

# IMPLEMENTASI GOOGLE SPEECH API PADA APLIKASI KOREKSI HAFALAN AL-QUR'AN BERBASIS ANDROID

(The Implementation of the Google Speech on Qur'an Recitation Correction Application Based on Android)

Affandy Akbar, Ario Yudo Husodo\*, Ariyan Zubaidi  
Program Studi Teknik Informatika, Fakultas Teknik, Universitas Mataram  
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA  
Email: affanalafasy@gmail.com, ario@ti.ftunram.ac.id, zubaidi13@unram.ac.id

## Abstract

*Muroja'ah is a method of repeating new memorization and old memorization that be heard to other people. This method is very popular in Indonesia. In order to make the learning process isn't boring, it is required a learning media that can be used anytime, one of them are using Android application. The research describes how to build a Qur'an recitation correction application based on Android and integrate it with Google Speech so that application built more interesting and interactive with users. Google Speech be integrate as voice input media, users requested recite Qur'an verses then Google Speech will convert their voice to text. Text result obtained then matched with text in source code. Result of testing the accuracy of Google Speech implementation on this application it's 100%, that means it's been as expected.*

**Keywords:** Muroja'ah, Google Speech, Android, Qur'an, Voice Recognition

\*Penulis korespondensi

## 1. PENDAHULUAN

Muroja'ah adalah metode mengulang hafalan, baik hafalan baru maupun hafalan lama yang disetorkan kepada orang lain. Dalam hal ini seorang *hafidz* Al-Qur'an dapat memperdengarkan hafalannya kepada guru ataupun orang lain. Metode ini sangat membantu, sebab terkadang jika mengulang sendiri hafalan Al-Qur'an terdapat kesalahan yang tidak disadari, akan berbeda jika melibatkan guru atau orang lain karena kesalahan-kesalahan yang terjadi akan mudah diketahui.

Cara yang biasa dilakukan oleh para *hafidz* Al-Qur'an untuk melakukan muroja'ah dengan mencari orang lain tergolong tidak fleksibel karena seorang *hafidz* Al-Qur'an harus mencari orang lain terlebih dahulu untuk menyetorkan hafalan mereka. Pada tahun 2016, Indonesia memiliki jumlah penghafal Al-Qur'an terbesar di dunia, tercatat penghafal Al-Qur'an di Indonesia pada tahun 2016 sekitar 35.000 orang, sedangkan di Arab Saudi hanya berjumlah 6.000 orang [1]. Oleh karena itu, para *hafidz* Al-Qur'an sangat membutuhkan suatu alat atau aplikasi yang dapat mengoreksi hafalan mereka tanpa harus bersusah payah mencari orang lain.

Pemanfaatan kemampuan *smartphone* untuk keperluan dibebberapa bidang pun dikembangkan dengan aplikasi-aplikasi yang mampu mendukung

dalam penggunaannya. Termasuk diantara penggunaan *smartphone* dalam bidang agama adalah untuk media pembelajaran. Pengguna *smartphone* di Indonesia berkembang dengan cepat. Jumlah penduduk di Indonesia mencapai 250 juta jiwa. Lembaga riset digital marketing Emarketer memperkirakan pada tahun 2018 jumlah pengguna aktif *smartphone* di Indonesia lebih dari 100 juta orang. Dengan jumlah sebesar itu, Indonesia akan menjadi negara pengguna aktif *smartphone* terbesar keempat di dunia setelah Cina, India, dan Amerika[5].

Aplikasi *mobile* memiliki beragam sistem operasi Android, Ios, Microsoft, Symbian dan lainnya. Pengguna aplikasi *mobile* lebih banyak menggunakan sistem operasi Android, seperti dilansir badan statistik dunia bahwa persentase pengguna sistem operasi Android pada tahun 2017 adalah 88,37% [2].

Pada penelitian ini, dibangun 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 para *hafidz* Al-Qur'an sebagai media untuk melakukan muroja'ah kapan saja dan dimana saja.

Batasan masalah dalam pengerjaan penelitian ini adalah :

1. Aplikasi ini memanfaatkan Google *Speech* API sebagai media *input* suara.
2. Aplikasi ini akan mengoreksi hafalan Al-Qur'an kemudian memberikan notifikasi jika hafalan benar ataupun salah.
3. Aplikasi ini terdiri dari 3 surah, yakni surah Al-Ikhlas, surah Al-Kautsar dan surah An-Naas.
4. Aplikasi harus terkoneksi dengan jaringan internet.
5. Aplikasi akan bekerja dengan maksimal jika jaringan internet baik.
6. Aplikasi ini dibatasi hanya untuk pengguna android saja.

## 2. TINJAUAN PUSTAKA

Pada penelitian yang dilakukan oleh Fauzan, peneliti menjelaskan bagaimana membangun aplikasi media pembelajaran iqro berbasis Android dan mengintegrasikannya dengan Google *Speech* menggunakan metode *Speech Recognition* (pengenalan suara)[3].

Pada penelitian yang dilakukan oleh Sutrisno, peneliti membuat *automatic Speech recognition* pengenalan nama hewan berbasis Android. Dari uji coba yang dilakukan, aplikasi tersebut membantu orang tua dalam mengajarkan anak tentang nama hewan di sekitarnya[10].

Pada penelitian yang dilakukan oleh Saputra, peneliti memanfaatkan *cloud Speech* API untuk pengembangan media pembelajaran bahasa Inggris menggunakan teknologi *speech recognition*. Dalam uji coba yang dilakukan peneliti berhasil merancang, membuat, dan mengimplementasikan aplikasi dengan memanfaatkan *cloud Speech* API untuk pembelajaran bahasa Inggris[8].

Pada penelitian yang dilakukan oleh Susilo, peneliti menerapkan *Speech recognition* pada permainan teka-teki silang menggunakan metode *Hidden Markov Model* (HMM) berbasis dekstop[11]. Penelitian ini memerlukan tempat pelatihan dan pengenalan suara yang jauh dari gangguan.

Pada penelitian yang dilakukan oleh Suryadharma, peneliti merancang aplikasi *Speech to text* bahasa Inggris ke bahasa Bali berbasis Android. Aplikasi ini menerapkan *Speech recognition* dan tidak memerlukan akses internet sehingga dapat digunakan dimana saja bagi para wisatawan yang berniat mengetahui sedikit kata dalam bahasa Bali[13].

Pada penelitian yang dilakukan oleh Hariyani, dibuat suatu sistem yang mengendalikan lampu dari jarak jauh berbasis suara yang didukung oleh Google *voice recognition* dan menggunakan Android. Sistem ini dapat mematikan dan menghidupkan lampu, selain

itu dapat juga mengatur lampu dengan beberapa tingkat intensitas cahaya. Peneliti ini menggunakan Arduino Uno R3 sebagai piranti pengendali dengan bluetooth sebagai media komunikasi antara *smartphone* dan mikrokontroler. Berdasarkan pengujian sistem dan alat yang dibuat, untuk pengujian di dalam ruangan dengan penghalang, jangkauan maksimal kontrol mencapai 10 meter dengan respon lampu rata-rata 1,426 detik[4].

Pada penelitian yang dilakukan oleh Supriyanta, peneliti membangun aplikasi yang mengkonversi suara ke dalam teks dengan bahasa Indonesia. Aplikasi ini dibuat menggunakan Google *Speech* API. Aplikasi ini dapat mengurangi kesalahan inputan pada bidang teks. Selain itu aplikasi ini bisa digunakan untuk merekam pembicaraan yang hasil rekamannya sudah langsung berbentuk teks[12].

Pada penelitian yang dilakukan oleh Latief, peneliti berhasil membuat aplikasi *voice command* sebagai penghubung untuk mengendalikan perangkat elektronik rumah tangga. Hanya saja pada aplikasi ini belum ada fitur untuk menambah perangkat yang bisa dikendalikan oleh sistem pengendali dan belum ada fitur untuk menambahkan perintah suara[6].

Pada penelitian yang dilakukan oleh Shinwani, dirancang dan dibangun aplikasi *voice translator* berbasis Android menggunakan *Hidden Markov Model*. Aplikasi *voice translator* ini memanfaatkan suara sebagai inputan yang akan diproses sehingga pengguna tidak perlu mengetikkan kata yang akan dikirimkan. Aplikasi *voice translator* juga akan menerjemahkan hasil konversi suara ke teks ke dalam bahasa Inggris sehingga pengguna yang kurang bisa berbahasa Inggris dapat memanfaatkan aplikasi ini[9].

Pada penelitian yang dilakukan oleh Nurhidayat, dilakukan optimalisasi sistem pengenalan wicara dengan *Levesthein Edit Distance* untuk trading forex online. Pada penelitian ini, digunakan *socket* untuk komunikasi dengan *software* metatrader 4 dan *HTTP Request/Response* untuk komunikasi dengan Google *Speech engine*. Dari uji coba yang telah dilakukan, pengenalan suara menggunakan Google *Speech* API memberikan hasil yang lebih cepat dan tentunya kemudahan dibandingkan dengan pengoperasian manual. Hanya saja sistem aplikasi pengenalan suara pada penelitian ini menggunakan aplikasi yang berbeda *platform* dengan *software* metatrader sehingga memerlukan *socket* untuk dapat berkomunikasi dan terdapat jeda waktu ketika mengirimkan perintah transaksi[7].

**2.1 Dasar Teori**

**2.2.1 Sistem Pakar**

Google *Speech* API diperkenalkan pada tahun 2008 di Amerika Serikat untuk beberapa tipe *smartphone*. Google *Speech* API adalah sebuah kerangka kerja yang dibuat oleh Google untuk mengenali suara, dirubah menjadi *string* (teks) dan dimasukkan ke dalam halaman pencarian Google sehingga akan menampilkan hasil pencarian berdasarkan masukan suara tersebut. Pengenalan suara dilakukan pada *server* Google menggunakan algoritma Hidden Markov Model (HMM). Dengan kata lain *input* suara yang diterima oleh perangkat Android akan dikirimkan ke *server* Google, yang selanjutnya *server* Google melakukan pengenalan dan mengubahnya menjadi teks menggunakan algoritma HMM. Hasil konversi berupa teks kemudian dimasukkan ke dalam halaman pencarian Google kemudian *server* Google akan mengirimkan hasil pencariannya tersebut ke perangkat Android.

*Speech Recognition* atau Google *Speech* adalah suatu API yang disediakan oleh Google untuk mengidentifikasi suara dengan menggunakan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola yang tersimpan di dalam *database* Google. Teknologi *Speech Recognition* merupakan teknologi pengenalan kata yang memanfaatkan sinyal suara manusia sebagai masukan untuk kemudian dikenali oleh sistem komputer. Teknologi ini merupakan pengembangan interaksi antara manusia dengan komputer untuk meminimalisir peralatan *input device* seperti *mouse*, *keyboard* maupun peralatan *interface* lainnya.

**2.2.2 Hubungan Pengenalan Suara dengan AI**

Kecerdasan buatan (*Artificial Intelegent*) merupakan cabang ilmu pengetahuan yang berhubungan dengan pemanfaatan mesin untuk memecahkan persoalan yang rumit. Pemahaman ucapan/suara (*Speech/Voice Understanding*) adalah teknik agar komputer dapat mengenali dan memahami bahasa ucapan. Proses ini mengizinkan seseorang berkomunikasi dengan komputer dengan cara berbicara kepadanya. Istilah pengenalan suara mengandung arti bahwa tujuan utamanya adalah mengenali kata yang diucapkan tanpa harus tahu artinya. Secara umum prosesnya adalah usaha untuk menerjemahkan apa yang diucapkan seorang manusia menjadi kata atau kalimat yang dapat dimengerti oleh komputer.

**3. METODE PENELITIAN**

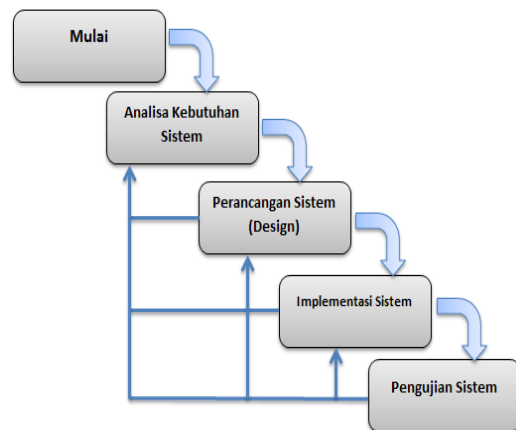
**3.1 Alat dan Bahan**

Penelitian ini menggunakan alat dan bahan yang terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*), sebagai berikut:

1. Laptop Asus K45A *processor* Intel Core i3 dengan *clock* frekuensi 2,40 GHz, Ram 6 GB.
2. Sistem operasi Windows 10 Pro.
3. *Smartphone* Xiaomi Note 5A, RAM 2 GB.
4. Google *Speech* API sebagai *framework* pengenalan kata.
5. Android Studio sebagai editor.

**3.2 Metode Pengembangan Sistem**

Pengembangan sistem yang akan dibangun ini menggunakan model *waterfall*. Model *waterfall* digunakan dalam pengembangan sistem ini karena prosesnya mengalir secara sistematis dari satu tahap ke tahap yang lain sehingga mudah untuk digunakan dalam pengembangan suatu sistem. Gambar 1 adalah diagram alur atau proses penggambaran pembuatan sistem.



Gambar 1. Pengembang sistem model *waterfall*

**3.2.1 Analisis Kebutuhan Sistem**

Pada tahap ini dilakukan penjabaran kebutuhan fungsional pengguna aplikasi sehingga aplikasi dapat dirancang dengan baik dan dapat berjalan dengan optimal. Secara umum kebutuhan fungsional yang harus ada di dalam aplikasi koreksi hafalan Al-Qur'an dapat dilihat pada Tabel I.

TABEL I. DESKRIPSI KEBUTUHAN

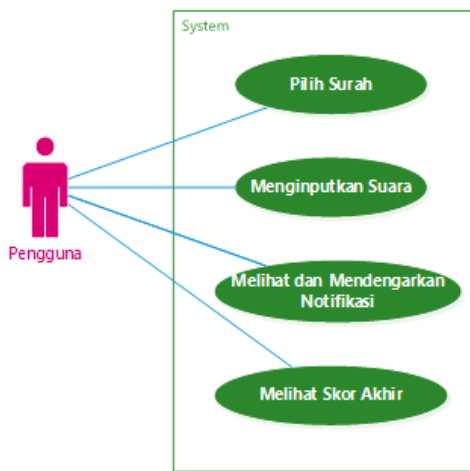
Nama Fungsi	Deskripsi Kebutuhan
Mengoreksi Hafalan	Aplikasi harus mengeluarkan notifikasi berupa audio dan gambar jika bacaan benar ataupun salah.
Mengetahui Poin	Aplikasi harus menghitung total poin yang didapatkan oleh pengguna dari total bacaan yang benar.

**3.2.2 Perancangan Sistem**

Dalam tahap ini, terdapat lima tahap perancangan yang dilakukan, yaitu perancangan *use case*, *class diagram*, *sequence diagram*, arsitektur sistem, perancangan tampilan aplikasi dan perancangan *database*.

**3.2.2.1 Perancangan use case diagram**

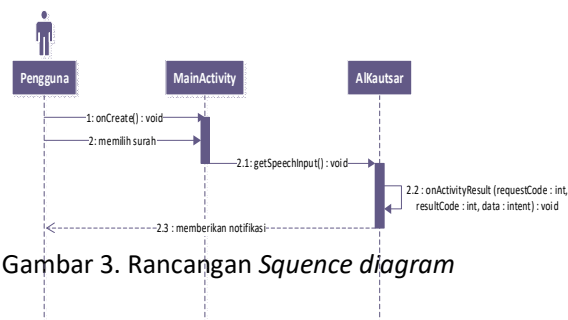
*Use case diagram* menggambarkan proses-proses yang dapat dilakukan dilakukan oleh aktor. *Use case diagram* aplikasi koreksi hafalan Al-Qur’an dapat dilihat pada Gambar 2.



Gambar 2. Use Case Diagram

**3.2.2.2 Perancangan Sequence Diagram**

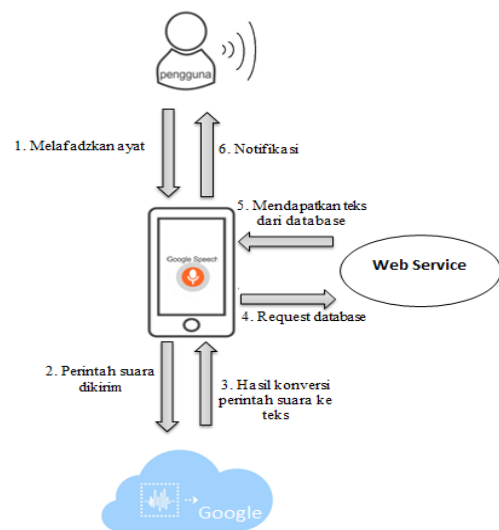
Sistem pertama kali akan menjalankan *method onCreate()*, Setelah itu pengguna akan memilih surat yang ingin dimuroja’ah. Pada surat yang telah dipilih, sistem menjalankan *method getSpeechInput()* kemudian pengguna melafadzkan ayat tertentu. Selanjutnya sistem menjalankan *method onActivityResult()* yang bertujuan untuk mengubah perintah suara menjadi teks lalu teks tersebut akan disesuaikan dengan kondisi yang telah dibuat sebelumnya kemudian sistem akan menampilkan notifikasi kepada pengguna. Gambaran dari *sequence diagram* pada aplikasi koreksi hafalan Al-Qur’an dapat dilihat pada Gambar 3.



Gambar 3. Rancangan Sequence diagram

**3.2.2.3 Arsitektur sistem dan alur kerja sistem**

Gambaran arsitektur dan alur kerja sistem dari aplikasi koreksi hafalan Al-Qur’an yang akan dibangun memiliki 5 alur kerja sistem. Pertama, pengguna melafadzkan ayat sesuai dengan surah yang dipilih. Kedua, sistem akan memproses perintah tersebut melalui *Google Speech*. Ketiga, *Google Speech* mengubah perintah suara tersebut menjadi teks. Keempat, Sistem akan merequest *string* dari *database*. Kelima, *String* yang didapatkan dari *database* akan disesuaikan dengan *string* yang didapatkan dari *Google*. Keenam, pengguna mendapatkan notifikasi dari hasil bacaan yang dilafadzkan. Gambaran arsitektur dan alur kerja sistem dapat dilihat pada Gambar 4.



Gambar 4. Arsitektur dan alur kerja sistem

**3.2.3 Implementasi**

Setelah melakukan tahap perancangan maka akan dilakukan proses implementasi yaitu pembangunan *Mobile Application*. Pada tahap pembangunan *mobile application*, rancangan *mobile application* akan diimplementasikan ke dalam *smartphone* menggunakan bahasa pemrograman Java dan *Android Studio* sebagai alat bantu dalam proses implementasi.

**3.2.4 Pengujian Sistem**

Teknik pengujian yang digunakan untuk melakukan pengujian pada aplikasi koreksi hafalan Al-Qur’an yaitu *black box* yang berfungsi untuk mengetahui kinerja dari fitur-fitur yang ada pada sistem. Selanjutnya pengujian dilakukan dengan menggunakan tabel pengujian akurasi *Google speech* untuk masing-masing surah. Tabel pengujian akurasi *Google speech* dapat dilihat pada Gambar 5.

TABEL II. PENGUJIAN APLIKASI PADA MASING-MASING SURAH

Surah	Ayat	Potongan Ayat	Penguji				
			A	B	C	D	E

Tabel pengujian (Tabel II) terdiri dari 4 kolom, yaitu :

1. Surah : Berisi nama surah
2. Ayat : Berisi ayat-ayat pada surah yang diuji
3. Potongan ayat : Berisi potongan ayat dalam bentuk kata
4. Penguji : Berisi hasil akurasi yang didapatkan dari masing-masing penguji

#### 4. HASIL DAN PEMBAHASAN

##### 4.1 Implementasi

Dalam penelitian ini, terdapat beberapa hal yang akan dibahas terkait implementasi rancangan aplikasi yang telah dijabarkan sebelumnya.

##### 4.1.1 Implementasi Kelas

Implementasi kelas pada Aplikasi Koreksi Hafalan Al-Qur'an merupakan kelas yang berisi bahasa pemrograman yang dibuat untuk mengimplementasikan sistem sesuai dengan perancangan yang dilakukan berupa *source code (coding)*. Implementasi kelas pada Aplikasi Koreksi Hafalan Al-Qur'an dapat dilihat pada Tabel III.

TABEL III. IMPLEMENTASI KELAS

Nama file Program	Keterangan
CSplashScreen	Kelas yang dijalankan pertama kali ketika aplikasi dibuka.
CMainActivity	Kelas Utama yang dijalankan pertama kali setelah kelas CSplashScreen
CAyat	Kelas untuk melakukan pengecekan <i>string</i>
CBenar	Kelas untuk menampilkan notifikasi jika bacaan benar
CSalah	Kelas untuk menampilkan notifikasi jika bacaan salah
CSkor	Kelas untuk menampilkan skor akhir

Untuk dapat mengenali suara yang dimasukkan ke aplikasi, digunakan Google API yang akan menerjemahkan suara masukan menjadi string. Dari

string ini, akan dicocokkan dengan string yang ada di *database*.

##### 4.1.2 Implementasi Interface Aplikasi

*Interface* merupakan antar muka yang akan berinteraksi secara langsung dengan pengguna aplikasi. Dalam implementasi *interface* yang telah dikembangkan berdasarkan perancangan yang telah dibuat sebelumnya, tampilan aplikasi dibagi menjadi beberapa *interface* sebagai berikut :

##### 4.1.2.1 Interface halaman muroja'ah

Pada halaman ini terdapat sebuah tombol yang berfungsi untuk memulai dan mengakhiri muroja'ah pada ayat tertentu. Implementasi *Interface* halaman muroja'ah dapat dilihat pada Gambar 5.



Gambar 5. *Interface* halaman muroja'ah

Dari halaman *input* suara, pengguna diminta menekan tombol 2 kali. Secara *default* tombol *input* suara memiliki warna merah, jika tombol ditekan maka warna tombol akan berubah menjadi abu-abu kemudian sistem akan merekam suara yang diucapkan oleh pengguna. Setelah selesai mengucapkan/ melafadzkan ayat maka pengguna harus menekan kembali tombol *input* suara tersebut hingga warna tombol kembali ke dalam kondisi *default* (merah).

##### 4.1.2.2 Interface jika bacaan benar

Pada halaman ini aplikasi akan menampilkan *layout* disertai dengan *audio* ketika bacaan yang telah dilafadzkan pada halaman muroja'ah bernilai benar, selanjutnya pengguna dapat melanjutkan muroja'ah ke ayat berikutnya dengan memilih tombol lanjut. *Interface* jika bacaan benar dapat dilihat pada Gambar 6.



Gambar 6. *Interface* jika bacaan benar

**4.1.2.3 *Interface* jika bacaan salah**

Pada halaman ini, aplikasi akan menampilkan *layout* disertai *audio* koreksi bacaan ketika bacaan yang telah dilafadzkan pada halaman muroja'ah bernilai salah. Selanjutnya pengguna dapat mengulang kembali melafadzkan ayat dengan menekan tombol "ok". Parameter bacaan dikatakan salah adalah jika hasil konversi teks Google *Speech* tidak sesuai dengan teks pada *source code*. *Interface* halaman jika bacaan salah dapat dilihat pada Gambar 7.



Gambar 7. *Interface* jika bacaan salah

**4.1.2.4 *Interface* hasil skor akhir hafalan**

Pada halaman ini, setelah pengguna telah selesai melakukan muroja'ah, maka sistem akan mengkalkulasikan skor akhir pengguna dari total bacaan yang benar pada setiap ayatnya. *Interface* halaman skor akhir hafalan dapat dilihat pada Gambar 8.



Gambar 8. *Interface* halaman skor akhir hafalan

**4.2 Pengujian**

Aplikasi yang telah menempuh tahap implementasi akan memasuki tahap pengujian untuk mengetahui sejauh mana keberhasilan aplikasi dan kualitasnya. Pengujian menggunakan metode *black box* dan tabel perhitungan akurasi.

**4.2.1 Hasil Pengujian *Black Box***

Metode ini menguji fungsi-fungsi pada aplikasi koreksi hafalan Al-Qur'an untuk menentukan apakah fungsi-fungsi tersebut sudah berjalan sesuai harapan atau tidak. Pengujian *black box* dapat dilihat pada Tabel IV.

TABEL IV. HASIL PENGUJIAN BLACK BOX

Skenario pengujian	Hasil yang di harapkan	Hasil pengujian	Kesimpulan
I	II	III	IV
Menekan tombol "klik" pada halaman muroja'ah	Sistem akan terhubung dengan Google <i>Speech</i>	Sesuai	Valid
Melafadzkan bacaan dengan baik dan benar.	Sistem akan mengoreksi bacaan yang dilafadzkan dengan cara melakukan perbandingan <i>String</i> kemudian akan memberikan notifikasi bahwa bacaan yang dilafadzkan pengguna adalah benar	Sesuai	Valid
Melafadzkan bacaan dengan suara yang samar dan bacaan yang salah	Sistem akan mengoreksi bacaan yang dilafadzkan dengan cara melakukan perbandingan <i>String</i> kemudian akan memberikan notifikasi bahwa bacaan yang dilafadzkan pengguna adalah salah. Selanjutnya sistem akan memperdengarkan bacaan dengan lafadz yang benar lalu pengguna diminta melafadzkan kembali bacaan tersebut.	Sesuai	Valid
Berhasil menyelesaikan hafalan	Sistem akan menampilkan skor akhir pengguna	Sesuai	Valid

**4.2.2 Tabel Perhitungan pengujian aplikasi**

Pada proses pengujian ini, aplikasi akan diuji oleh 5 orang santri, 3 diantaranya yang sudah fasih dan baik dalam melafadzkan Al-Qur'an yaitu A, B, dan C. kemudian 2 diantaranya memiliki kemampuan bacaan yang standar (tidak fasih dan tidak pula terbata-bata) yaitu D dan E. Pengujian dilakukan pada koneksi internet yang baik dan jauh dari gangguan. Tabel Pengujian Akurasi Google *Speech* dapat dilihat pada Tabel V, VI dan VII.

TABEL V. PENGUJIAN APLIKASI PADA SURAH AL-IKHLAS

Ayat	Lafadz	Penguji				
		A	B	C	D	E
1	Qul Huwallahu Ahad	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
2	Allahu Samad	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
3	Lam Yalid Walam Yulad	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
4	Walam Yakullahu Kufuwan Ahad	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Salah (Sesuai)	Salah (Sesuai)
Akurasi (%)		100%	100%	100%	100%	100%
Rata-Rata Akurasi(%)		100%				

Berikut ini merupakan rumus untuk menentukan rata-rata akurasi dari pengujian yang telah dilakukan untuk Surat Al Ikhlas.

$$\text{Rata - rata akurasi} = \frac{(100) + (100) + (100) + (100) + (100)}{5} = 100\%$$

TABEL VI. PENGUJIAN APLIKASI PADA SURAH AL-KAUTSAR

Ayat	Lafadz	Penguji				
		A	B	C	D	E
1	Inna a' thoina Kal Kautsar	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
2	Fasholli Lirobbika Wanhar	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
3	Innasya Niaka Huwal Abtar	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
Akurasi (%)		100%	100%	100%	100%	100%
Rata-Rata Akurasi(%)		100%				

Ini merupakan rumus yang digunakan untuk menghitung rata-rata akurasi aplikasi untuk Surat Al-Kautsar.

$$\text{Rata - rata akurasi} = \frac{(100) + (100) + (100) + (100) + (100)}{5} = 100\%$$

TABEL VII. PENGUJIAN APLIKASI SPEECH PADA SURAH AN-NAAS

Ayat	Lafadz	Penguji				
		A	B	C	D	E
1	Qul A'udzu Birabbinnas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
	Malikinnas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
	Ilahinnas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
	Min Syaril was waasil Khannas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Salah (Sesuai)	Salah (Sesuai)
5	Alladzi yuwaswisu fii Sudurinnas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Salah (Sesuai)	Salah (Sesuai)
6	Minal Jinnati Wannas	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)	Benar (Sesuai)
Akurasi (%)		100%	100%	100%	100%	100%
Rata-Rata Akurasi(%)		100%				

$$\text{Rata - rata akurasi} = \frac{(100) + (100) + (100) + (100) + (100)}{5} = 100\%$$

$$\text{Rata - rata akurasi keseluruhan} = \frac{(1.6) + (2.6) + (3.6)}{\text{total data}} \times 100\%$$

$$\text{Rata - rata akurasi keseluruhan} = \frac{(100) + (100) + (100)}{3} \times 100\% = 100\%$$

Jadi rata-rata akurasi aplikasi ini dengan kasus surah Al-Ikhlas, Al-Kautsar dan An-Naas adalah 100%, artinya sudah sesuai dengan harapan.

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan pembangunan aplikasi dan implementasi Google *Speech* API yang telah dilakukan pada Aplikasi Koreksi Hafalan Al-Qur'an maka dapat disimpulkan bahwa:

1. Pembuatan aplikasi ini sudah memenuhi kebutuhan pengguna sesuai dengan yang dijabarkan pada kebutuhan sistem.
2. Aplikasi Koreksi Hafalan Al-Qur'an dapat mengolah inputan suara dengan mengimplementasikan Google *Speech* API.
3. Dari Hasil pengujian *black box* menunjukkan bahwa seluruh fungsi telah berjalan sesuai dengan aplikasi yang diharapkan.
4. Google *Speech* API tidak bisa mendeteksi dengan maksimal lafadz ayat yang cenderung lebih panjang dan melafadzkan dengan suara yang kurang jelas.
5. Rata-rata akurasi Google *Speech* API dalam mengkonversi inputan suara menjadi teks pada

studi kasus surah Al-Ikhlas, Al-Kautsar dan An-Naas adalah 100%.

## 5.2 Saran

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

1. Aplikasi masih memanfaatkan Google *Speech* sebagai media *input* suara, maka untuk penelitian selanjutnya dapat menciptakan inputan suara yang lain tanpa harus menggunakan Google *Speech*
2. Aplikasi masih bersifat *online*, maka diharapkan peneliti selanjutnya mampu mengembangkan Aplikasi Koreksi Hafalan Al-Qur'an versi *offline*.
3. Diharapkan dapat meningkatkan fitur pengkoreksian menjadi lebih detail yang meliputi pengkoreksian hukum tajwid.
4. Diharapkan dapat menambah surah Al-Qur'an menjadi lebih banyak lagi.

- [9] Shinwani. 2016, "Rancang Bangun Aplikasi Voice Translator Berbasis Android Menggunakan Hidden Markov Model", Teknik Informatika Universitas Negeri Maulana Malik Ibrahim Malang.
- [10] Sutrisno, J. 2016, Aplikasi Pengenalan Nama Hewan Dengan Metode Speech Recognition Berbasis Android, Teknik Informatika, Universitas Muhammadiyah Sidoarjo.
- [11] Susilo, B. 2016, Penerapan Speech Recognition Pada Permainan Teka-Teki Silang Berbasis Desktop, Teknik Informatika, Universitas Bengkulu, Vol. 4, No.1 Maret 2016.
- [12] Supriyanta. 2014, Aplikasi Konversi Suara Ke Teks Berbasis Android Menggunakan Google Speech API, AMIK BSI Yogyakarta, Vol. 2, No. 2 September 2014.
- [13] Suryadharma. 2014, Perancangan Aplikasi Speech To Text Bahasa Inggris Ke Bahasa Bali Menggunakan Pocketsphinx Berbasis Android, Teknik Telekomunikasi, Universitas Telkom Bandung.

## DAFTAR PUSTAKA

- [1] Anonim, Jumlah Penghafal Al-Qur'an Di Indonesia, <https://quranpedia.id/blog/jumlah-penghafal-quran-di-indonesia-terbesar-di-dunia.html>, diakses pada 20 Maret 2018 12.03 PM.
- [2] Anonim, Smartphone, <https://www.statista.com/topics/840/smartphones.html> diakses pada 20 Maret 2018 10.07 AM.
- [3] Fauzan, A. 2018, Pembangunan Aplikasi Iqro' Berbasis Android Menggunakan Google Speech, Teknik Informatika, Universitas Brawijaya, Vol. 2, No.1, Januari 2018, hlm.29-35.
- [4] Hariyani. 2014, Realisasi Pengendali Intensitas Cahaya Lampu Dengan Kontrol Suara dan Google Android Speech Recognition API, Teknik Telekomunikasi, Universitas Telkom Bandung.
- [5] Kominfo. <https://kominfo.go.id/index/jumlah-pengguna-smarphone-meningkat.php> diakses pada 20 Maret 2018 10.30 AM.
- [6] Latief. 2015, Voice Command Pengendali Perangkat Elektronik Rumah Tangga Menggunakan Raspberry PI, UIN Sunan Kalijaga Yogyakarta.
- [7] Nurhidayat. 2014, "Optimalisasi Sistem Pengenalan Wicara dengan Levesthein Edit Distance Untuk Trading Forex Online". Teknik Telekomunikasi Universitas Telkom Bandung.
- [8] Saputra, S. 2017, Pemanfaatan Cloud Speech API Untuk Pengembangan Media Pembelajaran Bahasa Inggris Menggunakan Teknologi Speech Recognition, Teknik Informatika, STMIK Amikom Purwokerto.