

**PENERAPAN SPEECH RECOGNITION PADA APLIKASI  
ANDROID PENENTUAN LEVEL PEMBELAJARAN YANBU'A  
JILID 1 UNTUK ANAK USIA DINI MENGGUNAKAN  
GOOGLE SPEECH API**

**SKRIPSI**



Oleh :

**Helmi Holida Putri Puspita Ningrum**

**NIM E41170891**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI JEMBER**

**2021**

**PENERAPAN SPEECH RECOGNITION PADA APLIKASI  
ANDROID PENENTUAN LEVEL PEMBELAJARAN YANBU'A  
JILID 1 UNTUK ANAK USIA DINI MENGGUNAKAN  
GOOGLE SPEECH API**

**SKRIPSI**



sebagai salah satu syarat memperoleh gelar Sarjana Terapan Komputer  
(S.Tr.Kom) di Program Studi Teknik Informatika

Jurusan Teknologi Informasi

Oleh

**Helmi Holida Putri Puspita Ningrum**

**E41170891**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI JEMBER  
2021**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET  
DAN TEKNOLOGI  
POLITEKNIK NEGERI JEMBER  
JURUSAN TEKNOLOGI INFORMASI**

**Penerapan *Speech Recognition* Pada Aplikasi Android Penentuan Level  
Pembelajaran Yanbu'a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech  
API**

**Helmi Holida Putri Puspita Ningrum (NIM E41170891)**

Ketua Penguji

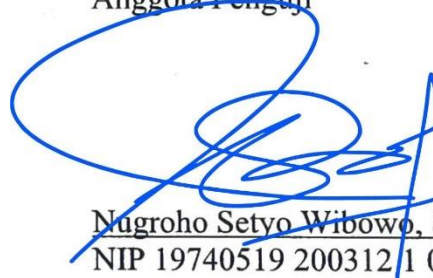
Adi Heru Utomo, S.Kom, M.Kom  
NIP. 19711115 199802 1 001

Sekretaris Penguji

Anggota Penguji



Trismayanti Dwi P, S.Kom, M.Cs  
NIP 19900227 201803 2 001



Nugroho Setyo Wibowo, ST, MT.  
NIP 19740519 2003121 002

Dosen Pembimbing



Trismayanti Dwi P, S.Kom, M.Cs  
NIP 19900227 201803 2 001

Mengesahkan,  
Ketua Jurusan Teknologi Informasi



Henra Yuli Riskiawan, S.Kom. M.Cs.  
NIP 19830203 200604 1 003

## SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

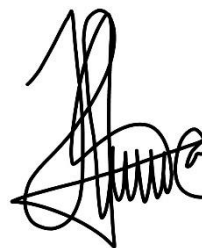
Nama : Helmi Holida Putri Puspita Ningrum

NIM : E41170891

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Laporan Skripsi saya yang berjudul “Penerapan *Speech Recognition* Pada Aplikasi Android Penentuan Level Pembelajaran Yanbu’a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech API” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam daftar pustaka di bagian akhir Laporan Skripsi ini .

Jember, 05 Juli 2021



Helmi Holida Putri Puspita Ningrum

NIM E41170891



**PERNYATAAN  
PERSETUJUAN PUBLIKASI  
KARYA ILMIAH UNTUK KEPENTINGAN  
AKADEMIS**

Yang bertandatangan di bawah ini, saya:

Nama : Helmi Holida Putri Puspita Ningrum  
NIM : E41170891  
Program Studi : Teknik Informatika  
Jurusan : Teknologi Informasi

Demi pengembangan Ilmu Pengetahuan, saya menyetujui untuk memberikan kepada UPT. Perpustakaan Politeknik Negeri Jember, Hak Bebas Royalti Non-Eksklusif (Non-Exclusive Royalty Free Right) atas Karya Ilmiah **berupa Laporan Skripsi saya yang berjudul:**

**PENERAPAN SPEECH RECOGNITION PADA APLIKASI ANDROID  
PENENTUAN LEVEL PEMBELAJARAN YANBU'A JILID 1 UNTUK  
ANAK USIA DINI MENGGUNAKAN GOOGLE SPEECH API**

Dengan Hak Bebas Royalti Non-Eksklusif ini UPT. Perpustakaan Politeknik Negeri Jember berhak menyimpan, mengalih media atau format, mengelola dalam bentuk pangkalan data (Database), mendistribusikan karya dan menampilkan atau mempublikasikannya di Internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau penciptanya.

Saya bersedia untuk menanggung secara pribadi tanpa melibatkan pihak Politeknik Negeri Jember, Segala bentuk tuntutan hukum yang timbul atas Pelanggaran Hak Cipta dalam Karya ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jember

Pada Tanggal : 05 Juli 2021

Yang menyatakan,

A handwritten signature in black ink, consisting of several loops and a long horizontal stroke at the bottom.

Nama : Helmi Holida Putri Puspita N.

NIM : E41170891

## **MOTTO**

“Kesuksesan tanpa doa, perjuangan dan pengorbanan adalah omong kosong”  
(helmihppn)

## **PERSEMBAHAN**

Bismillahirrahmanirrahim.

Dengan mengucap “Alhamdulillah robil a’lamin” sebagai ucapan rasa syukur yang sebesar-besarnya kepada Allah SWT. Skripsi ini saya persembahkan untuk:

1. Allah SWT yang telah memberikan kesehatan, rahmat, hidayah, rezeki dan semua yang saya butuhkan.
2. Teruntuk Bapak dan Ibuku tercinta, Bapak Ahmad Holip dan Ibu Rosida, terima kasih atas do’a, motivasi, semangat, cinta, kasih sayang dan pengorbanan yang telah diberikan
3. Terimakasih juga saya persembahkan kepada Dosen Pembimbing Ibu Trismayanti Dwi P.S., S.Kom, M.Cs selaku dosen pembimbing
4. Untuk Budi Yunior Valentino, terimakasih bantuannya dan kesabarannya dalam mengajari saya banyak hal tentang Android
5. Untuk mbak Nida Amalia, mbak Hasna terimakasih atas bantuannya memberikan saya kritik dan saran dalam menyelesaikan skripsi ini.
6. Untuk Merdina Marsa, terimakasih banyak atas saran dan bantuannya dalam menguji aplikasi ini.
7. Untuk TIF Golongan F dan G tahun angkatan 2017 yang menjadi teman sekaligus keluarga bagi saya, terimakasih atas doa, semangat dan dukungannya. Terimakasih juga atas kekompakan kalian selama empat tahun ini.



**Penerapan Speech Recognition Pada Aplikasi Android Penentuan  
Level Pembelajaran Yanbu'a Jilid 1 Untuk Anak Usia Dini Menggunakan  
Google Speech Api**

*Application of Speech Recognition in Android Applications Determines Level 1 of  
Yanbu'a Learning for Early Childhood Using Google Speech Api*

**Helmi Holida Putri Puspita Ningrum**  
**Study Program of Informatics Engineering**  
**Majoring of Information Technology**  
Program Studi Teknik Informatika  
Jurusan Teknologi Informasi

***ABSTRACT***

*The Yanbu'a method is one of the methods used in carrying out an activity to read the Qur'an which will later help carry out activities with good and maximum results. But the reality is that there are still many teenagers who do not understand the hijaiyah script. Therefore, Hijaya's illiteracy must be eradicated seriously. The purpose of this study is to create an android application for determining the level of learning yanbu'a volume 1 for early childhood using the Google Speech API so that it can be a forum for anyone who wants to learn and read hijaiyah letters in a fun way, especially for children. , to improve the efficiency and effectiveness of the learning system through Android smartphones. The test is carried out in 3 stages, namely Black box testing, UAT and testing by experts. Based on the results of the first test, the overall result was 51.42%. The total result for the second test was 71.42%. On the third try, the result is 100% which means all the letters have the correct answer based on the examiner's vote. Black box testing got a value of 100% while the UAT test got 70.75% results from the analysis results which amounted to*

*Keyword : speech recognition, google speech API, hidden markov model*

## RINGKASAN

**Penerapan Speech Recognition Pada Aplikasi Android Penentuan Level Pembelajaran Yanbu'a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech Api**, Helmi Holida Putri Puspita Ningrum, NIM E41170891, Tahun 2021, 129 hlm., Teknologi Informasi, Politeknik Negeri Jember, Trismayanti Dwi Puspitasari, S.Kom. M.Cs. (Dosen Pembimbing).

Metode Yanbu'a merupakan salah satu metode yang digunakan dalam melaksanakan suatu kegiatan membaca Al-Qur'an yang nantinya akan membantu melaksanakan kegiatan dengan hasil yang baik dan maksimal. Namun kenyataannya masih banyak remaja yang belum memahami aksara hijaiyah. Oleh karena itu, buta huruf Hijaya harus diberantas secara serius.

Tujuan dari penelitian ini adalah membuat aplikasi android untuk menentukan tingkat belajar yanbu'a jilid 1 untuk anak usia dini dengan menggunakan Google Speech API sehingga dapat menjadi wadah bagi siapa saja yang ingin belajar dan membaca huruf hijaiyah secara menyenangkan cara, terutama untuk anak-anak. , untuk meningkatkan efisiensi dan efektivitas sistem pembelajaran melalui smartphone Android.

Pengujian dilakukan dalam 3 tahap yaitu pengujian Black box, UAT dan pengujian oleh ahli. Berdasarkan hasil pengujian pertama, hasil keseluruhan adalah 51,42%. Hasil total untuk tes kedua adalah 71,42%. Pada percobaan ketiga, hasilnya 100% yang berarti semua huruf memiliki jawaban yang benar berdasarkan suara penguji. Pengujian black box mendapat nilai 100% sedangkan pengujian UAT mendapatkan hasil 70,75% dari hasil analisis yang sebesar 8.

## PRAKATA


Puji syukur kehadiran Allah Subhanahu wa ta'ala atas segala rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan Speech Recognition Pada Aplikasi Android Penentuan Level Pembelajaran Yanbu’a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech Api”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Diploma 4 (D-4) pada Jurusan Teknologi Informasi Program Studi Teknik Informatika di Politeknik Negeri Jember.

Penyusunan skripsi ini tidak terlepas dari berbagai pihak. Oleh karena itu penulis menyampaikan terima kasih kepada :

1. Ibu Trismayanti Dwi Puspitasari, S.Kom. M.Cs selaku Dosen Pembimbing.
2. Seluruh bapak dan ibu dosen beserta staff karyawan di program studi Teknik Informatika Politeknik Negeri Jember
3. Orang tua dan sahabat yang telah memberi semangat dan dukungannya
4. Semua pihak, teman-teman yang telah membantu dalam menyelesaikan penelitian ini.

Dengan harapan bahwa penelitian ini nantinya dapat berlanjut dan terus berkembang, penulis juga menerima segala kritik dan saran dari semua pihak demia kesempurnaan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember, 05 Juli 2021



Helmi Holida Putri Puspita Ningrum

## DAFTAR ISI

Halaman

<b>HALAMAN JUDUL</b> .....	<b>ii</b>
<b>HALAMAN PENGESAHAN</b> .....	<b>iii</b>
<b>SURAT PERNYATAAN</b> .....	<b>iv</b>
<b>PERNYATAAN</b> .....	<b>v</b>
<b>MOTTO</b> .....	<b>vii</b>
<b>PERSEMBAHAN</b> .....	<b>viii</b>
<b>ABSTRACT</b> .....	<b>ix</b>
<b>RINGKASAN</b> .....	<b>xi</b>
<b>PRAKATA</b> .....	<b>xiii</b>
<b>DAFTAR ISI</b> .....	<b>xii</b>
<b>DAFTAR GAMBAR</b> .....	<b>xvi</b>
<b>DAFTAR TABEL</b> .....	<b>xviii</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xix</b>
<b>BAB 1. PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan Penelitian .....	2
1.4 Manfaat Penelitian .....	3
<b>BAB 2. TINJAUAN PUSTAKA</b> .....	<b>4</b>
2.1 Penelitian Terdahulu ( <i>State of The Art</i> ) .....	4
2.2 Belajar Al-Qur'an.....	7
2.3 Metode Yanbu'a.....	7
2.4 Android .....	8
2.5 Android SDK .....	9
2.6 Android Studio .....	10
2.7 Bahasa Pemrograman Java.....	10
2.8 Bahasa Pemrograman Python .....	11
2.9 Jupyter Notebook .....	11

2.10	Matlab (Matrix Laboratory) .....	12
2.11	Google Speech API .....	13
2.12	Speech Recognition .....	14
2.7.1	Sistem Speech Recognition .....	15
2.7.2	Skema Utama Speech Recognition .....	17
2.13	Hidden Markov Model .....	23
2.14	Flowchart .....	24
<b>BAB 3. METODOLOGI PENELITIAN .....</b>		<b>27</b>
3.1	Tempat dan Waktu Penelitian .....	27
3.1.1	Tempat Penelitian .....	27
3.1.2	Waktu Penelitian .....	27
3.2	Alat dan Bahan .....	27
3.2.1	Alat .....	27
3.2.2	Bahan .....	28
3.3	Tahapan Penelitian .....	28
3.3.1	Studi Pendahuluan .....	30
3.3.2	Identifikasi Masalah .....	30
3.3.3	Perumusan Masalah .....	30
3.3.4	Tujuan Penelitian .....	31
3.3.5	Pengumpulan Data .....	31
3.3.6	Metode Pengembangan Sistem .....	31
3.3.7	Hasil dan Pembahasan .....	38
3.3.8	Kesimpulan dan Saran .....	39
<b>BAB 4. HASIL DAN PEMBAHASAN .....</b>		<b>40</b>
4.1	Studi Pendahuluan .....	40
4.2	Identifikasi Masalah .....	40
4.3	Perumusan Masalah .....	40
4.4	Tujuan Penelitian .....	41
4.5	Pengumpulan Data .....	41
4.6	Metode Pengembangan Sistem .....	44
4.6.1	Analisa Kebutuhan (Requirement) .....	44

4.6.2 Desain (Design) .....	45
4.6.3 Implementasi (Implementation).....	46
4.6.4 Pengujian (Testing).....	70
4.6.5 Pemeliharaan (Maintenance) .....	81
4.7 Hasil dan Pembahasan.....	83
4.8 Kesimpulan dan Saran.....	84
<b>BAB 5. KESIMPULAN DAN SARAN .....</b>	<b>86</b>
5.1 Kesimpulan .....	86
5.2 Saran.....	86
<b>DAFTAR PUSTAKA .....</b>	<b>87</b>
<b>LAMPIRAN.....</b>	<b>89</b>

## DAFTAR GAMBAR

	Halaman
Gambar 2. 1 <i>Speech Recognition</i> pada server google (B.R.Reddy & E.Mahender,2013).....	13
Gambar 2. 2 Model arsitektur suara ke <i>database server speech recognition</i> google (B.R.Reddy & E.Mahender, 2013).....	14
Gambar 2. 3 Proses MFCC .....	18
Gambar 2. 4 Skema <i>speech recognition</i> (Chelba, D., M., P.Nguyen, & S., 2012)23	23
Gambar 2. 5 Parameter <i>probabilistic</i> pada Hidden Markov Model.....	24
Gambar 3. 1 Tahapan penelitian .....	29
Gambar 3. 2 Metode <i>waterfall</i> .....	31
Gambar 3. 3 Flowchart <i>speech to text</i> .....	33
Gambar 3. 4 <i>Interface</i> halaman utama .....	34
Gambar 3. 5 <i>Interface</i> tentang yanbu'a.....	34
Gambar 3. 6 <i>Interface</i> keutamaan belajar Al-Qur'an.....	35
Gambar 3. 7 <i>Interface</i> latihan.....	35
Gambar 3. 8 <i>Interface</i> soal .....	36
Gambar 3. 9 <i>Interface</i> notifikasi jawaban benar .....	36
Gambar 3. 10 <i>Interface</i> notifikasi jawaban salah.....	37
Gambar 4. 1 Data suara.....	42
Gambar 4. 2 Tampilan splash screen .....	66
Gambar 4. 3 Tampilan halaman utama .....	66
Gambar 4. 4 Tampilan menu tentang yanbu'a .....	67
Gambar 4. 5 Tampilan menu belajar Al-Qur'an .....	67
Gambar 4. 6 Tampilan menu latihan.....	68
Gambar 4. 7 Tampilan menu soal .....	69
Gambar 4. 8 Tampilan <i>google speech api</i> .....	69
Gambar 4. 9 Tampilan jawaban salah .....	70
Gambar 4. 10 Tampilan jawaban benar .....	70
Gambar 4. 11 Source code mengganti tipe file .....	46
Gambar 4. 12 File data suara format mp3.....	46
Gambar 4. 13 File data suara format wav .....	47
Gambar 4. 14 Library MFCC.....	48
Gambar 4. 15 Visualisasi file audio .....	49



Gambar 4. 16 Hasil MFCC ..... 49

## DAFTAR TABEL

	Halaman
Tabel 2. 1 <i>State of the art</i> .....	5
Tabel 2. 2 Simbol <i>flowchart</i> .....	25
Tabel 3. 1 Jadwal Penelitian.....	<b>Error! Bookmark not defined.</b>
Tabel 4. 1 Huruf Hijaiyah .....	42
Tabel 4. 2 Flowchart Speech to Text .....	45
Tabel 4. 3 Akuisisi suara .....	47
Tabel 4. 4 Pengujian Black Box.....	71
Tabel 4. 5 Tabel Komponen Pilihan Jawaban Responden .....	74
Tabel 4. 6 Tabel Komponen Bobot Nilai Responden .....	74
Tabel 4. 7 Tabel Pertanyaan Kuisisioner .....	75
Tabel 4. 8 Tabel Data Kuesioner Santri Setelah Diolah .....	76
Tabel 4. 9 Pengujian akurasi implementasi Google Speech .....	79
Tabel 4. 10 Perbaikan pada Pengujian Akurasi tahap 1 .....	81
Tabel 4. 11 Perbaikan pada Pengujian Akurasi tahap 2 .....	82
Tabel 4. 12 Perbaikan pada Pengujian Akurasi tahap 3 .....	82

## DAFTAR LAMPIRAN

Lampiran 1. Form Pengujian Akurasi Implementasi Google Speech.....	89
Lampiran 2. Data sinyal suara.....	91
Lampiran 3. Kode program android.....	99

# **BAB 1. PENDAHULUAN**

## **1.1 Latar Belakang**

Metode Yanbu'a adalah salah satu cara yang digunakan dalam melaksanakan suatu kegiatan membaca Al-Qur'an yang nantinya akan membantu terlaksananya kegiatan dengan hasil yang baik dan maksimal. Tetapi kenyataannya saat ini masih banyak remaja yang belum mengerti aksara hijaiyah. Oleh karena itu buta huruf Hijayah harus diberantas secara serius.

Dewi Mulyani (2018) di dalam jurnalnya yang berjudul "Al-Quran Literacy for Early Childhood with Storytelling Techniques" menjelaskan bahwa, sebagaimana dilansir dalam Pikiran Rakyat 2017, dari sekitar 225 juta muslim, sebanyak 54% diantaranya termasuk kategori buta huruf Al-Quran, jadi baru 46% muslim yang mampu membaca Al-Quran. Jika dimasukkan pada indikator memahami Al-Quran tentu akan lebih kecil lagi. Hal ini sungguh memprihatinkan. Padahal, umat Islam masih mayoritas di Indonesia. Diungkap Republika (25 Juni 2018), meski merupakan negara mayoritas Muslim terbesar di dunia, namun hanya sekitar 0.5 persen umat Islam di Indonesia yang mampu membaca Al-Quran dengan baik. Berdasarkan riset IIQ (Institut Ilmu Al-Quran), tingkat buta huruf Al-Quran di Indonesia masih terbilang cukup tinggi, tercatat 65% masyarakat Indonesia buta huruf Al-Quran sebagaimana dilansir Republika (17 Januari 2018).

Pengenalan ucapan atau Google Speech API adalah platform yang dikembangkan oleh Google untuk mengubah pengenalan suara menjadi string (teks). Prinsip kerja sistem aplikasi ini adalah pengguna menginput suara, kemudian mengenali suara tersebut. Proses selanjutnya dikategorikan oleh sistem yang dibangun ke dalam aplikasi Android dan digunakan sebagai kriteria untuk menilai apakah layak memasuki level berikutnya atau tetap bertahan.

Pada penelitian yang dilakukan oleh Ach Fauzan yang berjudul Pembangunan Aplikasi Iqro' Berbasis Android Menggunakan Google Speech, peneliti memanfaatkan Google Speech API sebagai alat input suara. Dari pengujian yang dilakukan, aplikasi ini dapat digunakan sebagai sarana pembelajaran iqro sehingga

proses pembelajaran menjadi lebih menarik dan interaktif. Selain itu, dalam penelitian yang dilakukan oleh Dhimas Sena Rahmantara yang berjudul Aplikasi Pengenalan Nama Surah Pada Juz ke 30 Kitab Suci Al-Qur'an Menggunakan Speech Recognition, peneliti menerapkan algoritma Hidden Markov Model untuk memproses input pengguna dan menghitung kemungkinan nama surah. Aplikasi tersebut dapat menjadi media pembelajaran untuk mengingat dan mengetahui nama surah dan ayat-ayat pada juz ke-30.

Aplikasi ini diharapkan dapat menjadi wadah bagi siapa saja yang ingin belajar dan membaca huruf hijaiyah dengan cara yang menyenangkan, khususnya bagi anak-anak, untuk meningkatkan efisiensi dan efektifitas sistem pembelajaran melalui smartphone Android. Orang tua juga dapat menggunakan aplikasi ini untuk belajar bersama anaknya di rumah, sehingga orang tua dapat langsung mengetahui perkembangan membaca anaknya

## **1.2 Rumusan Masalah**

Rumusan masalah dalam penelitian ini, yaitu :

1. Bagaimana merancang sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API?
2. Bagaimana mengembangkan sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API?
3. Bagaimana tahapan-tahapan pengujian yang dilakukan agar dapat menemukan tingkat keakuratan antara suara pembicara dengan jawaban Google Speech API dalam aplikasi ini?

## **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini, yaitu

1. Merancang sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API

2. Mengembangkan sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API
3. Menjelaskan tahapan-tahapan pengujian yang dilakukan agar dapat menemukan tingkat keakuratan antara suara pembicara dengan jawaban Google Speech API dalam aplikasi ini

#### **1.4 Manfaat Penelitian**

Aplikasi ini diharapkan dapat menjadi wadah bagi siapa saja yang ingin belajar dan membaca huruf hijaiyah dengan cara yang menyenangkan, khususnya bagi anak-anak, untuk meningkatkan efisiensi dan efektifitas sistem pembelajaran melalui smartphone Android. Orang tua juga dapat menggunakan aplikasi ini untuk belajar bersama anaknya di rumah, sehingga orang tua dapat langsung mengetahui perkembangan membaca anaknya

## BAB 2. TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu (*State of The Art*)

*State of The Art* dimaksudkan untuk menganalisis penelitian sebelumnya yang pernah ada, yang sejalan dan mempunyai konsep yang hampir sama dengan penelitian saat ini. Kemudian melihat sejauh mana perbedaan masing-masing penelitian, sehingga masing-masing penelitian mempunyai tema yang original. Berikut ini merupakan penjabaran dari penelitian sebelumnya :

#### 1. PEMBANGUNAN APLIKASI IQRO' BERBASIS ANDROID MENGGUNAKAN GOOGLE SPEECH (Ach Fauzan, 2018)

Merupakan penelitian yang dilakukan untuk membangun aplikasi media pembelajaran iqro berbasis android dan mengintegrasikannya dengan google *speech* agar aplikasi yang dibangun lebih menarik dan interaktif dengan penggunaannya. Google *speech* diintegrasikan pada fitur soal, pengguna diminta untuk menjawab setiap soal menggunakan suara kemudian google *speech* berfungsi mengkonversi suara menjadi teks. Hasil teks yang diperoleh dicocokkan dengan kunci jawaban, dengan melibatkan pengguna dalam fitur soal menjadi aplikasi ini lebih interaktif. Pada total hasil pengujian ke 1 diperoleh total hasil 85% hal ini terjadi karena ada beberapa huruf yang memiliki kunci jawaban berbeda dengan apa yang diucapkan penguji. Setelah itu penulis melakukan perbaikan pada kunci jawaban. Pada pengujian ke 2 diperoleh total hasil 93%. Hasil ini memiliki kenaikan dibandingkan dengan pengujian pertama, namun untuk hasil yang lebih maksimal maka penulis kembali melakukan perbaikan pada kunci jawaban. Pada percobaan ke 3 diperoleh total hasil 100% artinya semua huruf sudah memiliki kunci jawaban yang benar berdasarkan suara yang diucapkan oleh penguji.

#### 2. APLIKASI PENGENALAN NAMA SURAH PADA JUZ KE 30 KITAB SUCI AL-QUR'AN MENGGUNAKAN SPEECH RECOGNITION (Dhimas Sena Rahmantara, 2018)

Merupakan penelitian yang dilakukan untuk merancang aplikasi pengenalan nama surah Al-Quran pada juz ke-30 berbasis *android* dengan memanfaatkan

teknologi *speech recognition*. Algoritma yang ditetapkan 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. Jawaban kuesioner menyatakan bahwa 87,1% responden mendengarkan dengan baik, 80,64% menjawab aplikasi menerima atau mengenali inputan suara *user* dengan akurat.

### 3. IMPLEMENTASI GOOGLE SPEECH API PADA APLIKASI KOREKSI HAFALAN AL-QURAN BERBASIS ANDROID (Affandy Akbar, 2019)

Merupakan penelitian yang dirancang untuk membangun sebuah aplikasi koreksi hafalan Al-Qur'an berbasis *android* dengan mengimplementasikan Google Speech API sebagai media *input* suara. Sehingga aplikasi ini nantinya dapat digunakan oleh pada hafidz Al-Qur'an sebagai media untuk melakukan muroja'ah kapan saja dan dimana saja. Dari hasil pengujian *black box* menunjukkan bahwa seluruh fungsi telah berjalan sesuai dengan aplikasi yang diharapkan. Google Speech API tidak bisa mendeteksi dengan maksimal lafadz ayat yang cenderung lebih panjang dan melafadzkan dengan suara yang kurang jelas.

Tabel 2. 1 *State of the art*

No	Peneliti	Ach (2018)	Fauzan Iqro' (2018)	Dhimas Rahmantara (2018)	Sena Nama (2019)	Affandy Akbar (2019)
1	Judul	Pembangunan Aplikasi Berbasis Android Menggunakan Google Speech	Aplikasi Iqro' Pengenalan Surah Pada 30 Kitab Suci Qur'an Menggunakan	Aplikasi Pengenalan Nama Surah Pada Juz ke 30 Kitab Suci Al-Qur'an Menggunakan	Implementasi Nama Google Speech API Pada Aplikasi Koreksi Hafalan Al-Quran Berbasis Android	Implementasi Google Speech API Pada Aplikasi Koreksi Hafalan Al-Quran Berbasis Android
2	Tema	Aplikasi	Aplikasi	Aplikasi	Aplikasi	Aplikasi
3	Platform	Android	Android	Android	Android	Android



---

4	Metode Penelitian	Google Speech API	Hidden Markov Model	Google Speech API
5	Hasil Penelitian	Hasil pengujian unit pada seluruh algoritma yang ada dalam sub bab pengujian menyatakan bahwa seluruh kasus uji memiliki hasil yang telah sesuai dengan yang diharapkan atau bernilai valid.	Berdasarkan hasil <i>blackbox testing</i> , keseluruhan fungsionalitas aplikasi telah berfungsi dengan baik. Berdasarkan hasil <i>control flow testing</i> , nilai <i>cyclomatic complexity</i> adalah 7 yang menunjukkan kode program tersusun sederhana dan ditulis dengan baik. Hasil pengujian kuesioner diperoleh 87,74% responden menyatakan aplikasi ini membantu <i>user</i> lebih mengetahui nama-nama surah pada juz ke-30	Dari hasil pengujian <i>blackbox</i> menunjukkan bahwa seluruh fungsi telah berjalan sesuai dengan aplikasi yang diharapkan. Google Speech API tidak bisa mendeteksi dengan maksimal lafadz ayat yang cenderung lebih Panjang dan melafadzkan dengan suara yang kurang jelas

---

## 2.2 Belajar Al-Qur'an

Al-Qur'an merupakan kitab suci umat Muslim yang diturunkan sebagai wahyu kepada Nabi Muhammad Saw. Al-Qur'an merupakan kitab penyempurna kitab lainnya, di antara kitab-kitab Allah Swt hanya Al-Qur'an yang wajib diyakini kebenarannya. Secara harfiah Al-Qur'an memiliki arti sebagai bacaan yang sempurna. Menurut Quraish Shihab, kata "Al-Qur'an" merupakan nama yang sangat tepat yang dipilih Allah Swt. karena tidak ada suatu bacaan pun sejak manusia mengenal tulis baca ribuan tahun yang lalu dapat menandingi Al-Qur'an. (Shihab, 1998, h.3)

Keutamaan membaca Al-Qur'an terdapat dalam Al-Qur'an itu sendiri yakni Al-Qur'an adalah kitab yang terpelihara dan terjaga sampai hari kiamat. Selain dalam surat Al-Hijr ayat 9, terdapat pula keterangan bahasa Al-Qur'an adalah kitab mulia, tidak ada kebathilan dan penyimpangan padanya. Qur'an Surat Fussilat ayat 41 dan 42 yang artinya: "...Dan sesungguhnya (Al-Qur'an) itu adalah kitab yang mulia, yang tidak akan didatangi oleh kebathilan baik dari depan maupun dari belakang (dari masa lalu dan masa yang akan datang)".

## 2.3 Metode Yanbu'a

Metode berarti suatu jalan yang dilalui untuk mencapai layanan, dalam bahasa Arab metode disebut "thoriqot". Thoriqot baca tulis Al-Qur'an Yanbu'a adalah suatu kitab thoriqoh (metode) untuk mempelajari baca tulis serta menghafal Al-Qur'an dengan cepat, mudah dan benar bagi anak maupun orang dewasa, yang dirancang dengan rasm Utsmani dan menggunakan tanda baca dan waqof yang ada di dalam Al-Qur'an. Al-Qur'an rasm Utsmani banyak dipelajari di negara-negara Arab dan negara Islam. Rasm Utsmani itu sendiri adalah tata cara atau kaidah penulisan huruf-huruf dan kata-kata Al-Qur'an yang disetujui pada masa Khalifah Utsman dan dipedomani oleh tim penyalin Al-Qur'an yang dibentuknya dan terdiri dari Zaid bin Tsabit, Abdullah Ibn Al-Zubair Ibn Hasyim. Metode yanbu'a adalah suatu metode pembelajaran membaca, menulis dan menghafal Al-Qur'an yang disusun

secara sistematis terdiri 7 jilid, cepat, tepat, benar dan tidak putus-putus sesuai dengan makhrojul huruf fan ilmu tajwid.

Berdasarkan berbagai pendapat tersebut disimpulkan bahwa metode Yanbu'a adalah salah satu cara yang digunakan dalam melaksanakan suatu kegiatan membaca Al-Qur'an yang nantinya akan membantu terlaksananya kegiatan dengan hasil yang baik dan maksimal. Dalam dunia Pendidikan, metode mempunyai peranan yang sangat penting terutama dalam kegiatan pembelajaran sehingga tercipta suasana yang kondusif baik di dalam maupun di luar kelas. Penerapan metode Yanbu'a dalam pembelajaran dalam pembelajaran membaca Al-Qur'an diharapkan dapat membantu anak didik dalam mempelajari bacaan Al-Qur'an, memberikan pencerahan di saat mengalami kejenuhan untuk membangkitkan motivasi membaca Al-Qur'an pada anak didik sehingga dapat membaca dengan lancar, benar dan fasih dalam suasana yang menyenangkan karena materinya/isinya diambil dari kumpulan ayat-ayat Al-Qur'an yang suci. (Suriah, n.d.)

## **2.4 Android**

Android adalah sistem operasi mobile berdasarkan modifikasi versi Linux. Android awalnya dikembangkan oleh sebuah perusahaan pengembang yang namanya sama, Android, Inc pada tahun 2005, sebagai bagian strateginya dalam memasuki pasar mobile, Google membeli Android dan mengambil alih pengembangnya sampai sekarang.

Google ingin Android terbuka dan bebas, oleh karena itu sebagian besar kode Android dilepas di bawah lisensi *open source* Apache, yang berarti bahwa siapapun yang ingin menggunakan Android dapat mendownload sumber kode Android secara penuh. Terlebih lagi bagi vendor (biasanya pabrikan hardware) dapat menambahkan ekstensi propietari pada Android dan menyesuaikan Android untuk membedakan produk Android mereka dengan lainnya. Model pengembangan yang sederhana ini membuat Android sangat menarik dan telah mengusik ketertarikan banyak vendor. Terutama bagi perusahaan yang terpengaruh dengan fenomena

iPhone Apple, sebuah kesuksesan yang luar biasa yang merevolusi industry smartphone. Perusahaan termasuk Motorola dan Sony yang selama bertahun-tahun mengembangkan sistem operasi mobile sendiri. Ketika iPhone diluncurkan, banyak perusahaan seperti ini berjuang untuk menemukan cara baru untuk merevitalisasi produk mereka. Pabrikan ini melihat Android sebagai sebuah solusi, mereka meneruskan untuk mendesain hardware mereka dan menggunakan Android sebagai Sistem Operasi yang memberikan kekuasaannya.

Keuntungan utama mengadopsi Android adalah Android menawarkan pendekatan terpadu pada pengembangan aplikasi. Pengembang hanya perlu mengembangkan untuk Android, dan aplikasi mereka dapat dijalankan pada banyak perangkat yang berbeda, sepanjang perangkat tersebut menggunakan Android. Pada dunia smartphone, aplikasi adalah bagian rantai sukses yang paling penting. Oleh karena itu pabrikan hardware melihat Android sebagai harapan terbaiknya untuk menantang serangan hebat iPhone, yang sudah memiliki dasar aplikasi yang besar.

Android telah mengalami sejumlah update sejak pertama kali diluncurkan pada tahun 2007 sampai yang diluncurkan terakhir tahun 2015. Kode nama versi Android menggunakan nama-nama kue agar mudah diingat. Android bersifat open source dan bebas tersedia bagi pabrikan untuk penyesuaian, sehingga tidak ada konfigurasi hardware dan software yang tetap. (Shinwani, 2016)

Versi android yang digunakan dalam penelitian ini yaitu Android 6.0 (Marshmallow). Android Marshmallow adalah generasi ke enam dari sistem operasi mobile milik Google, Android. Dirilis pada Oktober 2015, Marshmallow rilis dengan berbagai kelebihan yang menonjol di soal pengalaman pengguna dalam menggunakan Android.

## **2.5 Android SDK**

Android SDK (Software Development Kit) adalah tools Application Programming Interface (API) yang diperlukan untuk mengembangkan aplikasi pada platform Android. Android merupakan subset perangkat lunak ponsel yang meliputi sistem operasi, middleware, dan aplikasi kunci yang release oleh Google. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai

mengembangkan aplikasi pada platform Android dengan menggunakan bahasa pemrograman Java (Safaat, 2012).

## 2.6 Android Studio

Android Studio adalah sebuah IDE untuk Android Development yang diperkenalkan google pada acara Google I/O 2013. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android Studio merupakan IDE resmi untuk pengembangan aplikasi Android.

Sebagai pengembangan dari Eclipse, Android Studio mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai build environment. Fitur-fitur lainnya adalah sebagai berikut :

- a. Menggunakan Gradle-based build system yang fleksibel.
- b. Bisa mem-build multiple APK.
- c. Template support untuk Google Services dan berbagai macam tipe perangkat
- d. Layout editor yang lebih bagus.
- e. Built-in support untuk Google Cloud Platform, sehingga mudah untuk integrasi dengan Google Cloud Messaging dan App Engine.
- f. Import library langsung dari Maven repository (Shinwani, 2016).

## 2.7 Bahasa Pemrograman Java

Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh James Gosling di Sun Microsystem yang sekarang menjadi bagian dari Oracle Corporation. Sebagai bahasa pemrograman yang dinamis, Java dapat dijalankan di berbagai jenis *platform*. Untuk menjalankan aplikasi yang telah dibuat dengan bahasa pemrograman Java, setiap computer memerlukan JVM untuk menjalankannya Java dibagi kedalam tiga bentuk, yaitu (Syaputra,2017):

1. Java *Standard Edition* (Java SE) merupakan teknologi yang digunakan untuk mengembangkan *client-side stand alone application* seperti aplikasi *desktop*.
2. Java *Enterprise Edition* (Java EE) merupakan teknologi Java yang diterapkan untuk computer *server* seperti Java servlet, dan java *Server Page*.

3. Java *Micro Edition* (Java ME) merupakan teknologi yang digunakan untuk menerapkan Java pada *mobile phone*, *smarthphone* ataupun PDA.

## 2.8 Bahasa Pemrograman Python

Awal perkembangan Python dilakukan oleh Guido van Rossum pada tahun 1990 di Stichting Mathematisch Centrum (CWI), Amsterdam. Pada tahun 1995, Guido pindah ke CNRI di Virginia Amerika. Versi terakhir pada tahun 2000 dengan versi 1.6. Pada tahun 2000, Guido dan para pengembang inti Python pindah ke BeOpen.com yang merupakan sebuah perusahaan komersial dan membentuk BeOpen PythonLabs. Dari BeOpen PythonLabs inilah pengembangan Python 2.0. Setelah mengeluarkan Python 2.0, Guido dan beberapa anggota tim PythonLabs pindah ke DigitalCreations. Saat ini pengembangan Python terus dilakukan oleh sekumpulan pemrogram yang dikoordinir Guido dan Python Software Foundation. Python Software Foundation adalah sebuah organisasi non-profit yang dibentuk sebagai pemegang hak cipta intelektual Python sejak versi 2.1 dan dengan demikian mencegah Python dimiliki oleh perusahaan komersial. Saat ini distribusi Python sudah mencapai versi 2.7.14 dan versi 3.6.3 Penggunaan nama Python dipilih oleh Guido sebagai nama bahasa ciptaannya karena kecintaan Guido pada acara televisi Monty Python's Flying Circus. Oleh karena itu seringkali ungkapan-ungkapan khas dari acara tersebut seringkali muncul dalam korespondensi antar pengguna Python.

## 2.9 Jupyter Notebook

Jupyter Notebook diinstal dari sebuah software stack yang bernama Anaconda Data Science Platform. Anaconda Data Platform dapat diunduh melalui laman resminya yaitu <https://www.anaconda.com/>. Kelebihan dari Anaconda Data Platform ini yaitu dapat dengan mudah melakukan konfigurasi semua library yang berhubungan dengan modeling yang sedang dikerjakan, semuanya ada di satu tempat. Kemudian, kemudahan lainnya yaitu dapat environment sesuai kebutuhan, misalnya ketika masih mempunyai project yang menggunakan Python 2.x dapat menggunakan library khusus Python 2.x tanpa mengganggu project yang menggunakan versi Python 3.x. Kelebihan lain, di Anaconda juga menyediakan

akses ke tools data science, dan pengembangan software lainnya seperti Spyder, Microsoft Visual Studio Code, dan juga R Studio bagi yang juga terbiasa dengan bahasa pemrograman R.

### **2.10 Matlab (Matrix Laboratory)**

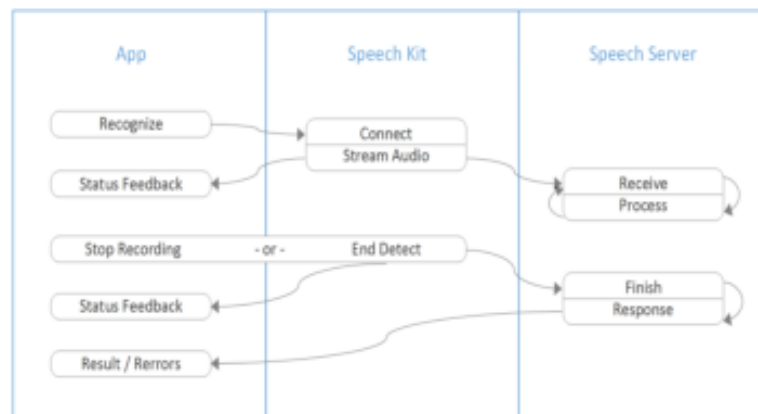
Matlab merupakan sebuah singkatan dari Matrix Laboratory, yang pertama kali dikenalkan oleh University of New Mexico dan University of Stanford pada tahun 1970. software ini pertama kali memang digunakan untuk keperluan analisis numerik, aljabar linier dan teori tentang matriks. Saat ini, kemampuan dan fitur yang dimiliki oleh Matlab sudah jauh lebih lengkap dengan ditambahkannya toolbox toolbox yang sangat luar biasa. Beberapa manfaat yang didapatkan dari Matlab antara lain:

- a) Perhitungan Matematika
- b) Komputasi numerik
- c) Simulasi dan pemodelan
- d) Visualisasi dan analisis data
- e) Pembuatan grafik untuk keperluan sains dan teknik
- f) Pengembangan aplikasi, misalnya dengan memanfaatkan GUI.

Matlab dapat dipandang sebagai sebuah kalkulator dengan fitur yang lengkap. Kita pernah menggunakan kalkulator dengan dengan fasilitas minimal, misalnya hanya terdapat fasilitas penambahan, pengurangan perkalian dan pembagian. Kalkulator yang lebih lengkap lagi adalah kalkulator scientific dimana fasilitas yang diberikan tidak hanya yang disebutkan di atas, melainkan sudah ada fungsi-fungsi trigonometri, bilangan kompleks, akar kuadrat dan logaritma. Nah, Matlab mirip dengan kalkulator tersebut, tetapi dengan fitur-fitur yang lengkap diantaranya dapat digunakan untuk memprogram, aplikasi berbasis GUI dan lengkap dengan toolbox yang dapat dimanfaatkan untuk memecahkan masalah sains dan teknik.

## 2.11 Google Speech API

Penggunaan Google Speech Recognition API memungkinkan developer untuk mengkonversi ucapan ke dalam teks. Layanan ini dapat di proses secara offline dan online. Namun pada pemrosesan secara offline, bahasa yang didukung terbatas pada bahasa yang terdapat pada masing-masing device/smartphone, selain itu dibebberapa versi Android fitur ini tidak dapat dioperasikan. Berbeda dengan perosesan secara online yang memungkinkan dukungan bahasa yang lengkap dan dapat dioperasikan hampir di semua versi Android. Namun dalam penggunaannya layanan ini memerlukan koneksi internet karena Speech Recognition terjadi di sever Google. Gambar 2.1 menunjukan alur *Speech Recognition* di *server* google.



Gambar 2. 1 *Speech Recognition* pada server google (B.R.Reddy & E.Mahender,2013)

Pada level application ada komponen utama yang tersedia untuk developer yaitu recognizer. Pada level speech kit ada beberapa proses yang terkoordinasi, yaitu :

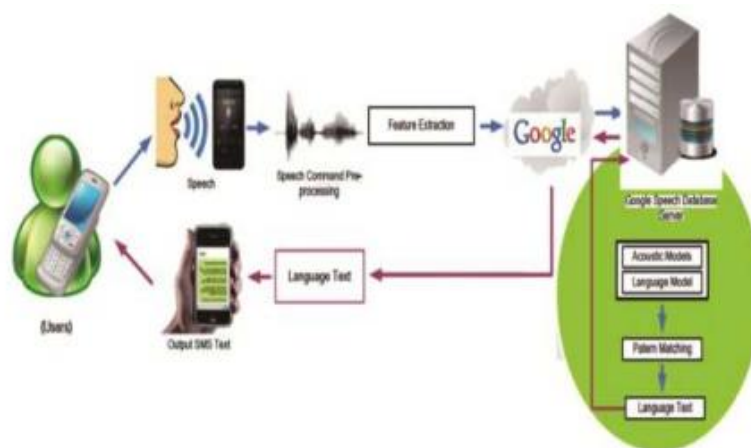
1. Library sepenuhnya mengelola sistem audio untuk recording
2. Komponen networking mengelola koneksi ke server
3. End detection menentukan bila pengguna telah berhenti berbicara dan secara otomatis menghentikan rekaman.

Google Speech API diluncurkan pada tahun 2008 di Amerika Serikat untuk beberapa tipe smartphone. Google Speech API adalah sebuah framework yang



dikembangkan oleh Google untuk mengenali suara, mengubahnya menjadi string (teks) dan memasukkannya ke dalam pencarian Google sehingga akan tampil hasil pencarian berdasarkan input suara. Pengenalan suara dilakukan pada server Google menggunakan algoritma Hidden Markov Model (HMM).

Dengan kata lain input suara yang diterima oleh perangkat Android (smartphone) akan dikirimkan ke server Google, yang selanjutnya server Google melakukan pengenalan dan mengubahnya menjadi teks menggunakan algoritma HMM. Hasil konversi suara menjadi teks kemudian dimasukkan dalam halaman pencarian Google kemudian server Google akan mengirimkan hasil pencariannya tersebut ke perangkat Android.



Gambar 2. 2 Model arsitektur suara ke *database server speech recognition* google (B.R.Reddy & E.Mahender, 2013)

## 2.12 Speech Recognition

*Speech Recognition* adalah proses otomatis mengenali kata-kata yang diucapkan kemudian mengubahnya ke dalam bentuk teks. Dengan kata lain *Speech Recognition* merupakan suatu proses yang memungkinkan untuk mengenali apa yang diucapkan oleh seseorang lalu mengubahnya ke bentuk tulisan. *Input* data yang diterima berupa suara atau ucapan manusia, kemudian sistem akan mengidentifikasi kata atau kalimat yang diucapkan dan menghasilkan keluaran atau *output* berupa teks yang sesuai dengan apa yang diucapkan. *Input* yang berupa kata-

kata tersebut diubah menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan angka lalu disesuaikan dengan kode-kode tertentu dan dicocokkan dengan suatu pola yang tersimpan dalam suatu perangkat. Untuk setiap ucapan yang berbeda akan dihasilkan pola ciri yang berbeda (Jurafsky, D., & Martin, J.H, 2009).

Penganalisis sintaks akan melakukan transformasi sinyal ucapan dari domain waktu ke domain frekuensi. Pada domain frekuensi, untuk kurun waktu yang singkat, setiap sinyal dapat terlihat memiliki ciri-ciri yang unik. Namun demikian, pengucapan suatu unit bunyi ucapan atau fonem seringkali bervariasi antar orang yang berbeda, juga terpengaruh oleh berbagai faktor, seperti : kondisi emosi, *noise*, dan faktor-faktor lainnya. Sistem *Speech Recognition* akan melakukan pengenalan untuk setiap unit bunyi pembentuk ucapan atau fonem, selanjutnya mencoba mencari kemungkinan kombinasi hasil ucapan yang paling dapat diterima. Bentuk sistem *Speech Recognition* yang sederhana adalah sistem yang hanya dapat mengenal sejumlah kata yang jumlahnya terbatas. Sistem ini biasanya lebih akurat dan lebih mudah dilatih, tetapi tidak dapat mengenal kata yang berada di luar kosa kata yang pernah diajarkan. Ukuran kosakata (*vocabulary*) dari sistem pengenalan suara mempengaruhi kompleksitas, parameter pelatihan dan akurasi sistem. Beberapa aplikasi pengenalan suara hanya memerlukan beberapa kata, sedangkan yang lainnya memerlukan kamus yang sangat besar (misalnya mesin pendiktean). Terdapat 4 jenis ukuran kosakata, yaitu :

- a. Kosakata ukuran kecil (*small vocabulary*) yang terdiri dari puluhan kata.
- b. Kosakata ukuran sedang (*medium vocabulary*) yang terdiri dari ratusan kata.
- c. Kosakata ukuran besar (*large vocabulary*) yang terdiri dari ribuan kata.
- d. Kosakata ukuran sangat besar (*very large vocabulary*) yang terdiri dari puluhan ribu kata.

### 2.7.1 Sistem Speech Recognition

Terdapat dua macam mode pada sistem pengenalan pembicaraan yaitu (Jurafsky, D., & Martin, J.H, 2009):

a. Mode Diktasi

Pada mode ini pengguna komputer dapat mengucapkan kata/kalimat yang hanya akan dikenali oleh mesin dan diubah menjadi data teks. Kemungkinan data yang dapat dikenali dibatasi oleh jumlah kata yang telah terdapat dalam *database*. Pengenalan mode ini bergantung pada pola suara dan aksentuasi pembicara serta yang telah dilakukan.

b. Mode Command dan Control

Pada mode ini pengguna mengucapkan kata/kalimat yang sudah terdefinisi terlebih dahulu pada database dan selanjutnya akan digunakan untuk menjalankan perintah tertentu pada aplikasi komputer. Jumlah perintah yang dapat dikenali tergantung dari aplikasi yang telah mendefinisikan terlebih dahulu pada database jenis-jenis perintah yang dapat dieksekusi. Mode ini merupakan speaker *independent* karena jumlah kata yang dikenali biasanya terbatas sekali dan ada kemungkinan pembicara tidak perlu melakukan pelatihan pada sistem sebelumnya.

Terdapat tiga proses utama pada sistem pengenalan pembicaraan. Baik pada mode diktasi ataupun *command and control* yaitu :

1. Pemisahan Kata

Pemisahan kata adalah proses untuk memisahkan suara yang diucapkan oleh pengguna menjadi beberapa bagian. Masing-masing bagian bisa berupa kalimat ataupun hanya sebuah kata.

2. Pencocokan kata

Pencocokan kata adalah proses untuk mencocokkan kata ucapan yang berhasil diidentifikasi dengan basis data yang dimiliki oleh sistem. Terdapat dua metode yang dapat dipakai pada proses pencocokan kata ini, yaitu : *whole-word matching* dan *phoneme matching*.

Pada *whole-word matching*, sistem akan mencari di basis data kata yang sama persis dengan kata hasil ucapan pengguna. Sedangkan pada *phoneme matching*, sistem mempunyai kamus fonem. Fonem ialah bagian terkecil dan unit dari suara yang membentuk sebuah kata.

3. Pembendaharaan Kata

Pembendaharaan kata ialah bagian terakhir dalam sebuah sistem pengenalan pembicaraan. Terdapat dua hal yang perlu diperhatikan pada pembendaharaan kata, yaitu ukuran dan keakuratan. Jika pembendaharaan kata, yaitu berjumlah banyak maka sebuah sistem akan mudah dalam melakukan pencocokan kata, tetapi dengan makin meningkatnya jumlah pembendaharaan kata, maka jumlah kata yang mempunyai ucapan hampir sama juga meningkat, dimana hal ini menurunkan keakuratan pengenalan.

Dan sebaiknya jika sebuah sistem mempunyai pembendaharaan kata sedikit, maka keakuratan pengenalan akan tinggi karena sedikitnya kata yang hampir sama, tetapi akan semakin banyak kata yang tidak dikenali. Untuk sistem pengenalan pembicaraan dengan metode *command and control*, akan lebih baik jika menggunakan jumlah pembendaharaan kata sedikit (kurang dari 100 kata/0, tetapi untuk mode diktasi akan membutuhkan jumlah pembendaharaan kata yang banyak.

### 2.7.2 Skema Utama Speech Recognition

Terdapat 3 langkah utama dalam sistem pengenalan ucapan, yaitu :

#### a. Proses Akuisisi Suara (*Voice Acquisition Process*).

Kata-kata yang diucapkan diubah menjadi sinyal digital dengan mengubah gelombang suara menjadi sekumpulan angka yang kemudian disesuaikan dengan kode-kode tertentu untuk mengidentifikasi. Hasil identifikasi kata yang diucapkan dapat ditampilkan dalam bentuk tertulis atau dapat dibaca oleh perangkat teknologi sebagai perintah untuk melakukan pekerjaan, misalnya menekan tombol pada ponsel genggam yang dilakukan secara otomatis dengan suara perintah. Dalam proses data suara akan dijelaskan dalam akuisisi data.

Sistem akuisisi data adalah sistem yang berfungsi untuk mengambil, mengumpulkan dan mempersiapkan data, mengolahnya untuk menghasilkan data yang diinginkan. Cara ini bertujuan untuk mempermudah setiap langkah yang diambil dalam keseluruhan proses.

Data sinyal suara diperoleh dengan merekam suara melalui mikrofon terhubung ke komputer. Rekaman suara dilakukan pada aplikasi menggunakan

frekuensi sampling standar 44100 Hz. Sinyal suara bisa menggunakan 16 bit/sample dan 1 untuk saluran mono. Durasi perekaman lebih pendek dalam waktu singkat lebih mudah ambil perbedaan di antara fitur-fiturnya.

Perhitungan untuk menentukan data sampling dapat menggunakan rumus berikut.

$$X = Fs \times dt(\text{seconds}) \times \left(\frac{\text{bit}}{8}\right) \times j \quad \text{-----2.1}$$

Keterangan :

X : data pengambilan sampel sinyal

Bit : jumlah bit

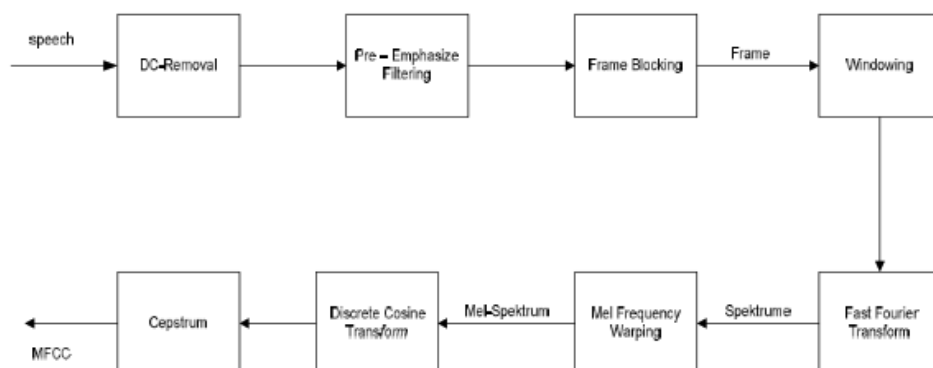
j : saluran (mono=1)

fs : frekuensi sampling

dt : durasi perekaman (detik)

#### b. Ekstraksi ciri (*feature extraction*).

Setelah mendapatkan suara digital melalui akuisisi data, proses selanjutnya adalah ekstraksi fitur. Dalam penelitian ini, Mel Frequency Cepstral Coefficient (MFCC) digunakan untuk mengekstrak karakteristik sinyal suara. Rekaman suara yang dimiliki telah masuk ke dalam sistem lalu diekstraksi menggunakan metode MFCC untuk dikenali oleh komputer. Langkah-langkah proses MFCC dijelaskan pada gambar berikut.



Gambar 2. 3 Proses MFCC

### 1. DC-Removal

DC-Removal bertujuan untuk menghitung data sampel suara rata-rata dan mengurangi nilai setiap sampel suara dengan nilai rata-rata dan mengurangi nilai setiap sampel suara dengan nilai rata-rata. Tujuannya adalah untuk mendapatkan file normalisasi input suara. Input suara proses ini adalah suara sampel berupa array dan hasilnya adalah normalisasi sampel suara array.

Rumus DC-Removal sebagai berikut.

$$y[n] = x[n] - \bar{x} ; 0 \leq n \leq N - 1 \quad \text{-----2.2}$$

Keterangan :

$y[n]$  : sampel sinyal dari DC-Removal

$x[n]$  : sampel sinyal asli

$\bar{x}$  : nilai rata-rata sampel sinyal asli

$N$  : panjang sinyal

### 2. Pre-Emphasize Filtering

Tujuan dari pemfilteran ini adalah untuk mendapatkan bentuk spektral frekuensi sinyal wicara yang lebih halus. Oleh karena itu, *pre emphasize* digunakan untuk mengoreksi sinyal dari gangguan *noise* dan meningkatkan akurasi proses ekstraksi.

Rumus Pre-Emphasize Filtering sebagai berikut.

$$y[n] = s[n] - \alpha s[n - 1] ; 0.9 \leq \alpha \leq 1.0 \quad \text{-----2.3}$$

Keterangan :

$y[n]$  : hasil sinyal *pre-emphasize*

$s[n]$  : sinyal sebelum *pre-emphasize*

Rumus sinyal *pre-emphasize* baru sebagai berikut.

$$N_n = N_n + Y_n$$

Keterangan :

$N_n$  :  $N_n$  baru

$N_n$  :  $N_n$  lama (nilai *pre-emphasize*)

$Y_n$  : hasil sinyal dari *pre-emphasize*

### 3. Frame Blocking

Frame blocking merupakan proses yang bertujuan untuk membagi sampel suara menjadi beberapa bingkai atau slot dengan panjang tertentu.

Rumus *frame blocking* sebagai berikut.

$$\text{Number of frames} = ((I - N) / M + 1) \quad \text{-----}2.4$$

Keterangan :

$I$  : laju sampel

$N$  : titik sampel

### 4. Windowing

Proses windowing digunakan untuk mengurangi diskontinuitas sinyal di awal dan akhir bingkai yang disebabkan oleh proses pemblokiran bingkai. Dalam proses windowing, sinyal akan diruncingkan menuju titik 0 (persimpangan nol) di awal dan akhir bingkai.

Proses windowing menggunakan jendela hamming dengan contoh titik 1764 sebagai berikut.

$$w(n) = 0.54 - 0.46 \cdot \cos(2 \cdot \pi \cdot j / (M - 1)) \quad \text{-----}2.5$$

Rumus windowing sebagai berikut.

$$x(n) = x_i(n) * w(n) \quad ; \quad n = 0, 1, \dots, N - 1 \quad \text{-----}2.6$$

Keterangan :

$x[n]$  : nilai sampel sinyal

$x_i[n]$  : nilai sampel bingkai sinyal 1

$N$  : ukuran bingkai, adalah kelipatan 2

$w[n]$  : fungsi jendela

### 5. FFT (Fast Fourier Transform)

FFT adalah salah satu metode untuk mengubah sinyal suara menjadi sinyal frekuensi.

Rumus FFT sebagai berikut.

$$F(K) = \sum_{n=1}^N f(n) \cos\left(\frac{2\pi nkT}{N}\right) - j \sum_{n=1}^N f(n) \sin\left(\frac{2\pi nkT}{N}\right) \quad \text{-----2.7}$$

#### 6. Mel Frequency Warping

Mel Frequency Warping umumnya menggunakan filterbank. Filterbank adalah salah satu bentuk filter bertujuan untuk mengetahui besarnya frekuensi pita tertentu pada suara sinyal.

Rumus Mel Frequency Warping sebagai berikut.

$$Y[t] = \sum_{j=1}^n S[j] H_i[j] \quad \text{-----2.8}$$

Keterangan :

N : besaran spektrum

S[j] : besaran spektrum pada frekuensi j

H<sub>i</sub>[j] : koefisien filterbank pada frekuensi j ( $1 \leq i \leq M$ )

M : jumlah saluran di filterbank

#### 7. Discrete Cosine Transform (DCT)

Proses DCT merupakan langkah terakhir dari proses ekstraksi fitur. DCT digunakan untuk mendapatkan beberapa nilai dimensi dari vektor yang sesuai.

Rumus DCT sebagai berikut.

$$C_n = \sum_{k=1}^K (\log SK) \cos \left[ n \left( K - \frac{1}{2} \right) \frac{\pi}{k} \right], \quad n = 1, 2 \dots K \quad \text{-----2.9}$$

Keterangan :

SK : output dari proses filterbank pada indeks K

K : jumlah koefisien yang diharapkan

#### 8. Cepstrum (Cepstral Liftering)

Hasil dari proses DCT adalah Cepstrum yang merupakan hasil akhir dari fitur proses ekstraksi. Cepstrum adalah istilah kebalikan dari spektrum. Itu sudah biasa memperoleh informasi dari sinyal suara ucapan manusia. Untuk dapat



meningkatkan kualitas proses pengenalan, maka hasil dari proses DCT haruslah melalui proses Cepstral Liftering.

Rumus DCT sebagai berikut.

$$w[n] = \begin{cases} 1 + \frac{L}{2} \sin\left(\frac{n\pi}{L}\right); & n = 1, 2, \dots, L \\ 0 & \end{cases} \text{-----2.10}$$

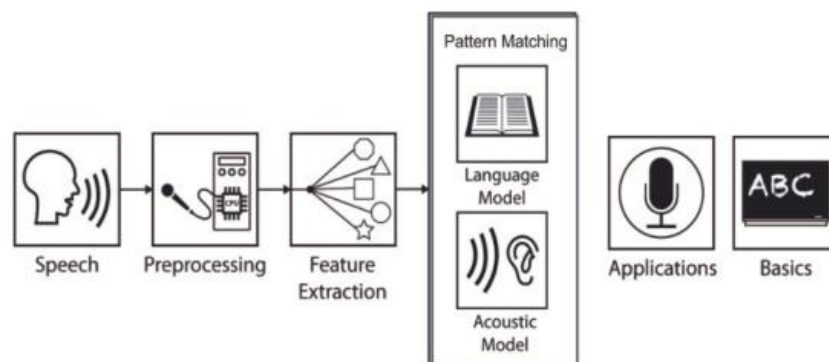
Keterangan :

- L : jumlah koefisien cepstral  
N : indeks koefisien cepstral

c. Pencocokan pola (*pattern matching*).

Merupakan proses akhir pada *Speech Recognition* yaitu tahap pencocokan data baru dengan data suara (pencocokan tata bahasa) pada *template*. Pencocokan pola merupakan pencocokan jenis suara melalui alat *speech recognizer* yang diidentifikasi berdasarkan ucapan yang dimasukkan pengguna dengan cara mengubah gelombang suara ke deretan angka-angka yang kemudian akan dicocokkan dengan data yang ada. Pencocokan pola digitalisasi suara dapat dicocokkan menggunakan sebuah algoritma yang kemudian dicocokkan dengan *template* suara yang sudah ada dalam *database server*. Hasil dari ekstraksi ciri lalu dimasukkan pada proses pengenalan pola ini. Metode yang digunakan dalam pengenalan pola adalah metode *Hidden Markov Model* (HMM). Pola yang didapat akan dicocokkan dengan berbagai macam model.

Adapun skema utama sistem pengenalan ucapan ditunjukkan pada Gambar 2.6.



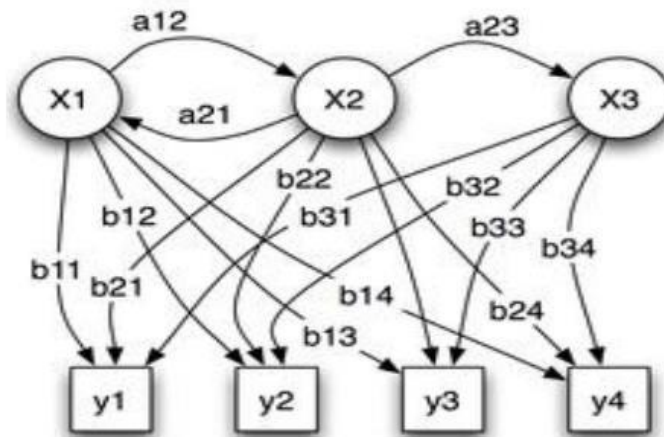
Gambar 2. 4 Skema *speech recognition* (Chelba, D., M., P.Nguyen, & S., 2012)

### 2.13 Hidden Markov Model

*Hidden Markov Model* (HMM) adalah suatu model statistik dari sebuah sistem yang diasumsikan sebuah proses Markov dengan parameter yang tak diketahui. Kita harus menentukan parameter-parameter tersembunyi (*state*) dari parameter-parameter tersembunyi (*state*) dari parameter-parameter yang dapat diamati. Parameter-parameter yang ditentukan kemudian dapat digunakan untuk analisis yang lebih jauh, misalnya untuk aplikasi *Pattern Recognition*.

Dalam HMM, *state* tidak dapat diamati secara langsung akan tetapi dapat diamati adalah variabel-variabel yang terpengaruh oleh *state*. Setiap *state* memiliki distribusi probabilitas atas token-token *output* yang mungkin muncul. Oleh karena itu rangkaian token yang dihasilkan oleh HMM memberikan sebagian informasi tentang sekuens *state-state*.

Pengenalan suara yang dilakukan pada *server* Google menggunakan algoritma *Hidden Markov Model* (HMM). Pengenalan suara berbasis HMM menggunakan konsep bahwa ucapan terdiri dari urutan suara dasar. Kata apapun di dalam kamus terdiri dari urutan suara dasar. Masing-masing suara dasar ini mempunyai model statistik. Oleh karena itu sebuah kata dapat diekspresikan sebagai urutan model statistik. Suara dijadikan sampel oleh sistem pengenalan suara untuk membentuk sebuah urutan vektor fitur suara (parameter numerik). Urutan vektor fitur suara ini adalah urutan observasi. Kemudian pengenal menentukan secara probabilistic model yang mana yang kemungkinan besar menghasilkan vektor suara. Mengidentifikasi urutan model menghasilkan identifikasi kata yang diucapkan. Ilustrasi algoritma Hidden Markov Model ditunjukkan pada gambar 2.7.



Gambar 2. 5 Parameter *probabilistic* pada Hidden Markov Model

Dimana :

x = kondisi

y = observasi yang mungkin

a = kemungkinan keadaan transisi

b = kemungkinan output

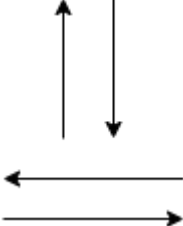
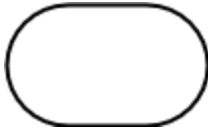
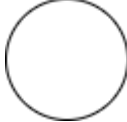

## 2.14 Flowchart

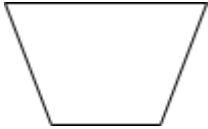
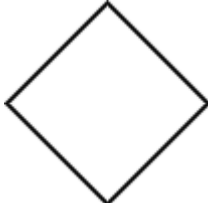

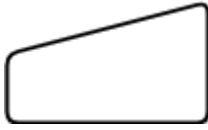


Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analisis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. Terdapat beberapa petunjuk yang harus diperhatikan dalam membuat flowchart, seperti :

1. Flowchart digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja, misalkan melakukan penggandaan diri.
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.

6. Lingkup dan range dari aktifitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan percabangannya diletakkan pada halaman yang terpisah atau hilangkan seluruhnya bisa percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol flowchart yang standar.  
Simbol yang digunakan dalam flowchart yaitu :

Tabel 2. 2 Simbol *flowchart*

No.	Nama Simbol	Gambar Simbol	Kegunaan
1	<i>Flow Direction</i>		Simbol yang digunakan untuk menghubungkan simbol satu dengan simbol yang lainnya.
2	<i>Terminator</i>		Simbol yang digunakan untuk penanda mulai dan selesainya sebuah proses.
3	<i>Connector</i>		Simbol yang digunakan untuk menyatakan sambungan proses ke proses dalam halaman yang sama.
4	<i>Processing</i>		Simbol untuk menggambarkan suatu proses yang dilakukan oleh komputer.

5	<i>Manual Operation</i>		Simbol yang digunakan untuk menyatakan sebuah proses yang tidak dilakukan oleh komputer.
6	<i>Decision</i>		Simbol yang digunakan untuk menunjukkan dua hasil yang memiliki kemungkinan ya atau tidak.
7	<i>Input/output</i>		Simbol yang menunjukkan sebuah proses input atau proses output.
8	<i>Manual Input</i>		Simbol yang menggambarkan sebuah proses input yang dilakukan secara manual dengan menggunakan online keyboard.
9	<i>Preparation</i>		Simbol yang digunakan untuk persiapan penyimpanan di dalam storage.
10	<i>Predefine Process</i>		Simbol yang digunakan untuk menunjukkan pelaksanaan suatu bagian prosedur yang belum detail.

## **BAB 3. METODOLOGI PENELITIAN**

### **3.1 Tempat dan Waktu Penelitian**

#### 3.1.1 Tempat Penelitian

Penelitian ini dilakukan di lingkungan Politeknik Negeri Jember Kampus Bondowoso dan Gedung Teknologi Informasi Politeknik Negeri Jember

#### 3.1.2 Waktu Penelitian

Pada penelitian ini dilaksanakan selama (6 bulan), terhitung dari Bulan Januari 2021 sampai Bulan Juni 2021

### **3.2 Alat dan Bahan**

#### 3.2.1 Alat

Pada penelitian ini menggunakan alat-alat yang terdiri dari perangkat keras (hardware) dan perangkat lunak (software) sebagai berikut :

##### 1. Perangkat Keras (Hardware)

Perangkat keras yang digunakan adalah sebagai berikut :

##### a. Perangkat keras yang digunakan adalah laptop dengan minimal spesifikasi :

- a) Processor : Intel Core 2 duo
- b) RAM : 4192 MB
- c) VGA : Intel HD Graphic
- d) Harddisk : 250 GB
- e) OS : Windows 7

##### b. Perangkat keras yang digunakan adalah smartphone dengan minimal spesifikasi :

- 1) Processor : 1,45Hz Snapdragon 425 Quad-core
- 2) RAM : 2GB

##### 2. Perangkat Lunak (Software)

Perangkat lunak yang digunakan dalam penelitian ini adalah :

##### a. Sistem Operasi *Windows 7*

b. Android Studio

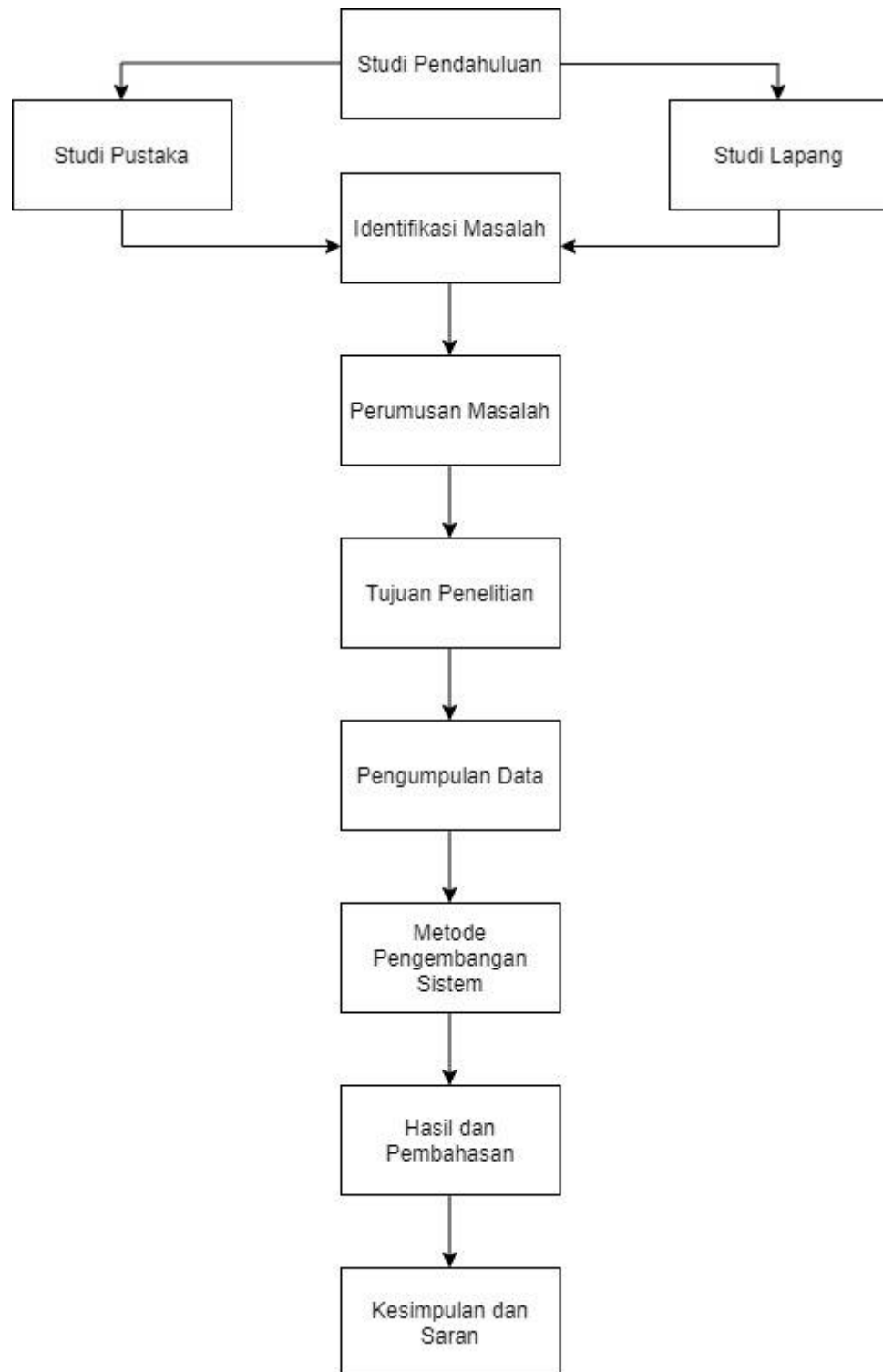
3.2.2 Bahan

Bahan yang digunakan dalam penelitian ini adalah sebagai berikut :

- a. Studi Literatur
- b. Data suara
- c. Data huruf hijaiyah yang akan digunakan (Yanbu'a jilid 1)

**3.3 Tahapan Penelitian**

Penelitian ini dilakukan beberapa bulan dengan melalui beberapa tahap yang dimulai dari studi pendahuluan, identifikasi masalah, perumusan masalah, tujuan penelitian, pengumpulan data, metode pengembangan sistem, hasil dan pembahasan, kesimpulan dan saran. Berikut konsep tahapan penelitian yang akan ditunjukkan pada gambar 3.1 dibawah



Gambar 3. 1 Tahapan penelitian



### 3.3.1 Studi Pendahuluan

Studi pendahuluan adalah studi yang dilakukan untuk memperoleh informasi tentang penelitian yang akan dilakukan. Studi pendahuluan dilakukan karena kelayakan penelitian berkenaan dengan prosedur penelitian dan hal lainnya yang masih belum jelas. Studi Pendahuluan ada dua macam, yaitu:

#### a. Studi Pustaka

Studi pustaka merupakan tahapan untuk mempelajari dan membaca buku referensi dan jurnal yang berhubungan dengan penelitian yang dilakukan yaitu sistem speech recognition dan metode pembelajaran yanbu'a. Literatur atau referensi yang dipelajari bersumber dari jurnal ilmiah atau buku-buku yang berhubungan dengan penelitian.

#### b. Studi Lapang

Studi lapang merupakan penelitian yang dilakukan dengan cara melakukan pengamatan langsung pada objek yang diteliti untuk memperoleh data primer. Lokasi untuk melakukan pengamatan langsung yaitu di TPQ Ar-Rahman, dimana pada TPQ tersebut mengadopsi metode pembelajaran Yanbu'a.

### 3.3.2 Identifikasi Masalah

Identifikasi masalah merupakan kegiatan untuk mengenali dan mengidentifikasi masalah dengan tepat yang berhubungan dengan penelitian yang akan dilakukan.

### 3.3.3 Perumusan Masalah

Setelah melakukan identifikasi masalah, selanjutnya adalah membuat rumusan masalah yang berkaitan dengan penelitian. Rumusan masalah memetakan suatu masalah menjadi rumus berupa pertanyaan, sehingga dari masalah itu dapat diteliti secara ilmiah untuk mendapatkan data dan informasi.

### 3.3.4 Tujuan Penelitian

Setelah memetakan rumusan masalah, selanjutnya adalah menentukan tujuan dari penelitian yang menunjukkan adanya hasil dari rumusan masalah dan adanya target dari sesuatu yang akan dicapai dalam penelitian.

### 3.3.5 Pengumpulan Data

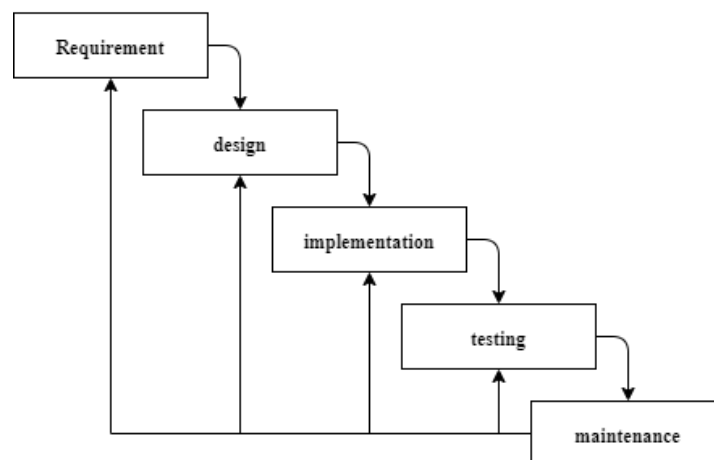
Tahap pengumpulan data bertujuan untuk memperoleh data dan informasi yang dibutuhkan dalam mencapai tujuan penelitian. Berikut cara dalam pengumpulan data :

#### 1. Observasi

Pada tahap ini, peneliti mengumpulkan data dan informasi yang dibutuhkan yaitu data huruf hijaiyah yang memiliki 29 huruf

### 3.3.6 Metode Pengembangan Sistem

Pada tahapan penelitian ini menggunakan metode SDLC (*System Development Life Cycle*). *System Development Life Cycle* (SDLC) adalah metode untuk merancang, membuat dan mengelola informasi dan proses sistem. Ada banyak model SDLC, salah satunya adalah model *Waterfall* yang terdiri dari lima langkah, yang harus diikuti untuk mengembangkan solusi perangkat lunak. Waterfall adalah model pengembangan sistem, yang menjadi fondasi atau awal dari model pengembangan sistem lainnya.



Gambar 3. 2 Metode *waterfall*

## 1. Analisa Kebutuhan (Requirement)

Proses Analisa kebutuhan ini merupakan tahapan untuk menganalisa apa saja yang akan dikerjakan pada proses penelitian.

### a. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berisi fitur-fitur apa saja yang bisa dilakukan oleh sistem.

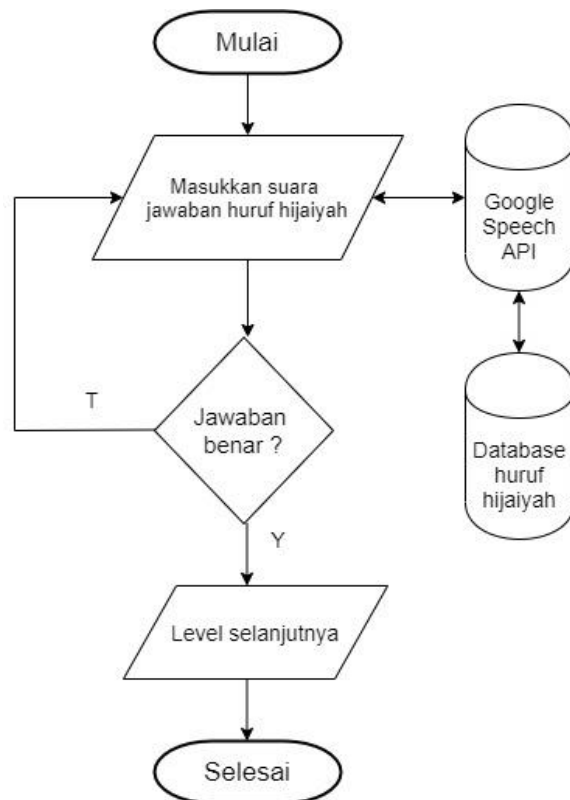
### b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang tidak berkaitan dengan fungsi sistem tapi diperlukan untuk mendukung aktivitas sistem dalam memenuhi kebutuhan fungsional.

## 2. Desain (*Design*)

Pada tahapan ini merupakan tahapan pendeskripsian sistem yang akan diterapkan dalam penerapan speech recognition pada aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan google speech API :

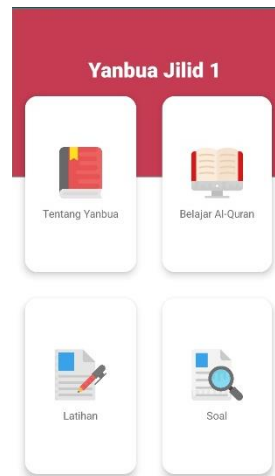
a. *Flowchart* adalah gambaran bagaimana sistem berjalan.



Gambar 3. 3 Flowchart *speech to text*

*b. Desain Antar Muka (Design User Interface)*

Rancangan antar muka digunakan untuk sistem dapat dengan mudah diakses, dipahami, dan digunakan oleh pengguna sistem. Dibawah ini merupakan tampilan antar muka aplikasinya :



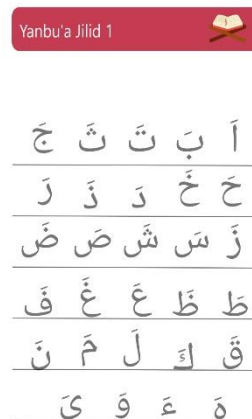
Gambar 3. 4 *Interface* halaman utama



Gambar 3. 5 *Interface* tentang yanbu'a



Gambar 3. 6 *Interface* keutamaan belajar Al-Qur'an



Gambar 3. 7 *Interface* latihan



Gambar 3. 8 *Interface* soal



Gambar 3. 9 *Interface* notifikasi jawaban benar



Gambar 3. 10 *Interface* notifikasi jawaban salah

### 3. Implementasi (*Implementation*)

Setelah dilakukan analisis dan perancangan sistem secara detail maka tahap selanjutnya yaitu tahap implementasi. Implementasi adalah fase implementasi proyek berdasarkan hasil analisis dalam bahasa pemrograman tertentu.

Pada implementasi penerapan *speech recognition* pada aplikasi penentuan level pembelajaran Yanbu'a dilakukan dengan menggunakan bahasa pemrograman java berbasis android.

Implementasi dibagi menjadi dua yaitu Implementasi Program dan Analisis Sistem. Implementasi program merupakan hasil dari perancangan sistem sedangkan analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan yang terjadi yang diharapkan dapat diusulkan perbaikan-perbaikannya.

### 4. Pengujian (*Testing*)

#### a. Pengujian *Black Box Testing*



Tahap ini merupakan uji coba fungsional terhadap sistem yang sudah dibuat. Pengujian dilakukan untuk memastikan output sesuai dengan kebutuhan sistem.

b. Pengujian UAT.

UAT (*User Acceptance Test*) adalah tahap akhir dari pengujian yang dilakukan oleh pengguna aplikasi yang dibangun untuk mengetahui apakah aplikasi berjalan sesuai dengan proses yang telah ditentukan, yang dirancang untuk memperbaiki apapun yang terjadi sebelum aplikasi dibuat. Biasanya dievaluasi oleh pengguna aplikasi menggunakan kuesioner berdasarkan kasus uji yang diperlukan.

c. Pengujian Akurasi Implementasi Google Speech.

Pengujian ini melibatkan seorang ahli membaca Al-Qur'an (guru mengaji) untuk mengkaji akurasi *google speech* yang diimplementasikan dalam media pembelajaran yanbu'a. Penguji akan memberikan nilai terhadap sistem yang telah diuji cobakan. Seorang ahli diminta untuk menggunakan fitur soal dan menjawabnya dengan inputan benar. Setelah itu seorang ahli diminta mengisi form yang berisi kesesuaian huruf hijaiyah yang diucapkan dengan apa yang ditampilkan dalam antarmuka hasil jawaban, apakah sudah benar atau belum.

5. Pemeliharaan (*Maintenance*)

Pemeliharaan sistem adalah suatu kombinasi dari berbagai tindakan yang dilakukan untuk menjaga suatu sistem atau memperbaikinya sampai suatu kondisi yang bisa diterima.

### 3.3.7 Hasil dan Pembahasan

Hasil dan penjelasan dapat dijelaskan sebagai pemikiran asli untuk memberikan penjelasan dan interpretasi atas hasil penelitan yang telah dianalisis guna menjawab pertanyaan pada penelitiannya. Jadi, pembahasan hasil penelitian merupakan pembahasan terhadap temuan yang diperoleh

### 3.3.8 Kesimpulan dan Saran

Kesimpulan adalah pernyataan singkat, jelas, dan sistematis dari keseluruhan hasil analisis, pembahasan, dan pengujian hipotesis dalam sebuah penelitian. Saran adalah usul atau pendapat dari seorang peneliti yang berkaitan dengan pemecahan masalah yang menjadi objek penelitian ataupun kemungkinan penelitian lanjutan.

## **BAB 4. HASIL DAN PEMBAHASAN**

### **4.1 Studi Pendahuluan**

#### a. Studi Pustaka

Studi pustaka merupakan tahapan untuk mempelajari dan membaca buku referensi dan jurnal yang berhubungan dengan penelitian yang dilakukan yaitu sistem speech recognition dan metode pembelajaran yanbu'a. Literatur atau referensi yang dipelajari bersumber dari jurnal ilmiah atau buku-buku yang berhubungan dengan penelitian.

#### b. Studi Lapang

Studi lapang merupakan penelitian yang dilakukan dengan cara melakukan pengamatan langsung pada objek yang diteliti untuk memperoleh data primer. Lokasi untuk melakukan pengamatan langsung yaitu di TPQ Ar-Rahman, dimana pada TPQ tersebut mengadopsi metode pembelajaran Yanbu'a.

### **4.2 Identifikasi Masalah**

Pada tahapan ini, akan disebutkan beberapa masalah yang menjadi dasar penelitian ini. Berikut hasil dari identifikasi masalah penelitian :

1. Banyaknya remaja yang belum mengerti aksara hijaiyah
2. Metode yang diajarkan dalam mempelajari Al-Qur'an bersifat monoton sehingga tidak menyenangkan

### **4.3 Perumusan Masalah**

Berdasarkan identifikasi masalah yang sudah diuraikan, dapat dirumuskan permasalahan sebagai berikut :

1. Bagaimana merancang sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API?

2. Bagaimana mengembangkan sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API?
3. Bagaimana tahapan-tahapan pengujian yang dilakukan agar dapat menemukan tingkat keakuratan antara suara pembicara dengan jawaban Google Speech API dalam aplikasi ini?

#### **4.4 Tujuan Penelitian**

1. Merancang sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API
2. Mengembangkan sebuah aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan Google Speech API
3. Menjelaskan tahapan-tahapan pengujian yang dilakukan agar dapat menemukan tingkat keakuratan antara suara pembicara dengan jawaban Google Speech API dalam aplikasi ini

#### **4.5 Pengumpulan Data**

Pada pengumpulan data ini dilakukan pada 10 Narasumber dengan jenis kelamin laki-laki dan perempuan. Pada proses ini narasumber diminta membacakan daftar kata yang sudah diberikan satu per satu. Daftar kata tersebut berjumlah 27 huruf hijaiyah. Pengambilan suara tersebut dilakukan menggunakan perekam suara *smartphone*. Perekaman dilakukan dengan pembacaan huruf hijaiyah pada daftar huruf hijaiyah satu per satu sehingga mendapatkan 27 data suara pada setiap narasumber. Setiap data suara rata-rata berdurasi sekitar 1 detik.

- ☐ Huruf Ain
- ☐ Huruf Alif
- ☐ Huruf Ba
- ☐ Huruf Dal
- ☐ Huruf Dhot
- ☐ Huruf Dzal
- ☐ Huruf Dzo
- ☐ Huruf Fa
- ☐ Huruf Goin
- ☐ Huruf Ha
- ☐ Huruf Haa
- ☐ Huruf Hamzah
- ☐ Huruf Ho
- ☐ Huruf Jim
- ☐ Huruf Kal
- ☐ Huruf Kof
- ☐ Huruf Lam
- ☐ Huruf Mim
- ☐ Huruf Nun
- ☐ Huruf Ro
- ☐ Huruf Shod
- ☐ Huruf Sin
- ☐ Huruf Syin
- ☐ Huruf Ta
- ☐ Huruf To
- ☐ Huruf Tsa
- ☐ Huruf Ya
- ☐ Huruf Za

Gambar 4. 1 Data suara

Pengumpulan data huruf hijaiyah jilid 1 diperoleh dari buku panduan yanbu'a. Dalam hal ini hanya mengambil data 29 huruf hijaiyah. Data 29 huruf hijaiyah ditampilkan pada tabel 4.1.

Tabel 4. 1 Huruf Hijaiyah

No	Huruf Hijaiyah	Pengucapan Huruf Hijaiyah
1.	ا	Alif
2.	ب	Ba
3.	ت	Tak
4.	ث	Sak
5.	ج	Jim
6.	ح	Hak

7.	خ	Kho'
8.	د	Dal
9.	ذ	Dzal
10.	ر	Ro
11.	ز	Za
12.	س	Sin
13.	ش	Syim
14.	ص	Sot
15.	ض	Dot
16.	ط	Tok
17.	ظ	Dok
18.	ع	Ain
19.	غ	Goin
20.	ف	Vak
21.	ق	Kof
22.	ك	Kaf
23.	ل	Lam
24.	م	Mim
25.	ن	Nun
26.	ه	Ha
27.	ء	Hamzah
28.	و	Wau
29.	ي	Ya

## 4.6 Metode Pengembangan Sistem

### 4.6.1 Analisa Kebutuhan (Requirement)

Proses Analisa kebutuhan ini merupakan tahapan untuk menganalisa apa saja yang akan dikerjakan pada proses penelitian. Tujuan dari fase analisis adalah memahami dengan sebenar-benarnya kebutuhan dari sistem baru dan mengembangkan sebuah sistem yang mawadahi kebutuhan tersebut, baik untuk kebutuhan sistem maupun dilihat dari segi pengguna.

#### a. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berisi fitur-fitur apa saja yang bisa dilakukan oleh sistem. Kebutuhan fungsional yang terdapat pada sistem speech recognition pada aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan google speech api sebagai berikut :

1. Sistem mampu mengolah data suara dengan menggunakan *google speech API*
2. Sistem dapat menampilkan halaman utama, menu tentang yanbu'a, menu belajar al-qur'an, menu latihan dan menu soal
3. Sistem mampu menerapkan fitur *text to speech*

#### b. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang tidak berkaitan dengan fungsi sistem tapi diperlukan untuk mendukung aktivitas sistem dalam memenuhi kebutuhan fungsional. Kebutuhan non-fungsional yang terdapat pada sistem speech recognition pada aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan google speech api sebagai berikut :

1. Desain *user interface* dan *user friendly* sehingga memudahkan pengguna dalam menggunakan sistem.
2. Sistem berbasis *android*.

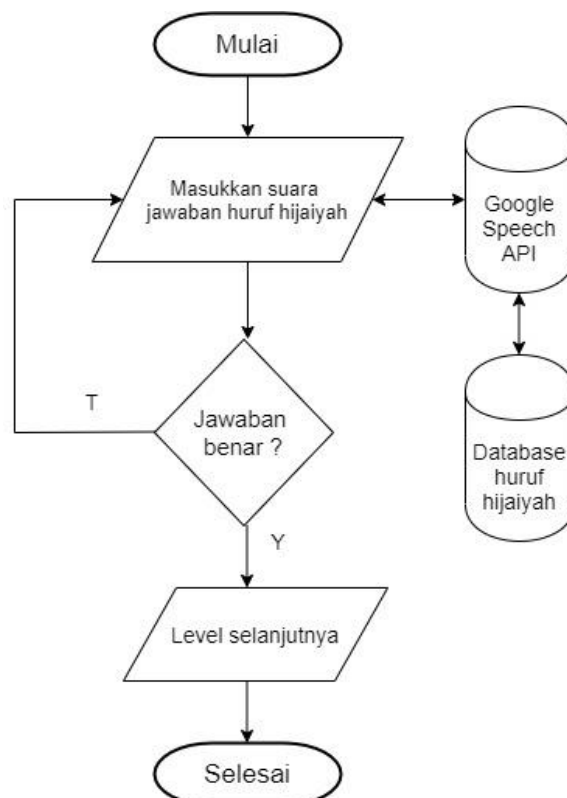
#### 4.6.2 Desain (Design)

Tahapan desain bertujuan untuk memberikan gambaran lengkap tentang apa yang harus dikerjakan dan bagaimana tampilan dari sebuah sistem yang diinginkan. Pada tahapan ini merupakan tahapan pendeskripsian sistem yang akan diterapkan dalam penerapan speech recognition pada aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan google speech API. Tahapan desain ini hanya terdiri dari flowchart dan desain antar muka.

##### 1. Flowchart

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program.

##### a. Flowchart Speech to Text



Tabel 4. 2 Flowchart Speech to Text



### 4.6.3 Implementasi (Implementation)

#### a. Analisis Sistem

##### 1. Proses Mengganti Tipe File

Pada tahap pengambilan data suara menghasilkan banyak data suara dengan format mp3. Namun, format mp3 harus dikonversi ke format WAV untuk melanjutkan pemrosesan. WAV banyak digunakan dalam rekaman musik dan pembuatan film. Pengeditan sangat mudah dan kualitas suaranya juga tinggi. WAV juga mendukung perekaman multi-layer dan paling sering digunakan di pengeditan non-liner (NLE).

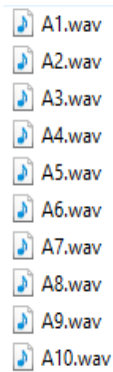
Untuk merubah file dengan format mp3 menjadi format WAV, pada penelitian ini menggunakan matlab dengan source code:

```
mp3filename = 'D:\Helmione\PCD\Ba.mp3';
wavfilename = 'D:\Helmione\PCD\Ba.wav';
sinyal = audioread(mp3filename);
info = audioinfo(mp3filename);
Fs = info.SampleRate;
audiowrite (wavfilename, sinyal, Fs);
```

Gambar 4. 2 Source code mengganti tipe file

Name	#	Title	Contributing artists	Album
 A1.mp3				
 A2.mp3				
 A3.mp3				
 A4.mp3				
 A5.mp3				
 A6.mp3				
 A7.mp3				
 A8.mp3				
 A9.mp3				
 A10.mp3				

Gambar 4. 3 File data suara format mp3



Gambar 4. 4 File data suara format wav

## 2. Proses Akuisisi Suara (*Voice Acquisition Process*).

Rumus sampling :

$$X = FS \cdot dt \left( \frac{bit}{8} \right) \cdot J \quad \text{-----} 4.1$$

Dimana : X = Jumlah data sampel

FS = Frekuensi Sampling (Hz)

Dt = Durasi (detik)

Bit = Jumlah data tiap sampel

J = Channel (mono = 1 dan stereo =2)

Analisis ini menggunakan contoh durasi rekaman 1 detik.

Tabel 4. 3 Akuisisi suara

Frekuensi (Hz)	Bits / Sample	Channel	Perhitungan	Jumlah data sample (byte)
8000	16	Mono	$8000 * 1 * (16/8) * 1$	16000
11025	16	Mono	$11025 * 1 * (16/8) * 1$	22050
22025	16	Mono	$22025 * 1 * (16/8) * 1$	44100
44100	16	Mono	$44100 * 1 * (16/8) * 1$	88200

### 3. Ekstraksi Ciri (feature extraction)

Dalam penelitian ini, rekaman suara yang dimiliki telah masuk ke dalam sistem lalu diekstraksi menggunakan metode MFCC untuk dikenali oleh komputer.

Pada penelitian ini, proses MFCC dapat diamati dengan menggunakan bahasa Python, yang mana menggunakan beberapa *library* yang terdapat pada python untuk memudahkan proses MFCC. Library yang dipakai dapat dilihat pada Gambar 4.11

```
import librosa
import librosa.display
import IPython.display as ipd
import matplotlib.pyplot as plt
import numpy as np
```

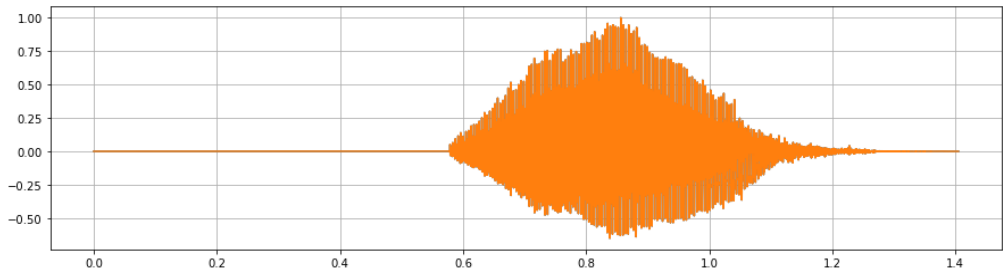
Gambar 4. 5 Library MFCC

Library di atas berfungsi untuk proses pembacaan file, penulisan file, menyimpan file dan proses perhitungan. Penjelajasan dari library tersebut yaitu:

- a. Library numpy yang digunakan sebagai pengambil nilai numerik untuk dijadikan perhitungan selanjutnya.
- b. Library librosa digunakan sebagai pembaca file audio dan juga sebagai pengambil nilai dari MFCC feature.

Library tersebut digunakan pada proses yang dilakukan pada tahap selanjutnya. Proses selanjutnya yang dilakukan setelah deklarasi library tersebut adalah membuat fungsi untuk mendapatkan MFCC feature dari file wav.

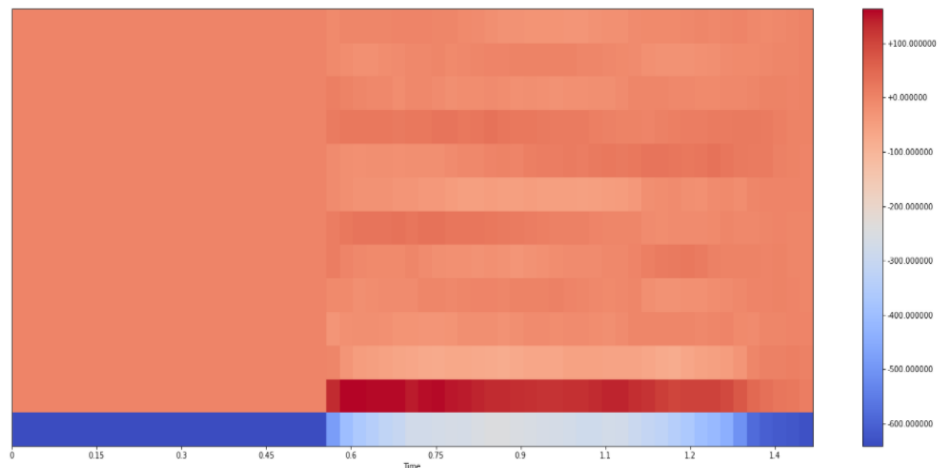
Pada proses pengambilan nilai dari MFCC feature tahap pertama yang dilakukan adalah pembacaan file audio dengan menggunakan library librosa. File audio tersebut dapat diambil panjang gelombang atau amplitudo dan waktu



rambat dari gelombang tersebut. Kedua parameter tersebut bisa disebut juga dengan frekuensi. Visualisasi dari file audio tersebut dapat dilihat pada Gambar 4.15

Gambar 4. 6 Visualisasi file audio

Hasil MFCC tersebut dapat dilihat pada Gambar 4.16



Gambar 4. 7 Hasil MFCC

Berikut ini adalah perhitungan manual dari proses ekstraksi fitur yang mana data sampel yang digunakan menggunakan data dummy.

#### b. DC-Removal

Berdasarkan perhitungan DC-Removal, berikut adalah contohnya dari beberapa sampel data yang akan digunakan untuk perhitungan dengan

menggunakan rumus-rumus yang terdapat pada metode MFCC. Data sampel terdiri dari 10,11,13,14,15,12,17,18,9,8 dihitung sebagai data referensi.

$$\bar{x} = \frac{10+11+13+14+15+12+17+18+9+8}{10} = 12.7$$

Setelah diketahui nilai rata-rata, maka nilai sinyal awal adalah dikurangi dengan hasil DC Removal, sehingga dihasilkan nilai sinyal sebagai berikut.

$$y[n] = x[n] - \bar{x} ; 0 \leq n \leq N - 1 \quad \text{-----4.2}$$

$$y_0 = 10 - 12.7 = -2.7$$

$$y_5 = 12 - 12.7 = -0.7$$

$$y_1 = 11 - 12.7 = -1.7$$

$$y_6 = 17 - 12.7 = 4.3$$

$$y_2 = 13 - 12.7 = 0.3$$

$$y_7 = 18 - 12.7 = 5.3$$

$$y_3 = 14 - 12.7 = 1.3$$

$$y_8 = 9 - 12.7 = -3.7$$

$$y_4 = 15 - 12.7 = 2.3$$

$$y_9 = 8 - 12.7 = -4.7$$

Jadi, hasil data sinyal setelah DC Removal adalah -2.7, -1.7, 0.3, 1.3, 2.3, -0.7, 4.3, 5.3, -3.7, -4.7

### c. Pre-emphasise

Jika hasil DC Removal adalah (-2.7, -1.7, 0.3, 1.3, 2.3, -0.7, 4.3, 5.3, -3.7, -4.7) maka *pre-emphasize* dihitung menggunakan rumus *pre-emphasize* dan nilai  $\alpha = 0.97$  sebagai berikut.

$$y[n] = s[n] - \alpha s[n - 1] ; 0.9 \leq \alpha \leq 1.0 \quad \text{-----4.3}$$

$$Y_0 = -2.7$$

$$Y_1 = (-1.7 - (-2.7 * 0.97)) = 0.919$$

$$Y_2 = (0.3 - (-1.7 * 0.97)) = 1.949$$

$$Y_3 = (1.3 - (0.3 * 0.97)) = 1.009$$

$$Y_4 = (2.3 - (1.3 * 0.97)) = 1.039$$

$$Y_5 = (-0.7 - (2.3 * 0.97)) = -2.931$$

$$Y_6 = (4.3 - (-0.7 * 0.97)) = 4.979$$

$$Y_7 = (5.3 - (4.3 * 0.97)) = 1.129$$

$$Y8 = (-3.7) - (5.3 * 0.97) = -8.841$$

$$Y9 = (-4.7) - (-3.7 * 0.97) = -1.111$$

Sinyal baru setelah *pre-emphasize* diperoleh dari data DC-Removal yang ditambahkan dengan data *pre-emphasize* menggunakan rumus sinyal baru yaitu

$$N_n = N_n + Y_n \quad \text{-----4.4}$$

$$N0 = -2.7 + (-2.7) = -5.4$$

$$N5 = -0.7 + (-2.931) = -3.631$$

$$N1 = -1.7 + (0.919) = -0.781$$

$$N6 = 4.3 + (4.979) = 9.279$$

$$N2 = 0.3 + (1.949) = 2.249$$

$$N7 = 5.3 + (1.129) = 6.429$$

$$N3 = 1.3 + (1.009) = 2.309$$

$$N8 = -3.7 + (-8.841) = -12.541$$

$$N4 = 2.3 + (1.039) = 3.339$$

$$N9 = -4.7 + (-1.111) = -5.811$$

Jadi nilai sinyal setelah *pre-emphasize* sebagai berikut (-5.4, -0.781, 2.249, 2.309, 3.339, -3.631, 9.279, 6.429, -12.541, -5.811).

#### d. Frame Blocking

Dalam penelitian ini, suara dipotong sepanjang 20 ms di setiap shift. Rumus Frame Blocking digunakan untuk menghitung jumlah frame dan jumlah sampel per frame. Dalam penelitian ini didapatkan jumlah frame dengan waktu = 20 ms, dan laju sampel = 88200 Hz,  $N = 88200 * 0,02 = 1764$  titik sampel dan  $M = 1764/2 = 882$ . Dihitung menggunakan rumus *frame blocking* yaitu sebagai berikut.

$$\text{Number of frames} = ((I - N)/ M+1) \quad \text{-----4.5}$$

$$\text{Waktu} : 20\text{ms (milidetik)} = 0.02 \text{ s (detik)}$$

$$I : \text{laju sampel} = 88200 \text{ Hz}$$

$$N : \text{titik sampel}$$

$$\text{Jumlah frame yaitu} = ((88200-1764)/882+1) = 99$$

#### e. Windowing

Proses windowing menggunakan jendela hamming dengan contoh titik 1764 sebagai berikut.

$$W(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{M-1}\right) \quad \text{-----4.6}$$

$$W_0 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 0}{882-1}\right) = 0.08$$

$$W_1 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 1}{882-1}\right) = 0.08$$

$$W_2 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 2}{882-1}\right) = 0.08$$

$$W_3 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 3}{882-1}\right) = 0.08$$

$$W_4 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 4}{882-1}\right) = 0.08$$

$$W_5 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 5}{882-1}\right) = 0.08$$

$$W_6 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 6}{882-1}\right) = 0.08$$

$$W_7 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 7}{882-1}\right) = 0.08$$

$$W_8 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 8}{882-1}\right) = 0.08$$

$$W_9 = 0.54 - 0.46 \cos\left(\frac{2 \times 3.14 \times 9}{882-1}\right) = 0.08$$

Hasil proses *pre-emphasize* (-5.4, -0.781, 2.249, 2.309, 3.339, -3.631, 9.279, 6.429, -12.541, -6.921) akan dilakukan proses windowing menggunakan rumus sebagai berikut.

$$x(n) = xi(n) * w(n) \quad ; \quad n = 0, 1, \dots, N - 1 \quad \text{-----4.7}$$

$$X_0 = -5.4 * 0.08 = -0.43$$

$$X_5 = -3.631 * 0.08 = -0.29$$

$$X_1 = -0.781 * 0.08 = -0.06$$

$$X_6 = 9.279 * 0.08 = 0.74$$

$$X_2 = 2.249 * 0.08 = 0.18$$

$$X_7 = 6.429 * 0.08 = 0.51$$

$$X_3 = 2.309 * 0.08 = 0.18$$

$$X_8 = -12.541 * 0.08 = -1.012$$

$$X_4 = 3.339 * 0.08 = 0.26$$

$$X_9 = -5.811 * 0.08 = -0.47$$

Pada proses windowing diperoleh nilai-nilai sebagai berikut (-0.43, -0.06, 0.18, 0.26, -0.29, 0.74, 0.51, -1.012, -0.47).

f. FFT

$$----- F(K) = \sum_{n=1}^N f(n) \cos\left(\frac{2\pi nkT}{N}\right) - j \sum_{n=1}^N f(n) \sin\left(\frac{2\pi nkT}{N}\right) -----4.8$$

$$\begin{aligned} f_0 &= \frac{1}{4} \left[ -0.43 \left( \cos\left(\frac{2\phi i * 0 * 0}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 0}{10}\right) \right) + \\ &\quad \left[ -0.06 \left( \cos\left(\frac{2\phi i * 0 * 1}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 1}{10}\right) \right) + \\ &\quad \left[ 0.17 \left( \cos\left(\frac{2\phi i * 0 * 2}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 2}{10}\right) \right) + \\ &\quad \left[ 0.18 \left( \cos\left(\frac{2\phi i * 0 * 3}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 3}{10}\right) \right) + \\ &\quad \left[ 0.18 \left( \cos\left(\frac{2\phi i * 0 * 4}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 4}{10}\right) \right) + \\ &\quad \left[ -0.29 \left( \cos\left(\frac{2\phi i * 0 * 5}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 5}{10}\right) \right) + \\ &\quad \left[ 0.74 \left( \cos\left(\frac{2\phi i * 0 * 6}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 6}{10}\right) \right) + \\ &\quad \left[ 0.51 \left( \cos\left(\frac{2\phi i * 0 * 7}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 7}{10}\right) \right) + \\ &\quad \left[ -1.003 \left( \cos\left(\frac{2\phi i * 0 * 8}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 8}{10}\right) \right) + \\ &\quad \left[ -0.55 \left( \cos\left(\frac{2\phi i * 0 * 9}{10}\right) \right) \right] - j \left( \sin\left(\frac{2\phi i * 0 * 9}{10}\right) \right) \\ &= -0.103 - 0j \\ &= -0.103 \end{aligned}$$

Untuk perhitungan F1 sampai F9 sama dengan F0 hanya berbeda pada bagian F seperti  $F1 = \frac{2}{4}$ ,  $F2 = \frac{3}{4}$ ,  $F3 = \frac{4}{4}$ ,  $F4 = \frac{5}{4}$ ,  $F6 = \frac{7}{4}$ ,  $F7 = \frac{8}{4}$ ,  $F8 = \frac{9}{4}$ ,  $F9 = \frac{10}{4}$

Jadi, data yang diperoleh dari hasil sinyal FFT sebagai berikut ( $F0 = -0.108$ ,  $F1 = -0.031$ ,  $F2 = 0.135$ ,  $F3 = 0.184$ ,  $F4 = 0.334$ ,  $F5 = -0.43$ ,  $F6 = 1.305$ ,  $F7 = 1.036$ ,  $F8 = -2.27$ ,  $F9 = -1.175$ )

#### g. Mel Frequency Warping

Untuk mendapatkan  $H_i$  menggunakan rumus

$$H_i = 2595 * \log(1+1000/700)/S_i/2 -----4.9$$

$$H_0 = 2595 * \log(1+1000/700) / -0.108/2 = -4629.56$$

$$H_1 = 2595 * \log(1+1000/700) / -0.031/2 = -16002.6$$

$$H_2 = 2595 * \log(1+1000/700) / 0.135/2 = 3703.133$$



$$H3 = 2595 * \log (1+1000/700) /0.184//2 = 2703.206$$

$$H4 = 2595 * \log (1+1000/700) /0.334/2 = 1493.941$$

$$H5 = 2595 * \log (1+1000/700) /-0.43/2 = -1142.33$$

$$H6 = 2595 * \log (1+1000/700) /1.305/2 = 382.8749$$

$$H7 = 2595 * \log (1+1000/700) /1.036//2 = 482.6177$$

$$H8 = 2595 * \log (1+1000/700) /-2.27/2 = -219.441$$

$$H9= 2595 * \log (1+1000/700) /-1.175//2 = -425.183$$

Kemudian perhitungan filterbank dapat menggunakan rumus sebagai berikut.

$$Y[t] = \sum_{j=1}^n S[j]Hi[j] \quad \text{-----4.10}$$

$$Y0 = -0.108 * -4629.56 = 499.99 \quad Y5 = -0.437 * -1143.33 = 499.99$$

$$Y1 = -0.031 * -16002.6 = 499.99 \quad Y6 = 1.305 * 382.8749 = 499.99$$

$$Y2 = 0.135 * 3703.133 = 499.99 \quad Y7 = 1.036 * 482.6177 = 499.99$$

$$Y3 = 0.184 * 2703.206 = 499.99 \quad Y8 = -2.2784 * -219.441 = 499.99$$

$$Y4 = 0.334 * 1493.941 = 499.99 \quad Y9 = -1.175 * -425.183 = 499.99$$

#### h. Discrete Cosine Transform

$$Cn = \sum_{k=1}^k (\log SK) \text{Cos} \left[ n \left( K - \frac{1}{2} \right) \frac{\pi}{k} \right], \quad n = 1, 2 \dots K \quad \text{-----4.11}$$

$$\begin{aligned} C0 &= \log (499.99) \cos[0(0-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(1-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(2-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(3-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(4-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(5-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(6-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(7-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(8-1/2)3.14/10] + \\ &\quad \log (499.99) \cos[0(9-1/2)3.14/10] \\ &= 26.989 \end{aligned}$$

Jadi nilai DCT = 26,989

i. Cepstrum (Cepstral Liftering)

Hasil dari proses DCT adalah Cepstrum yang merupakan hasil akhir dari fitur proses ekstraksi. Cepstrum adalah istilah kebalikan dari spektrum. Itu sudah biasa memperoleh informasi dari sinyal suara ucapan manusia. Untuk dapat meningkatkan kualitas proses pengenalan, maka hasil dari proses DCT haruslah melalui proses Cepstral Liftering.

Rumus DCT sebagai berikut.

$$w[n] = \begin{cases} 1 + \frac{L}{2} \sin\left(\frac{n\pi}{L}\right); & n = 1, 2, \dots, L \\ 0 & \end{cases} \quad \text{-----4.12}$$

4. Pengenalan Pola (*pattern recognition*)

Merupakan proses akhir pada *Speech Recognition* yaitu tahap pencocokan data baru dengan data suara (pencocokan tata bahasa) pada *template*. Metode yang digunakan dalam pengenalan pola adalah metode *Hidden Markov Model* (HMM). Pola yang didapat akan dicocokkan dengan berbagai macam model.

a. Hidden Markov Model

*Hidden Markov Model* (HMM) adalah suatu model statistik dari sebuah sistem yang diasumsikan sebuah proses Markov dengan parameter yang tak diketahui. Hidden Markov Model juga merupakan sebuah metode yang tugasnya mengelompokkan sifat-sifat *spectral* dari bagian suara dengan beberapa pola berdasarkan nilai probabilitas kemiripan (*likelihood*).

Berikut adalah beberapa elemen dalam HMM, yaitu sebagai berikut:

1. Jumlah state (N)

State dilabelkan {1,2,3,...N} dan state pada waktu dinotasikan dengan  $q_t$ .

9. Parameter model (M)

10. Menginisialkan *state* awal

$$\pi_i = P(q_t = i) \quad 1 \leq i \leq N \quad \text{-----4.13}$$

Keterangan :

$\pi$  = state awal

P = probabilitas

$q_t$  = label *state* pada waktu t

N = jumlah *state*

11. Probabilitas transisi ( $A = [a_{ij}]$ )

$$a_{ij} = P(q_{t+1} = j | q_t = i) \quad 1 \leq i, j \leq N \quad \text{-----4.14}$$

Keterangan :

$a_{ij}$  = probabilitas transisi

P = probabilitas

$q_t$  = label *state* pada waktu t

N = jumlah *state*

12. Probabilitas simbol observasi ( $B = \{b_j(k)\}$ )

$$b_j(k) = P(o_t = k | q_t = j)$$

$$1 \leq j \leq N, \leq k \leq M \quad \text{-----4.15}$$

Keterangan :

B = probabilitas simbol observasi

P = probabilitas

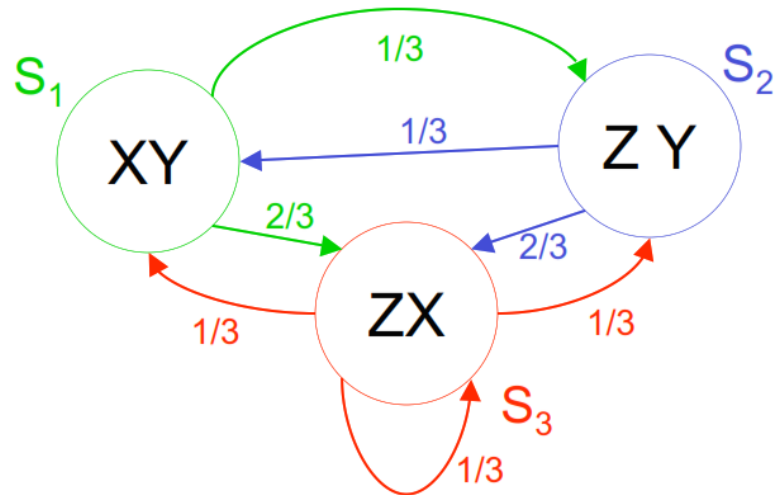
$o_t$  = label observasi pada waktu t

$q_t$  = label *state* pada waktu t

N = jumlah *state*

M = parameter model

Contoh penyelesaian rantai markov model :



Diketahui :  $N = 3$

$M = 3$

$$\pi_1 = 1/2$$

$$\pi_2 = 1/2$$

$$\pi_3 = 0$$

$$a_{11} = 0$$

$$a_{12} = 1/3$$

$$a_{31} = 2/3$$

$$a_{12} = 1/3$$

$$a_{22} = 0$$

$$a_{32} = 2/3$$

$$a_{13} = 1/3$$

$$a_{32} = 1/3$$

$$a_{33} = 1/3$$

$$b_1 (X) = 1/2$$

$$b_1 (Y) = 1/2$$

$$b_1 (X) = 0$$

$$b_2 (X) = 0$$

$$b_2 (Y) = 1/2$$

$$b_2 (X) = 1/2$$

$$b_3 (X) = 1/2$$

$$b_3 (Y) = 0$$

$$b_3 (X) = 1/2$$

## 5. Pembuatan model training dan testing

Hal pertama yang harus dilakukan untuk menjalankan proses pemodelan pada Google Colab yaitu menyimpan file audio wav didalam folder yang telah tersedia di Google Colab. Setelah itu import packages yang diperlukan di dalam proses pemodelan.

```

import os
import pathlib
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import tensorflow as tf
from tensorflow.keras.layers.experimental import preprocessing
from tensorflow.keras import layers
from tensorflow.keras import models
from IPython import display
from google_drive_downloader import GoogleDriveDownloader as gdd
# Set seed for experiment reproducibility
seed = 42
    tf.random.set_seed(seed)
    np.random.seed(seed)

```

File audio wav berada di folder var/local. Folder var/local hanya berisi 27 folder yang masing-masing folder berisi 10 suara berbeda

```
data_dir='/var/local'
```

Periksa statistik dasar tentang kumpulan data

```

commands = np.array(tf.io.gfile.listdir(str(data_dir)))
commands = commands[commands != 'README.md']
print('Commands:', commands)

```

Ekstrak file audio ke dalam daftar

```

filenames = tf.io.gfile.glob(str(data_dir) + '/*/*')
filenames = tf.random.shuffle(filenames)
num_samples = len(filenames)
print('Number of total examples:', num_samples)
print('Number of examples per label:',
      len(tf.io.gfile.listdir(str(data_dir))))
print('Example file tensor:', filenames[0])

```

Pisahkan file menjadi set pelatihan, validasi, dan pengujian.

```

train_files = filenames[:216]
val_files = filenames[216: 216 + 40]
test_files = filenames[-40:]

print('Training set size', len(train_files))
print('Validation set size', len(val_files))
print('Test set size', len(test_files))

```

File audio awalnya akan dibaca sebagai file biner, yang ingin diubah menjadi tensor numerik. Untuk memuat file audio akan menggunakan `tf.audio.decode_wav` yang mengembalikan audio yang disandikan WAV sebagai Tensor dan laju sampel

```

def decode_audio(audio_binary):
    audio, _ = tf.audio.decode_wav(audio_binary)
    return tf.squeeze(audio, axis=-1)

```

Label untuk setiap file WAV adalah direktori induknya

```

def get_label(file_path):
    parts = tf.strings.split(file_path, os.path.sep)
    return parts[-2]

```

Definisikan metode yang akan mengambil nama file dari file WAV

```

def get_waveform_and_label(file_path):
    label = get_label(file_path)
    audio_binary = tf.io.read_file(file_path)
    waveform = decode_audio(audio_binary)
    return waveform, label

```

Penerapan `process_path` untuk membangun set pelatihan

```

AUTOTUNE = tf.data.AUTOTUNE
files_ds = tf.data.Dataset.from_tensor_slices(train_files)
waveform_ds = files_ds.map(get_waveform_and_label, num_parallel_calls=AUTOTUNE)

```

Periksa beberapa bentuk gelombang audio dengan label yang sesuai waktu

```
rows = 10
cols = 3
n = rows*cols
fig, axes = plt.subplots(rows, cols, figsize=(10, 12))
for i, (audio, label) in enumerate(waveform_ds.take(n)):
    r = i // cols
    c = i % cols
    ax = axes[r][c]
    ax.plot(audio.numpy())
    ax.set_yticks(np.arange(-1.2, 1.2, 0.2))
    label = label.numpy().decode('utf-8')
    ax.set_title(label)
plt.show()
```

Menerapkan FFT untuk mengubah audio menjadi domain frekuensi

```
def get_spectrogram(waveform):
    zero_padding = tf.zeros([111216] - tf.shape(waveform), dtype=tf.
float32)
    waveform = tf.cast(waveform, tf.float32)
    equal_length = tf.concat([waveform, zero_padding], 0)
    spectrogram = tf.signal.stft(
        equal_length, frame_length=255, frame_step=128)
    spectrogram = tf.abs(spectrogram)
    return spectrogram
```

Bandingkan bentuk gelombang, spektogram dan audio sebenarnya dari satu cont

```
for waveform, label in waveform_ds.take(1):
    label = label.numpy().decode('utf-8')
    spectrogram = get_spectrogram(waveform)

print('Label:', label)
print('Waveform shape:', waveform.shape)
print('Spectrogram shape:', spectrogram.shape)
print('Audio playback')
display.display(display.Audio(waveform, rate=16000))
```

```

def plot_spectrogram(spectrogram, ax):
    log_spec = np.log(spectrogram.T)
    height = log_spec.shape[0]
    width = log_spec.shape[1]
    X = np.linspace(0, np.size(spectrogram), num=width, dtype=int)
    Y = range(height)
    ax.pcolormesh(X, Y, log_spec)
fig, axes = plt.subplots(2, figsize=(12, 8))
timescale = np.arange(waveform.shape[0])
axes[0].plot(timescale, waveform.numpy())
axes[0].set_title('Waveform')
axes[0].set_xlim([0, 111216])
plot_spectrogram(spectrogram.numpy(), axes[1])
axes[1].set_title('Spectrogram')
plt.show()

```

Ubah dataset bentuk gelombang untuk memiliki gambar spektrogram dan label yang sesuai sebagai id bilangan bulat

```

def get_spectrogram_and_label_id(audio, label):
    spectrogram = get_spectrogram(audio)
    spectrogram = tf.expand_dims(spectrogram, -1)
    label_id = tf.argmax(label == commands)
    return spectrogram, label_id
spectrogram_ds = waveform_ds.map(
    get_spectrogram_and_label_id, num_parallel_calls=AUTOTUNE
)

```

Periksa gambar spektrogram untuk sampel yang berbeda dari kumpulan data

```

rows = 10 cols = 3
n = rows*cols
fig, axes = plt.subplots(rows, cols, figsize=(10, 10))
for i, (spectrogram, label_id) in enumerate(spectrogram_ds.take(n))
:
    r = i // cols
    c = i % cols
    ax = axes[r][c]
    plot_spectrogram(np.squeeze(spectrogram.numpy()), ax)
    ax.set_title(commands[label_id.numpy()])
    ax.axis('off')
plt.show()

```



Sebelum membangun dan melatih model harus mengulangi *preprocessing* set pelatihan pada set validasi dan pengujian

```
def preprocess_dataset(files):
    files_ds = tf.data.Dataset.from_tensor_slices(files)
    output_ds = files_ds.map(get_waveform_and_label, num_parallel_calls=AUTOTUNE)
    output_ds = output_ds.map(
        get_spectrogram_and_label_id, num_parallel_calls=AUTOTUNE)
    return output_ds
```

```
train_ds = spectrogram_ds
val_ds = preprocess_dataset(val_files)
test_ds = preprocess_dataset(test_files)
```

```
batch_size = 64
train_ds = train_ds.batch(batch_size)
val_ds = val_ds.batch(batch_size)
```

```
train_ds = train_ds.cache().prefetch(AUTOTUNE)
val_ds = val_ds.cache().prefetch(AUTOTUNE)
```

```
for spectrogram, _ in spectrogram_ds.take(1):
    input_shape = spectrogram.shape
    print('Input shape:', input_shape)
    num_labels = len(commands)
    norm_layer = preprocessing.Normalization()
    norm_layer.adapt(spectrogram_ds.map(lambda x, _: x))
    model = models.Sequential([
        layers.Input(shape=input_shape),
        preprocessing.Resizing(32, 32),
        norm_layer,
        layers.Conv2D(32, 3, activation='relu'),
        layers.Conv2D(64, 3, activation='relu'),
        layers.MaxPooling2D(),
        layers.Dropout(0.25),
        layers.Flatten(),
        layers.Dense(128, activation='relu'),
        layers.Dropout(0.5),
        layers.Dense(num_labels),
    ])
```

```

model.compile(
    optimizer=tf.keras.optimizers.Adam(),
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits
=True),
    metrics=['accuracy'],
    \

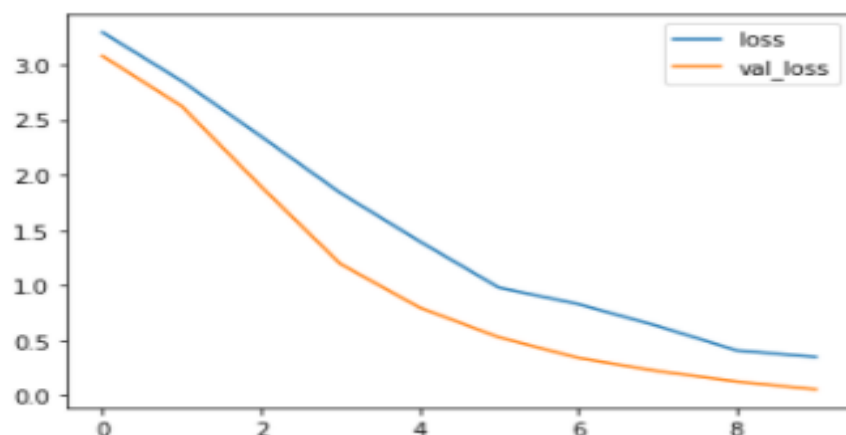
```

```

EPOCHS = 10
history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=EPOCHS,
    callbacks=tf.keras.callbacks.EarlyStopping(verbose=1, patience
=2),

```

Periksa kurva pelatihan dan validasi untuk melihat peningkatan model selama pelatihan



```

metrics = history.history
plt.plot(history.epoch, metrics['loss'], metrics['val_loss'])
plt.legend(['loss', 'val_loss'])
plt.show()

```

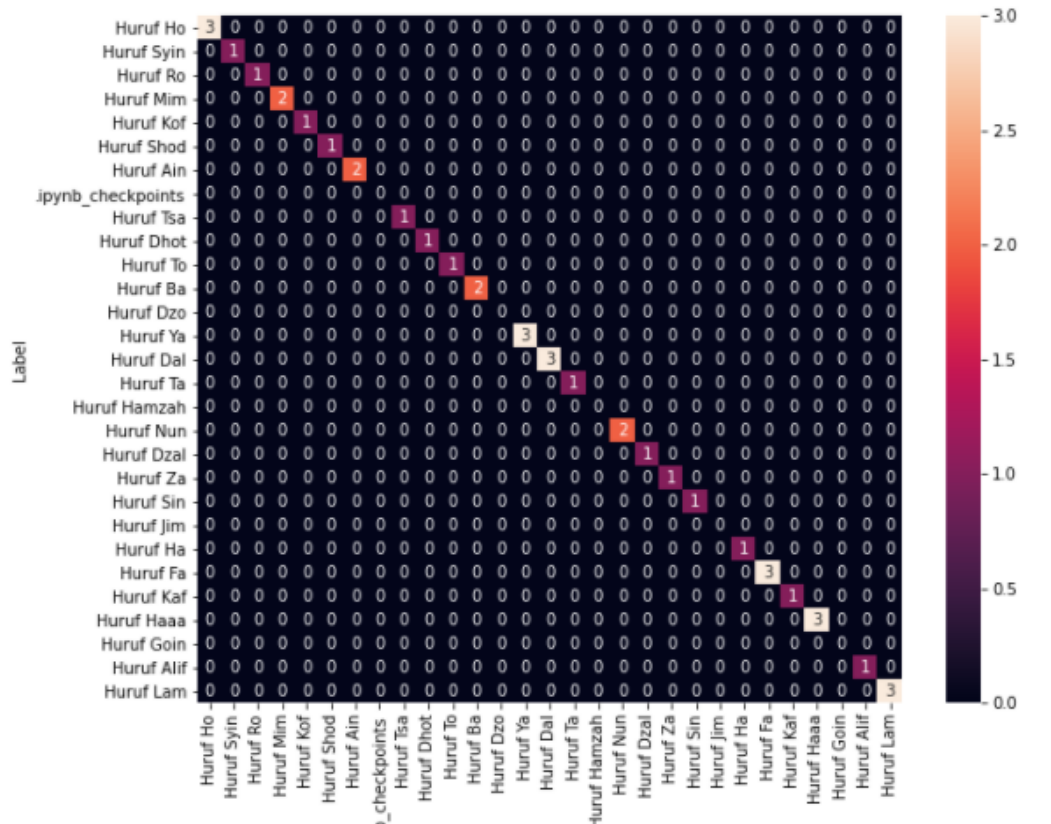
Jalankan model pada set pengujian dan periksa kinerjanya

```
test_audio = []
test_labels = []
for audio, label in test_ds:
    test_audio.append(audio.numpy())
    test_labels.append(label.numpy())
test_audio = np.array(test_audio)
test_labels = np.array(test_labels)
```

```
y_pred = np.argmax(model.predict(test_audio), axis=1)
y_true = test_labels
test_acc = sum(y_pred == y_true) / len(y_true)
print(f'Test set accuracy: {test_acc:.0% }')
```

```
confusion_mtx = tf.math.confusion_matrix(y_true, y_pred)
plt.figure(figsize=(10, 8))
sns.heatmap(confusion_mtx, xticklabels=commands, yticklabels=
=commands,
            annot=True, fmt='g')
plt.xlabel('Prediction')
plt.ylabel('Label')
```

```
confusion_mtx = tf.math.confusion_matrix(y_true, y_pred)
plt.figure(figsize=(10, 8))
sns.heatmap(confusion_mtx, xticklabels=commands, yticklabels=
=commands,
            annot=True, fmt='g')
plt.xlabel('Prediction')
plt.ylabel('Label')
```



## 2. Implementasi Program

### a. Tampilan Splash Screen

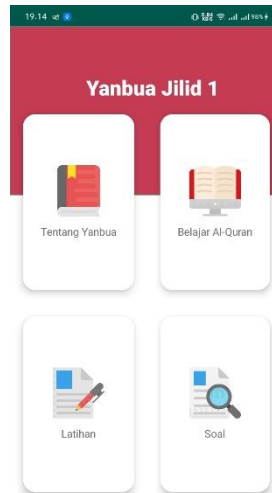


**YANBU'A**

Gambar 4. 8 Tampilan splash screen

b. Tampilan Halaman Utama

Pada halaman utama terdapat empat menu, yaitu menu tentang yanbu'a, menu belajar al-qur'an, menu latihan dan menu soal.



Gambar 4. 9 Tampilan halaman utama

c. Tampilan Menu Tentang Yanbu'a

Pada menu ini berisikan informasi singkat mengenai pengertian metode yanbu'a



Gambar 4. 10 Tampilan menu tentang yanbu'a

#### d. Tampilan Menu Belajar Al-Qur'an

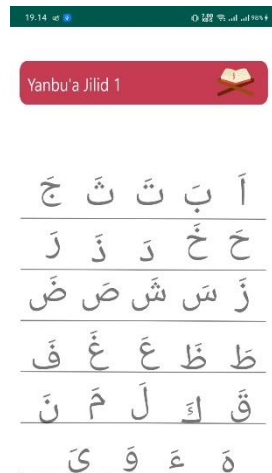
Pada menu ini berisikan informasi betapa pentingnya mempelajari al-Qur'an. Pentingnya belajar Al-Qur'an dijelaskan dalam Al-Qur'an surah Al-Faathir ayat 20



Gambar 4. 11 Tampilan menu belajar Al-Qur'an

e. Tampilan Menu Latihan

Pada menu ini terdapat 29 huruf hijaiyah yang apabila hurufnya ditekan akan mengeluarkan suara. Menu ini menerapkan *text to speech*.



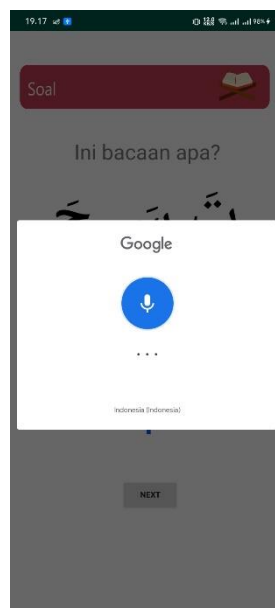
Gambar 4. 12 Tampilan menu latihan

f. Tampilan Menu Soal

Menu ini menampilkan beberapa soal huruf hijaiyah secara acak yang terdiri dari 35 soal yang 6 diantaranya merupakan gabungan tiga huruf hijaiyah.



Gambar 4. 13 Tampilan menu soal



Gambar 4. 14 Tampilan *google speech api*





Gambar 4. 15 Tampilan jawaban salah






Gambar 4. 16 Tampilan jawaban benar

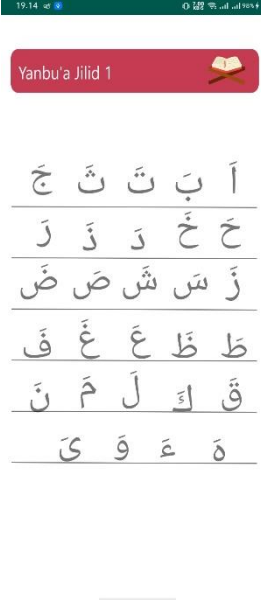

- 4.6.4 Pengujian (Testing)
- Pengujian *Black Box*

Berikut tabel 4.5 merupakan hasil pengujian fungsionalitas menggunakan teknik *black box testing* pada Aplikasi Android Penentuan Level Pembelajaran Yanbu'a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech Api

Tabel 4. 4 Pengujian Black Box

No	Aksi	Hasil	Output
1	Buka aplikasi	<p>Menampilkan tampilan <i>splash screen</i>, lalu langsung menuju halaman utama</p> 	Sesuai

2	Tekan menu tentang yanbu'a	<p>Menampilkan menu tentang yanbu'a</p>  <p>Metode Yanbu'a adalah suatu metode hasil tulis dan mangratil al-Quran untuk mendapatkan saniri tidak boleh mengaja membaca langsung dengan cepat tepat dan tidak putus-putus disesuaikan dengan kaidah maabazul huruf. Timbulnya metode yanbu'a adalah usulan dan dorongan Aswaja Haidak, kashih Yakusli Quran suraja mereka selalu ada hubungan dengan pentak, disamping usulan dan masyarakat baik juga dari Lembaga Pendidikan Masjid serta Musajid seluruhnya dari cabang Kudus dan Jepara.</p>	Sesuai
3	Tekan menu belajar Al-Qur'an	<p>Menampilkan menu belajar Al-Qur'an</p>  <p>Sesungguhnya orang-orang yang selalu membaca kitab Allah dan mengerjakan salat dan menafkahkan kepada mereka dengan diam-diam dan terang-terangan, mereka itu mengharapkan perniagaan yang tidak akan merugi, agar Allah menyempurnakan kepada mereka pahala mereka dan menambah mereka dan karunia-Nya. Sesungguhnya Allah Maha Pengampun lagi Maha Menyukuri. (Baqah:20)</p>	Sesuai
4	Tekan menu Latihan	Menampilkan menu latihan	Sesuai

			
5	Tekan menu Soal		Sesuai

Berdasarkan hasil pengujian fungsional dengan menggunakan teknik *black box testing* pada Aplikasi Android Penentuan Level Pembelajaran Yanbu'a Jilid 1 Untuk Anak Usia Dini Menggunakan Google Speech Api menunjukkan bahwa dari aksi yang dilakukan dengan hasil yang ditampilkan sudah sesuai.

#### b. Pengujian UAT

Untuk mengetahui tanggapan responden (user) terhadap aplikasi pembelajaran yang akan diimplementasikan, maka dilakukan pengujian dengan memberikan 8 pertanyaan kepada responden 10 (santri) dimana jawaban dari pertanyaan tersebut terdiri dari tingkatan yang dapat dipilih sebagai berikut:

Tabel 4. 5 Tabel Komponen Pilihan Jawaban Responden

A	Sangat : Mudah/Bagus/Sesuai/Jelas
B	Mudah/Bagus/Sesuai/Jelas
C	Netral
D	Cukup : Sulit/Bagus/Sesuai/Jelas
E	Sangat : Sulit/Jelek/Tidak Sesuai/Tidak Jelas

Tabel 4. 6 Tabel Komponen Bobot Nilai Responden

Jawaban	Bobot
A. Sangat : Mudah/Bagus/Sesuai/Jelas	5
B. Mudah/Bagus/Sesuai/Jelas	4
C. Netral	3
D. Cukup : Sulit/Bagus/Sesuai/Jelas	2
E. Sangat : Sulit/Jelek/Tidak Sesuai/Tidak Jelas	1

No	Pertanyaan	A	B	C	D	E
1	Apakah tampilan media pembelajaran aplikasi ini menarik?					
2	Apakah menu-menu media pembelajaran aplikasi ini mudah dipahami?					
3	Apakah materi media pembelajaran aplikasi ini mudah dipahami?					

4	Apakah adanya aplikasi Yanbu'a Jilid 1 membantu memahami dalam mengenali huruf hijaiyah?					
5	Apakah aplikasi ini sudah berjalan dengan baik pada perangkat mobile Android?					
6	Apakah aplikasi dapat menampilkan fitur google speech dengan baik?					
7	Apakah media pembelajaran aplikasi ini dapat dijadikan media bantu belajar?					
8	Apakah media pembelajaran aplikasi ini sudah cukup baik?					

Tabel 4. 7 Tabel Pertanyaan Kuisisioner

No	Pertanyaan	Jawaban					Presentase				
		A	B	C	D	E	A	B	C	D	E
1	Apakah tampilan media pembelajaran aplikasi ini menarik?	3	3	3	1	0	30 %	30 %	30 %	10 %	0 %
2	Apakah menu-menu media pembelajaran aplikasi ini mudah dipahami?	4	2	3	1	0	40 %	20 %	10 %	10 %	0 %
3	Apakah materi media pembelajaran aplikasi ini mudah dipahami?	2	3	4	1	0	20 %	30 %	40 %	10 %	0 %
4	Apakah adanya aplikasi Yanbu'a Jilid 1 membantu memahami dalam mengenali huruf hijaiyah?	2	3	1	1	3	20 %	30 %	10 %	10 %	30 %

5	Apakah aplikasi ini sudah berjalan dengan baik pada perangkat mobile Android?	5	2	2	1	0	50 %	20 %	20 %	10 %	0 %
6	Apakah aplikasi dapat menampilkan fitur google speech dengan baik?	2	3	2	2	1	20 %	30 %	20 %	20 %	10 %
7	Apakah media pembelajaran aplikasi ini dapat dijadikan media bantu belajar?	3	2	3	2	0	30 %	20 %	30 %	20 %	0 %
8	Apakah media pembelajaran aplikasi ini sudah cukup baik?	2	2	3	2	1	20 %	20 %	30 %	20 %	10 %

Setelah mendapatkan data di atas, diolah dengan cara mengalikan setiap poin jawaban dengan bobot yang sudah ditentukan sesuai dengan tabel bobot nilai jawaban. Dari hasil perhitungan dengan mengalikan setiap jawaban bobot yang sudah ditentukan maka didapat hasil sebagai berikut

Tabel 4. 8 Tabel Data Kuesioner Santri Setelah Diolah

No	Pertanyaan	Nilai					Jumlah
		Ax5	Bx4	Cx3	Dx2	Ex1	
1	Apakah tampilan media pembelajaran aplikasi ini menarik?	15	12	9	2	0	38
2	Apakah menu-menu media pembelajaran aplikasi ini mudah dipahami?	20	8	9	2	0	39

3	Apakah materi media pembelajaran aplikasi ini mudah dipahami?	10	12	12	2	0	36
4	Apakah adanya aplikasi Yanbu'a Jilid 1 membantu memahami dalam mengenali huruf hijaiyah?	10	12	3	2	3	30
5	Apakah aplikasi ini sudah berjalan dengan baik pada perangkat mobile Android?	25	8	6	2	0	41
6	Apakah aplikasi dapat menampilkan fitur google speech dengan baik?	10	12	6	2	1	31
7	Apakah media pembelajaran aplikasi ini dapat dijadikan media bantu belajar?	15	8	9	4	0	36
8	Apakah media pembelajaran aplikasi ini sudah cukup baik?	10	8	9	4	1	32

- Analisa pertanyaan pertama

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan pertama adalah 38. Nilai rata-ratanya adalah  $38/10 = 3.8$ .

Presentase nilainya adalah  $3.8/5 \times 100 = 76\%$ .

- Analisa pertanyaan kedua

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan kedua adalah 39. Nilai rata-ratanya adalah  $39/10 = 3.9$ .

Presentase nilainya adalah  $3.9/5 \times 100 = 78\%$ .

- Analisa pertanyaan ketiga



Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan ketiga adalah 36. Nilai rata-ratanya adalah  $36/10 = 3.6$ .

Presentase nilainya adalah  $3.6/5 \times 100 = 72\%$ .

- Analisa pertanyaan keempat

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan keempat adalah 30. Nilai rata-ratanya adalah  $30/10 = 3$ .

Presentase nilainya adalah  $3/5 \times 100 = 60\%$ .

- Analisa pertanyaan kelima

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan kelima adalah 41. Nilai rata-ratanya adalah  $41/10 = 4.1$ .

Presentase nilainya adalah  $4.1/5 \times 100 = 82\%$ .

- Analisa pertanyaan keenam

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan keenam adalah 31. Nilai rata-ratanya adalah  $31/10 = 3.1$ .

Presentase nilainya adalah  $3.1/5 \times 100 = 62\%$ .

- Analisa pertanyaan ketujuh

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan ketujuh adalah 36. Nilai rata-ratanya adalah  $36/10 = 3.6$ .

Presentase nilainya  $3.6/5 \times 100 = 72\%$ .

- Analisa pertanyaan kedelapan

Dari tabel di atas dapat dilihat bahwa jumlah nilai dari 10 responden untuk pertanyaan kedelapan adalah 32. Nilai rata-ratanya adalah  $32/10 = 3.2$ .

Presentase nilainya adalah  $3.2/5 \times 100 = 64\%$ .

c. Pengujian Akurasi Implementasi Google Speech

Pengujian Akurasi Implementasi Google Speech diuji oleh Ustadzah Merdina Marsa. Beliau merupakan Ustadzah dari TPQ Ar-rahman yang mengajar jilid 1. Hasil pengujian akurasi implementasi Google Speech ditunjukkan pada tabel

Tabel 4. 9 Pengujian akurasi implementasi Google Speech

Huruf Hijaiyah (Yanbu'a jilid 1)	Pengujian ke		
	1	2	3
Alif	Sesuai	Sesuai	Sesuai
Ba	Tidak sesuai	Sesuai	Sesuai
Tak	Sesuai	Sesuai	Sesuai
Sak	Tidak Sesuai	Tidak sesuai	Sesuai
Jim	Sesuai	Sesuai	Sesuai
Hak	Sesuai	Sesuai	Sesuai
Kho'	Tidak Sesuai	Tidak Sesuai	Sesuai
Dal	Sesuai	Sesuai	Sesuai
Dzal	Tidak sesuai	Tidak sesuai	Sesuai
Ro	Sesuai	Sesuai	Sesuai
Za	Sesuai	Sesuai	Sesuai
Sin	Sesuai	Sesuai	Sesuai
Syin	Sesuai	Sesuai	Sesuai
Sot	Tidak Sesuai	Tidak Sesuai	Sesuai
Dot	Tidak Sesuai	Tidak Sesuai	Sesuai
Tok	Tidak sesuai	Sesuai	Sesuai
Dok	Tidak Sesuai	Tidak Sesuai	Sesuai
Ain	Tidak sesuai	Tidak sesuai	Sesuai
Goin	Tidak Sesuai	Sesuai	Sesuai
Vak	Sesuai	Sesuai	Sesuai
Kof	Tidak Sesuai	Sesuai	Sesuai
Kaf	Sesuai	Sesuai	Sesuai
Lam	Tidak Sesuai	Sesuai	Sesuai
Mim	Sesuai	Sesuai	Sesuai
Nun	Sesuai	Sesuai	Sesuai
Ha	Tidak sesuai	Tidak sesuai	Sesuai
Hamzah	Sesuai	Sesuai	Sesuai

Wau	Sesuai	Sesuai	Sesuai
Ya	Tidak sesuai	Sesuai	Sesuai
Total Hasil *)	51,72%	71,41%	89,65%

\*) Total Hasil = jumlah huruf yang sesuai / jumlah huruf hijaiyah x 100%

Berdasarkan bentuk hasil pengujian pada tabel...Berdasarkan hasil pengujian pertama diperoleh hasil keseluruhan sebesar 51,72%. Hal ini dikarenakan ada beberapa kunci jawaban yang berbeda dengan apa yang diucapkan oleh penguji. Selain itu, Google Speech memiliki kelemahan dalam mengkonversi ucapan ke teks dalam huruf hijaiyah yang tidak ditemukan dalam kamus bahasa Indonesia dan bahasa Arab. Oleh karena itu, apa yang dikatakan penguji tidak selalu sama dengan jawaban di aplikasi. Penelitian ini kemudian memperbaiki kunci jawabannya. Total hasil untuk pengujian kedua adalah 71,41%. Hasil ini akan meningkat dari test pertama, tetapi untuk mendapatkan hasil terbaik, jawabannya telah diperbaiki. Pada percobaan ketiga, hasilnya adalah 100% yang berarti semua huruf memiliki jawaban yang benar berdasarkan suara pemeriksa,

#### 4.6.5 Pemeliharaan (Maintenance)

Pemeliharaan sistem adalah suatu kombinasi dari berbagai tindakan yang dilakukan untuk menjaga suatu sistem atau memperbaikinya sampai suatu kondisi yang bisa diterima.

Berikut maintenance yang telah dilakukan pada aplikasi android penentuan level pembelajaran yanbu'a jilid 1 untuk anak usia dini menggunakan google speech API

Tabel 4. 10 Perbaikan pada Pengujian Akurasi tahap 1

No	Huruf Hijaiyah	Pengujian ke 1	No	Huruf Hijaiyah	Pengujian ke 1
1	Alif	Sesuai	19	Tok	Tidak sesuai
2	Ba	Tidak sesuai	20	Dok	Tidak Sesuai
3	Tak	Sesuai	21	Ain	Tidak sesuai
4	Sak	Tidak Sesuai	22	Goin	Tidak Sesuai
5	Jim	Sesuai	23	Vak	Sesuai
6	Hak	Sesuai	24	Kof	Tidak Sesuai
7	Kho'	Tidak Sesuai	25	Kaf	Sesuai
8	Dal	Sesuai	26	Lam	Tidak Sesuai
9	Dzal	Tidak sesuai	27	Mim	Sesuai
10	Ro	Sesuai	28	Nun	Sesuai
11	Za	Sesuai	29	Ha	Tidak sesuai
12	Sin	Sesuai	30	Hamzah	Sesuai
13	Syim	Sesuai	31	Wau	Sesuai
14	Sot	Tidak Sesuai	32	Ya	Tidak sesuai
15	Dot	Tidak Sesuai	Total Hasil		51,72%

Pada total hasil pengujian ke 1 diperoleh total hasil 51,72%. Hal ini terjadi karena ada beberapa huruf yang memiliki kunci jawaban berbeda dengan apa yang

diucapkan oleh penguji. Maka dilakukan *maintenance* pada kunci jawaban agar nanti pada pengujian ke 2 mendapatkan hasil yang lebih baik.

Tabel 4. 11 Perbaikan pada Pengujian Akurasi tahap 2

No	Huruf Hijaiyah	Pengujian ke 2	No	Huruf Hijaiyah	Pengujian ke 2
1	Alif	Sesuai	19	Tok	Sesuai
2	Ba	Tidak sesuai	20	Dok	Tidak Sesuai
3	Tak	Sesuai	21	Ain	Tidak sesuai
4	Sak	Tidak Sesuai	22	Goin	Sesuai
5	Jim	Sesuai	23	Vak	Sesuai
6	Hak	Sesuai	24	Kof	Sesuai
7	Kho'	Tidak Sesuai	25	Kaf	Sesuai
8	Dal	Sesuai	26	Lam	Sesuai
9	Dzal	Tidak sesuai	27	Mim	Sesuai
10	Ro	Sesuai	28	Nun	Sesuai
11	Za	Sesuai	29	Ha	Tidak sesuai
12	Sin	Sesuai	30	Hamzah	Sesuai
13	Syim	Sesuai	31	Wau	Sesuai
14	Sot	Tidak Sesuai	32	Ya	Sesuai
15	Dot	Tidak Sesuai	Total Hasil		71,41%

Pada total hasil pengujian ke 2 diperoleh total hasil 71,41%. Agar mendapatkan hasil yang lebih maksimal, maka dilakukan *maintenance* kunci jawaban pada tahap ke 3

Tabel 4. 12 Perbaikan pada Pengujian Akurasi tahap 3

No	Huruf Hijaiyah	Pengujian ke 3	No	Huruf Hijaiyah	Pengujian ke 3
1	Alif	Sesuai	19	Tok	Sesuai
2	Ba	Sesuai	20	Dok	Sesuai

3	Tak	Sesuai	21	Ain	Sesuai
4	Sak	Sesuai	22	Goin	Sesuai
5	Jim	Sesuai	23	Vak	Sesuai
6	Hak	Sesuai	24	Kof	Sesuai
7	Kho'	Sesuai	25	Kaf	Sesuai
8	Dal	Sesuai	26	Lam	Sesuai
9	Dzal	Sesuai	27	Mim	Sesuai
10	Ro	Sesuai	28	Nun	Sesuai
11	Za	Sesuai	29	Ha	Sesuai
12	Sin	Sesuai	30	Hamzah	Sesuai
13	Syim	Sesuai	31	Wau	Sesuai
14	Sot	Sesuai	32	Ya	Sesuai
15	Dot	Sesuai	Total Hasil		100 %

Pada total hasil pengujian ke 3 diperoleh total hasil 100%. Hasil tersebut jauh lebih baik daripada hasil pengujian ke 1 dan ke 2. *Google speech* memiliki kelemahan dalam mengkonversi suara ke teks huruf-huruf hijaiyah yang tidak ada dalam kamus bahasa indonesia maupun bahasa arab.

#### 4.7 Hasil dan Pembahasan

Dari penelitian yang telah dilakukan diperoleh hasil dan pembahasan sebagai berikut :

1. Pada tahap desain menjelaskan tentang proses *speech to text* dengan menggunakan flowchart
2. Tahap implementasi ada 2 macam, yaitu tahap analisis sistem dan implementasi program. Pada analisis sistem menjelaskan tentang perhitungan manual dari proses *speech recognition* yang dimulai dari proses mengganti tipe file, proses akuisisi suara, ekstraksi ciri hingga pencocokan pola. Sedangkan pada implementasi program menjelaskan

tentang awal pembuatan aplikasi android hingga pembuatan program *speech recognition* di python.

3. Pengujian dilakukan 3 tahap, yaitu pengujian Black box, UAT dan pengujian akurasi oleh ahli. Berdasarkan hasil pengujian pertama diperoleh hasil keseluruhan sebesar 51,42%. Total hasil untuk pengujian kedua adalah 71,42%. Pada percobaan ketiga, hasilnya adalah 100% yang berarti semua huruf memiliki jawaban yang benar berdasarkan suara pemeriksa. *Black box testing* memperoleh nilai 100% sedangkan pengujian UAT mendapatkan hasil 70,75% dari hasil analisa yang berjumlah 8
4. Pada tahap pemeliharaan dilakukan 3 tahap karena ada beberapa yang harus diperbaiki terutama pada pengujian akurasi *google speech API*. *Google speech* memiliki kelemahan dalam mengkonversi suara ke teks huruf-huruf hijaiyah yang tidak ada dalam kamus bahasa indonesia maupun bahasa arab.

#### **4.8 Kesimpulan dan Saran**

##### a. Kesimpulan

Dari penelitian yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Perancangan aplikasi android dengan *speech recognition* pada penelitian ini dilakukan dengan 6 tahap, yaitu identifikasi masalah, studi pustaka, pengumpulan data metode pengembangan sistem, hasil pembahasan dan analisa dan kesimpulan
2. Pengembangan aplikasi android dengan *speech recognition* pada penelitian ini menggunakan metode pengembangan waterfall, yang terdiri dari analisa kebutuhan, desain, implementasi, pengujian dan pemeliharaan.
3. Pengujian dilakukan 3 tahap, yaitu pengujian Black box, UAT dan pengujian akurasi oleh ahli. Berdasarkan hasil pengujian pertama diperoleh hasil keseluruhan sebesar 51,42%. Total hasil untuk pengujian kedua adalah

71,42%. Pada percobaan ketiga, hasilnya adalah 100% yang berarti semua huruf memiliki jawaban yang benar berdasarkan suara pemeriksa. *Black box testing* memperoleh nilai 100% sedangkan pengujian UAT mendapatkan hasil 70,75% dari hasil analisa yang berjumlah 8

b. Saran

Terdapat beberapa saran yang perlu disampaikan berdasarkan hasil pembahasan pada penelitian ini yaitu:

1. Aplikasi masih memanfaatkan Google Speech API sebagai media input suara, maka untuk penelitian selanjutnya dapat menciptakan inputan suara yang lain tanpa harus menggunakan Google Speech API.
2. Diharapkan dapat menambah huruf hijaiyah yang lebih lengkap misal jilid 2 atau seterusnya



## **BAB 5. KESIMPULAN DAN SARAN**

### **5.1 Kesimpulan**

Dari penelitian yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Perancangan aplikasi android dengan speech recognition pada penelitian ini dilakukan dengan 6 tahap, yaitu identifikasi masalah, studi pustaka, pengumpulan data metode pengembangan sistem, hasil pembahasan dan analisa dan kesimpulan
2. Pengembangan aplikasi android dengan speech recognition pada penelitian ini menggunakan metode pengembangan waterfall, yang terdiri dari analisa kebutuhan, desain, implementasi, pengujian dan pemeliharaan.
3. Pengujian dilakukan 3 tahap, yaitu pengujian Black box, UAT dan pengujian akurasi oleh ahli. Berdasarkan hasil pengujian pertama diperoleh hasil keseluruhan sebesar 51,42%. Total hasil untuk pengujian kedua adalah 71,42%. Pada percobaan ketiga, hasilnya adalah 100% yang berarti semua huruf memiliki jawaban yang benar berdasarkan suara pemeriksa. *Black box testing* memperoleh nilai 100% sedangkan pengujian UAT mendapatkan hasil 70,75% dari hasil analisa yang berjumlah 8

### **5.2 Saran**

Terdapat beberapa saran yang perlu disampaikan berdasarkan hasil pembahasan pada penelitian ini yaitu:

3. Aplikasi masih memanfaatkan Google Speech API sebagai media input suara, maka untuk penelitian selanjutnya dapat menciptakan inputan suara yang lain tanpa harus menggunakan Google Speech API.
4. Diharapkan dapat menambah huruf hijaiyah yang lebih lengkap misal jilid 2 atau seterusnya

## DAFTAR PUSTAKA

- Gapar, I. K. K., Widagda, I. G. A., & Suarbawa, K. N. (2020). *Human Voice Recognition by Using Hebb Artificial Neural Network Method Pengenalan Suara Manusia dengan Menggunakan Metode Jaringan Syaraf Tiruan Hebb Human Voice Recognition by Using Hebb Artificial Neural Network Method. March*. <https://doi.org/10.24843/BF.2018.v19.i01.p04>
- Izzudin, F. (2017). *Pengenalan Suara Bacaan Ayat Al-Qur'an Menggunakan Model WAVENET*.
- Jaya, M. T. S., Puspitaningrum, D., & Susilo, B. (2016). *Penerapan Speech Recognition Pada Permainan Teka-Teki Silang Menggunakan Metode Hidden Markov Model ( Hmm ) Berbasis Desktop*. 4(1), 119–129.
- Mata, P., Qur, P. A.-, Hadist, A. N., Mts, D. I., & Malang, A. D. K. (2015). *Aplikasi Metode Yanbu'a Dalam Meningkatkan Kefasihan Dan Kelancaran Baca Siswa Kelas Vii A Pada Mata Pelajaran Al-Qur'an Hadist Di Mts Al-Hidayah Donowarih Kabupaten Malang*.
- Mulyani, D., Pamungkas, I., & Inten, D. N. (2018). *Jurnal Obsesi : Jurnal Pendidikan Anak Usia Dini Al-Quran Literacy for Early Childhood with Storytelling Techniques*. 2(2), 202–210. <https://doi.org/10.31004/obsesi.v2i2.72>
- Purba. (2019) Mata, P., Qur, P. A.-, Hadist, A. N., Mts, D. I., & Malang, A. D. K. (2015). *Aplikasi Metode Yanbu'a Dalam Meningkatkan Kefasihan Dan Kelancaran Baca Siswa Kelas Vii A Pada Mata Pelajaran Al-Qur'an Hadist Di Mts Al-Hidayah Donowarih Kabupaten Malang*.
- Ramadhan, Z., & Endah, S. N. (2016). *Perintah Suara Berbahasa Indonesia untuk Membuka dan Menutup Aplikasi dalam Sistem Operasi Windows Menggunakan Metode Mel Frequency Cepstrum Coefficient dan Metode Backpropagation*. Snik, 33–41.

- Safaat, N (2012). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika
- Shinwani, M. F. (2016). *Rancang Bangun Aplikasi Voice Translator Berbasis Android Menggunakan Hidden Markov Model*
- Studi, P., Fisika, S., Surabaya, U. N., & Noortjahja, D. A. (2015). *Implementasi Hidden Markov Models ( Hmm ) Sebagai Filter Untuk Mereduksi Noise Pada Esophageal Speech Aprilia Fitriatul Aisyah. 04.*
- Suriah, M. (n.d.). *Metode Yanbu ' A Untuk Meningkatkan Kemampuan Membaca Al-Quran Pada Kelompok B-2 RA Permata Hati Al-Mahalli Bantul. 3(November 2018), 291–299.*

## LAMPIRAN

Lampiran 1. Form Pengujian Akurasi Implementasi Google Speech

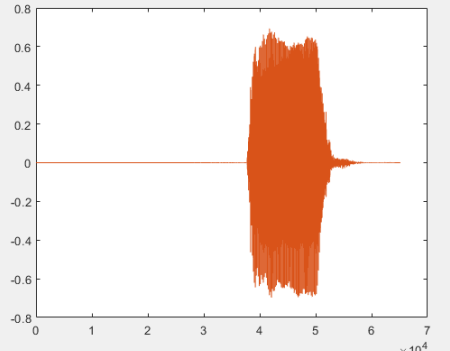
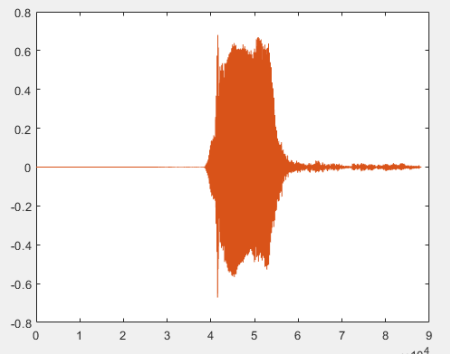
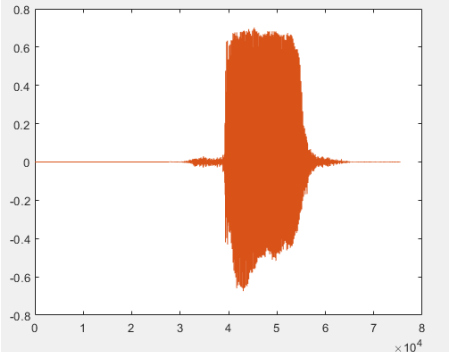
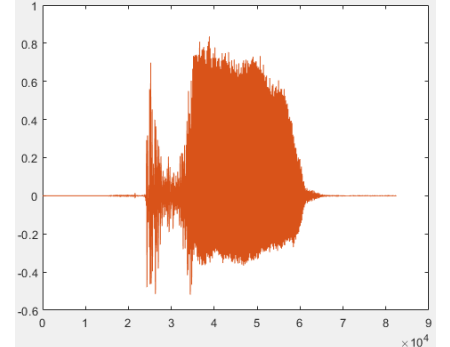
Huruf Hijaiyah (Yanbu'a jilid 1)	Pengujian ke		
	1	2	3
Alif	✓	✓	✓
Ba	TS	✓	✓
Tak	✓	✓	✓
Sak	TS	TS	✓
Jim	✓	✓	✓
Hak	✓	✓	✓
Kho'	TS	TS	✓
Dal	✓	✓	✓
Dzal	TS	TS	✓
Ro	✓	✓	✓
Za	✓	✓	✓
Sin	✓	✓	✓
Syim	✓	✓	✓
Sot	TS	TS	✓
Dot	TS	TS	✓
Tok	TS	✓	✓
Dok	TS	TS	✓
Ain	TS	TS	✓
Goin	TS	✓	✓
Vak	✓	✓	✓
Kof	TS	✓	✓

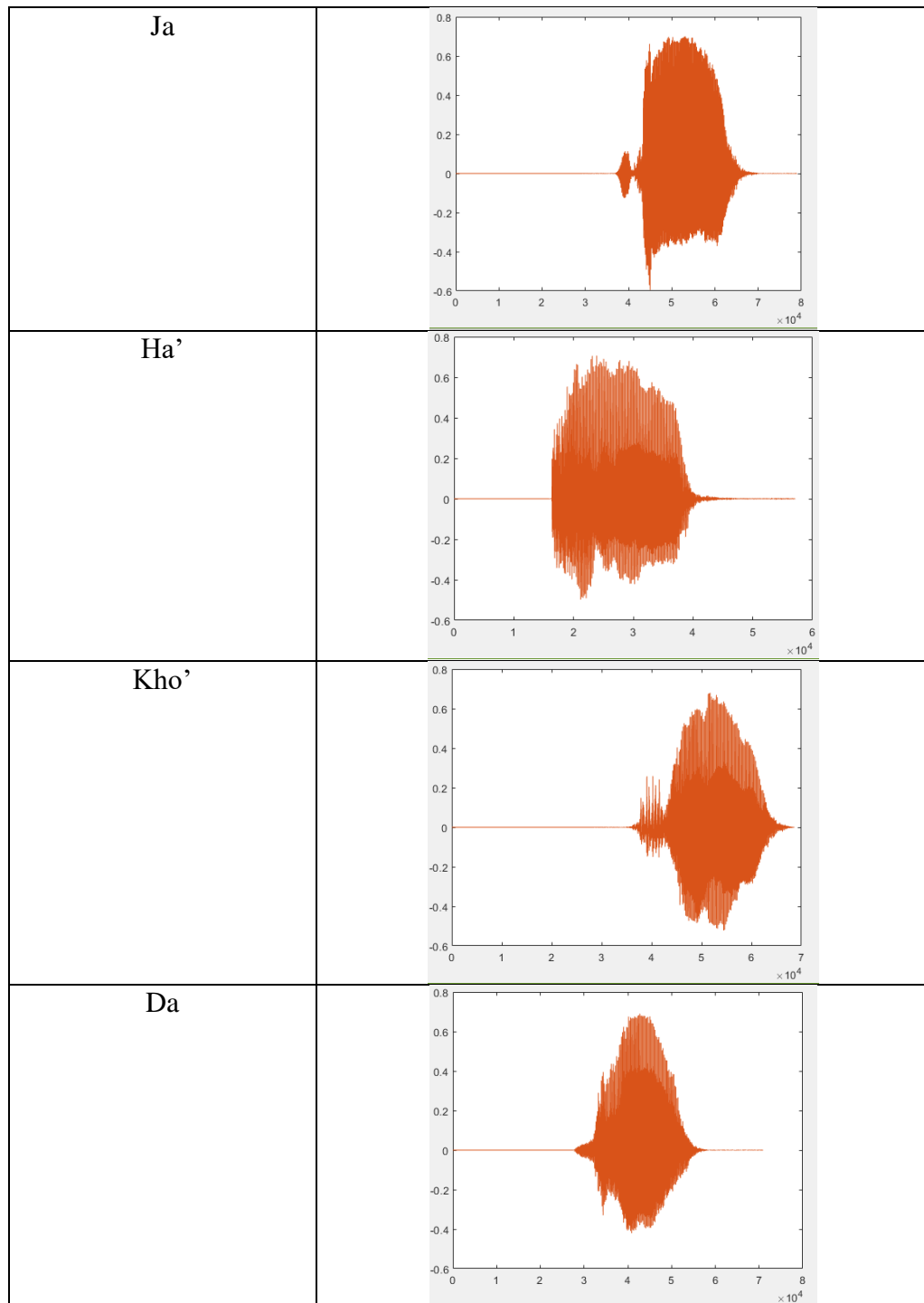
Kaf	5	5	5
Lam	TS	5	5
Mim	5	5	5
Nun	5	5	5
Ha	TS	TS	5
Hamzah	5	5	5
Wau	5	5	5
Ya	TS	5	5
Total Hasil *)	51,72%	71,41%	89,65%

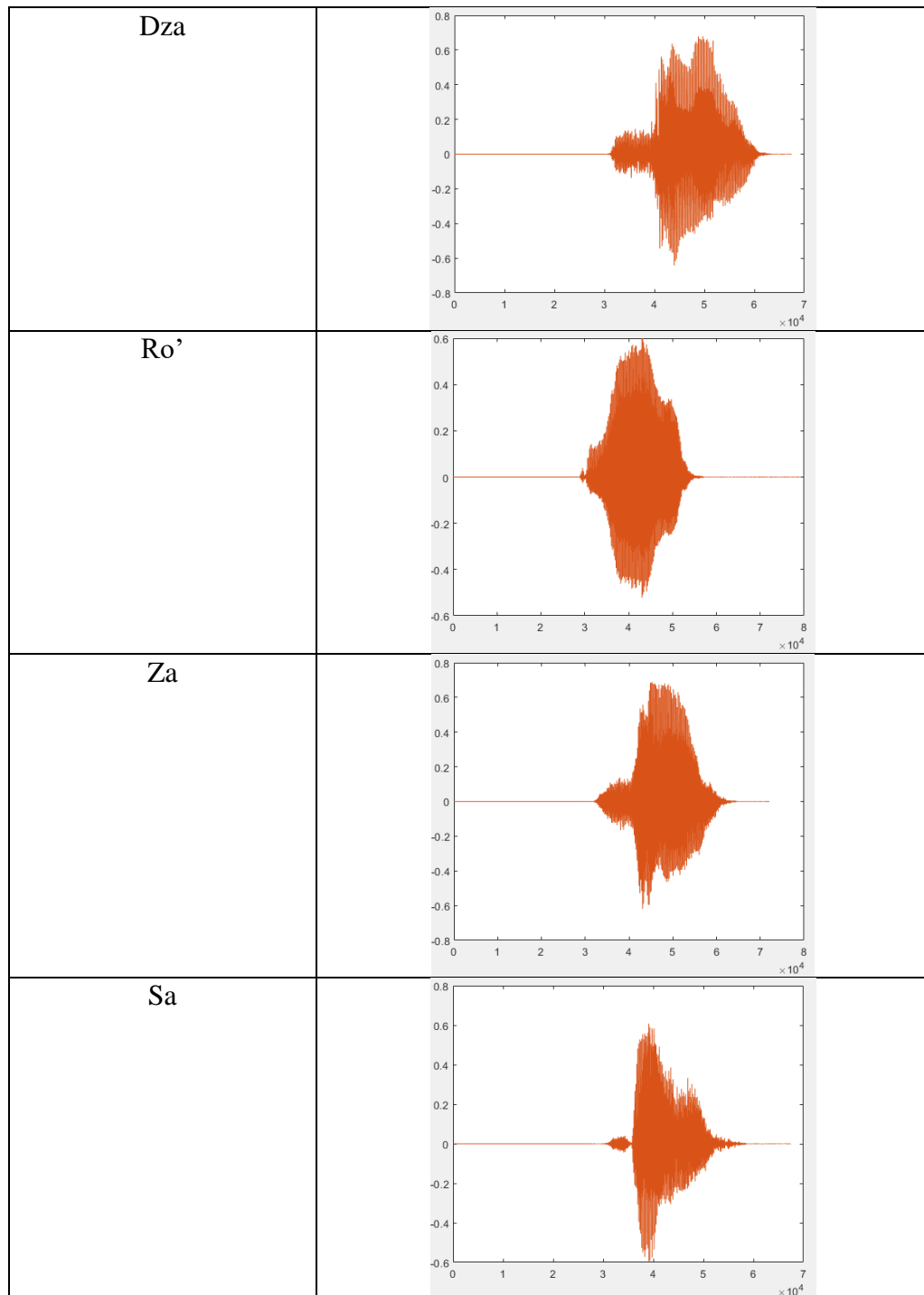
Catatan:

*M. Qul*  
merdin

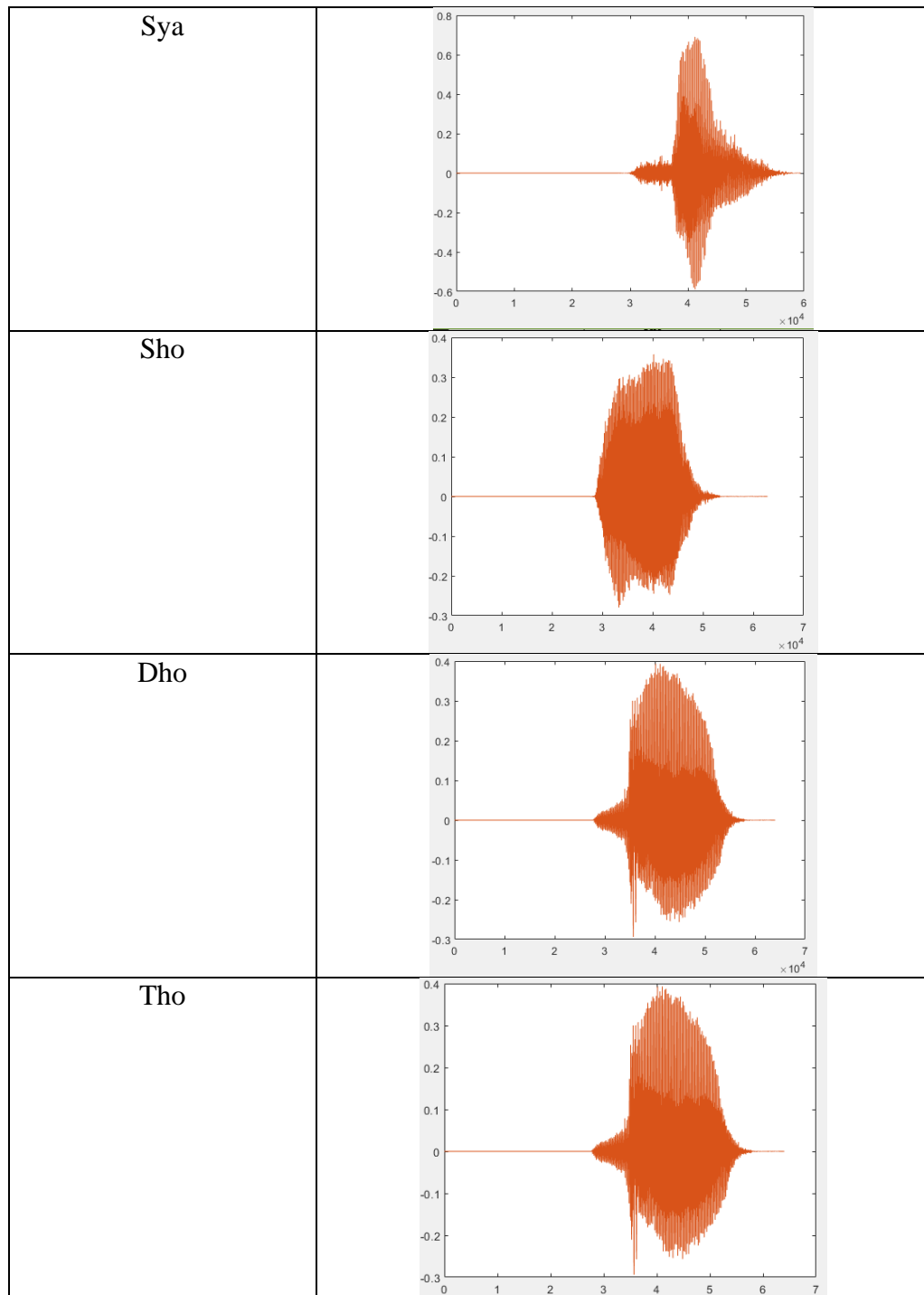
## Lampiran 2. Data sinyal suara

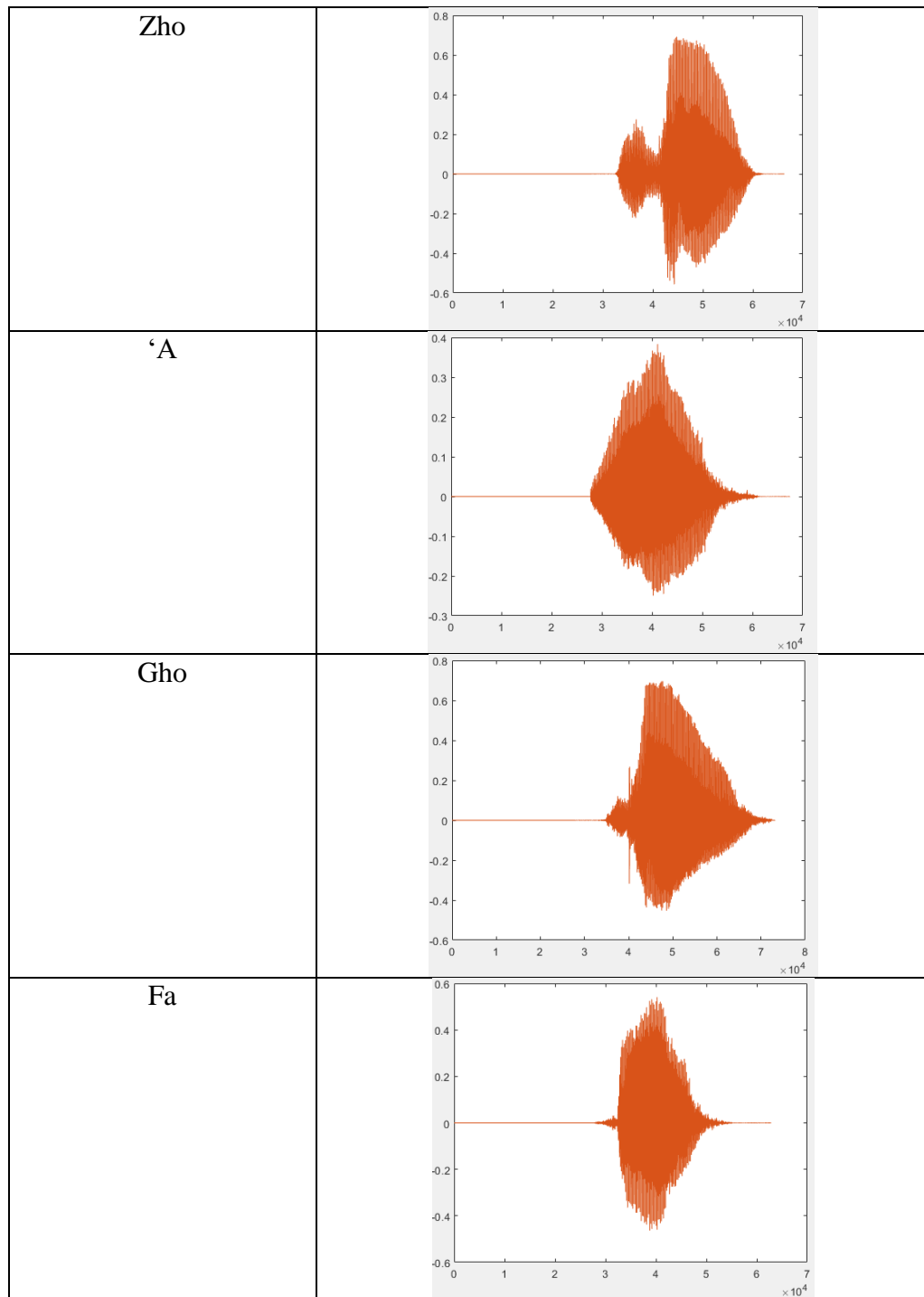
Huruf	Data Sinyal Suara
A	 <p>The plot shows a signal with a duration of approximately 7 x 10^4 samples. The amplitude ranges from -0.8 to 0.8. The signal is mostly zero, with a significant burst of activity between 4 x 10^4 and 6 x 10^4 samples, reaching a peak amplitude of about 0.7.</p>
Ba	 <p>The plot shows a signal with a duration of approximately 9 x 10^4 samples. The amplitude ranges from -0.8 to 0.8. The signal is mostly zero, with a significant burst of activity between 4 x 10^4 and 6 x 10^4 samples, reaching a peak amplitude of about 0.7.</p>
Ta	 <p>The plot shows a signal with a duration of approximately 8 x 10^4 samples. The amplitude ranges from -0.8 to 0.8. The signal is mostly zero, with a significant burst of activity between 4 x 10^4 and 6 x 10^4 samples, reaching a peak amplitude of about 0.7.</p>
Tsa	 <p>The plot shows a signal with a duration of approximately 9 x 10^4 samples. The amplitude ranges from -0.6 to 1.0. The signal is mostly zero, with a significant burst of activity between 2 x 10^4 and 6 x 10^4 samples, reaching a peak amplitude of about 0.8.</p>

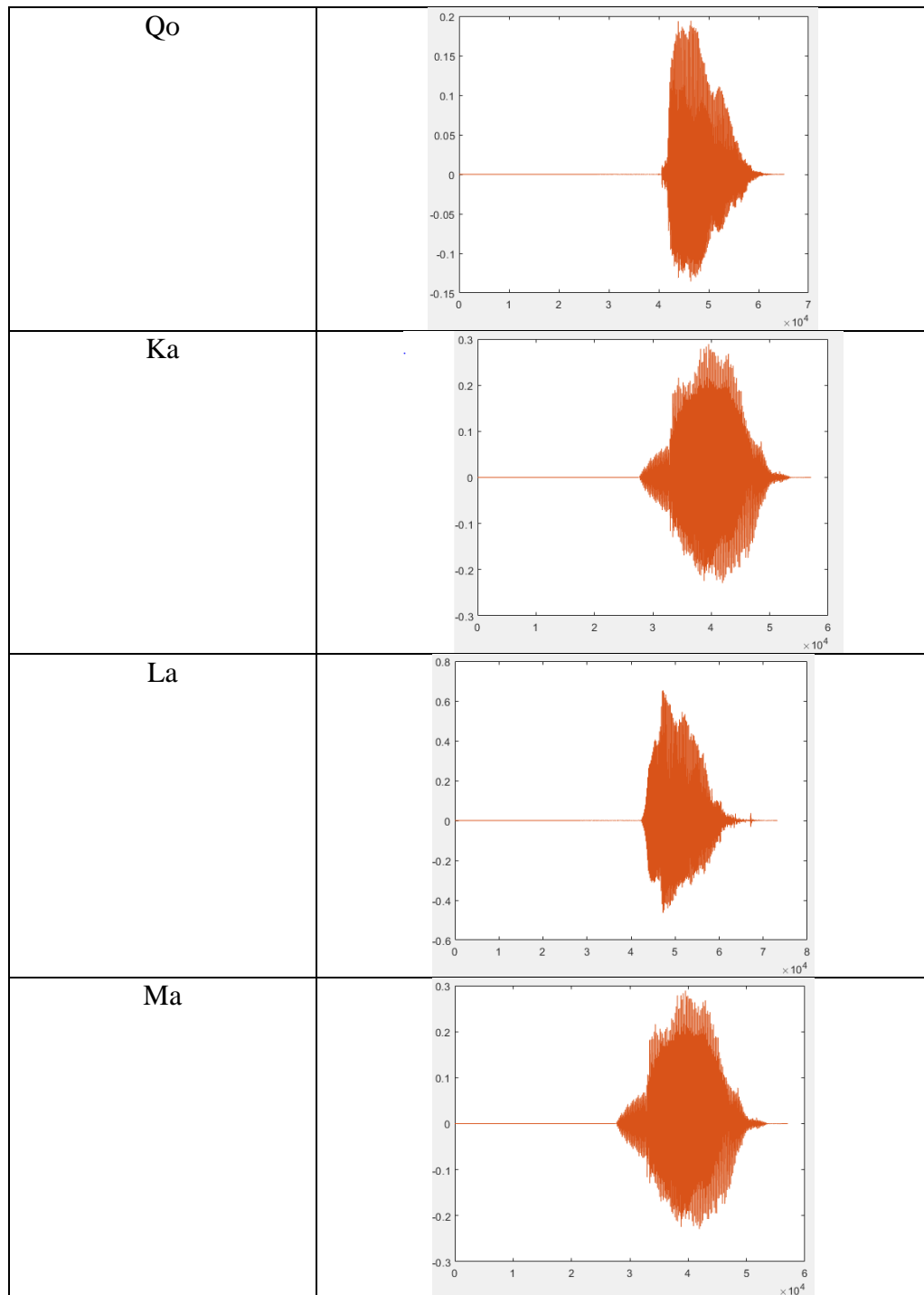


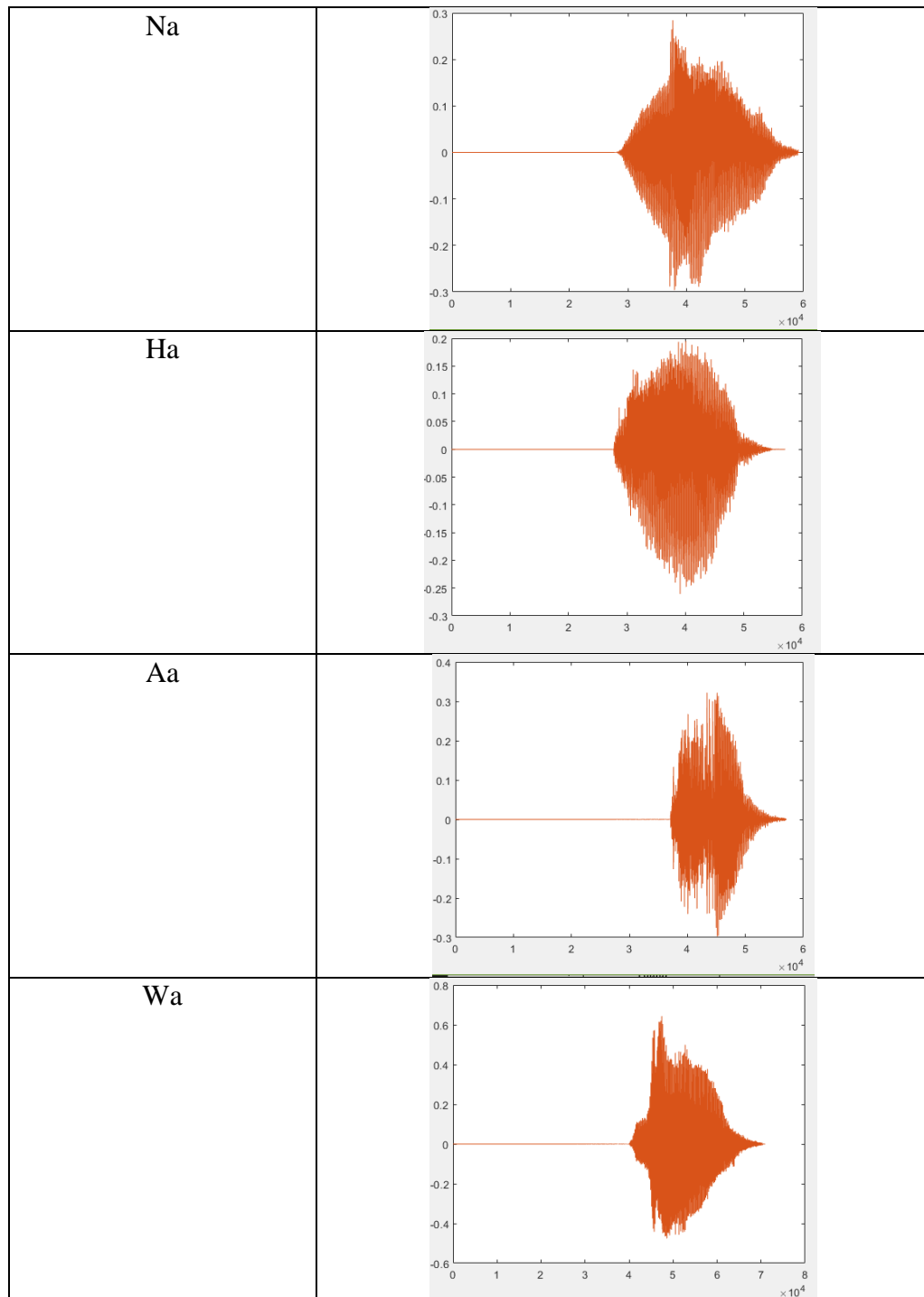


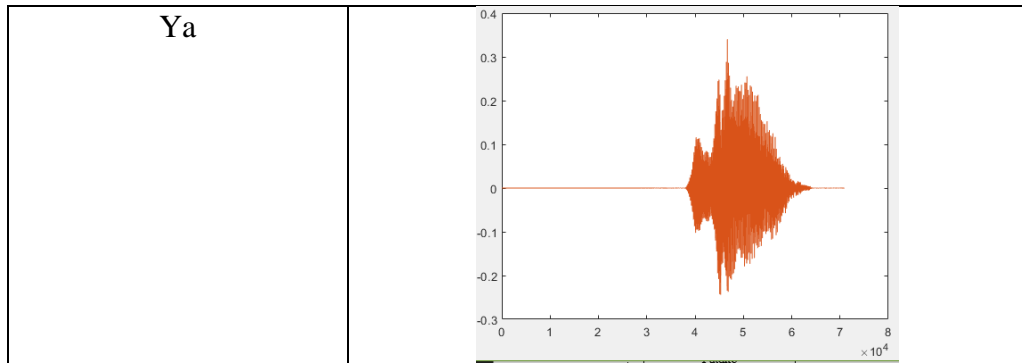












### Lampiran 3. Kode program android

#### 1. Main Activity.java

```

package com.example.cobanimasi;

import android.content.Intent;
import android.os.Handler;
import android.speech.RecognizerIntent;
import android.support.annotation.Nullable;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.Locale;
import java.util.Random;

public class MainActivity extends AppCompatActivity {

    private static final int SPLASH_SCREEN = 3000;
    private static final int REQUEST_CODE_SPEECH_INPUT = 1000 ;

    //Variables
    Animation topAnim, bottomAnim;
    ImageView image;
    TextView logo, mTextTv, mTextHasil, mtextQuiz, mtextQuizA;
    Button btnNext;
    ImageButton mVoiceButton;

    private Quiz mQuiz = new Quiz();

    private int mQuizLenght = mQuiz.mKunci_id.length;

    Random random;
    // String hasil = "";

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
WindowManager.LayoutParams.FLAG_FULLSCREEN);
    setContentView(R.layout.activity_main);

    //Animations
    topAnim =
AnimationUtils.loadAnimation(this,R.anim.top_animation);
    bottomAnim =
AnimationUtils.loadAnimation(this,R.anim.bottom_animation);

    //Hooks
    image = findViewById(R.id.imageView);
    logo = findViewById(R.id.textView);

    image.setAnimation(topAnim);
    logo.setAnimation(bottomAnim);

    new Handler().postDelayed(new Runnable() {
        @Override
        public void run() {
            Intent intent = new Intent(MainActivity.this,Dashboard.class);
            startActivity(intent);
            finish();
        }
    },SPLASH_SCREEN) ;

}
}

```

## 2. Dashboard.java

```

package com.example.cobanimasi;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.CardView;
import android.view.View;

```

```

public class Dashboard extends AppCompatActivity implements
View.OnClickListener {
    private CardView tentang_yanbua, belajar, latihan, soal;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dashboard);
//    Definisi Cardview
        tentang_yanbua = (CardView) findViewById(R.id.tentang_yanbua);
        belajar = (CardView) findViewById(R.id.belajar);
        latihan = (CardView) findViewById(R.id.latihan);
        soal = (CardView) findViewById(R.id.soal);
//    Menambahkan ClickListener ke CardView
        tentang_yanbua.setOnClickListener(this);
        belajar.setOnClickListener(this);
        latihan.setOnClickListener(this);
        soal.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        Intent i ;
        switch (v.getId()) {
            case R.id.tentang_yanbua : i = new Intent(this,
TentangYanbua.class); startActivity(i); break;
            case R.id.belajar : i = new Intent(this, Belajar.class);
startActivity(i); break;
            case R.id.latihan : i = new Intent(this, Latihan.class);
startActivity(i); break;
            case R.id.soal : i = new Intent(this, Soal.class); startActivity(i);
break;
            default:break;
        }
    }
}

```

### 3. TentangYanbua.java

```

package com.example.cobanimasi;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;

```



```

public class TentangYanbua extends AppCompatActivity {
    ImageView yanbua;
    TextView txtyanbua;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tentang_yanbua);
    }
}

```

#### 4. Belajar.java

```

package com.example.cobanimasi;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class Belajar extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_belajar);
    }
}

```

#### 5. Latihan.java

```

package com.example.cobanimasi;

import android.media.MediaPlayer;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

public class Latihan extends AppCompatActivity {
    private TextView ring1, ring2, ring3, ring4, ring5, ring6, ring7, ring8,
    ring9, ring10, ring11, ring12, ring13, ring14, ring15, ring16, ring17,
    ring18, ring19, ring20, ring21, ring22, ring23, ring24, ring25, ring26,
    ring27, ring28, ring29;

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_latihan);
    ring2 = findViewById(R.id.ring2);
    ring3 = findViewById(R.id.ring3);
    ring4 = findViewById(R.id.ring4);
    ring5 = findViewById(R.id.ring5);
    ring7 = findViewById(R.id.ring7);
    ring8 = findViewById(R.id.ring8);
    ring10 = findViewById(R.id.ring10);
    ring11 = findViewById(R.id.ring11);
    ring12 = findViewById(R.id.ring12);
    ring13 = findViewById(R.id.ring13);
    ring16 = findViewById(R.id.ring16);
    ring17 = findViewById(R.id.ring17);
    ring18 = findViewById(R.id.ring18);
    ring19 = findViewById(R.id.ring19);
    ring21 = findViewById(R.id.ring21);
    ring23 = findViewById(R.id.ring23);
    ring24 = findViewById(R.id.ring24);
    ring25 = findViewById(R.id.ring25);
    ring28 = findViewById(R.id.ring28);
    ring29 = findViewById(R.id.ring29);

    final MediaPlayer mdring2 = MediaPlayer.create(this, R.raw.ba);
    final MediaPlayer mdring3 = MediaPlayer.create(this, R.raw.ta);
    final MediaPlayer mdring4 = MediaPlayer.create(this, R.raw.sa);
    final MediaPlayer mdring5 = MediaPlayer.create(this, R.raw.ja);
    final MediaPlayer mdring7 = MediaPlayer.create(this, R.raw.ho);
    final MediaPlayer mdring8 = MediaPlayer.create(this, R.raw.da);
    final MediaPlayer mdring10= MediaPlayer.create(this, R.raw.ro);
    final MediaPlayer mdring11= MediaPlayer.create(this, R.raw.za);
    final MediaPlayer mdring12= MediaPlayer.create(this, R.raw.sa);
    final MediaPlayer mdring13= MediaPlayer.create(this,
R.raw.sya);
    final MediaPlayer mdring16= MediaPlayer.create(this, R.raw.to);
    final MediaPlayer mdring17= MediaPlayer.create(this,
R.raw.dzo);
    final MediaPlayer mdring18= MediaPlayer.create(this, R.raw.al);
    final MediaPlayer mdring19= MediaPlayer.create(this, R.raw.go);
    final MediaPlayer mdring21= MediaPlayer.create(this, R.raw.ko);
    final MediaPlayer mdring23= MediaPlayer.create(this, R.raw.la);
    final MediaPlayer mdring24= MediaPlayer.create(this, R.raw.ma);
    final MediaPlayer mdring25= MediaPlayer.create(this, R.raw.na);
    final MediaPlayer mdring28= MediaPlayer.create(this, R.raw.wa);
    final MediaPlayer mdring29= MediaPlayer.create(this, R.raw.ya);

```

```
ring2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring2.start();
    }
});
ring3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring3.start();
    }
});
ring4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring4.start();
    }
});
ring5.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring5.start();
    }
});
ring7.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring7.start();
    }
});
ring8.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring8.start();
    }
});
ring10.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring10.start();
    }
});
ring11.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
```

```
        mdring11.start();
    }
});
ring12.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring12.start();
    }
});
ring13.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring13.start();
    }
});
ring16.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring16.start();
    }
});
ring17.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring17.start();
    }
});
ring18.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring18.start();
    }
});
ring19.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring19.start();
    }
});
ring21.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        mdring21.start();
    }
});
ring23.setOnClickListener(new View.OnClickListener() {
```

```

        @Override
        public void onClick(View v) {
            mdring23.start();
        }
    });
    ring24.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mdring24.start();
        }
    });
    ring25.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mdring25.start();
        }
    });
    ring28.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mdring28.start();
        }
    });
    ring29.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            mdring29.start();
        }
    });
}
}

```

## 6. Quiz.java

```

package com.example.cobanimasi;

public class Quiz {

    public String mKunci_id[] = {
        "a" //1
        "ba" //2
        "ta" //3
        "sa" //4
        "ja" //5
        "ha" //6
        "ho" //7
    }
}

```

```

"da",//8
"da",//9
"ro",//10
"za",//11
"sa",//12
"sya",//13
"so",//14
"do",//15
"to",//16
"do",//17
"a",//18
"go",//19
"fa",//20
"ko",//21
"ka",//22
"la",//23
"ma",//24
"na",//25
"ha",//26
"a",//27
"wa",//28
"ya",//29
"tatata",//30
"tajata",//31
"ababa",//32
"fatako",//33
"toroko",//34
"gofaro",//35
};

```

```

public String mKunci_ar[]={
    "أ",//1
    "ب",//2
    "ت",//3
    "ث",//4
    "ج",//5
    "ح",//6
    "خ",//7
    "د",//8
    "ذ",//9
    "ر",//10
    "ز",//11
    "س",//12
    "ش",//13
    "ص",//14
    "ض",//15

```

```

        "ط" //16
        "ظ" //17
        "ع" //18
        "غ" //19
        "ف" //20
        "ق" //22
        "ل" //23
        "م" //24
        "ن" //25
        "ة" //26
        "ء" //27
        "و" //28
        "ي" //29
        "ت ت ت" //30
        "ب ب أ" //31
        "ج س ت" //32
        "ق ت ف" //33
        "ق ر ظ" //34
        "ز ف ع" //35
    };

    public String getKunci_id(int i){
        String kunci_id = mKunci_id[i];
        return kunci_id;
    }

    public String getKunci_ar(int i){
        String kunci_ar = mKunci_ar[i];
        return kunci_ar;
    }
}

```

## 7. Soal.java

```

package com.example.cobanimasi;

import android.content.Intent;
import android.speech.RecognizerIntent;
import android.support.annotation.Nullable;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;

```

```

import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.Locale;
import java.util.Random;

public class Soal extends AppCompatActivity {

    private static final int REQUEST_CODE_SPEECH_INPUT =1000;

    //views from activity
    TextView mTextTv, mTextHasil, mtextQuiz, mtextQuizA;
    EditText inputVoice;
    Button btnNext;
    ImageButton mVoiceButton;

    private Quiz mQuiz = new Quiz();
    private String mAnswer;
    private int mQuizLenght = mQuiz.mKunci_id.length;

    private String[] quiz;
    private String[] quiz_arab;
    Random random;
    // String hasil = "";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_soal);

        random= new Random();

        mTextTv = findViewById(R.id.textTv);
        mTextHasil = findViewById(R.id.textHasil2);
        mtextQuiz = findViewById(R.id.textQuiz);
        mtextQuizA = findViewById(R.id.textQuizA);
        btnNext = findViewById(R.id.btnNext);
        mVoiceButton = findViewById(R.id.voiceBtn);

        mVoiceButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                speak();
            }
        });
    }
}

```



```

    }
  });

  btnNext.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
      startActivity(new Intent(getApplicationContext(), Soal.class));
    }
  });
  updateQuiz(random.nextInt(mQuizLenght));
}

private void updateQuiz (int num) {
  mtextQuiz.setText(mQuiz.getKunci_id(num));
  mtextQuizA.setText(mQuiz.getKunci_ar(num));
}

private void speak(){
  Intent intent = new Intent
(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);

  intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,Recogni
zerIntent.LANGUAGE_MODEL_FREE_FORM);
  //    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, "ar-SA");
  intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
Locale.getDefault());
  intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "Hi speak
something" );

  //start intent
  try {
    //in there was no error
    //show dialog
    startActivityForResult(intent,
REQUEST_CODE_SPEECH_INPUT);
  }
  catch (Exception e) {
    //if there was some error
    //get message of error and show
    Toast.makeText(this, ""+e.getMessage(),
Toast.LENGTH_SHORT).show();
  }
}
}

```

```

//receive voice input and handle it

@Override
protected void onActivityResult (int requestCode, int resultCode,
@Nullable Intent data){
    super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode){
        case REQUEST_CODE_SPEECH_INPUT:{
            if (resultCode == RESULT_OK && null!=data) {
                //get text array from voice intent
                ArrayList<String> result =
data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                //set to text view

                mTextTv.setText(result.get(0));
                // GANTI DISINI =====
                String hasil = mtextQuiz.getText().toString();
                if (mTextTv.getText().toString().compareTo(hasil) == 0){
                    Log.e("hasil sukses", mTextTv.getText().toString() );
                    mTextHasil.setText("Benar");
                }else {
                    mTextHasil.setText("Salah");
                    Log.e("hasil gagal", mTextTv.getText().toString() );
                }
            }
        }
        break;
    }
}
}
}

```