BAB III
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

3.1.1 Analisis Masalah

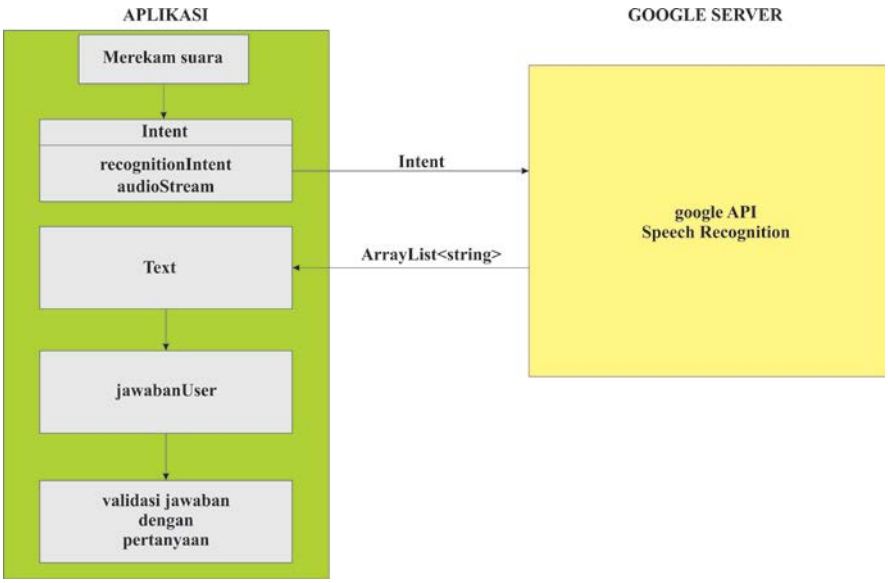
Kemampuan membaca sangat perlu diperhatikan khususnya dari segi aspek kebahasaan yang meliputi penggunaan lafal. Tujuan agar anak memahami tentang penggunaan bahasa ketika membaca, sehingga tidak terjadi kesalahan penggunaan dalam pemakaian bahasa.

Aplikasi *mobile* bisa menjadi alternatif, karena *mobile* merupakan pengguna tertinggi bila dibandingkan dengan pengguna *internet.* Kelebihan menggunaan aplikasi *mobile* bisa digunakan dimanapun dan kapanpun, dapat memperdengarkan suara/bunyi dan pengecekan kata yang diucapkan sehingga pengguna bisa lebih tertarik. Aplikasi *mobile* ini menggunakan platform *android* yang merupakan sistem operasi yang paling banyak digemari untuk saat ini, dengan menempati peringkat pertama android 79%, iOS 14.2%, Microsoft 3.3%, Blackberry 2.7% [10].

3.1.2 Analisis *Speech Recognition*

Aplikasi merekam suara yang dimasukkan pengguna. *Audio stream* hasil perekaman dimasukkan kedalam data *intent* dengan *action recognitionintent*, *intent* tersebut dikirim ke *server google* untuk dilakukan proses *speech recognition*. *Server google* mengirimkan hasil *speech recognition* berupa *ArrayList<String>*, string tersebut kemudian di bandingkan dengan pertanyaan sebagai proses verifikasi.

Berikut gambar proses tentang *speech recognition* pada aplikasi lafal bahasa Indonesia:



Gambar 3.1 Proses *Speech Recognition*

3.1.2.1 Struktur Data pada Proses *Speech Recognition*

Berikut tabel yang menjelaskan tentang struktur data yang ada pada proses *speech recognition*:

Table 3.1 Struktur data pada proses *speech recognition*

Nama Variabel	Tipe Data
jawabanUser	String
pertanyaan	String
res	Resources
r	Random
i	Int
listPertanyaan	Array(string)
intent	Intent
text	arrayList<string>

3.1.2.2 Parameter pada Proses *Speech Recognition*

Berikut tabel yang menjelaskan tentang parameter yang ada pada proses *speech recognition*:

Tabel 3.2 Parameter pada proses *speech recognition*

Nama method	Nama var parameter	Tipe data
startActivityFixResult	intent	intent
	requestCode	int
onActivityResult	requestCode	int
	requestCode	int
	data	intent
setJawabanUser	jawabanUser	string
setPertanyaan	pertanyaan	string
setContext	context	context

3.1.3 Analisis Aplikasi Sejenis Terhadap Fitur yang Sudah Ada

Sebelum merancang aplikasi yang akan dibuat, terlebih dahulu dilakukan analisis pada aplikasi sejenis yang sudah ada. Ini bertujuan agar aplikasi yang sudah dibuat nanti memiliki solusi dari permasalahan aplikasi yang sudah ada.

Aplikasi edukasi pengucapan bahasa sudah ada, diantaranya *Speak English by Appjungs GmbH & Co.KG*, *Speak Arabic by Holfeld.com*, *Speak English Correctly by Orange Duck Studio* Setelah melakukan observasi terhadap aplikasi tersebut, didapat kesimpulan seperti yang bisa dilihat pada tabel 3.3

Berikut tabel perbandingan aplikasi sejenis dalam hal fitur:

Tabel 3.3 Perbandingan Aplikasi Sejenis

Nama Aplikasi	Jenis Aplikasi	Kelebihan	Kekurangan
Speak English by Appjungs GmbH	*Speak*	Mengulang *speak* yang keluar dari aplikasi	Hanya *speak* saja, tidak ada konfirmasi benar atau salah, kata yang tersedia terbatas.
Speak Arabic by Holfeld.com	*Speak*	-	Tidak bisa mengucapkan bahasa atau melafalkan

Tabel 3.3 Perbandingan Aplikasi Sejenis (lanjutan)

Nama Aplikasi	Jenis Aplikasi	Kelebihan	Kekurangan
			bahasa yang tersedia, Tidak ada konfirmasi benar atau salah, hanya tersedia beberapa kata dan beberapa angka atau terbatas.
Speak English Correctly by Orange Duck Studio	*Speak*	Bisa mengulang speak yang keluar dari aplikasi	Tidak ada konfirmasi kata benar atau salah saat pengucapan diulang, hanya pengulangan kata dan pencarian kata.

3.1.4 Analisis Aplikasi yang Akan Dibangun

Aplikasi ini dibangun sebagai media pembelajaran untuk mempelajari bahasa Indonesia. Dengan adanya mengucapkan lafal bahasa indonesia, konfirmasi benar atau salah dari kata yang telah diucapkan.

Tabel 3.4 Fitur aplikasi

No	Fungsi	Deskripsi
1	Tes Pelafalan Kata	Fitur untuk memasukkan suara.
2	Mengecek Pelafalan	Fitur untuk mengecek ke Google API
3	Contoh Pelafalan Kata	Fitur untuk mengetahui cara baca kata.
4	Hasil Kata	Fitur untuk mengetahui benar atau salah dari hasil speech yang telah diucapkan.

Tabel 3.4 Fitur aplikasi (lanjutan)

No	Fungsi	Deskripsi
5	Info aplikasi	Untuk melihat cara menggunakan aplikasi.

3.1.5 Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

Spesifikasi kebutuhan perangkat lunak berisi deskripsi dari kebutuhan perangkat lunak yang akan dibangun. SKPL dari pembelajaran bahasa Indonesia bisa dilihat pada table berikut :

Tabel 3.5 Kebutuhan Fungsional

Kode Kebutuhan	Deskripsi Kebutuhan
SKPL-F01	Menampilkan materi pembelajaran lafal bahasa Indonesia
SKPL-F02	Memasukan contoh pengucapan oleh pengguna
SKPL-F03	Menampilkan info aplikasi

Tabel 3.6 Kebutuhan Non Fungsional

Kode Kebutuhan	Deskripsi Kebutuhan
SKPL-NF01	Ketersediaan koneksi internet untuk dapat diakses
SKPL-NF02	Desain aplikasi yang memudahkan penggunaan
SKPL-NF03	Kesesuaian kebutuhan spesifikasi perangkat keras agar aplikasi bisa berjalan dengan optimal

3.1.6 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang menitik beratkan pada properti prilaku yang dimiliki oleh sistem, diantaranya kebutuhan perangkat keras, perangkat lunak dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.1.6.1 Analisis Kebutuhan Perangkat Keras

Perangkat keras yang dibutuhkan untuk implementasi sistem yang akan dibuat. Adapun perangkat keras yang digunakan antara lain:

Tabel 3.7 Perangkat Keras Komputer

Komputer		
No	Spesifikasi	
1	Processor	2.3 GHz
2	Memory	4 GB
3	VGA	512 MB
4	Harddisk	500 GB

Tabel 3.8 Perangkat Keras Smartphone

Smartphone		
No	Spesifikasi	
1	CPU	1.2 GHz Dual-Core
2	Memory Internal	1.7 GB storage, 1 GB RAM, 4 GB ROM
3	Memory Eksternal	2 GB
4	Layar	480 x 800 pixels, 4.3 inches
5	Speaker	Standar

3.1.6.2 Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak digunakan dalam sebuah sistem merupakan kumpulan perintah-perintah yang diberikan kepada perangkat keras agar saling berinteraksi untuk melakukan suatu tugas. Perangkat lunak yang dibutuhkan untuk membangun aplikasi pembelajaran untuk pengucapan lafal bahasa Indonesia adalah sebagai berikut:

1. Sistem Operasi Windows
2. JDK (*Java Development Kit*) 1.6, SDK (*Standard Development Kit*)
3. ADT (Android Development Tools) V22.6.2-1085508.
4. Eclipse Kepler 4.3.
5. Android emulator versi 2.3 (Gingerbread)
6. Coreldraw X6.

7. Sound and Voice Recorder yaitu ASR

3.1.6.3 Analisis Pengguna Sistem

Analisis pengguna sistem merupakan analisis terhadap pengguna yang akan menggunakan aplikasi yang telah dibangun. Perangkat keras dan perangkat lunak yang ada tidak akan berguna apabila tidak ada pengguna yang mengoperasikannya. Oleh karena itu dibuatlah analisis karakteristik pengguna seperti yang bisa dilihat pada table 3.5.

Tabel 3.9 Karakteristik Pengguna

Pengguna	Karakteristik	
Pengguna	Umur	6 – 12 tahun.
	Edukasi	Sekolah Dasar
	Keterampilan	Bisa mengoperasikan smartphone Android.
	Kriteria	Pengguna tidak memiliki cacat pada pengucapannya, seperti bibir sumbing dan autis sehingga mempengaruhi kejelasan kata-kata yang diucapkan.
	Jenis kelamin	Laki-laki dan perempuan.
	Motivasi	Menengah
	Koneksi	Harus menggunakan Internet

3.1.7 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan. Analisis kebutuhan fungsional ini meliputi analisis kebutuhan sistem, analisis kebutuhan data, spesfikasi sistem dan pemodelan sistem.

Pemodelan sistem dimodelkan dengan menggunakan UML (Unified Modeling Language). Tahap-tahap pemodelan dalam analisis tersebut antar lain

Use Case Diagram, *Class Diagram*, *Activity Diagram*, *Sequence Diagram*, *Component Diagram*, dan *Deployment Diagram*.

3.1.7.1 Analisis Kebutuhan Sistem

Analisis kebutuhan data bertujuan untuk menganalisis kebutuhan data yang diperlukan dan memudahkan perancangan informasi setelah system yang dibutuhkan telah diketahui. Adapun analisis kebutuhan data dengan kriteria sebagai berikut :

1. Data Daftar Kata, menampilkan data daftar kata.
2. Data Rekaman Suara, menampilkan data rekaman suara dari daftar kata.

3.1.7.2 Spesifikasi Sistem

Spesifikasi sistem yang dibutuhkan sebagai berikut :

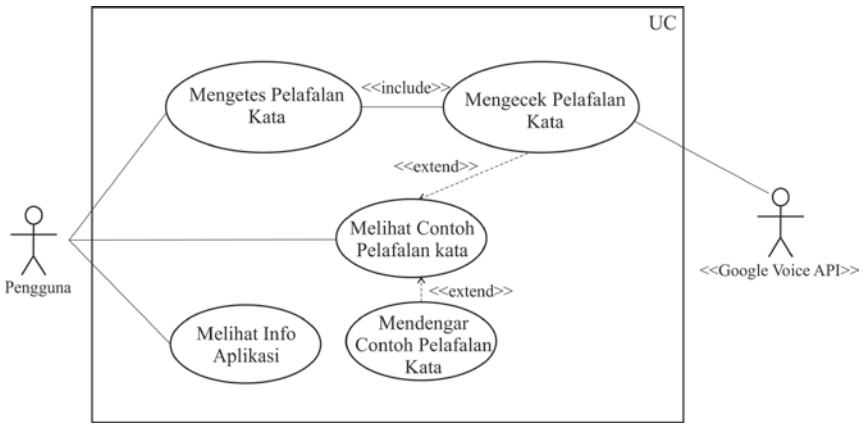
Spesifikasi sistem pada pengguna yaitu Pengguna berada dalam wilayah cakupan jaringan seluler agar tidak mengalami gangguan saat menjalankan aplikasi.

3.1.7.3 Pemodelan Sistem

Pemodelan sistem berfungsi untuk menentukan fungsi-fungsi yang dapat dilakukan oleh sistem pada aplikasi serta menentukan kelas yang dibutuhkan untuk realisasi fungsi-fungsi sistem yang telah dianalisis sebelumnya dan mendeskripsikannya kedalam bentuk diagram.

3.1.7.3.1 Use Case Diagram

Use case diagram Secara umum interaksi antara penggguna dengan sistem perintah suara ini digambarkan pada *use case* diagram berikut ini :



Gambar 3.2 *Use case* Diagram Aplikasi Pelafalan Bahasa Indonesia

Use case terdiri dari tiga bagian, yaitu: definisi aktor, definisi *use case* dan skenario *use case.*

1. Definisi *Actor*

Definisi *actor* berfungsi untuk menjelaskan *actor* yang terdapat pada Diagram *use case.* Definisi *actor* dijelaskan pada tabel 3.10

Tabel 3.10 Definisi *Actor*

No	*Use Case*	Deskripsi
1	Pengguna	Orang yang menggunakan aplikasi lafal pembelajaran bahasa Indonesia.
2	<<Google Voice API>>	Sistem yang hanya menyediakan database.

2. Defenisi Use Case

Definisi *use case* berfungsi untuk menjelaskan fungsi *use case* yang terdapat pada diagram *use case.* Definisi *use case* dijelaskan pada tabel 3.11:

Tabel 3.11 Definisi *Use Case*

No	*Use Case*	Deskripsi
1	Mengetes Pelafalan Kata	Proses untuk mengetes pengecekan pelafalan kata dengan memasukkan suara.
2	Mengecek Pelafalan Kata	Proses untuk mengecek pelafalan kata masukan dengan kata yang ada di google.
3	Melihat Contoh Pelafalan Kata	Proses untuk melihat pelafalan kata berupa list kata.
4	Melihat Info Aplikasi	Proses menampilkan menu info aplikasi.
5	Mendengar Pelafalan Kata	Proses mendengarkan pelafalan kata.

3. Skenario *Use Case*

Skenario pada diagram *use case* berisi mengenai alur kejadian yang didalamnya menjelaskan urutan interaksi antara aktor dengan use case tersebut sedangkan diagram aktifitas atau diagram *activity* menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis.

Berikut adalah tabel 3.12 skenario *use case* mengetes pelafalan kata telah didefinisikan sebelumnya:

Nomer Use Case	1	
Nama Use Case	Mengetes Pelafalan Kata	
Tujuan	Untuk mengetes pelafalan kata	
Kondisi Awal	Pengguna berada di menu utama	
Kondisi Akhir	Sistem menampilkan tes pelafalan	
Pemicu	Pengguna memilih menu mengetes pelafalan kata	
Aliran Utama	Step	Aksi
	1.	Pengguna memilih menu tes pelafalan kata
	2.	Sistem menampilkan kata secara acak dan tombol tes pelafalan
	3.include::mengecek pelafalan kata	Sistem mengecek pelafalan kata
	4.	Sistem menampilkan dialog benar
	5.	Pengguna memilih tombol selesai
Ekstensi	4.1.	Sistem menampilkan dialog salah
	4.1.1	Pengguna memilih tombol selesai
	4.1.2	Pengguna memilih tombol ulangi
	5.1	Pengguna memilih tombol berikutnya

Berikut adalah skenario jalannya *use case* yang telah didefinisikan sebelumnya

Tabel 3.13 Skenario *Use Case* Mengecek Pelafalan Kata

Nomer Use Case	2	
Nama Use Case	Mengecekan Pelafalan Kata	
Tujuan	Untuk mengecekan pelafalan kata	
Kondisi Awal	Pengguna berada di menu tes pelafalan	
Kondisi Akhir	Sistem menampilkan hasil cek pelafalan	
Pemicu	Pengguna memilih menu mengecekan pelafalan kata	
Aliran Utama	Step	Aksi
	1.	Pengguna menekan tombol cek pelafalan
	2.	Sistem memunculkan tampilan untuk merekam suara (selama 10 detik)
	3.	Pengguna memasukkan kata berupa suara
	4.	Sistem mengecek kata ke Google API
	5.	Sistem mencocokkan hasil kata dari Google API dengan kata yang ditanyakan
	6.	Jika jawaban pengguna benar maka tampil dialog benar
	7.	Pengguna memilih tombol “selesai”
	8.	Sistem kembali ke menu Utama

Tabel 3.13 Skenario *Use Case* Mengecek Pelafalan Kata (lanjutan)

Nomer Use Case	2	
Nama Use Case	Mengecekan Pelafalan Kata	
Tujuan	Untuk mengecekan pelafalan kata	
Kondisi Awal	Pengguna berada di menu tes pelafalan	
Kondisi Akhir	Sistem menampilkan hasil cek pelafalan	
Pemicu	Pengguna memilih menu mengecekan pelafalan kata	
Ekstensi	Step	Aksi Cabang
	1.1	Sistem menampilkan Perangkat tidak mendukung
	2.1	Pengguna tidak memasukkan kata berupa suara (diam / tidak ada aktifitas sama sekali)
	2.1.1	Sistem kembali ke tampilan
	2.2	Jika sinyal tidak ada atau gangguan koneksi maka sistem menampilkan dialog tidak dapat menjangkau google saat ini.
	2.2.1	Pengguna menekan tombol “batalkan”

Tabel 3.13 Skenario *Use Case* Mengecek Pelafalan Kata (lanjutan)

Nomer Use Case	2	
Nama Use Case	Mengecekan Pelafalan Kata	
Tujuan	Untuk mengecekan pelafalan kata	
Kondisi Awal	Pengguna berada di menu tes pelafalan	
Kondisi Akhir	Sistem menampilkan hasil cek pelafalan	
Pemicu	Pengguna memilih menu mengecekan pelafalan kata	
Ekstensi	Step	Aksi Cabang
	2.2.1.1	Sistem menampilkan tampilan
	2.2.2	Pengguna menekan “coba lagi”
	2.2.2.1	Sistem mencari koneksi dan kembali ke rekaman
	4.1	Jika suara yang di masukkan tidak jelas maka sistem menampilkan dialog tidak mengerti, coba ucapkan lagi !
	4.1.1	Pengguna memilih tombol “batalkan”
	4.1.2	Sistem kembali ke tampilan
	4.2.1	Pengguna memilih tombol “coba lagi”

Tabel 3.13 Skenario *Use Case* Mengecek Pelafalan Kata (lanjutan)

Nomer Use Case	2	
Nama Use Case	Mengecekan Pelafalan Kata	
Tujuan	Untuk mengecekan pelafalan kata	
Kondisi Awal	Pengguna berada di menu tes pelafalan	
Kondisi Akhir	Sistem menampilkan hasil cek pelafalan	
Pemicu	Pengguna memilih menu mengecekan pelafalan kata	
Ekstensi	Step	Aksi Cabang
	4.2.2	Sistem kembali ke tampilan rekaman
	6.1	Jika jawaban salah maka tampil dialog salah
	6.1.1	Pengguna memilih tombol “selesai”
	6.1.1.1	Sistem kembali ke menu utama
	6.1.2	Pengguna memilih tombol “ulangi”
	6.1.2.1	Sistem menampilkan tampilan
	6.2	Pengguna memilih tombol “kata berikutnya”
	6.2.1	Sistem menampilkan acak kata

Berikut adalah skenario *use case* melihat contoh pelafalan kata telah didefinisikan sebelumnya:

Tabel 3.14 Skenario *Use Case* Melihat Contoh Pelafalan Kata

Nomer Use Case	3	
Nama Use Case	Melihat Contoh Pelafalan Kata	
Tujuan	Untuk Melihat Contoh Pelafalan Kata	
Kondisi Awal	Pengguna berada di menu utama	
Kondisi Akhir	Sistem menampilkan daftar yank ada di indek kata	
Pemicu	Pengguna memilih menu melihat pelafalan kata	
Aliran Utama	Step	Aksi
	1.	Pengguna memilih menu contoh pelafalan
	2.	Sistem menampilkan daftar indek kata
	3.	Pengguna menekan daftar indek kata
	4.extends::mendengar pelafalan kata	Sistem menampilkan indek kata yang sudah di pilih
	5.	Pengguna menekan tombol “mendengarkan”
	6.	Sistem memainkan suara
	7.	Pengguna mengecek pelafalan
	8.	Sistem menampilkan dialog benar
	9.	Sistem menampilkan dialog selesai
Ekstensi	Step	Aksi Cabang
	5.1.	Pengguna menekan tombol kembali
	8.1.	Sistem menampilkan dialog salah
	8.1.1.	Pengguna memilih coba lagi
	8.1.2.	Pengguna memilih selesai

Berikut adalah skenario *use case* melihat info aplikasi telah didefinisikan sebelumnya:

Tabel 3.15 Skenario *Use Case* Melihat Info Aplikasi

Nomer Use Case	4	
Nama Use Case	Melihat Info Aplikasi	
Tujuan	Untuk menampilkan menu Info Aplikasi	
Kondisi Awal	Pengguna berada di menu utama	
Kondisi Akhir	Sistem menampilkan menu info aplikasi	
Pemicu	Pengguna memilih menu Info Aplikasi	
Aliran Utama	Step	Aksi
	1.	Pengguna memilih menu info aplikasi
	2.	Sistem menampilkan menu info aplikasi

Berikut adalah skenario *use case* mendengar contoh pelafalan kata telah didefinisikan sebelumnya:

Tabel 3.16 Skenario *Use Case* Mendengar Contoh Pelafalan Kata

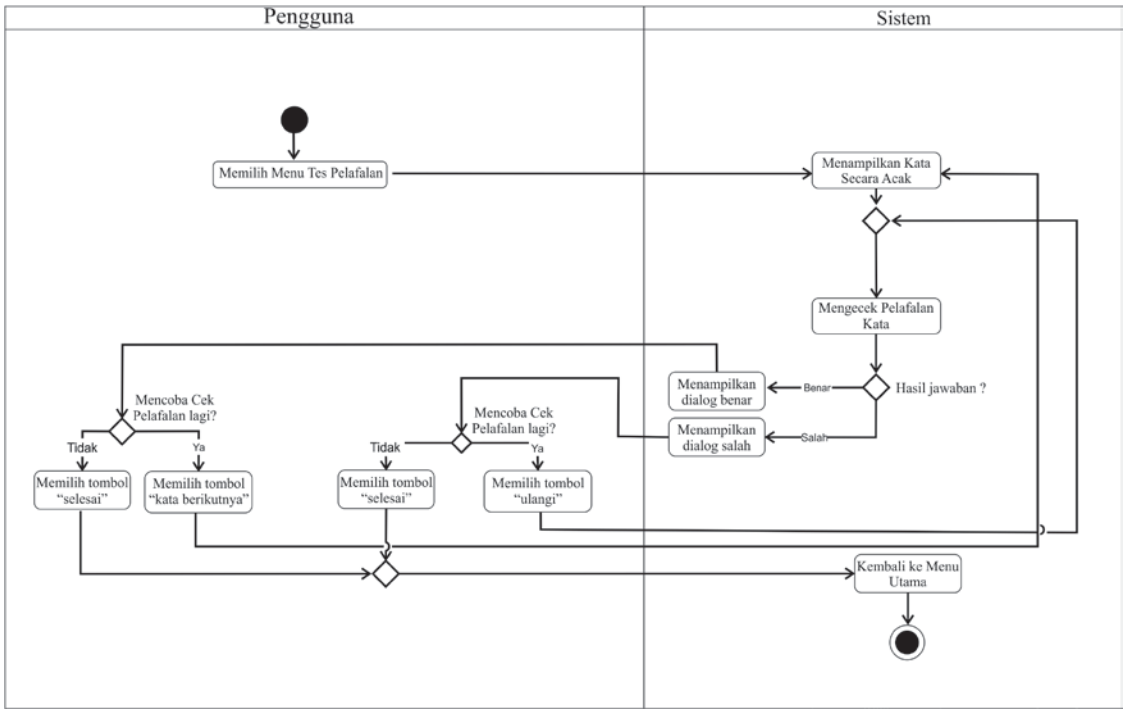
Nomer Use Case	5	
Nama Use Case	Mendengar Contoh Pelafalan Kata	
Tujuan	Untuk mendengar contoh pelafalan kata	
Kondisi Awal	Pengguna sudah mengklik daftar indek kata	
Kondisi Akhir	Sistem menampilkan suara rekaman dari daftar kata yang sudah di pilih sebelumnya	
Pemicu	Pengguna memilih daftar kata	
Aliran Utama	Step	Aksi
	1.	Sistem menerima masukan pilihan kata
	2.	Sistem mengambil ke indek kata
	3.	Sistem mengeluarkan suara sesuai dengan kata yang sudah di pilih.

3.1.7.3.2 *Activity Diagram*

Berikut adalah gambar diagram *activity* untuk *use case* yang telah didefinisikan sebelumnya :

1. *Activity Diagram* Tes Pelafalan

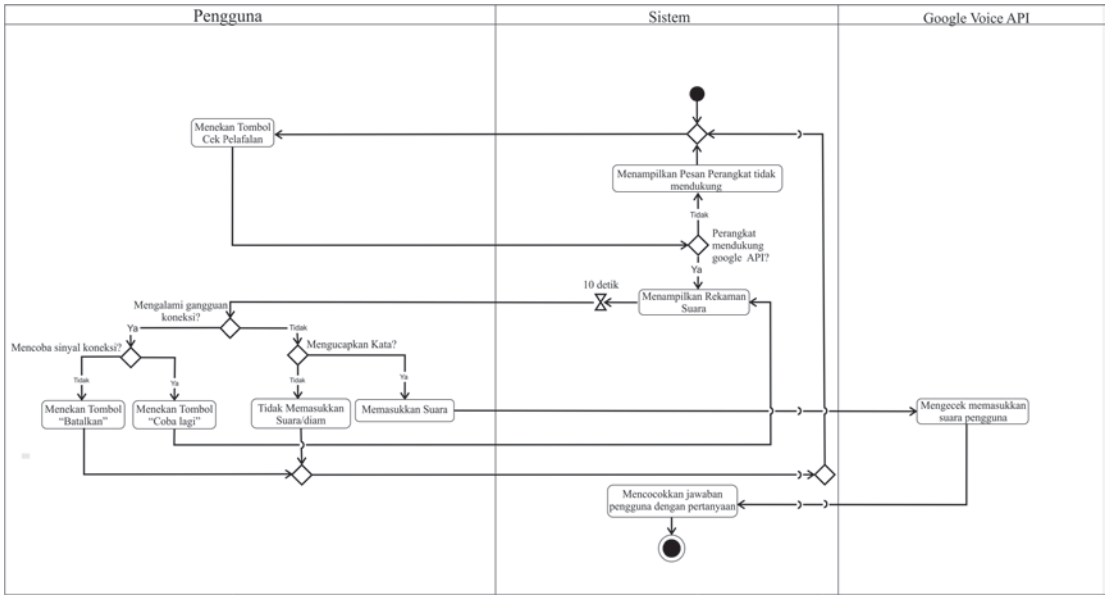
Activity Diagram Tes Pelafalan pada aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.3 *Activity Diagram* Tes Pelafalan Kata

2. *Activity Diagram* Mengecek Pelafalan Kata

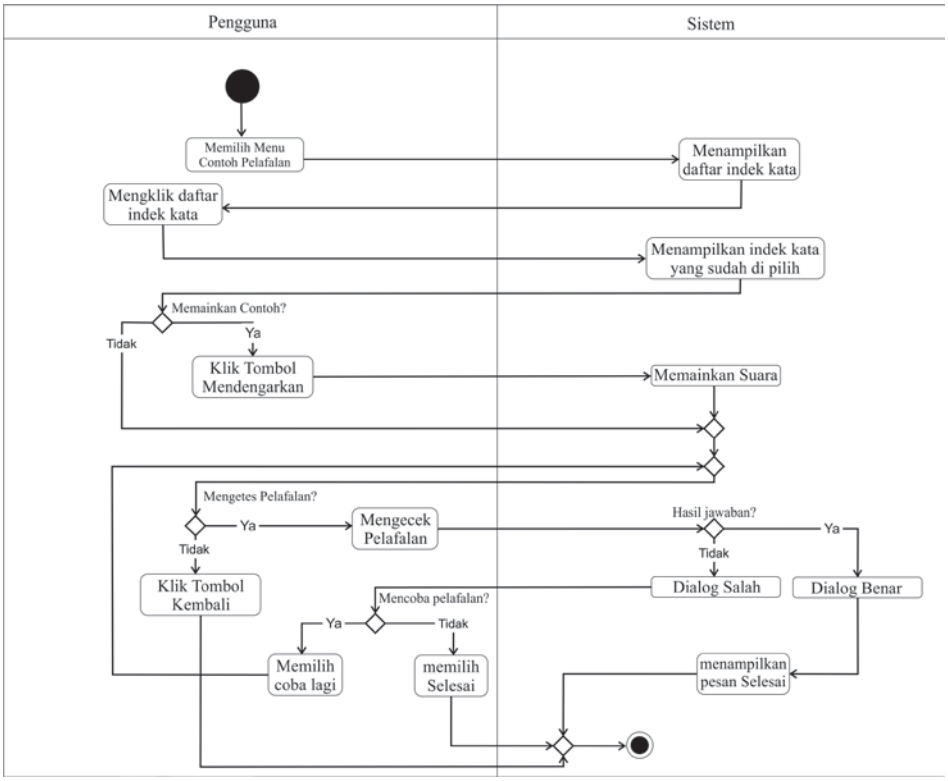
Activity Diagram mengecek pelafalan kata pada aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.4 *Activity Diagram* Mengecek Pelafalan Kata

3. *Activity Diagram* Contoh Pelafalan Kata

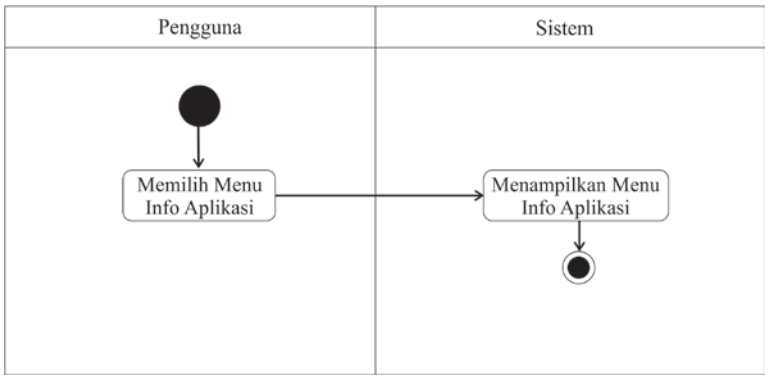
Activity Diagram contoh pelafalan kata pada aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.5 *Activity Diagram* Contoh Pelafalan Kata

4. *Activity Diagram* Info Aplikasi

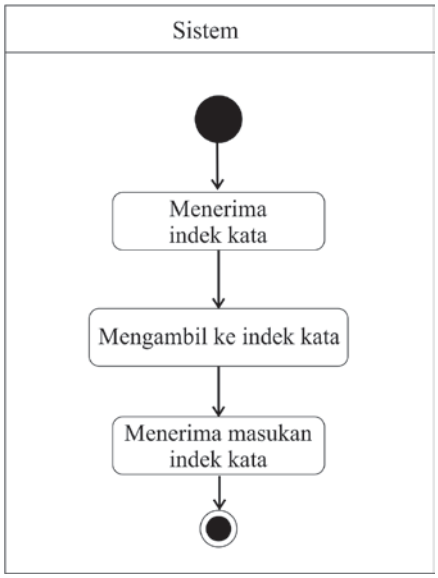
Activity Diagram Info Aplikasi pada aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.6 *Activity Diagram* Info Aplikasi

5. *Activity Diagram* Mendengarkan Contoh Pelafalan

Activity Diagram Mendengarkan Contoh Pelafalan pada aplikasi ini dapat dilihat pada gambar berikut :

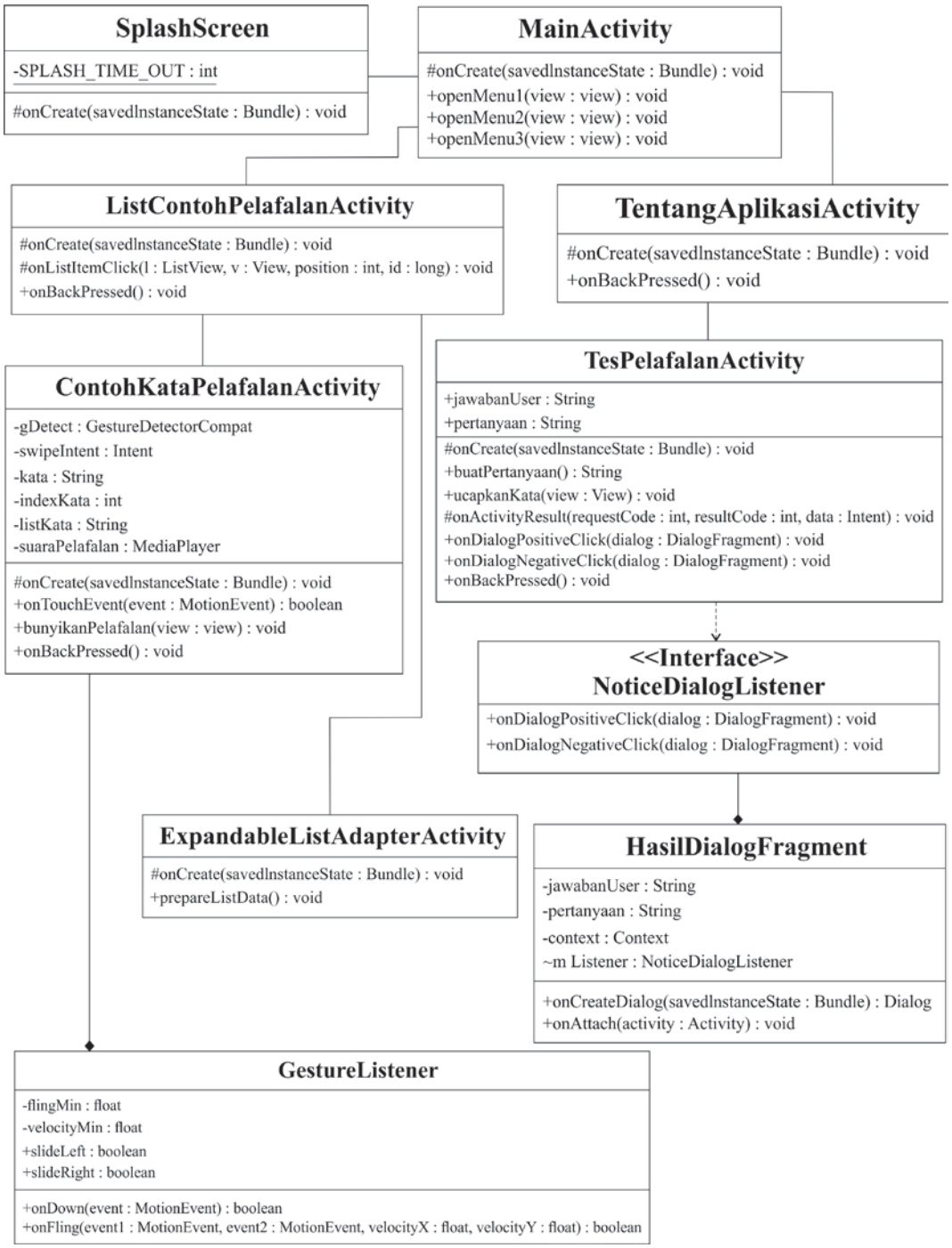


Gambar 3.7 *Activity Diagram* Mendengarkan Contoh Pelafalan

3.1.7.3.3 Class Diagram

Class diagram atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas seperti : kelas *main*, kelas yang menangangi tampilan sistem, kelas yang diambil dari pendefinisian *use case*, dan kelas yang diambil dari pendefinisian data.

Berikut gambaran *class diagram* lafal bahasa Indonesia :



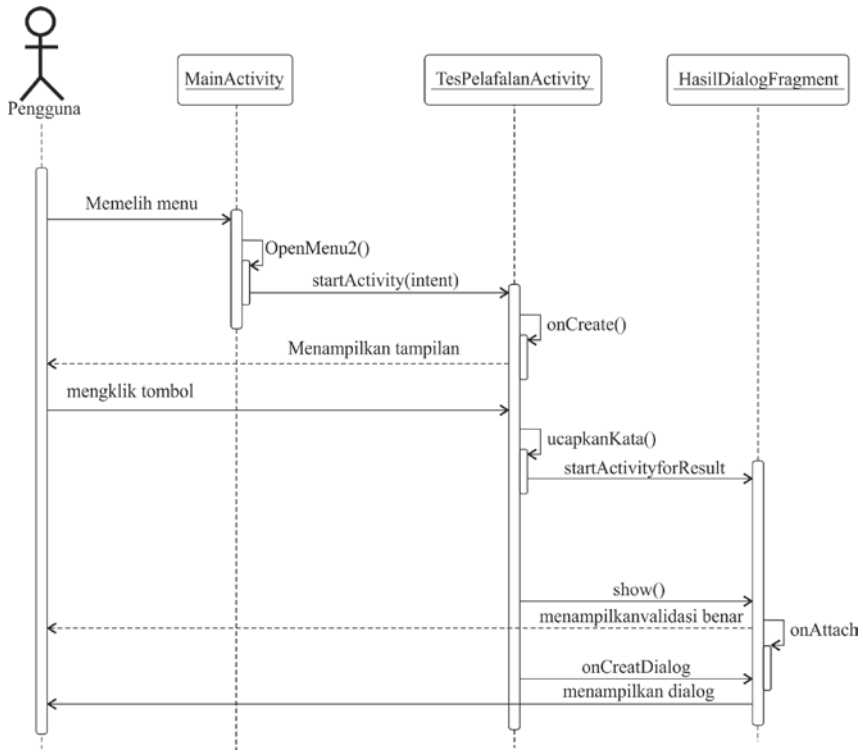
Gambar 3.8 Diagram kelas Aplikasi Lafal Bahasa Indonesia

3.1.7.3.4 Sequence Diagram

Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antarobjek. Untuk menggambar *sequence diagram* harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang dinstalasi menjadi objek itu.

1. *Sequence Diagram* Tes Pelafalan Kata

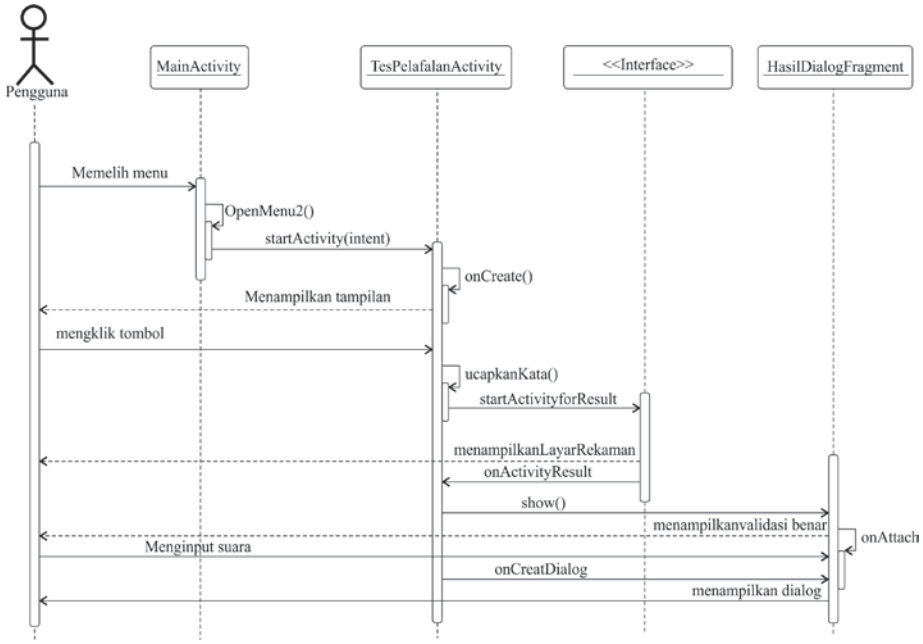
Berikut gambaran sequence diagram tes pelafalan kata :



Gambar 3.9 *Sequence Diagram* Tes Pelafalan Kata

2. *Sequence Diagram* Mengecek Pelafalan Kata

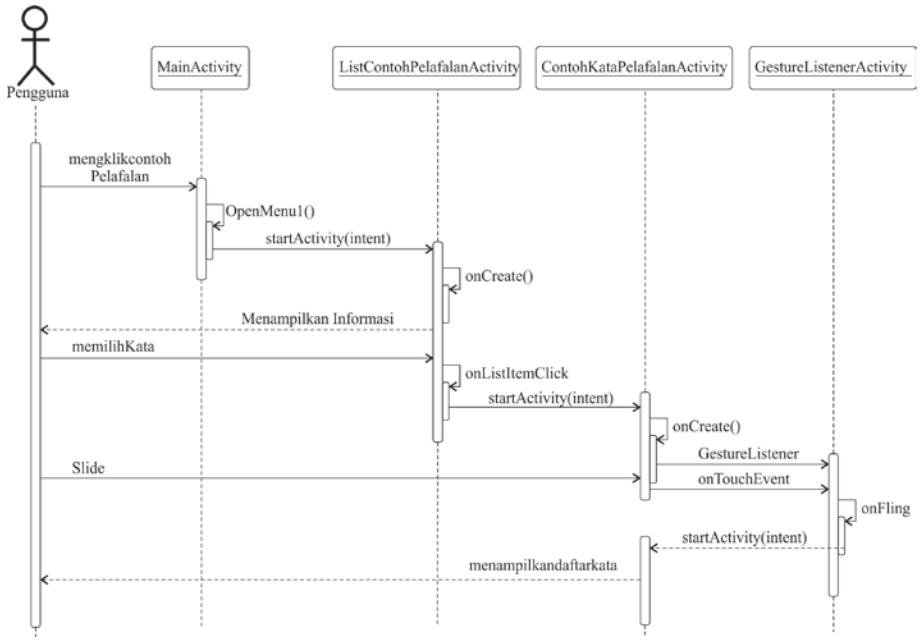
Berikut gambaran sequence diagram mengecek pelafalan kata :



Gambar 3.10 *Sequence Diagram* Mengecek Pelafalan Kata

3. *Sequence Diagram* Contoh Pelafalan Kata

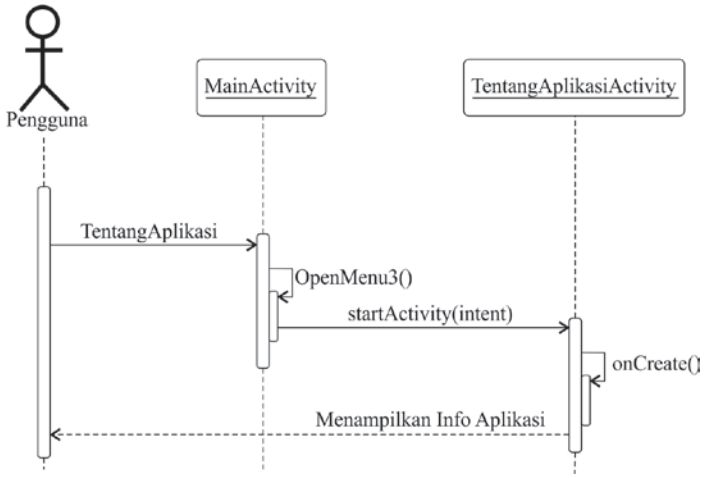
Berikut gambaran sequence diagram contoh pelafalan kata :



Gambar 3.11 *Sequence Diagram* Contoh Pelafalan Kata

4. *Sequence Diagram* Info Aplikasi

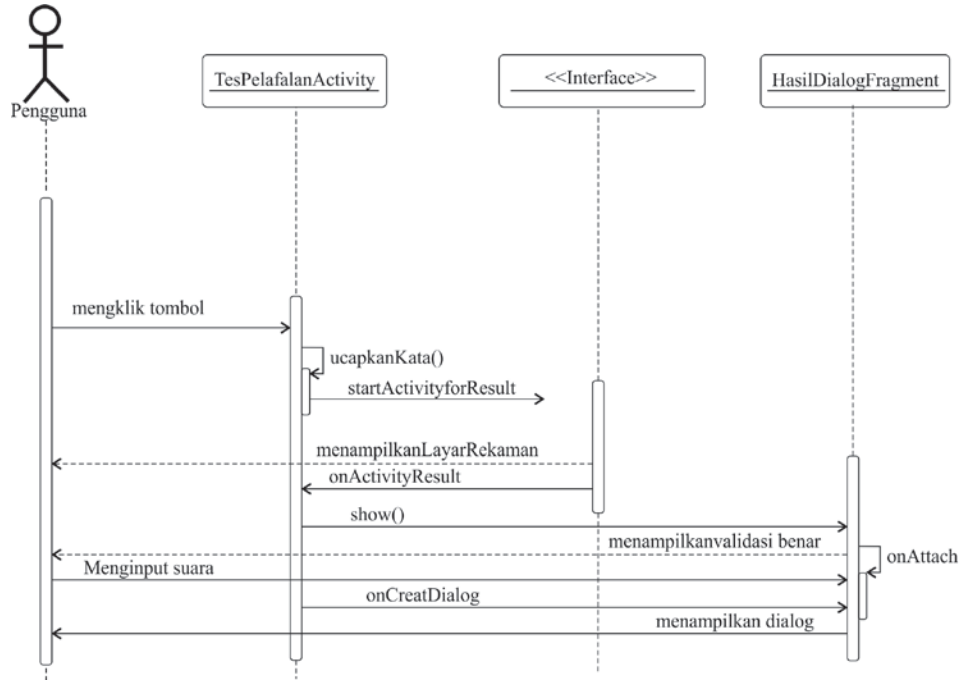
Berikut gambaran sequence diagram info aplikasi:



Gambar 3.12 *Sequence Diagram* Info Aplikasi

5. *Sequence Diagram* Mendengar Pelafalan Kata

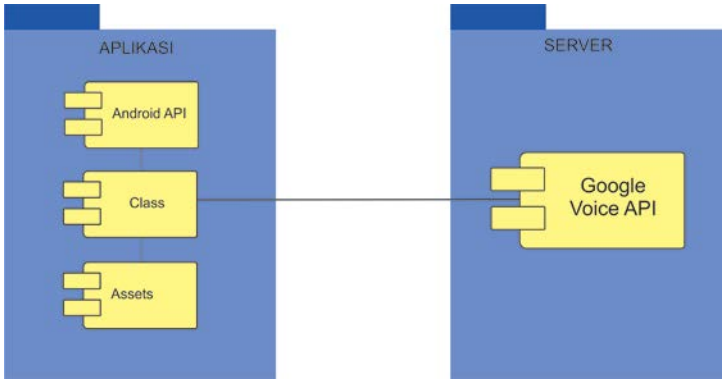
Berikut gambaran sequence diagram mendengarkan pelafalan kata:



Gambar 3.13 *Sequence Diagram* Mendengarkan Pelafalan Kata

3.1.7.3.5 Component Diagram

Component diagram dibuat untuk menunjukkan organisasi dan ketergantungan di antara kumpulan komponen dalam sebuah sistem.



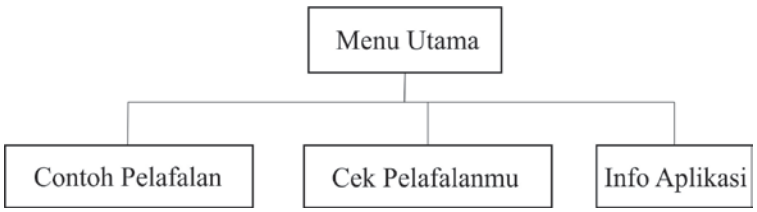
Gambar 3.14 *Component Diagram*

3.2 Perancanan Sistem

Perancangan menu aplikasi digunakan untuk menjelaskan menu, submenu yang terdapat di dalam aplikasi pemlafalaan bahasa Indonesia agar pengguna dapat lebih mudah dalam menggunakannya.

3.2.1 Perancangan Menu Aplikasi

Menu ini dapat diakses oleh pengguna, menu ini berfungsi untuk pengguna melihat aplikasi, melafalkan kata, dan mengecek kata. Perancangan struktur menu diperlihatkan pada gambar berikut.



Gambar 3.15 Perancangan struktur menu

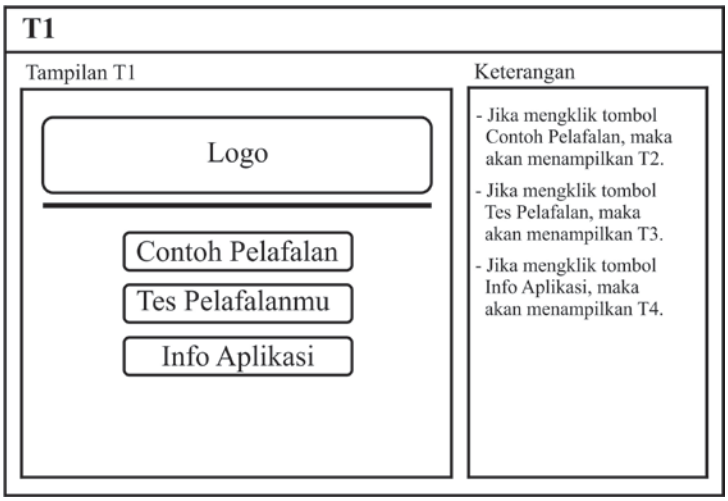
3.2.2 Perancangan Antarmuka

Interface atau antar muka merupakan tampilan dari suatu program aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara program dengan user. Sistem yang akan dibangun diharapkan menyediakan interface yang mudah dipahami dan digunakan oleh user. Berikut ini adalah perancangan antarmuka pada sistem yang akan dibuat. Berikut adalah gambaran

umum dari perancangan antarmuka aplikasi pengguna pada aplikasi pelafalan bahasa Indonesia.

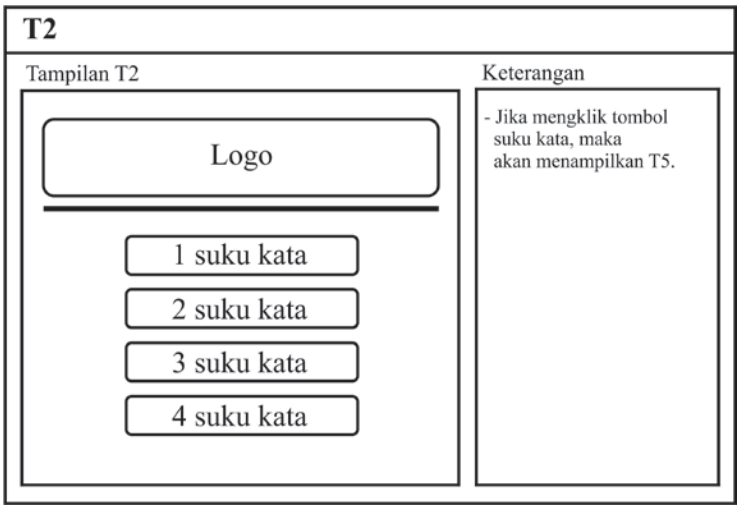
3.2.2.1 Perancangan Antarmuka Aplikasi Pengguna

1. Perancangan Antarmuka Menu Awal Aplikasi



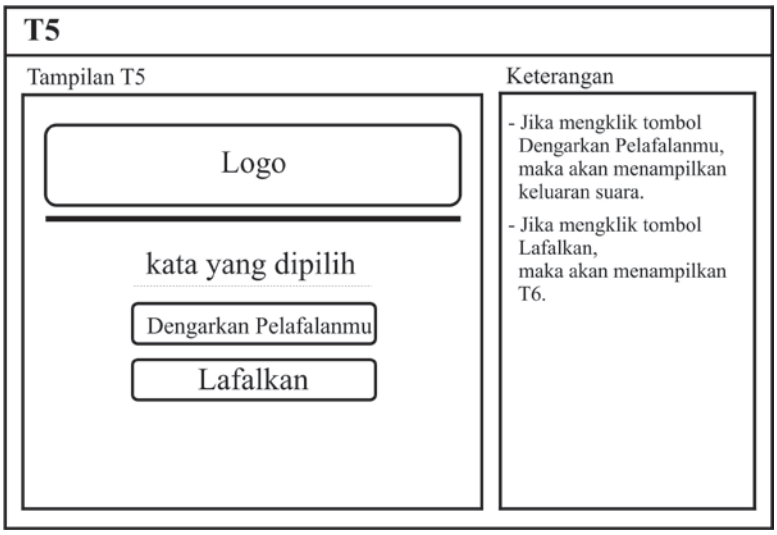
Gambar 3.16 Perancangan Antarmuka Menu Awal Aplikasi

2. Perancangan Antarmuka Contoh Pelafalan Kata



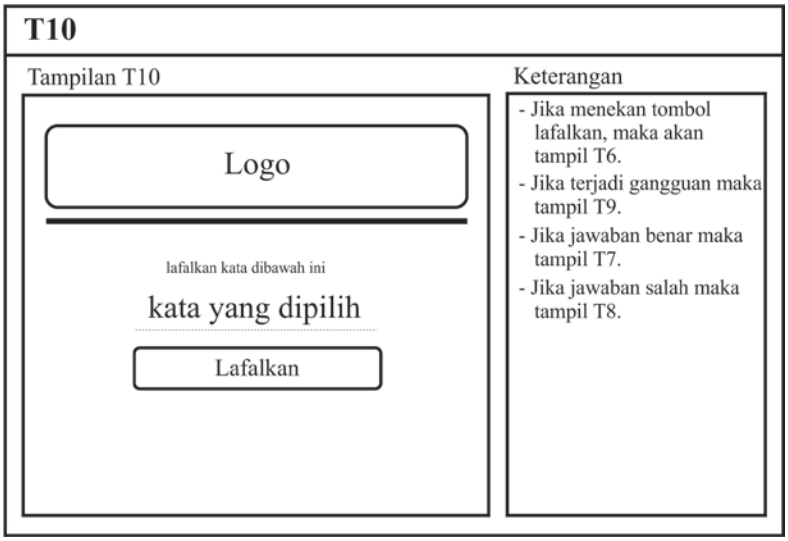
Gambar 3.17 Perancangan Antarmuka Contoh Pelafalan Kata

3. Perancangan Antarmuka Pelafalan Kata Yang Dipilih



Gambar 3.18 Perancangan Antarmuka Pelafalan Kata Yang Dipilih

4. Perancangan Antarmuka Tes Pelafalan Kata



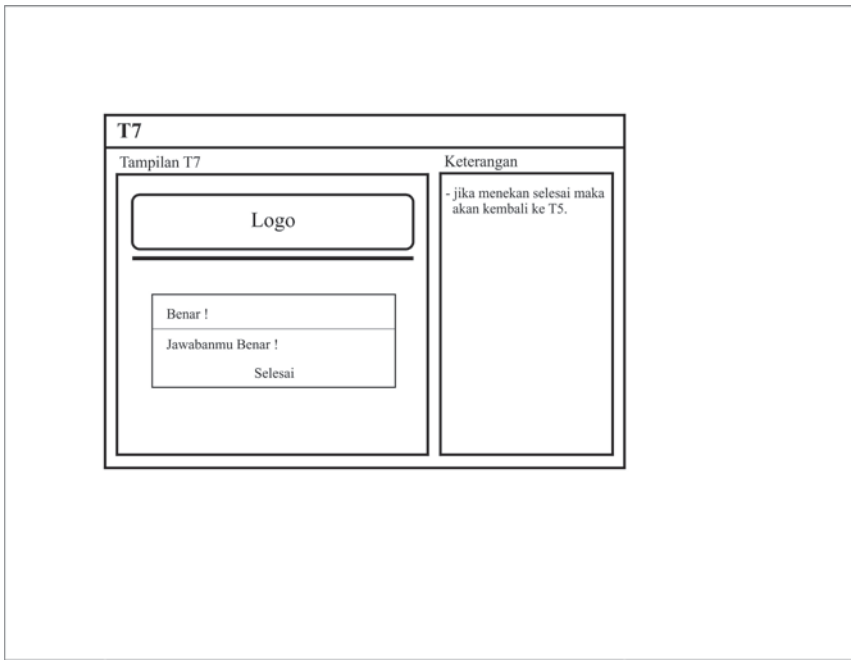
Gambar 3.19 Perancangan Antarmuka Tes Pelafalan Kata

5. Perancangan Antarmuka Pesan Saat Memasukkan Suara



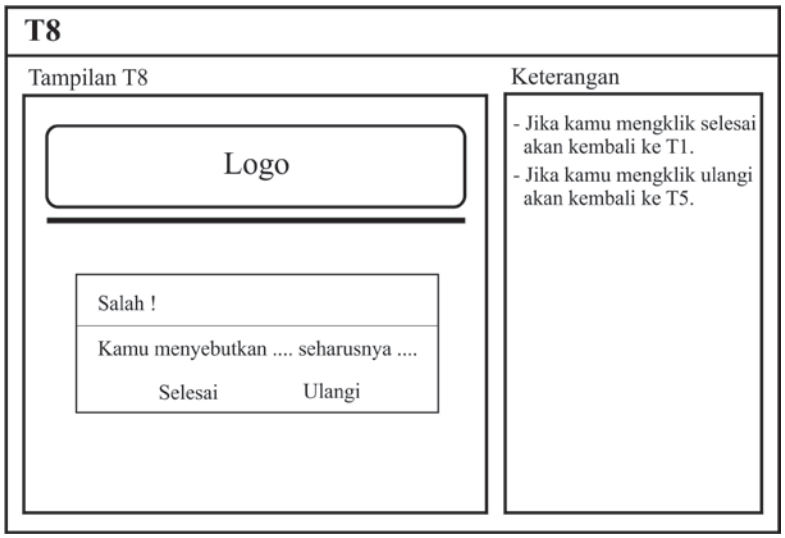
Gambar 3.20 Perancangan Antarmuka Pesan Saat Memasukkan Suara

6. Perancangan Antarmuka Pesan Saat Memasukkan Suara Benar



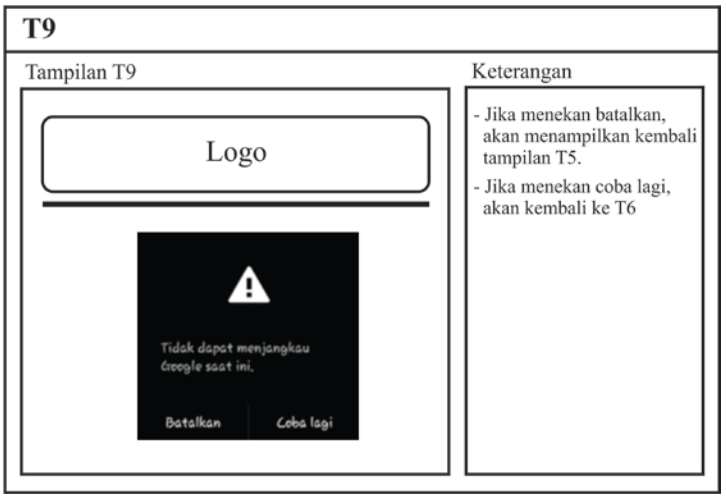
Gambar 3.21 Perancangan Antarmuka Pesan Saat Memasukkan Suara Benar

7. Perancangan Antarmuka Pesan Saat Memasukkan Suara Salah



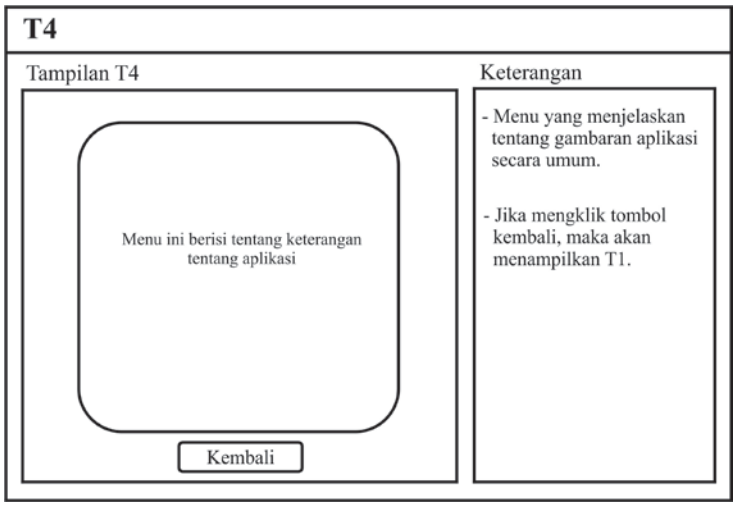
Gambar 3.22 Perancangan Antarmuka Pesan Saat Memasukkan Suara Salah

8. Perancangan Antarmuka Pesan Saat Gangguan Sinyal Internet



Gambar 3.23 Perancangan Antarmuka Pesan Saat Gangguan Sinyal Internet

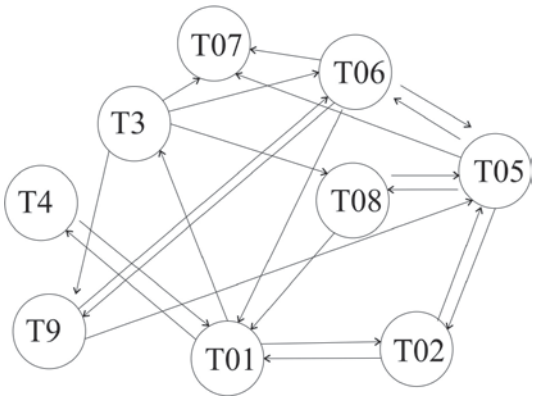
9. Perancangan Antarmuka Info Aplikasi



Gambar 3.24 Perancangan Antarmuka Info Aplikasi

3.2.3 Jaringan Semantik

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukan hubungan antar objek (lingkaran) dan grafis yang menggambarkan formasi antar objek tersebut. Adapun jaringan semantik untuk aplikasi pelafalan bahasa Indonesia sebagai berikut :



Gambar 3.25 Jaringan Semantik