



Aplikasi Pengenalan Nama Surah pada Juz ke 30 Kitab Suci Al-Qur'an Menggunakan *Speech Recognition*

Dhimas Sena Rahmantara^a, Kartina Diah Kesuma Wardhani^b, Maksu Ro'is Adin Saf^c

^aProgram Studi Teknik Informatika, Politeknik Caltex Riau, email: dhimas14ti@mahasiswa.pcr.ac.id

^bProgram Studi Teknik Informatika, Politeknik Caltex Riau, email: diah@pcr.ac.id

^cProgram Studi Teknik Informatika, Politeknik Caltex Riau, email: maksu@pcr.ac.id

Al-Qur'an is a scripture which contains the saying of Allah Subhanahu Wa Ta'aala and was revealed to Prophet Muhammad. The 30th juz is the juz that exists in the Al-Qur'an. When studying how to read Al-Qur'an well, the first thing that is learned is reading and memorizing surahs in the 30th juz. Nevertheless, there is a problem in remembering or knowing the surah name and the verse which are in the 30th juz. An android application was developed in order to recognize the surah names in the 30th juz by utilizing speech recognition technology to overcome that problem. Markov Model (Markov Chain) algorithm was implemented in this application. This algorithm will process user's speech and compute probability of the surah name that was spoken. Speech detection testing gave result that the highest accuracy of application in recognizing the speeches was in the environment without noise with the accuracy of 100% in the most ideal distance is 50 cm for male and for female user. Based on the blackbox testing result, all functionalities of the application have functionated well. Control flow testing gave result that the value is 7 which indicates that the code is simple and well written. 87,74% respondents answered, by filling up the questionnaires, that the application is useful in order to make user knows better about the surah names in the 30th juz.

Keywords: Al-Qur'an, 30th Juz, Markov Model, Speech Recognition.

Abstrak

Al-Qur'an adalah kitab suci yang berisi firman Allah Subhanahu Wa Ta'aala yang diwahyukan kepada Nabi Muhammad. Juz ke-30 merupakan juz yang terdapat pada kitab suci Al-Qur'an. Ketika belajar membaca Al-Qur'an, hal pertama yang dipelajari adalah membaca dan menghafal surah-surah yang terdapat pada juz ke-30. Meskipun demikian, terdapat permasalahan dalam hal mengingat atau mengetahui nama surah dan ayat-ayat pada juz ke-30. Untuk mengatasi masalah tersebut, maka dirancang aplikasi pengenalan nama surah Al-Qur'an pada juz ke-30 berbasis *android* dengan memanfaatkan teknologi *speech recognition*. Algoritma yang diterapkan pada aplikasi ini adalah algoritma *Markov Model (Markov Chain)*. Algoritma ini mengolah inputan suara yang diberikan oleh *user* dan menghitung probabilitas nama surah yang diucapkan. Setelah dilakukan pengujian keberhasilan deteksi suara, akurasi tertinggi aplikasi dalam menerima inputan suara adalah pada lingkungan tanpa *noise* dengan akurasi sebesar 100% pada jarak yang paling ideal adalah sejauh 50 cm untuk suara *user* laki-laki dan perempuan. Berdasarkan hasil *blackbox testing*, keseluruhan fungsionalitas aplikasi telah berfungsi dengan baik. Berdasarkan hasil *control flow testing*, nilai *cyclomatic complexity* adalah 7 yang menunjukkan kode program tersusun sederhana dan ditulis dengan baik. Hasil pengujian kuesioner diperoleh 87,74% responden menyatakan aplikasi ini dapat membantu *user* lebih mengetahui nama-nama surah pada juz ke-30.

Kata Kunci: Al-Qur'an, Juz ke-30, *Markov Model, Speech Recognition*.

© 2018 Jurnal RESTI

1. Pendahuluan

Pesatnya perkembangan teknologi informasi dewasa ini menghadirkan teknologi yang memungkinkan interaksi antara manusia dan komputer terjadi melalui perantara suara sebagai media inputan. Teknologi ini lebih umum dikenal sebagai *speech recognition*. Definisi dari *speech recognition* adalah kemampuan

suatu perangkat elektronik untuk mengenali kata-kata yang diucapkan [1].

Teknologi *speech recognition* telah diimplementasikan di beberapa bidang. Salah satu contohnya adalah di bidang telekomunikasi yang mengimplementasikan *speech recognition* sebagai media untuk mengetikkan pesan singkat (SMS)

melalui *smartphone*. Di bidang *home automation*, *speech recognition* diterapkan sebagai media untuk memonitor dan mengontrol kondisi rumah dengan perintah suara [2]. Salah satu algoritma yang dapat diterapkan pada *speech recognition* adalah algoritma *Markov Model*. *Markov Model* dapat digunakan sebagai algoritma pencarian data dan mencari nilai probabilitas suatu kata pada teknologi *speech recognition* ini. Teknologi *speech recognition* juga telah diimplementasikan pada perangkat lunak kitab suci Al-Qur'an dalam versi *digital*.

Kitab suci Al-Qur'an sendiri terdiri dari 30 juz dan 114 surah yang terdapat di dalamnya. Saat memulai untuk menghafalkan nama surah dan ayat Al-Qur'an, hal yang pertama dilakukan adalah dengan membaca surah-surah yang terdapat pada juz ke-30. Hal ini dikarenakan surah-surah pada juz ke-30 memiliki jumlah ayat yang relatif lebih sedikit dibandingkan dengan jumlah ayat pada surah-surah yang terdapat pada juz lainnya [3].

Dari uraian di atas, untuk membantu *user* lebih mengetahui nama-nama surah, ayat surah, jumlah ayat, terjemahan, isi kandungan dan latar belakang/asbabun yang terdapat pada juz ke-30, maka dibangun sebuah aplikasi *android* yang memanfaatkan teknologi *speech recognition*. Aplikasi ini akan memperdengarkan audio secara *random* atau acak kepada *user*. Kemudian *user* memberikan *input* dengan menebak nama surah yang diperdengarkan. Perangkat *android* akan menangkap inputan suara nama surah yang diberikan oleh *user*. Algoritma *Markov Model* akan mengolah inputan suara tersebut dengan mencocokkannya dengan nama surah yang ada pada aplikasi. Jika tebakan *user* benar, maka *output* yang akan dihasilkan adalah mengenai nama surah, jumlah ayat, isi kandungan surah, dan latar belakang turunnya (asbabun nuzul) surah.

2. Tinjauan Pustaka

2.1 Penelitian Terdahulu

Penelitian tentang *speech recognition* telah dilakukan untuk *unlock* pada perangkat *android*. Suara yang diberikan akan berfungsi sebagai media untuk membuka (*unlock*) perangkat *android* yang sebelumnya dalam keadaan terkunci (*locked*) [4].

Penelitian lainnya adalah penerapan *speech recognition* untuk menampilkan animasi 3 dimensi yang berbentuk robot dengan menggunakan *speech recognition* sebagai media untuk menghasilkan aksi obyek robot 3 dimensi tersebut [5].

Penelitian selanjutnya adalah penelitian untuk memudahkan presentasi dengan menggunakan inputan berupa suara. Tujuan dirancangnya aplikasi ini adalah untuk menggantikan fungsi *mouse* dan *keyboard* saat *presenter* sedang melakukan presentasi menggunakan *Microsoft PowerPoint* [6].

Speech recognition juga dapat diimplementasikan untuk pengontrolan lampu. Penelitian yang dilakukan ini bertujuan untuk mengenali ucapan yang diberikan *user*, kemudian ucapan akan dicocokkan dan diproses untuk menghidupkan dan mematikan lampu [7].

2.2 Al-Qur'an dan Juz ke-30

Al-Qur'an adalah kitab suci umat Islam yang berisi firman Allah Subhanahu Wa Ta'aala yang diturunkan kepada Nabi Muhammad Shallallahu 'Alaihi Wasallam dengan perantaraan Malaikat Jibril untuk dibaca, dipahami, dan diamalkan sebagai petunjuk atau pedoman hidup bagi umat manusia [8].

Juz ke-30 atau disebut juga juz amma, merupakan juz terakhir yang terdapat pada kitab suci Al-Qur'an dan bagian yang paling sering didengar dan paling sering dibaca. Ketika belajar membaca Al-Qur'an, hal pertama yang dipelajari adalah membaca dan menghafal surah-surah yang terdapat dalam juz ke-30 ini [3]. Juz ke-30 merupakan bagian di dalam Al-Qur'an yang memiliki jumlah surah terbanyak, dengan jumlah 37 surah di dalamnya.

2.3 Speech Recognition

Speech recognition adalah sebuah teknologi yang memakai peralatan seperti *microphone* untuk menangkap inputan suara untuk berinteraksi dengan komputer. Teknologi ini tidak sama dengan teknologi *voice recognition* yang mengenali suara sebagai identifikasi keamanan [6]. Noertjahyana & Adipranata menjelaskan bahwa terdapat dua mode atau cara pada *speech recognition* [9], yaitu mode diktasi dan mode *command and control*.

Pada mode diktasi, jumlah kata yang dapat dikenali dibatasi oleh jumlah kata yang telah terdapat pada *database*. Pengenalan mode diktasi ini merupakan *speaker dependent*. Keakuratan mode ini bergantung pada pola suara dan akses pembicara serta pelatihan yang telah dilakukan. Sedangkan pada mode *command and control*, kata atau kalimat yang sudah didefinisikan terlebih dahulu pada *database*. Mode ini merupakan *speaker independent* karena jumlah kata yang dikenali biasanya terbatas dan pengguna tidak perlu melakukan pelatihan pada sistem.

2.4 Android

Android merupakan kumpulan perangkat lunak yang ditujukan bagi perangkat bergerak yang mencakup sistem operasi, *middleware*, dan aplikasi. Android berjalan di *Dalvik Virtual Machine* (DVM) dimana *Virtual Machine* ini dirancang sebagai *runtime environment* untuk memastikan bahwa beberapa fitur berjalan lebih efisien pada perangkat *mobile* [10]. Supardi [10] menjelaskan bahwa arsitektur android terdiri dari *Applications & Widgets*, *Applications Frameworks*, *Libraries*, *Android Run Time*, serta *Linux Kernel*.

2.5 Markov Model

Markov Model atau yang juga dikenal sebagai *markov chain* adalah suatu teknik matematika yang digunakan untuk *modelling* berbagai macam sistem dan *business process*. Teknik ini dapat digunakan untuk meramalkan perubahan yang terjadi di masa depan berdasarkan variabel-variabel dinamis pada kejadian dan pengamatan di masa lalu. *Markov Model* tidak memberikan keputusan rekomendasi, tetapi hanya informasi probabilitas atau peluang yang dapat membantu pengambil keputusan. Dengan kata lain, *Markov Model* bukanlah teknik untuk optimasi, melainkan teknik deskriptif mengenai informasi probabilitas atau peluang di masa depan [11].

Sebuah rantai *markov* adalah urutan variabel-variabel acak seperti X_1, X_2, X_3, \dots hingga X_n dengan sifat Markov yaitu, mengingat keadaan masa depan dan masa lalu dengan keadaan yang independen, dengan kata lain:

$$P(X_{n+1} = j | X_1 = x_1, X_2 = x_2, X_3 = x_3, \dots, X_n = i) = P(X_{n+1} = j | X_n = i) = P_{ij} \quad (1)$$

Dimana i adalah variabel yang akan dicari dan j adalah variabel yang diketahui. Sehingga didapatkan peluang munculnya variabel i jika nilai variabel j diketahui [12].

3. Metodologi Penelitian

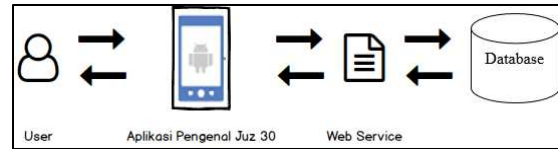
Metodologi penelitian yang terdapat pada penelitian ini adalah sebagai berikut.

3.1 Studi Literatur

Studi literatur yang dilakukan yaitu dengan mempelajari buku-buku, jurnal-jurnal, dan situs forum serta sumber literatur lainnya yang digunakan untuk mendukung perancangan dan pembangunan aplikasi.

3.2 Perancangan Arsitektur Sistem

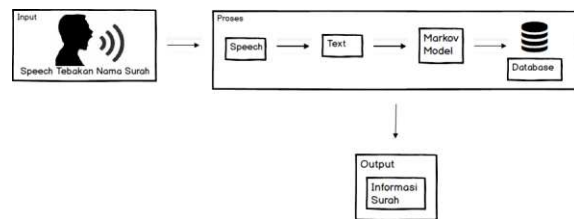
Arsitektur sistem merupakan suatu gambaran yang memetakan bentuk dari suatu sistem atau aplikasi. Gambar 1 menunjukkan perancangan arsitektur sistem yang dibangun.



Gambar 1. Perancangan Arsitektur Sistem

3.3 Perancangan Block Diagram

Block diagram atau diagram blok merupakan gambaran yang menjelaskan mengenai tiga hal, yaitu *input*, proses, dan *output* yang terdapat pada aplikasi yang akan dibangun. *Block diagram* aplikasi ini dapat dilihat pada Gambar 2.



Gambar 2. Perancangan Block Diagram

3.4 Perancangan Flowchart Penggunaan Aplikasi

Alur atau *flowchart* penggunaan aplikasi dapat dilihat pada Gambar 3.

3.5 Perancangan Use Case Diagram

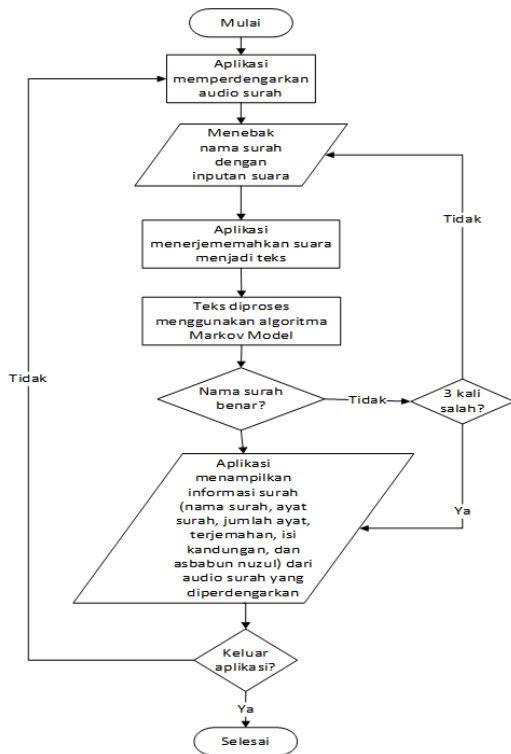
Use case diagram adalah diagram yang menggambarkan sisi fungsionalitas dari aplikasi yang akan dirancang. Gambar 4 berikut ini memperlihatkan *use case diagram* yang terdapat pada aplikasi. Lihat Gambar 4.

3.6 Perancangan Tabel Database

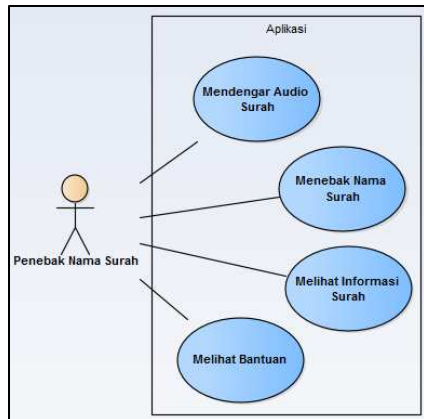
Tabel 1 adalah tabel yang digunakan untuk pembuatan *database* pada aplikasi yang dibangun.

3.7 Pembangunan Aplikasi

Pembangunan aplikasi *android* ini menggunakan perangkat lunak IDE *Android Studio* dengan *Java* sebagai bahasa pemrograman dan basis data (*database*) *MySQL*.



Gambar 3. Flowchart Penggunaan Aplikasi



Gambar 4. Perancangan Use Case Diagram

Tabel 1. Perancangan Tabel Database

Nama Atribut	Tipe Data	Keterangan
No_surah	Int (3)	Primary Key
Nama_surah	Varchar (100)	Nama surah
Ayat_surah	Text	Ayat surah
Audio_surah	Varchar (100)	Path link audio surah
Informasi_surah	Text	Informasi surah

3.8 Pengujian

Pengujian yang dilakukan pada penelitian ini meliputi pengujian fungsionalitas (*blackbox testing*), pengujian kode program (*control flow testing*), pengujian keberhasilan deteksi suara oleh perangkat *android*, dan pengujian kuesioner.

4. Hasil dan Pembahasan

Pada bagian ini akan dijelaskan mengenai hasil perancangan tampilan aplikasi dan mengenai hasil pengujian

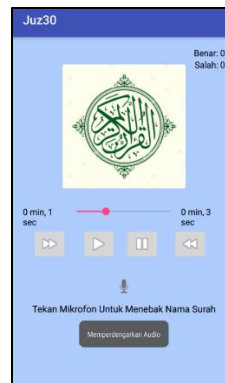
4.1 Hasil Perancangan Tampilan Aplikasi

Gambar 5 merupakan tampilan menu utama yang terdapat di dalam aplikasi. Terdapat tiga menu utama, yaitu menu Dengarkan dan Tebak Audio Surah, menu Bantuan Penggunaan Aplikasi, dan menu Tentang Aplikasi.



Gambar 5. Tampilan Menu Utama

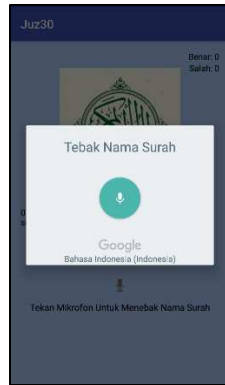
Gambar 6 adalah tampilan yang digunakan oleh *user* untuk mendengarkan audio surah yang akan diperdengarkan secara *random* atau acak.



Gambar 6. Tampilan Mendengarkan Audio Surah

Gambar 7 adalah tampilan yang akan memunculkan *prompt* yang digunakan oleh *user* untuk menebak

nama surah dari audio surah yang telah diperdengarkan.



Gambar 7. Tampilan Menebak Nama Surah

Gambar 8 adalah tampilan yang akan muncul jika *user* yang menebak nama surah dengan tebakan nama surah yang salah.



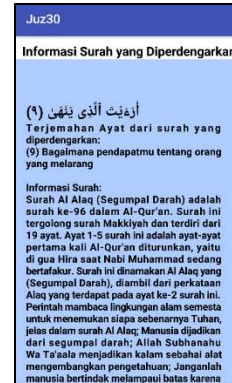
Gambar 8. Tampilan Jika Tebakan Nama Surah Salah

Jika *user* yang menebak nama surah dengan tebakan nama surah yang benar, maka aplikasi akan memberikan notifikasi atau tampilan tebakan nama surah benar. Tampilan jika tebakan user benar dapat dilihat pada Gambar 9.



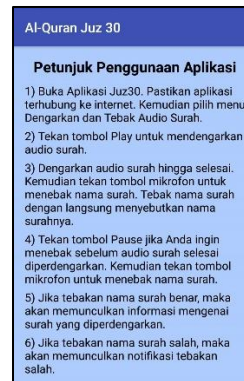
Gambar 9. Tampilan Jika Tebakan Nama Surah Benar

Gambar 10 adalah tampilan informasi surah yang diperdengarkan akan dapat dilihat oleh *user* jika tebakan nama surah yang adalah tebakan yang benar. Pada tampilan ini akan berisi informasi tentang nama surah, jumlah ayat, isi kandungan surah, dan sebab turunnya (asbabun nuzul) surah yang diperdengarkan tersebut.



Gambar 10. Tampilan Informasi Surah yang Diperdengarkan

Gambar 11 adalah tampilan yang dapat digunakan oleh *user* untuk melihat petunjuk penggunaan aplikasi agar *user* menjadi lebih mudah dalam menggunakan aplikasi ini.



Gambar 11. Tampilan Bantuan Penggunaan Aplikasi

4.2 Pengujian Fungsionalitas (*Blackbox Testing*)

Pengujian fungsionalitas (*blackbox testing*) merupakan pengujian yang memfokuskan pada setiap fungsi dari aplikasi sudah berjalan sebagaimana seharusnya. Pengujian dilakukan pada fungsionalitas mendengarkan audio surah, menebak nama surah, melihat informasi surah, dan melihat bantuan yang terdapat pada aplikasi. Tabel 2 menunjukkan hasil pengujian fungsionalitas (*blackbox testing*).

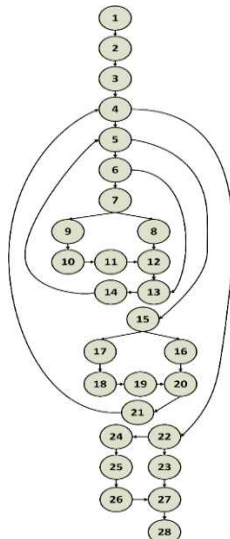
Tabel 2. Pengujian Fungsionalitas (*Blackbox Testing*)

No	Nama Use Case	Jenis Path Testing	Hasil yang Diharapkan	Hasil Pengujian	Keterangan
1	Mendengar Audio Surah	<i>Basic Path:</i> User berada pada tampilan menu utama. Kemudian user memilih menu Dengarkan dan Tebak Audio Surah pada menu utama. Aplikasi memunculkan tampilan Dengarkan dan Tebak Audio Surah. Lalu user memilih tombol <i>play</i> , maka aplikasi akan memperdengarkan audio surah secara <i>random/acak</i> . <i>Alternate path:</i> Jika aplikasi tidak terhubung ke internet, maka akan memunculkan notifikasi "Network not connected" saat user memilih menu Dengarkan dan Tebak Audio Surah.	Aplikasi akan memperdengarkan audio surah secara <i>random/acak</i> . Aplikasi memunculkan notifikasi "Network not connected".	Sesuai Harapan	Valid
2	Menebak Nama Surah	<i>Basic Path:</i> User memilih tombol Dengarkan dan Tebak Audio Surah. Kemudian aplikasi memunculkan <i>prompt</i> kepada user untuk menebak nama surah yang telah diperdengarkan. <i>Alternate path:</i> Jika aplikasi tidak terhubung ke internet, maka akan memunculkan notifikasi "Network not connected" saat user memilih tombol mikrofon.	Aplikasi memunculkan <i>prompt</i> kepada user untuk menebak nama surah yang telah diperdengarkan. Lalu aplikasi mengolah suara tebakan user. Aplikasi memunculkan notifikasi "Network not connected".	Sesuai Harapan	Valid
3	Melihat Informasi	<i>Basic Path:</i> User telah menebak nama surah dengan tebakan yang benar.	Aplikasi memunculkan notifikasi	Sesuai Harapan	Valid

Surah	Kemudian aplikasi memunculkan notifikasi "Tebakan Nama Surah Benar" dan memunculkan <i>output</i> informasi dari surah yang ditebak.	"Tebakan Nama Surah Benar" dan memunculkan <i>output</i> informasi dari surah yang ditebak.	n		
4	Melihat Bantuan	<i>Alternate path:</i> Jika user menebak dengan tebakan nama surah yang salah, maka aplikasi memunculkan notifikasi "Tebakan Nama Surah Salah". <i>Basic Path:</i> User berada pada tampilan menu utama. Kemudian user memilih menu Bantuan Penggunaan Aplikasi. Lalu aplikasi memunculkan tampilan Bantuan Penggunaan Aplikasi.	Aplikasi memunculkan notifikasi "Tebakan Nama Surah Salah". Aplikasi memunculkan tampilan Bantuan Penggunaan Aplikasi.	Sesuai Harapan	Valid

4.3 Pengujian Kode Program (*Control Flow Testing*)

Control Flow Testing adalah pengujian yang dilakukan terhadap alur kerja kode program untuk menentukan kode program tersebut sudah sesuai atau belum terhadap suatu proses algoritma. Pengujian ini dilakukan pada algoritma *Markov Model*. Gambar 13 menunjukkan graf kode program yang telah dibuat.



Gambar 12. Graf Kode Program Algoritma *Markov Model*

Berdasarkan graf kode program yang dihasilkan, maka nilai *cyclomatic complexity* dihitung dengan persamaan sebagai berikut:

$$V(G) = E - N + 2 \quad (2)$$

E = banyak Edge (panah); N = banyak Node.

$$V(G) = 33 - 28 + 2 = 5 + 2 = 7$$

Terdapat 7 jalur eksekusi program yang terjadi. Seluruh jalur eksekusi tersebut adalah sebagai berikut:

1. 1-2-3-4-5-6-7-8-12-13-14-5-15-16-20-21-4-22-23-27-28
2. 1-2-3-4-5-6-7-8-12-13-14-5-15-17-18-19-20-21-4-22-23-27-28
3. 1-2-3-4-5-6-7-9-10-11-12-13-14-5-15-16-20-21-4-22-23-27-28
4. 1-2-3-4-5-6-7-9-10-11-12-13-14-5-15-17-18-19-20-21-4-22-23-27-28
5. 1-2-3-4-5-6-13-14-5-15-16-20-21-4-22-23-27-28
6. 1-2-3-4-5-6-13-14-5-15-17-18-19-20-21-4-22-23-27-28
7. 1-2-3-4-5-6-7-9-10-11-12-13-14-5-15-17-18-19-20-21-4-22-24-25-26-27-28

4.4 Pengujian Akurasi dan Keberhasilan Deteksi

Hasil pengujian deteksi perangkat *android* dengan variabel jarak, suara *user* laki-laki, dan kondisi lingkungan (tanpa *noise* / dengan *noise*). Setiap *user* melakukan percobaan deteksi suara sebanyak lima kali untuk setiap satuan jarak pengujian. Hasilnya dapat dilihat pada Tabel 3 dan Tabel 4.

Tabel 3. Rata-rata Keberhasilan Deteksi *User* Laki-laki pada Lingkungan Tanpa *Noise*

<i>User</i>	50 CM	1M	2M	3M	4M	5M	6M
<i>User1</i>	100%	100%	60%	40%	0%	0%	0%
<i>User2</i>	100%	100%	80%	60%	40%	20%	0%
<i>User3</i>	100%	100%	40%	40%	0%	0%	0%
<i>User4</i>	100%	100%	100%	20%	0%	0%	0%
<i>User5</i>	100%	100%	100%	100%	80%	0%	0%
Rata-rata Keberhasilan Tiap Satuan Jarak	100%	100%	76%	52%	24%	4%	0%

Tabel 4. Rata-rata Keberhasilan Deteksi *User* Laki-laki pada Lingkungan yang Memiliki *Noise*

<i>User</i>	50CM	1M	2M	3M	4M	5M	6M
<i>User1</i>	100%	80%	0%	0%	0%	0%	0%
<i>User2</i>	100%	100%	40%	20%	0%	0%	0%
<i>User3</i>	80%	80%	0%	0%	0%	0%	0%
<i>User4</i>	80%	20%	0%	0%	0%	0%	0%
<i>User5</i>	100%	20%	0%	0%	0%	0%	0%
Rata-rata Keberhasilan Tiap Satuan Jarak	92%	60%	8%	4%	0%	0%	0%

Seperti halnya pengujian deteksi pada *user* laki-laki, pengujian yang dilakukan pada *user* perempuan menggunakan variabel jarak, suara *user* perempuan, dan kondisi lingkungan (tanpa *noise* / dengan *noise*) dimana setiap *user* melakukan percobaan deteksi suara sebanyak lima kali untuk setiap satuan jarak pengujian. Untuk hasil pengujian dengan *user* perempuan dapat dilihat pada Tabel 5 dan Tabel 6.

Tabel 5. Rata-rata Keberhasilan Deteksi *User* Perempuan pada Lingkungan Tanpa *Noise*

<i>User</i>	50 CM	1M	2M	3M	4M	5M	6M
<i>User1</i>	100%	100%	100%	0%	0%	0%	0%
<i>User2</i>	100%	100%	100%	20%	20%	0%	0%
<i>User3</i>	100%	60%	40%	20%	0%	0%	0%
<i>User4</i>	100%	80%	20%	0%	0%	0%	0%
<i>User5</i>	100%	100%	80%	20%	0%	0%	0%
Rata-rata Keberhasilan Tiap Satuan Jarak	100%	88%	68%	12%	4%	0%	0%

Tabel 6. Rata-rata Keberhasilan Deteksi *User* Perempuan pada Lingkungan yang Memiliki *Noise*

<i>User</i>	50CM	1M	2M	3M	4M	5M	6M
<i>User1</i>	40%	0%	0%	0%	0%	0%	0%
<i>User2</i>	20%	0%	0%	0%	0%	0%	0%
<i>User3</i>	0%	0%	0%	0%	0%	0%	0%
<i>User4</i>	0%	0%	0%	0%	0%	0%	0%
<i>User5</i>	20%	20%	0%	0%	0%	0%	0%
Rata-rata Keberhasilan Tiap Satuan Jarak	16%	4%	0%	0%	0%	0%	0%

4.5 Pengujian Kuesioner

Kuesioner ini dibagikan kepada sebanyak 31 orang responden untuk mendapatkan respons dari masing-masing responden tersebut. Jumlah 31 orang responden ini didasarkan pada pendapat Roscoe dalam Sekaran & Bougie yang menyatakan bahwa minimal 30 sampel responden adalah cocok untuk kebanyakan penelitian [13]. Rekapitulasi jawaban dari pertanyaan yang diajukan dapat dilihat pada Tabel 7 berikut ini.

Tabel 7. Rekapitulasi Jawaban Kuesioner

No	Pertanyaan	STS (1)	TS (2)	N (3)	S (4)	SS (5)
1	Apakah audio surah pada aplikasi ini dapat didengarkan dengan baik?	0	0	2	16	13
2	Apakah aplikasi ini dapat menerima atau mengenali suara yang diucapkan dengan akurat?	0	1	5	17	8
3	Apakah aplikasi memberikan <i>output</i> /keluaran berupa informasi surah yang sesuai dengan audio surah yang diperdengarkan?	0	0	4	15	12

No	Pertanyaan	STS (1)	TS (2)	N (3)	S (4)	SS (5)
5	Apakah dengan aplikasi ini dapat membantu Anda mengetahui informasi surah yang berupa nama surah, ayat surah, jumlah ayat, terjemahan, isi kandungan dan latar belakang/asbabun nuzul dari ayat-ayat atau surah-surah yang terdapat pada juz ke-30?	0	0	4	11	16

Dari rekapitulasi jawaban kuesioner di atas, maka dilakukan perhitungan menggunakan skala likert. Rumus perhitungan skala likert adalah sebagai berikut:

$$\text{Nilai rata-rata (mean)} = \frac{\sum f_i \cdot x_i}{\sum f_m} \times 100\% \quad (3)$$

f_i = jumlah *user* ke- i ;

x_i = nilai bobot ke- i ;

f_m = total *user* dikali dengan nilai bobot tertinggi.

Dengan menggunakan rumus skala likert, responden menjawab sangat setuju bahwa audio surah pada aplikasi dapat didengarkan dengan baik. Nilai rata-rata jawaban pertanyaan ini adalah sebesar 87,1%.

Selanjutnya responden memberikan tanggapan sangat setuju bahwa aplikasi dapat menerima atau mengenali suara yang diucapkan dengan akurat. Nilai rata-rata jawaban pertanyaan ini adalah sebesar 80,64%.

Kemudian responden menjawab sangat setuju bahwa aplikasi memberikan *output*/keluaran berupa informasi surah yang sesuai dengan audio surah yang diperdengarkan. Nilai rata-rata jawaban pertanyaan ini adalah sebesar 85,16%.

Selanjutnya responden merasa sangat setuju bahwa aplikasi ini dapat membantu *user* lebih mengetahui nama-nama surah pada Juz ke-30. Nilai rata-rata jawaban pertanyaan ini adalah sebesar 87,74%.

Kemudian responden menjawab sangat setuju bahwa audio surah pada aplikasi ini dapat membantu *user* mengetahui informasi surah yang berupa nama surah, ayat surah, jumlah ayat, terjemahan, isi kandungan dan latar belakang/asbabun nuzul dari ayat-ayat atau surah-surah yang terdapat pada juz ke-30. Nilai rata-rata jawaban pertanyaan ini adalah sebesar 87,74%.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut:

1. Keseluruhan fitur aplikasi yang dibangun telah berfungsi dengan baik dengan pengujian *blackbox testing*.
2. Algoritma *Markov Model* telah diimplementasikan dengan baik pada aplikasi ini.
3. Algoritma *Markov Model* yang digunakan memiliki nilai *cyclomatic complexity* sebesar 7 dalam pengujian *control flow testing* yang menunjukkan kode program tersusun sederhana dan ditulis dengan baik.
4. Jarak paling ideal aplikasi *android* dalam menerima inputan suara *user* laki-laki dan perempuan adalah pada lingkungan tanpa *noise*, yaitu dengan jarak sejauh 50 cm dengan rata-rata keberhasilan deteksi sebesar 100%.
5. Perangkat *android* tidak dapat menerima inputan suara *user* laki-laki pada jarak 6 meter atau lebih pada lingkungan tanpa *noise* dan jarak 4 meter atau lebih pada lingkungan yang memiliki *noise*. Sedangkan untuk suara *user* perempuan, perangkat *android* tidak dapat mendeteksi suara pada jarak 5 meter atau lebih pada lingkungan tanpa *noise* dan jarak 2 meter atau lebih pada lingkungan yang memiliki *noise*.
6. Jawaban kuesioner menyatakan bahwa 87,1% responden menyatakan audio surah dapat didengarkan dengan baik, 80,64% menjawab aplikasi menerima atau mengenali inputan suara *user* dengan akurat, 85,16% menjawab aplikasi memberikan *output*/keluaran berupa informasi surah yang sesuai dengan audio surah yang diperdengarkan, 87,74% menyatakan bahwa dengan aplikasi ini dapat membantu *user* lebih mengetahui nama-nama surah pada Juz ke-30, dan 87,74% *user* menyatakan bahwa aplikasi ini dapat membantu *user* mengetahui informasi surah yang berupa nama surah, ayat surah, jumlah ayat, terjemahan, isi kandungan dan latar belakang/asbabun nuzul dari ayat-ayat atau surah-surah yang terdapat pada juz ke-30.

5.2 Saran

Saran yang diberikan untuk dapat digunakan pada penelitian-penelitian selanjutnya adalah sebagai berikut:

1. Mengambil studi kasus surah-surah kitab suci Al-Qur'an pada juz yang lainnya.
2. Pembuatan aplikasi dengan atau sistem dengan menggunakan algoritma *speech recognition* yang lainnya.
3. Pembuatan aplikasi *speech recognition* dapat dilakukan pada platform perangkat *mobile* lainnya, seperti iOS.

Daftar Rujukan

- [1] P. Christensson, "Speech Recognition," 2014. [Online]. Available: https://techterms.com/definition/speech_recognition. [Accessed 23 April 2017].
- [2] A. W. Dani, "Perancangan Aplikasi Voice Command Recognition Berbasis Android dan Arduino Uno," *Jurnal Teknologi Elektro Universitas Mercu Buana*, pp. Volume 1, No 1, Halaman 11-19, 2016.
- [3] A. A. Zuhdi, "Upaya Peningkatan Kemampuan Menghafal Juz 'Amma Melalui Metode Jama' Siswa Kelas IV SDIT Al-Ma'ruf Tegalrejo Magelang," Sekolah Tinggi Agama Islam Negeri Salatiga, Salatiga, 2011.
- [4] D. Natalia, "Perancangan dan Implementasi Speech Recognition Sistem sebagai Fungsi Unlock pada Handset Android," Universitas Telkom, Bandung, 2013.
- [5] T. Porwasih, "Aplikasi Speech To Text Pada Animasi Robot Pintar Berbasis Android," *Jurnal Aksara Komputer Terapan Politeknik Caltex Riau*, pp. Volume 2, No 2, Halaman 209-215, 2013.
- [6] F. P. Putra, "Aplikasi Kontrol Slide Microsoft Office Powerpoint Dengan Suara Menggunakan Teknologi Windows Speech Recognition," *Jurnal Aksara Komputer Terapan Politeknik Caltex Riau*, pp. Volume 3, No 1, Halaman 96-103, 2014.
- [7] M. Hasbi, "Speech Recognition Menggunakan Algoritma Markov Model Untuk Mengontrol Lampu," *Jurnal Aksara Komputer Terapan Politeknik Caltex Riau*, pp. Volume 5, No 1, Halaman 376-384, 2016.
- [8] Kemdikbud (Pusat Bahasa), "Alquran," 2017. [Online]. Available: <http://kbbi.web.id/Alquran>. [Accessed 24 April 2017].
- [9] A. Noertjahyana and R. Adipranata, "Implementasi Sistem Pengenalan Suara Menggunakan SAPI 5.1 dan Delphi 5," *Jurnal Informatika Universitas Petra*, pp. Volume 4, No 2, Halaman 107-114, 2003.
- [10] Y. Supardi, *Semua Bisa Menjadi Programmer Android*, Jakarta: PT Elex Media Komputindo, 2014.
- [11] M. I. Mas'ud, "Pendekatan Rantai Markov Dalam Pemilihan Universitas di Pasuruan," *Journal Knowledge Industrial Engineering (JKIE) Universitas Yudharta*, pp. Volume 4, No 1, Halaman 63-70, 2017.
- [12] E. Abdurachman, "Konsep Dasar Markov Chain Serta Kemungkinan Penerapannya di Bidang Pertanian," *Jurnal Informatika Pertanian*, pp. Volume 8, Halaman 499-505, 1999.
- [13] Sekaran, Uma & Bougie, Roger, *Research Methods for Business*, Chichester: John Wiley & Sons Ltd, 2009.