan Sosial Tunanetra. In *Jurnal Komunitas : Jurnal Pengabdian kepada Masyarakat* (Vol. 6, Issue 2). <http://ojs.stiami.ac.id>
- Kamel, A., Dardian Marindani, E., & Sujaini, H. (2023). Qur'an Smart Speaker with Voice Control Using Raspberry Pi and Google Assistant Library. *Journal of Electrical Engineering, Energy, and Information Technology (J3EIT)*, 11(3), 128–136. <https://doi.org/10.26418/j3eit.v11i3.68960>
- Kini, S., Patil, A. P., Pooja, M., & Balasubramanyam, A. (2022). SQL Injection Detection and Prevention using Aho-Corasick Pattern Matching Algorithm. *2022 3rd International Conference for Emerging Technology (INCET)*, 1–6. <https://doi.org/10.1109/INCET54531.2022.9825040>

- Kurnia, A., & Khaerani, I. F. S. R. (2020). Perancangan Digitalisasi Tafsir Al-Qur'an Untuk Disabilitas Netra. *Jurnal TEDC*, 14(2), 128–133.
- Marliana, I., Ikhwan, A., & Fawaati, T. M. (2023). *IMPLEMENTASI MIT APP INVENTOR DALAM GAME MENGENAL HURUF BERBASIS ANDROID* (Vol. 17, Issue 1). <https://ejurnal.teknokrat.ac.id/index.php/teknoinfo/index>
- Megawati, S., & Lawi, A. (2021). Pengembangan Sistem Teknologi Internet of Things Yang Perlu Dikembangkan Negara Indonesia. *Journal Information Engineering and Educational Technology*, 5(1).
- Nizam, M., Yuana, H., & Wulansari, Z. (2022). MIKROKONTROLER ESP 32 SEBAGAI ALAT MONITORING PINTU BERBASIS WEB. In *Jurnal Mahasiswa Teknik Informatika* (Vol. 6, Issue 2).
- Nur Kholifah, S., Heryana, N., Bagja Nugraha, H., Singaperbangsa Karawang, U., Ronggo Waluyo, J. H., Teluk Jambe, P., & Barat, J. (2023). ANALISIS USABILITY PADA APLIKASI HIMFO MENGGUNAKAN METODE SYSTEM USABILITY SCALE (SUS) (STUDI KASUS HIMPUNAN MAHASISWA TEKNIK INFORMATIKA UNSIKA). In *Jurnal Mahasiswa Teknik Informatika* (Vol. 7, Issue 2).
- Nuryasin, & Ferina, A. T. (2024). EVALUASI USABILITY APLIKASI SPOTIFY MENGGUNAKAN METODE SYSTEM USABILITY SCALE (SUS). *Jurnal Perangkat Lunak*, 6(2).
- Pandey, R. K., & Taruna, S. (2021). Prevalent Exact String-Matching Algorithms in Natural Language Processing: A Review. *Journal of Physics: Conference Series*, 1854(1). <https://doi.org/10.1088/1742-6596/1854/1/012042>
- Rabbani, M. K. (2021). *SISTEM KENDALI PAGAR RUMAH BERBASIS INTERNET OF THINGS* [Universitas Islam Negeri Maulana Malik Ibrahim]. <http://etheses.uin-malang.ac.id/28330/7/17650057.pdf>
- Ramadhan, D. W., Soedijono, B., & Pramono, E. (2019). PENGUJIAN USABILITY WEBSITE TIME EXCELINDO MENGGUNAKAN SYSTEM USABILITY SCALE (SUS) (STUDI KASUS: WEBSITE TIME EXCELINDO). *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 4(2), 139–147. <https://excelindo.co.id>
- Safwani, H., Suendri, & Triase. (2021). Pencarian Arti Ayat Al-Qur'an dengan Speech Recognition Menggunakan Algoritma Berry Ravindran Berbasis Android. *JISTech (Journal of Islamic Science and Technology)*, 6(2).
- Sahib, H. I., Rahman, N. H. A., Al-Qaysi, A. K., & Attiah, M. L. (2021). Comparison of data recovery techniques on master file table between Aho-Corasick and logical data recovery based on efficiency. *Telkomnika (Telecommunication Computing Electronics and Control)*, 19(1), 73–78. <https://doi.org/10.12928/TELKOMNIKA.V19I1.16276>

- Salamah, I., Handayani, S., & Youlanda, A. (2020). Aplikasi Untuk Al-Qur'an Audio Juz 30 Bagi Penyandang Tunanetra Menggunakan Voice Recognition Berbasis Android. In *Journal of Information Sistem Research (JOSH)* (Vol. 1, Issue 4).
- Sari, S. S., & Ginting, G. (2021). Implementasi Algoritma Boyer Moore Pada Kamus Perbedaan Kata Dalam Bahasa Inggris British dan Bahasa Inggris America. *Journal of Informatics Management and Information Technology*, 1(2), 74–78.
- Satria, M. N. D., Saputra, F., & Pasha, D. (2020). MIT APP INVERTOR PADA APLIKASI SCORE BOARD UNTUK PERTANDINGAN OLAHRAGA BERBASIS ANDROID. *Jurnal Teknoinfo*, 14(2), 81. <https://doi.org/10.33365/jti.v14i2.665>
- Siahaan, B. P., & Paradongan. (2020). *IMPLEMENTASI ALGORITMA AHO – CORASICK UNTUK PENCARIAN DOCUMENT FULL TEKS TIPE PDF* [Doctoral dissertation, Institut Teknologi Indonesia]. <http://repository.iti.ac.id/jspui/handle/123456789/480>
- Siahaan, M., Harsana Jasa, C., Anderson, K., Rosiana, M. V., Lim, S., & Yudianto, W. (2020). Penerapan Artificial Intelligence (AI) Terhadap Seorang Penyandang Disabilitas Tunanetra. In *Journal of Information System and Technology* (Vol. 01).
- Siliwangi, J. P., Irwan, A., & Kiswantono, A. (2023). MEMBUAT SPEAKER BLUETOOTH HELM DENGAN MODUL PENERIMA BLUETOOTH 4.1. *Jurnal Pengabdian Siliwangi*, 9(1).
- Suhartono, M. (2018). Speaker recognition in content-based image retrieval for a high degree of accuracy. *Bulletin of Electrical Engineering and Informatics*, 7(3), 350–358. <https://doi.org/10.11591/eei.v7i3.957>
- Sulaeman, N. F., & Murnawan, M. (2023). Implementasi Algoritma Aho-Corasick pada Pencarian di Aplikasi Lost and Found. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, 9(3), 509–515.
- Susanto, F., Komang Prasiani, N., & Darmawan, P. (2022). IMPLEMENTASI INTERNET OF THINGS DALAM KEHIDUPAN SEHARI-HARI. In *Jurnal IMAGINE* (Vol. 2, Issue 1). Online. <https://jurnal.std-bali.ac.id/index.php/imagine>
- Trivedi, U. (2020). An Optimized Aho-Corasick Multi-Pattern Matching Algorithm for Fast Pattern Matching. *2020 IEEE 17th India Council International Conference (INDICON)*, 1–5. <https://doi.org/10.1109/INDICON49873.2020.9342041>
- Ulum, M. (2022). *Sistem Kontrol Gorden Berbasis Internet of Things menggunakan Algoritma Knuth Morris Pratt*. Universitas Islam Negeri Maulana Malik Ibrahim.

Wagyana, A. (2019). Prototipe Modul Praktik untuk Pengembangan Aplikasi Internet of Things (IoT). *Setrum: Sistem Kendali-Tenaga-Elektronika-Telekomunikasi-Komputer*, 8(2), 238–247.

Wisnu, A. (2024). *SISTEM KENDALI PINTU RUMAH BERBASIS INTERNET OF THINGS MENGGUNAKAN ALGORITMA RABIN-KARB* [Universitas Islam Negeri Maulana Malik Ibrahim]. <http://etheses.uin-malang.ac.id/65986/>

LAMPIRAN

Lampiran 1

VIDEO DEMONSTRASI SISTEM

Link Drive:

<https://bit.ly/DemoSistemQirabox>

Lampiran 2

LEMBAR PENGUJIAN ERROR SISTEM

HASIL PENGUJIAN ERROR SISTEM				
A.n Suratna				
No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	nyalakan speaker	✓		
Perintah 2	naikkan volume	✓		
Perintah 3	turunkan volume	✓		
Perintah 4	Surah al-ikhlas	✓		respon lambat
Perintah 5	Putar surah al-qadr	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

A.n. Manik Setyani

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	hentikan speaker	✓		
Perintah 2	nyala speaker	✓		tidak sesuai karena sudah nyala
Perintah 3	mulai speaker	✓		
Perintah 4	putar surat al ashr	✓		tidak sesuai pattern "surat"
Perintah 5	putar surat al ashr	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Lasimon

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	naikkan volume	✓		
Perintah 2	Surat al annas	✓		tidak sesuai pattern "Surah"
Perintah 3	Surah al an-nas	✓		
Perintah 4	Surah ad dhuhā	✓		
Perintah 5	Stop speaker ini	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Imam Taufik

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	Hentikan speaker	✓		
Perintah 2	nyala speaker	✓		"Speaker sudah nyala
Perintah 3	putar surah an-naba	✓		
Perintah 4	putar surah al-abiyat	✓		
Perintah 5	matikan speaker	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Wati

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	mulai speaker	✓		
Perintah 2	putar surah annas	✓		
Perintah 3	naikkan volume	✓		
Perintah 4	tambah naik volume	✓		
Perintah 5	Matikan speaker	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Wandi

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	Nyalakan Speaker	~		
Perintah 2	putar surah an-nas	✓		
Perintah 3	putar surah ad duha	✓		
Perintah 4	stop speaker	✓		
Perintah 5	Mulai Lagi	~		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Gardianto.

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	nyalakan speaker	✓		
Perintah 2	nyala speaker	✓		"Speaker sudah menyala"
Perintah 3	putarkan surah an-naba	✓		
Perintah 4	matikan speaker	✓		
Perintah 5	nyalakan speaker ini	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM

An. Winik Nur'aini

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	Nyalakan speaker	✓		
Perintah 2	Matikan speaker	✓		
Perintah 3	tolong nyalakan speaker	✓		
Perintah 4	putar surat an-naba			
Perintah 5	putar surah an-naba	✓		
Total Error Sistem		0		

HASIL PENGUJIAN ERROR SISTEM				
Bapak M. Muryono				
No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	tolong nyalakan speaker		✓	di dlm speaker sudah nyala. kondisi real = mati
Perintah 2	nyalakan speaker	✓		bisa nyala setelah dlm diubah
Perintah 3	matikan speaker	✓		
Perintah 4	tolong nyalakan speaker	✓		
Perintah 5	Putarkan surah an-naba	✓		
Total Error Sistem		1		

HASIL PENGUJIAN ERROR SISTEM

Tri Marta.

No. Perintah	Perintah	Respon Sistem		Catatan
		Merespon	Tidak Merespon	
Perintah 1	Turunkan volume nya	✓		
Perintah 2	Turunkan volume lagi	✓		
Perintah 3	putar surah al-qariah	✓		
Perintah 4	putarkan surah alail	✓		
Perintah 5	matikan speakernya	✓		
Total Error Sistem		0		

Lampiran 3

LEMBAR PENGUJIAN SUS

NAMA RESPONDEN : Bapak Suratno

USIA : 51 tahun

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra				✓	
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.				✓	
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.				✓	
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.				✓	
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.					✓
10.	Saya merasa sistem kontrol speaker ini membingungkan	✓				

NAMA RESPONDEN : Ibu Hanik Setyaning

USIA : 50 tahun

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra					✓
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.	✓				
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.				✓	
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.			✓		
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.					✓
10.	Saya merasa sistem kontrol speaker ini membingungkan				✓	

NAMA RESPONDEN : Lasiman

USIA : 45 th

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra				✓	
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.		✓			
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.				✓	
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an		✓			
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.				✓	
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.				✓	
10.	Saya merasa sistem kontrol speaker ini membingungkan			✓		

NAMA RESPONDEN : Imam Taufik

USIA : 42 th.

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra					✓
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.			✓		
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.				✓	
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an			✓		
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.		✓			
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.			✓		
10.	Saya merasa sistem kontrol speaker ini membingungkan		✓			

NAMA RESPONDEN : Wati

USIA : 42 tahun.

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra			✓		
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.			✓		
3.	Saya merasa nyaman menggunakan sistem ini				✓	
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.			✓		
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an					✓
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.			✓		
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.					✓
10.	Saya merasa sistem kontrol speaker ini membingungkan			✓		

NAMA RESPONDEN : Wandi

USIA : 41 tahun.

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra				✓	
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.			✓		
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.				✓	
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.				✓	
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.					✓
10.	Saya merasa sistem kontrol speaker ini membingungkan		✓			

NAMA RESPONDEN : Sardianto

USIA : 61 tahun

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra					✓
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.	✓				
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.		✓			
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.		✓			
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.				✓	
10.	Saya merasa sistem kontrol speaker ini membingungkan			✓		

NAMA RESPONDEN : Wiwik Nur Aini

USIA : 45 tahun

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra					✓
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.	✓				
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.			✓		
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.		✓			
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.				✓	
10.	Saya merasa sistem kontrol speaker ini membingungkan		✓			

NAMA RESPONDEN : Tri Marta Yuana
 USIA : 34 tahun.

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra					✓
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.	✓				
3.	Saya merasa nyaman menggunakan sistem ini					✓
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.	✓				
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an					✓
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an	✓				
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya					✓
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.	✓				
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.					✓
10.	Saya merasa sistem kontrol speaker ini membingungkan	✓				

NAMA RESPONDEN : Suryono
 USIA : 45 tahun

No	Pernyataan	1	2	3	4	5
1.	Saya merasa sistem ini akan sering digunakan tunanetra			✓		
2.	Saya merasa sistem ini terlalu rumit untuk penggunaan sehari-hari tunanetra.		✓			
3.	Saya merasa nyaman menggunakan sistem ini				✓	
4.	Saya merasa sistem ini membutuhkan terlalu banyak waktu untuk dipelajari.	✓				
5.	Saya merasa terbantu atas adanya sistem ini dalam mengakses speaker murottal Al-Qur'an				✓	
6.	Saya merasa sistem ini hanya mempersulit dalam kontrol speaker murottal Al-Qur'an				✓	
7.	Saya akan merekomendasikan sistem ini untuk penyandang tunanetra lainnya			✓		
8.	Saya merasa masih membutuhkan bantuan orang lain untuk menggunakan sistem ini.		✓			
9.	Saya merasa sistem ini memberikan pengalaman pengguna yang menyenangkan.				✓	
10.	Saya merasa sistem kontrol speaker ini membingungkan				✓	

Lampiran 4

**DOKUMENTASI PENGUJIAN TERHADAP PENYANDANG
TUNANETRA**









