

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem didefinisikan sebagai tahapan yang bertujuan untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan serta menentukan kebutuhan-kebutuhan sehingga dapat dilakukan perbaikan-perbaikannya. Tahapan analisis sistem membahas beberapa analisis, yaitu analisis masalah, analisis prosedur yang sedang berjalan, analisis aplikasi sejenis, analisis arsitektur sistem, analisis kebutuhan non fungsional, dan analisis kebutuhan fungsional.

3.1.1 Analisis Masalah

Analisis Masalah bertujuan untuk menggambarkan permasalahan dan kendala yang ada pada saat melakukan penelitian di SLB B Sukapura. Berdasarkan hasil observasi dan wawancara maka analisis masalah yg ada meliputi hal – hal sebagai berikut :

1. Sulitnya pengajar dan masyarakat untuk berkomunikasi dengan siswa penderita tunawicara.
2. Belum diterapkannya standarisasi bahasa isyarat yang diberikan pemerintah.

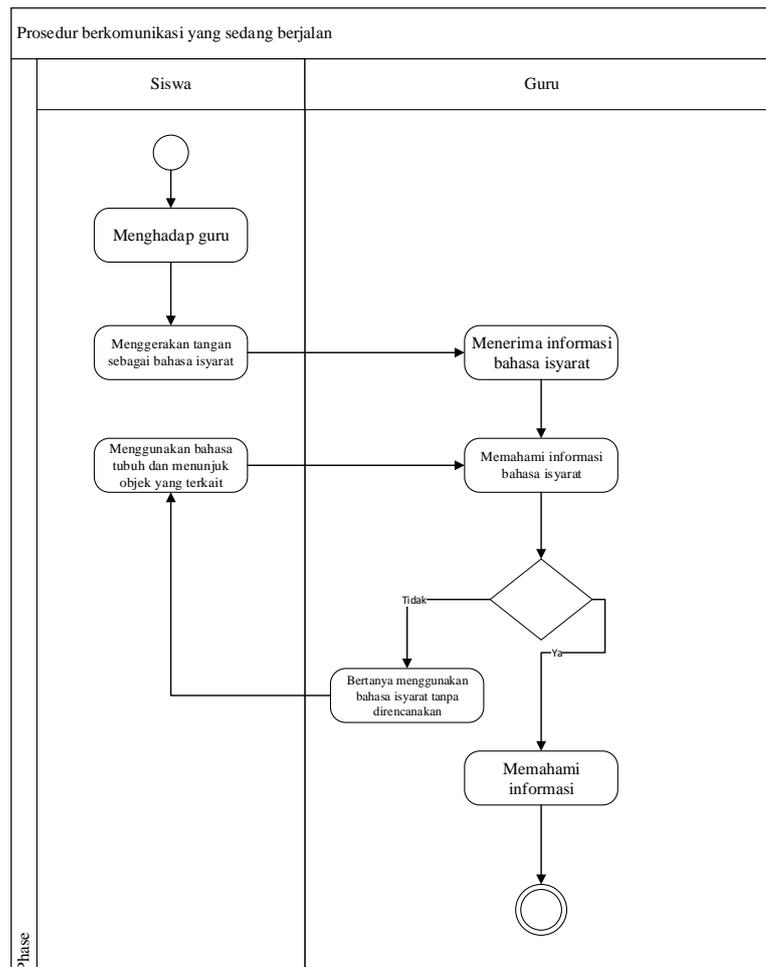
3.1.2 Analisis Prosedur Yang Berjalan

Analisis prosedur yang sedang berjalan bertujuan untuk mempelajari dan mengevaluasi sistem yang sedang berjalan dan merumuskan tujuan yang ingin dicapai dengan sistem yang baru, serta menyusun rencana pembangunan sistem.

Berdasarkan hasil observasi yang telah dilakukan dan wawancara dengan pihak terkait didapat prosedur berkomunikasi yang sedang berjalan yang dapat dilihat pada Gambar 3.1. :

1. Siswa menghadap guru.
2. Siswa menggerakkan tangan sebagai bahasa isyarat.
3. Guru menerima informasi bahasa isyarat dari siswa.
4. Guru memahami informasi bahasa isyarat.

5. Guru tidak memahami informasi bahasa isyarat.
6. Guru bertanya kepada siswa menggunakan bahasa isyarat tanpa direncanakan.
7. Siswa memberikan informasi dengan menggunakan bahasa tubuh dan menunjuk objek yang terkait.
8. Guru memahami informasi bahasa isyarat yang diberikan oleh siswa.

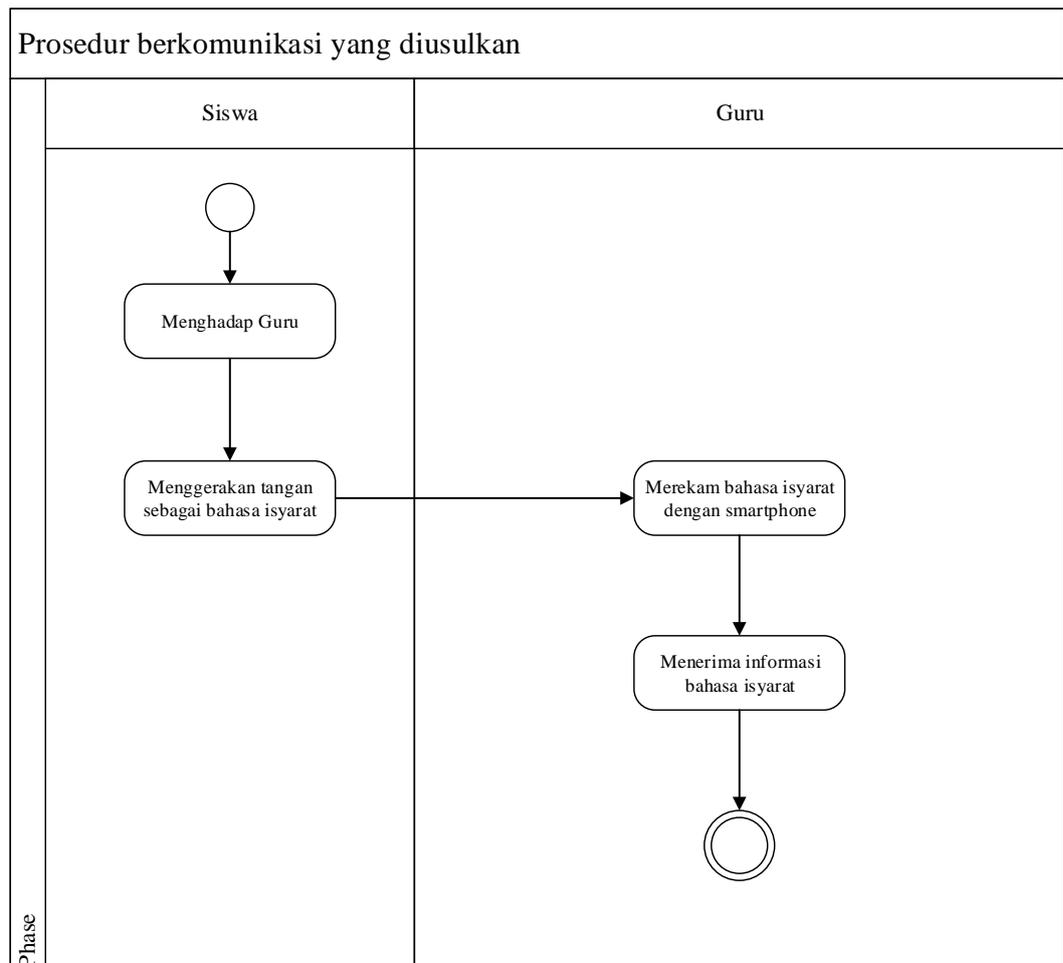


Gambar 3.1. Prosedur berkomunikasi yang sedang berjalan

3.1.3 Analisis Prosedur Yang Diusulkan

Berdasarkan prosedur yang sedang berjalan di SLB B Sukapura. Maka Penulis mengusulkan prosedur sesuai dengan aplikasi. Berikut adalah prosedur yang diusulkan yang dapat dilihat pada Gambar 3.2. :

1. Siswa menghadap guru.
2. Siswa menggerakkan tangan sebagai bahasa isyarat.
3. Guru merekam gerakan bahasa isyarat menggunakan smartphone.
4. Guru menerima informasi bahasa isyarat.



Gambar 3.2. Prosedur Yang Diusulkan

3.1.4 Analisis Jarak

Analisis Jarak adalah jarak ketika berkomunikasi. Berikut adalah analisis jarak ketika berkomunikasi yang dapat dilihat pada Tabel 3.1.

Tabel 3.1. Analisis Jarak

Zona	Jarak	Keterangan
Intim	0 – 45,5 cm	Berkomunikasi dengan keluarga dan orang terdekat.
Personal	45,5 cm – 124 cm	Berkomunikasi dengan teman – teman.
Sosial	124 cm – 378 cm	Berkomunikasi dengan orang yang tak terlalu dikenal.

Publik	378 cm - seterusnya	Berkomunikasi dihadapan banyak orang.
--------	---------------------	---------------------------------------

3.1.5 Analisis Cahaya

Analisis Cahaya adalah kondisi cahaya yang dibutuhkan untuk sistem mendeteksi *gesture* bahasa isyarat. Analisis Cahaya dapat dilihat pada Tabel 3.2.

Tabel 3.2. Analisis Cahaya

Kondisi	Lux	Keterangan
Gelap	0	Tempat gelap atau tidak ada cahaya
Normal	50 – 80	Di dalam ruangan dengan lampu dengan cahaya normal
Redup	20 – 50	Di dalam ruangan dengan lampu kurang cahaya

3.1.6 Analisis Gesture

Gesture SIBI adalah bahasa isyarat yang digunakan oleh penderita tunawicara. *Gesture* huruf dan angka yang digunakan pada aplikasi ini dapat dilihat pada Tabel 3.3.

Tabel 3.3. Analisis Gesture

No	Huruf/Angka	Gesture
1	A	
2	B	
3	C	
4	D	

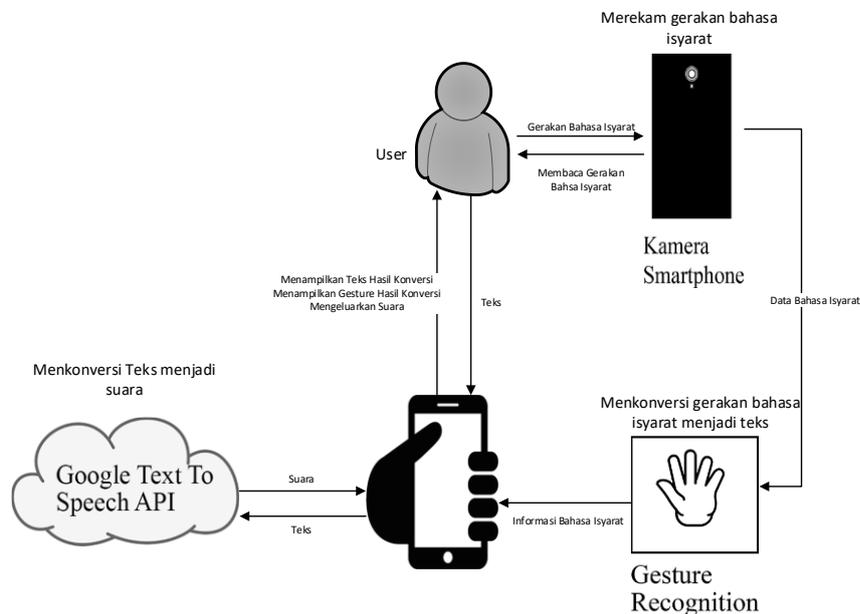
5	E	
6	F	
7	G	
8	H	
9	I	
10	J	
11	K	
12	L	
13	M	
14	N	
15	O	

16	P	
17	Q	
18	R	
19	S	
20	T	
21	U	
22	V	
23	W	
24	X	
25	Y	
26	Z	

3.1.7 Analisis Sistem Yang Diusulkan

Sistem yang akan dibangun adalah aplikasi *gesture to text* dan *text to speech*. Aplikasi ini bertujuan untuk membantu pengajar, masyarakat dan siswa tunawicara di SLB B Sukapura untuk berkomunikasi serta masyarakat agar dapat memahami bahasa isyarat SIBI. Aplikasi ini menggunakan fitur *Gesture Recognition* dan *Text*

To Speech menggunakan *API Google Text to Speech*. Berikut adalah cara kerja dari aplikasi yang akan di buat yang dapat dilihat pada Gambar 3.3 di bawah ini.



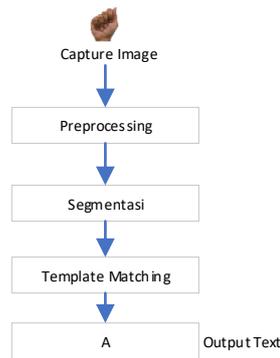
Gambar 3.3. Sistem Yang Diusulkan

Berikut adalah keterangan dari arsitektur sistem yang akan dibangun:

1. User menggerakkan *gesture* bahasa isyarat ke kamera smartphone
2. Kamera Smartphone menangkap *gesture* bahasa isyarat
3. Data *Gesture* Bahasa Isyarat dikirim ke *Gesture Recognition* untuk konversi.
4. Hasil Konversi ditampilkan
5. User menginputkan teks
6. Teks dikonversi menjadi *gesture* bahasa isyarat dan mengeluarkan suara

3.1.8 Analisis Metode *Gesture To Text*

Analisis metode *Gesture To Text* untuk menggambarkan bagaimana sistem yang akan dibuat dapat melakukan pendeteksian *gesture* bahasa isyarat. *Gesture To Text* dilakukan dengan mengenali bentuk *gesture* bahasa isyarat, lalu setelah mengenali bentuk *gesture* bahasa isyarat akan di olah menggunakan *Template Matching* untuk mencocokkan bentuk *gesture* bahasa isyarat masukkan dengan *gesture template*.



Gambar 3.4. Proses *Template Matching*

1. Pengambilan *Gesture* Bahasa Isyarat

Pengambilan *Gesture* Bahasa Isyarat ini adalah tahap proses pengambilan *Gesture* menggunakan sensor kamera pada smartphone android.

Pengambilan ini dilakukan pada ruangan yang memiliki cukup cahaya.

Tahap ini menghasilkan citra masukkan dengan format *bitmap* RGB.

2. *Preprocessing*

Didalam tahapan *Preprocessing* terdapat proses *grayscale*.

3. Segmentasi

Tahapan selanjutnya segmentasi untuk mendapatkan bentuk *gesture* tangan dengan deteksi tepi *canny*, *contour*. Setelah didapatkan *gesture* tangan maka akan memotong (*Cropping*) area *contour*.

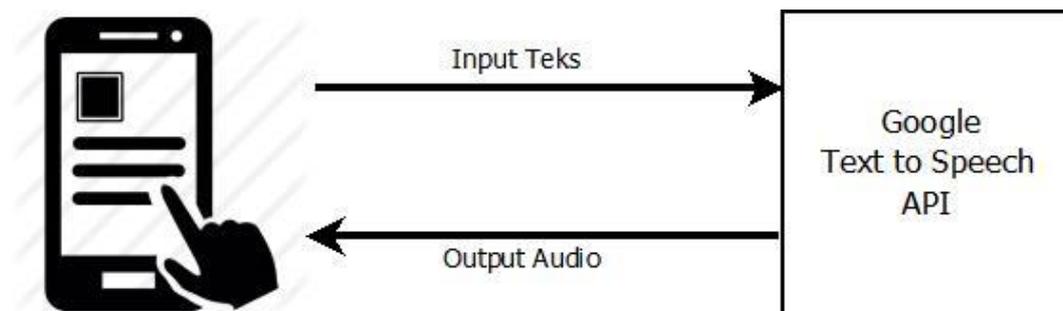
4. *Template Matching*

Tahap *Template Matching* untuk membandingkan bentuk *gesture* hasil segmentasi dengan *template gesture* bahasa isyarat. Jika bentuk *gesture* menemukan bentuk yang sama dengan *template* maka akan memunculkan huruf dari *gesture* tersebut.

3.1.9 Analisis Google Text To Speech API

Penggunaan Google *Text To Speech* API memungkinkan developer untuk mengkonversi teks menjadi suara. Layanan ini dapat diproses secara offline dan online. Pada pemrosesan secara offline, bahasa yang didukung terbatas pada bahasa yang terdapat pada masing-masing device/smartphone. Berbeda dengan pemrosesan secara online yang memungkinkan dukungan bahasa yang lengkap.

Namun dalam penggunaannya pengguna diharuskan memiliki koneksi internet untuk dapat mengakses database pada server Google secara online. Alur penggunaan Google *Text To Speech* API untuk dapat mengkonversi teks menjadi suara (ucapan).



Gambar 3.5. Alur Google Text To Speech API

Dibawah ini adalah penjelasan dari gambar 20 tentang cara kerja google *text to speech* :

1. Input teks dan bahasa dikirim device ke Google TTS melalui http request
2. *Google TTS* mengkonversi tulisan yang dikirimkan melalui *http request* sebelumnya kedalam bentuk suara (*audio*)
3. Setelah proses konversi selesai *Google TTS* akan menjalankan suara ke *device* secara *realtime*

Input teks pada *Google TTS* harus memperhatikan tanda baca dan juga jumlah karakternya. Titik dan juga koma akan mempengaruhi artikulasi. Setelah titik “.” waktu jeda akan lebih lama karena titik merupakan akhir dari sebuah kalimat. Tanda baca “!?,” juga menunjukkan akhir potongan pada kalimat yang mengandung unsur seru dan juga tanya, namun waktu jeda tidak terlalu lama.

Google Text to Speech mendukung 31 bahasa dalam proses *Text To Speech*. Bahasa yang didukung oleh *Google* dapat dilihat pada Tabel 3.4.

Tabel 3.4. Bahasa Yang Didukung Google Text To Speech API

No	Nama Bahasa	Google Speak Code	Google Speak Code (asumsi)
1	<i>Arabic</i>	ar	
2	<i>Bulgarian</i>	-	ru

3	<i>Catalan</i>	ca	
4	<i>Chinese Simplified</i>	zh-CN	
5	<i>Chinese Traditional</i>	zh-TW	
6	<i>Czech</i>	cs	
7	<i>Danish</i>	da	
8	<i>Dutch</i>	nl	
9	<i>English</i>	en	
10	<i>Estonian</i>	-	nl
11	<i>Finnish</i>	fi	
12	<i>French</i>	fr	
13	<i>German</i>	de	
14	<i>Greek</i>	el	
15	<i>Haitian Creole</i>	ht	
16	<i>Hebrew</i>	-	iw
17	<i>Hindi</i>	hi	
18	<i>Hmong Daw</i>	-	nl
19	<i>Hungarian</i>	hu	
20	<i>Indonesian</i>	id	
21	<i>Italian</i>	it	
22	<i>Japanese</i>	ja	
23	<i>Korean</i>	ko	
24	<i>Latvian</i>	lv	
25	<i>Lithuanian</i>	-	nl
26	<i>Malay</i>	-	id
27	<i>Norwegian</i>	no	
28	<i>Persian</i>	-	ar
29	<i>Polish</i>	pl	
30	<i>Portuguese</i>	pt	
31	<i>Romanian</i>	ro	

32	<i>Russian</i>	ru	
33	<i>Slovak</i>	sk	
34	<i>Slovenian</i>	-	nl
35	<i>Spanish</i>	es	
36	<i>Swedish</i>	sv	
37	<i>Thai</i>	th	
38	<i>Turkish</i>	tr	
39	<i>Ukrainian</i>	-	ru
40	<i>Urdu</i>	-	ar
41	<i>Vietnamese</i>	vi	

3.1.10 OpenCV For Android SDK

OpenCV Library Android adalah library yang digunakan dalam pengolahan citra masukkan pada aplikasi yang akan dibangun. Versi OpenCV yang digunakan adalah OpenCV Android. Penggunaan library OpenCV sangat vital bagi fungsionalitas utama pada aplikasi yang akan dibangun, tanpa penggunaan OpenCV, tujuan utama dari aplikasi ini tidak bisa tercapai.



Gambar 3.6. OpenCV For Android SDK

Cara menggunakan OpenCV pada android studio adalah sebagai berikut :

1. Download OpenCV for Android SDK
2. Buat Project Baru
3. Import Module opencv sdk yang sudah didownload

4. Edit build.gradle pada folder opencv library dan sesuaikan
5. Sinkron

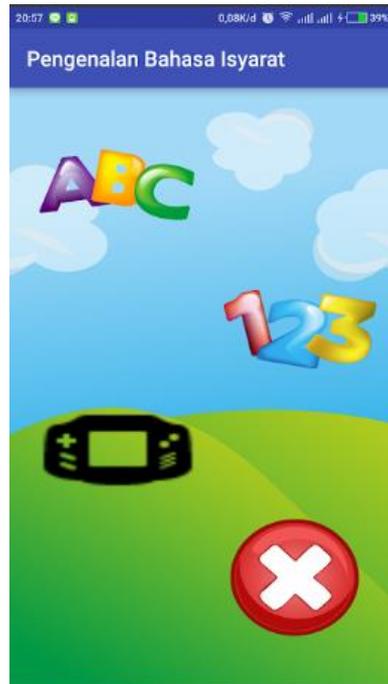
3.1.11 Analisis Aplikasi Sejenis

Analisis aplikasi sejenis merupakan analisis yang akan membahas mengenai aplikasi yang menjadi acuan dalam pembangunan sebuah aplikasi. Dalam aplikasi sejenis akan dibahas mengenai berbagai hal yang ada didalam aplikasi sejenis diantaranya konten, cara menggunakan dan komponen-komponen apa saja yang ada di dalam aplikasi tersebut sehingga dapat menjadi acuan bagi kebutuhan yang akan dibangun.

3.1.11.1 SIBI (Sistem Isyarat Bahasa Indonesia)

SIBI (Sistem Isyarat Bahasa Indonesia) adalah satu diantara aplikasi pengenalan bahasa isyarat SIBI di google playstore. Dibuat oleh LGA Indie Dev. Aplikasi ini membantu penderita tunarungu untuk mempermudah mengenal bahasa isyarat SIBI. Aplikasi ini juga memiliki materi pengenalan abjad, angka, dan materi permainan. Didalam materi permainan berisi susunan abjad atau angka. Aplikasi ini menjadi salah satu alternatif media pembelajaran bagi pengguna smartphone android.

Tampilan awal pada aplikasi SIBI (Sistem Isyarat Bahasa Indonesia) dapat dilihat pada Gambar 3.7.

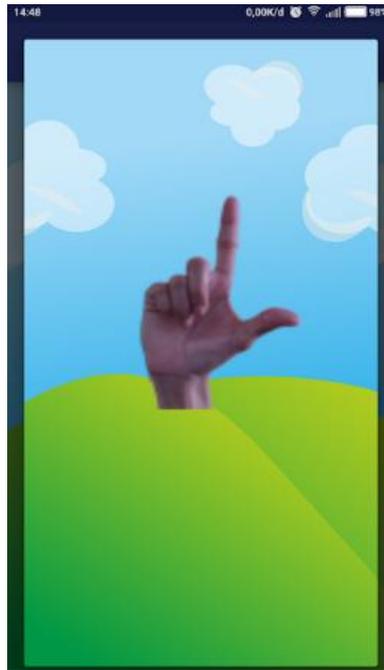


Gambar 3.7. Tampilan Awal

Berikut adalah menu melihat huruf bahasa isyarat SIBI. Disini pengguna dapat melihat huruf dalam bahasa isyarat SIBI. Tampilan menu huruf dapat dilihat pada Gambar 3.8 dan Gambar 3.9.



Gambar 3.8. Menu Huruf



Gambar 3.9. Menu Huruf *Gesture*

Berdasarkan analisis pada aplikasi SIBI (Sistem Isyarat Bahasa Indonesia) maka dapat disimpulkan kelebihan dan kekurangan pada aplikasi yang dapat dilihat pada Tabel 3.5.

Tabel 3.5. Kelebihan dan Kekurangan Aplikasi SIBI

No	Kelebihan	Kekurangan
1.	Mudah digunakan.	Hanya dapat dilihat, tidak dengan praktik.
2.	Cara mengenalkan bahasa isyarat SIBI sangat jelas.	Hanya bersifat seperti kamus.
3.	Mudah dipahami untuk belajar.	

3.1.11.2 G-Selfie

G-Selfie (*Gesture Selfie*) adalah satu diantara aplikasi fotografi yang menggunakan *gesture* untuk mengambil gambar. Dibuat oleh MACRON. Aplikasi ini dapat mengambil gambar tanpa menekan *tombol*, melainkan hanya dengan menggerakkan tangan maka aplikasi akan mengambil gambar secara *otomatis*.

Tampilan aplikasi G-Selfie (*Gesture Selfie*) ketika menggunakan mode menggerakkan tangan. Aplikasi akan mengambil secara otomatis pada saat tangan pengguna membentuk hati yang dapat dilihat pada Gambar 3.10.



Gambar 3.10. Tampilan Aplikasi

Berdasarkan analisis pada aplikasi G-Selfie (*Gesture Selfie*) maka dapat disimpulkan kelebihan dan kekurangan pada aplikasi yang dapat dilihat pada Tabel 3.6.

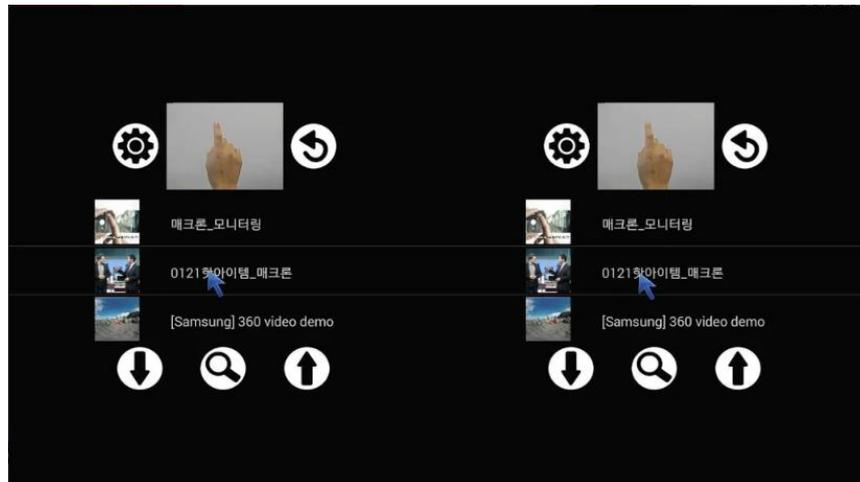
Tabel 3.6. Kelebihan dan Kekurangan Aplikasi G-Selfie

No.	Kelebihan	Kekurangan
1.	Mudah digunakan	Terkadang tidak terbaca <i>gesture</i> tangannya
2.	Pengenalan gerakan tangan sangat baik.	
3.	Mempermudah dalam melakukan foto selfie.	

3.1.11.3 VR Gesture Player

VR Gesture Player adalah aplikasi pemutar video yang menggunakan VR (*Virtual Reality*) dan menggunakan *gesture* tangan. Dibuat oleh MACRON. Aplikasi ini dapat memutar video dengan VR dan mengganti video dengan cara

menggerakkan tangan yang direkam oleh kamera belakang *smartphone*, selain itu *gesture* tangan juga dapat mengatur aplikasi. Tampilan dari aplikasi *VR Gesture Player* dapat dilihat pada Gambar 3.11 dan Gambar 3.12.



Gambar 3.11. Tampilan Aplikasi *VR Gesture Player*



Gambar 3.12. Tampilan Pengaturan Aplikasi *VR Gesture Player*

Berdasarkan analisis pada aplikasi *VR Gesture Player* maka dapat disimpulkan kelebihan dan kekurangan pada aplikasi yang dapat dilihat pada tabel Tabel 3.7.

Tabel 3.7. Kelebihan dan Kekurangan *VR Gesture Player*

No	Kelebihan	Kekurangan
1.	Mudah digunakan	Akurasi <i>pointer</i> dengan <i>gesture</i> kurang

2.	Pengenalan <i>gesture</i> tangan sangat baik.	
3.	Mempermudah mengatur aplikasi tanpa menyentuh layar <i>smartphone</i> .	

3.1.12 Spesifikasi Kebutuhan Perangkat Lunak

Berdasarkan analisis masalah yang telah dilakukan, maka didapat spesifikasi kebutuhan perangkat lunak pada aplikasi *gesture to text* dan *text to speech*. Spesifikasi kebutuhan perangkat lunak dilakukan berdasarkan kebutuhan pada aplikasi *gesture to text* dan *text to speech* sehingga perangkat lunak harus memenuhi kriteria yang dapat dilihat pada Tabel 3.8.

Tabel 3.8. Spesifikasi Kebutuhan Fungsional

No	Kode Kebutuhan	Deskripsi Kebutuhan
1	SKPL-F-01	Sistem menyediakan fasilitas melakukan <i>gesture to text</i> .
2	SKPL-F-02	Sistem menyediakan fasilitas melakukan <i>text to gesture</i> .
3	SKPL-F-03	Sistem menyediakan fasilitas <i>text to speech</i>

Spesifikasi kebutuhan non fungsional akan memberikan batasan layanan yang ditawarkan sistem seperti batasan waktu, batasan pengembangan proses standarisasi dan lain-lain. Spesifikasi kebutuhan non fungsional dapat dilihat pada tabel Tabel 3.9.

Tabel 3.9. Spesifikasi Kebutuhan Non Fungsional

Kode SKPL	Spesifikasi Kebutuhan Fungsional
SKPL-NF-01	Sistem yang dibangun menggunakan <i>platform</i> android
SKPL-NF-02	Sistem yang dibangun minimal menggunakan sistem operasi android versi 5.0. Lollipop.
SKPL-NF-03	Sistem yang dibangun menggunakan <i>Gesture Recognition</i>
SKPL-NF-04	Sistem yang dibangun menggunakan Google API seperti <i>Google Text to Spech API</i>
SKPL-NF-05	Sistem yang dibangun dengan menggunakan spesifikasi hardware yang memenuhi standar minimum kebutuhan

3.1.13 Analisis Kebutuhan Non-Fungsional

Analisis dan kebutuhan non-fungsional meliputi analisis dan kebutuhan perangkat keras serta analisis dan kebutuhan perangkat lunak. Adapun kebutuhan non-fungsional untuk menjalankan aplikasi ini meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan pengguna sistem yang akan menggunakan aplikasi. Analisis kebutuhan non-fungsional bertujuan agar aplikasi yang dibangun dapat digunakan sesuai dengan kebutuhan pengguna aplikasi dalam mencari informasi yang dibutuhkan.

3.1.13.1 Kebutuhan Perangkat Keras

Berikut adalah kebutuhan perangkat keras yang digunakan dalam membangun dan mengimplementasikan Aplikasi *gesture to text* dan *text to speech* yang dapat dilihat pada Tabel 3.10.

Tabel 3.10. Kebutuhan Perangkat Keras Smartphone

No	Perangkat Keras	Spesifikasi
1	<i>Processor</i>	1.2 GHz
2	<i>RAM</i>	2 GB
3	Media penyimpanan internal	8 GB
4	Media penyimpanan eksternal	8 GB
5	Kamera	0.3 Megapixel

3.1.13.2 Kebutuhan Perangkat Lunak

Berikut adalah spesifikasi minimum perangkat lunak yang dibutuhkan agar sistem dapat berjalan dengan baik yang dapat dilihat pada tabel Tabel 3.11.

Tabel 3.11. Kebutuhan Perangkat Lunak Smartphone

No	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Android Lollipop

3.1.13.3 Analisis Pengguna

Analisis pengguna yaitu analisis yang menjelaskan *user* atau pengguna dari aplikasi, dalam hal ini aplikasi *gesture to text* dan *text to speech*. Analisis pengguna dapat dilihat pada Tabel 3.12.

Tabel 3.12. Analisis Pengguna Sistem

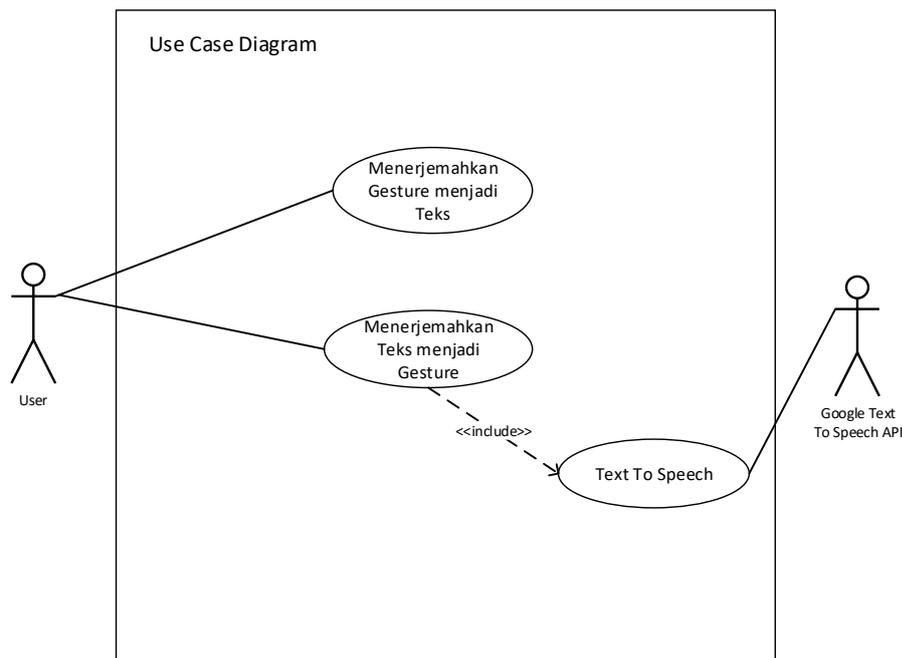
No	Pengguna	Usia	Pengalaman
1	Guru	Minimal 20 tahun	Sudah pernah dan bisa menggunakan aplikasi di smartphone bersistem operasi android
2	Siswa	Minimal 6 tahun	Sudah pernah menggunakan dan bisa aplikasi di smartphone bersistem operasi android
3	Masyarakat	Minimal 6 tahun	Sudah pernah menggunakan dan bisa aplikasi di smartphone bersistem operasi android

3.1.14 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan suatu tahapan analisis dengan melakukan penggambaran, perencanaan dan pembuatan sketsa, serta pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh. Analisis kebutuhan fungsional juga dapat diartikan sebagai alat bantu yang digunakan dalam menggambarkan secara umum sistem yang akan dibangun, yaitu dengan pendekatan berorientasi objek. Tahapan analisis ini menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik. Analisis kebutuhan fungsional pada sistem atau aplikasi akan digambarkan menggunakan *tools UML (Unified Modelling Language)* dalam bentuk diagram. Terdapat beberapa diagram pada *tools UML* yang umum digunakan untuk menggambarkan fungsional sistem antara lain *Use Case Diagram*, *Skenario Use Case*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*.

3.1.14.1 Use Case Diagram

Use Case Diagram mendeskripsikan sebuah interaksi antar satu atau lebih aktor dengan sistem yang akan dibuat. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Analisis *use case diagram* pada sistem yang akan dibangun dapat dilihat pada Gambar 3.13.



Gambar 3.13. Use Case Diagram

1. Identifikasi Aktor

Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dibangun. Berikut adalah aktor-aktor yang terdapat pada sistem yang akan dibangun. Identifikasi aktor pada aplikasi *gesture to text* dan *text to speech* dapat dilihat pada Tabel 3.13.

Tabel 3.13. Identifikasi Aktor

No	Aktor	Deskripsi
1	User	Merupakan aktor dari aplikasi yang dibangun yang berperan untuk menggunakan aplikasi.
2	Google Text To Speech API	Merupakan aktor yang berperan menkonversi teks menjadi suara.

2. Identifikasi *Use Case Diagram*

Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Berikut adalah use case yang terdapat pada sistem yang akan dibangun yang dapat dilihat pada Tabel 3.14.

Tabel 3.14. Identifikasi *Use Case Diagram*

No	Use Case	Deskripsi
1.	Menerjemahkan <i>Gesture</i> menjadi Teks	Proses yang dilakukan user untuk menginputkan gerakan bahasa isyarat dan merubah bentuk tangan menjadi teks.
2.	Menerjemahkan Teks menjadi <i>Gesture</i>	Proses yang dilakukan user untuk menginputkan teks dan merubah teks menjadi gambar bentuk tangan.
3.	<i>Text To Speech</i>	Proses merubah teks menjadi suara.

3. Skenario *Use Case*

Skenario *Use Case* mendeskripsikan urutan langkah-langkah dalam proses bisnis, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor.

1. Skenario *Use Case* Menerjemahkan *Gesture* menjadi Teks

Tabel 3.15. SKPL Fungsional Menerjemahkan *Gesture* Menjadi Teks

Kode SKPL	Deskripsi
SKPL-F-01	Sistem menyediakan fasilitas menerjemahkan <i>gesture</i> menjadi teks

Tabel 3.16. Skenario Menerjemahkan *Gesture* menjadi Teks

<i>Use Case Name</i>	Menerjemahkan <i>Gesture</i> Menjadi Teks
<i>Related Requirement</i>	SKPL-F-01
<i>Goal in Content</i>	Untuk menerjemahkan gerakan bahasa isyarat ke teks
<i>Precondition</i>	User belum melakukan gerakan bahasa isyarat
<i>Successful End Condition</i>	Sistem menampilkan teks hasil konversi dari gerakan bahasa isyarat
<i>Failed End Condition</i>	Sistem tidak menampilkan teks.

<i>Primary Actor</i>	User	
<i>Secondary Actor</i>	Sistem	
<i>Triger</i>	User melakukan gerakan bahasa isyarat	
<i>Included Cases</i>	-	
<i>Main Flow</i>	Step	Action
	1	Sistem menampilkan menu.
	2	User memilih menu <i>Gesture To Text</i> .
	3	Sistem menampilkan menu <i>Gesture To Text</i> .
	4	User melakukan gerakan bahasa isyarat.
	5	Sistem merekam gerakan bahasa isyarat.
	6	Sistem menampilkan teks hasil konversi dari gerakan bahasa isyarat
<i>Extention</i>	Step	Branching Action
	6.1	Sistem tidak menampilkan teks hasil konversi.

2. Skenario *Use Case* Menerjemahkan Teks menjadi *Gesture*

Tabel 3.17. SKPL Fungsional Menerjemahkan Teks menjadi *Gesture*

Kode SKPL	Deskripsi
SKPL-F-02	Sistem menyediakan fasilitas menerjemahkan teks menjadi <i>gesture</i> .

Tabel 3.18. Skenario Menerjemahkan Teks menjadi *Gesture*

<i>Use Case Name</i>	Menerjemahkan Teks Menjadi <i>Gesture</i>
<i>Related Requirement</i>	SKPL-F-02
<i>Goal in Content</i>	Untuk menerjemahkan teks menjadi gambar gerakan bahasa isyarat
<i>Precondition</i>	User belum memasukkan teks
<i>Successful End Condition</i>	Sistem menampilkan gambar bahasa isyarat dari teks yang diinputkan oleh user
<i>Failed End Condition</i>	Masukkan teks
<i>Primary Actor</i>	User
<i>Secondary Actor</i>	Sistem
<i>Triger</i>	Sistem memanggil gambar gerakan bahasa isyarat dari teks

<i>Included Cases</i>	-	
<i>Main Flow</i>	Step	Action
	1	User memilih menu <i>text to gesture</i>
	2	Sistem menampilkan menu <i>text to gesture</i>
	3	User menginputkan teks
	4	Sistem menampilkan gambar gerakan bahasa isyarat
<i>Extention</i>	Step	Branching Action
	3.1	Masukkan teks

3. Skenario Use Case Text To Speech

Tabel 3.19. SKPL Fungsional Text To Speech

Kode SKPL	Deskripsi
SKPL-F-03	Sistem menyediakan fasilitas <i>text to speech</i>

Tabel 3.20. Skenario Text To Speech

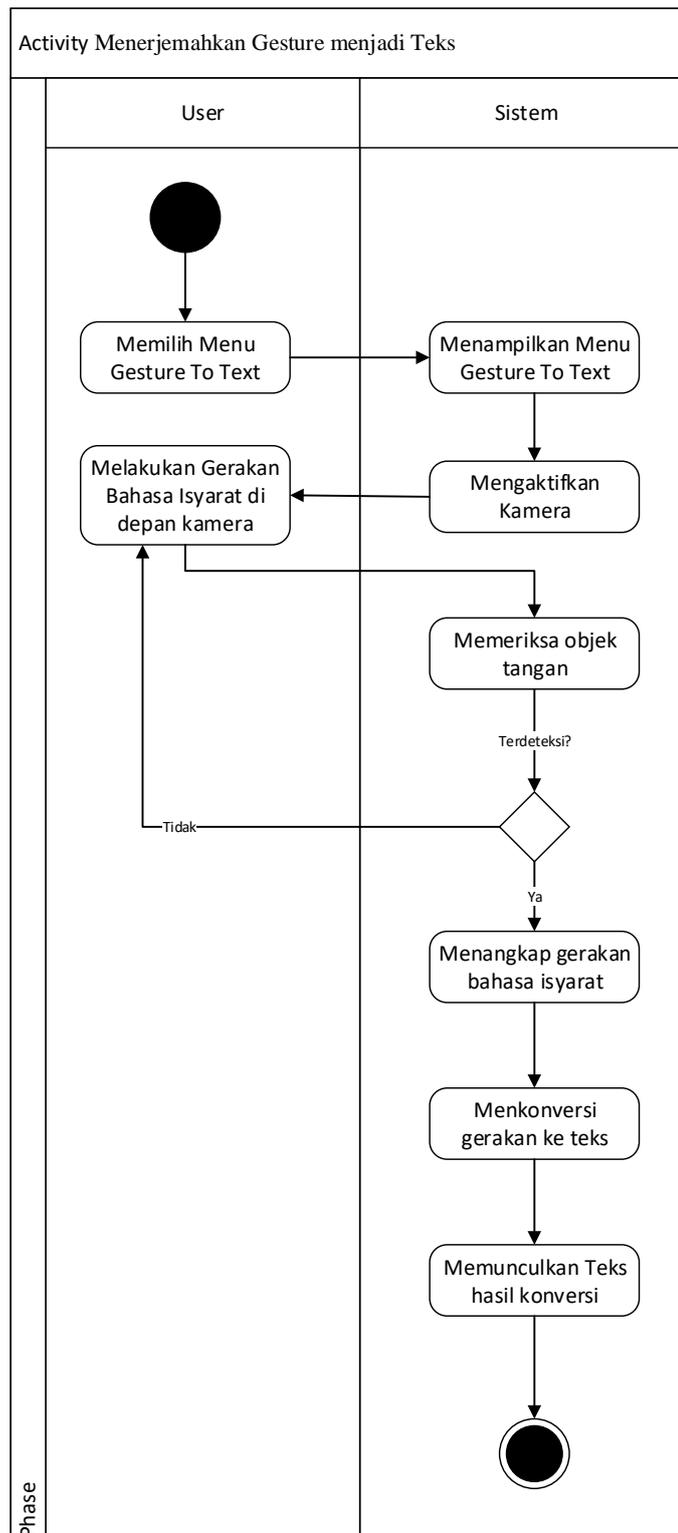
<i>Use Case Name</i>	<i>Text To Speech</i>	
<i>Related Requirement</i>	SKPL-F-03	
<i>Goal in Content</i>	Untuk mengkonversi teks hasil konversi ke suara	
<i>Precondition</i>	User belum menekan <i>tombol</i> suara	
<i>Successful End Condition</i>	Sistem mengeluarkan suara berdasarkan teks yang diinputkan	
<i>Failed End Condition</i>	-	
<i>Primary Actor</i>	User	
<i>Secondary Actor</i>	-	
<i>Triger</i>	-	
<i>Included Cases</i>	-	
<i>Main Flow</i>	Step	Action
	1	User menekan <i>tombol</i> suara.
	2	Sistem mengkonversi teks menjadi suara
	3	Sistem mengeluarkan suara.

3.1.14.2 Activity Diagram

Activity Diagram merupakan diagram yang menggambarkan hubungan aktifitas yang terjadi dalam sistem atau aplikasi antara pengguna yang menggunakan sistem atau aplikasi.

1. *Activity Menerjemahkan Gesture menjadi Teks*

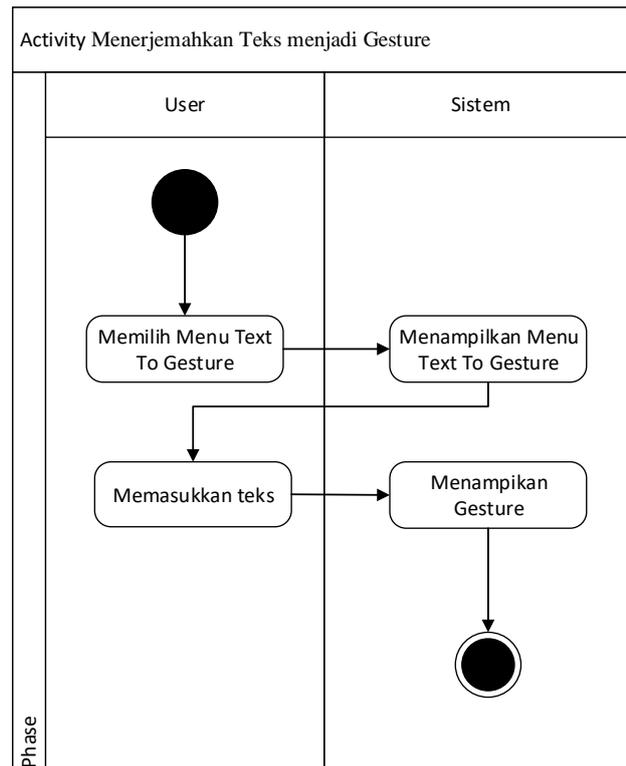
Activity Diagram Menerjemahkan Gesture menjadi Teks digunakan ketika user ingin menerjemahkan bahasa isyarat. *Activity Diagram Menerjemahkan Gesture menjadi Teks* dapat dilihat pada Gambar 3.14.



Gambar 3.14. Activity Diagram Menerjemahkan Gesture menjadi Teks

2. *Activity Menerjemahkan Teks menjadi Gesture*

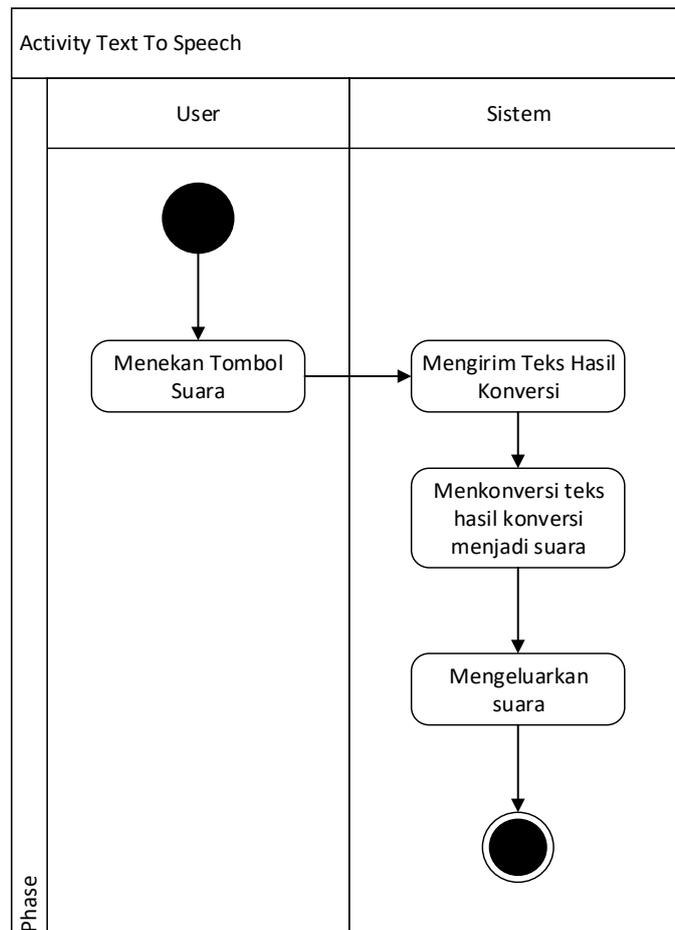
Activity Diagram Menerjemahkan Teks menjadi Gesture digunakan ketika user menginputkan teks dan menkonversinya ke gambar gerakan bahasa isyarat. *Activity Diagram Menerjemahkan Teks menjadi Gesture* dapat dilihat pada Gambar 3.15.



Gambar 3.15. Activity Diagram Menerjemahkan Teks menjadi Gesture

3. *Activity Text To Speech*

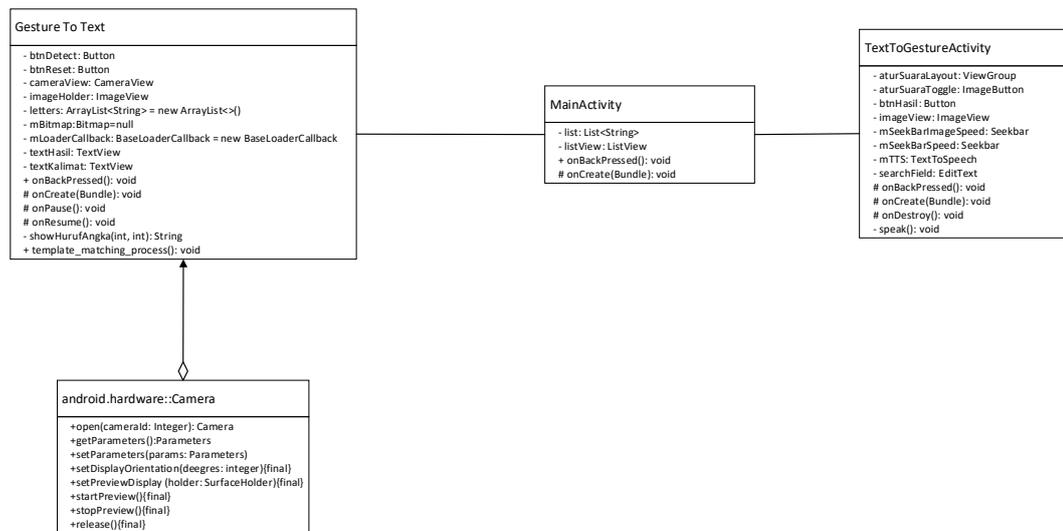
Activity Diagram Text To Speech digunakan ketika sistem menkonversi teks menjadi suara. *Activity Diagram Text To Speech* dapat dilihat pada Gambar 3.16.



Gambar 3.16. Activity Diagram Text To Speech

3.1.14.3 Class Diagram

Class Diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem. Struktur ini meliputi atribut-atribut dan metode-metode yang ada pada masing-masing class yang dapat dilihat pada Gambar 3.17.

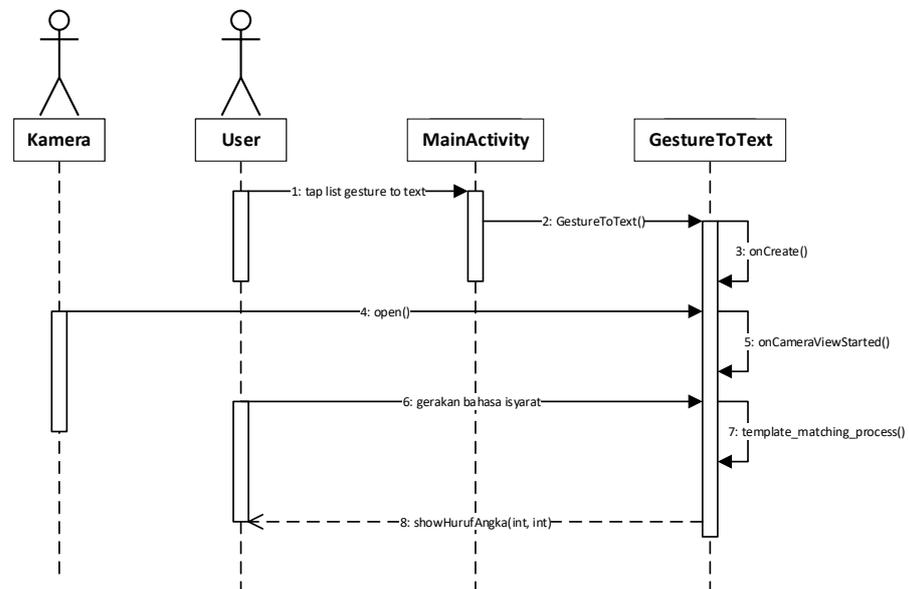


Gambar 3.17. *Class Diagram*

3.1.14.4 *Sequence Diagram*

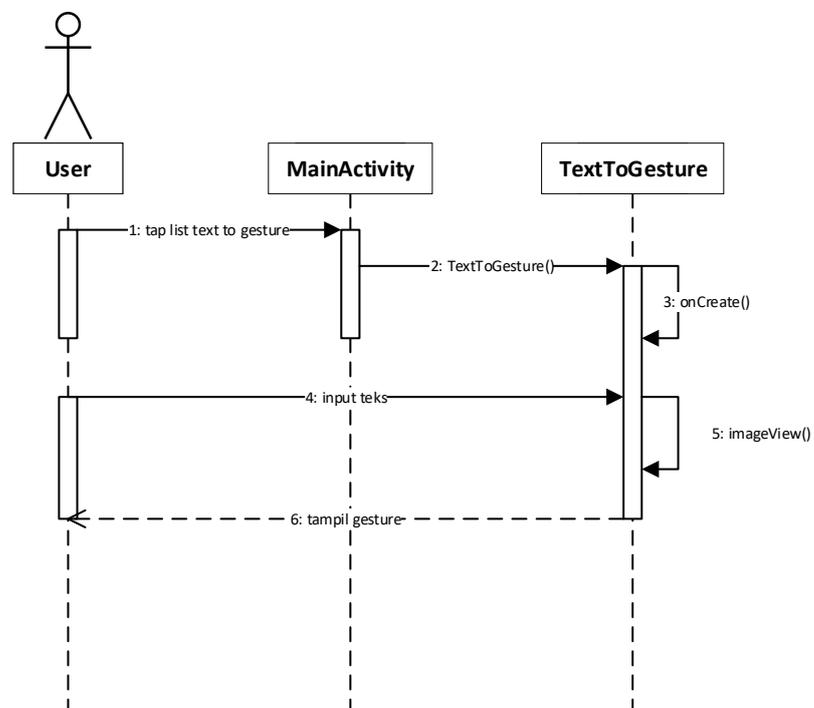
Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram jenis ini memberikan kejelasan sejumlah metode dan pesan-pesan yang diletakkan diantaranya di dalam sebuah *use case*. Berikut adalah *Sequence Diagram* yang terdapat pada aplikasi :

1. *Sequence Diagram* Menerjemahkan *Gesture* menjadi Teks yang dapat dilihat pada gambar Gambar 3.18.



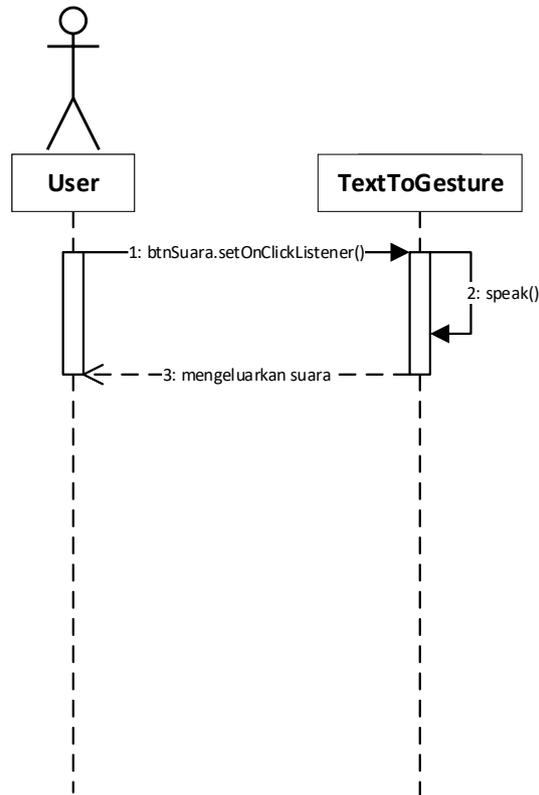
Gambar 3.18. Sequence Diagram Menerjemahkan Gesture menjadi Teks

2. *Sequence Diagram* Menerjemahkan Teks menjadi *Gesture* yang dapat dilihat pada Gambar 3.19.



Gambar 3.19. Sequence Diagram Menerjemahkan Teks menjadi Gesture

3. *Sequence Diagram Text To Speech* yang dapat dilihat pada gambar Gambar 3.20.



Gambar 3.20. *Sequence Diagram Text To Speech*

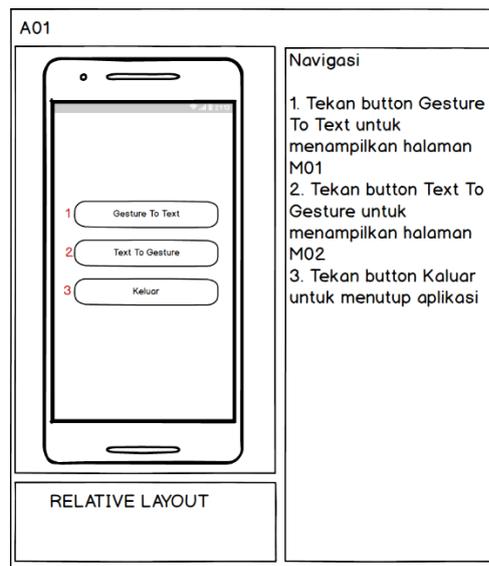
3.2 Perancangan Antarmuka

Perancangan sistem bertujuan untuk menspesifikasikan aspek-aspek teknik yang menjadi solusi dalam perencanaan. Pada tahap ini perancangan akan didefinisikan secara detail untuk mengatasi masalah-masalah yang lebih teknis, berkaitan dengan kegiatan implementasi seperti perancangan antarmuka, perancangan pesan, dan jaringan semantik.

3.2.1 Perancangan Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga mempermudah dalam mengimplementasikan aplikasi serta memudahkan dalam pembuatan aplikasi. Berikut adalah perancangan antarmuka aplikasi.

1. Perancangan Antarmuka Halaman Utama



Gambar 3.21. Tampilan Halaman Utama

2. Perancangan Antarmuka Halaman *Gesture To Text*



Gambar 3.22. Tampilan Halaman *Gesture To Text*

3. Perancangan Antarmuka Halaman *Text To Gesture*

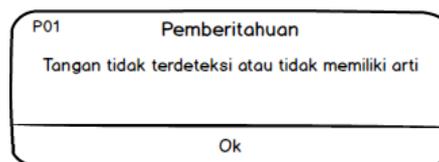


Gambar 3.23. Tampilan Halaman *Text To Gesture*

3.2.2 Perancangan Pesan

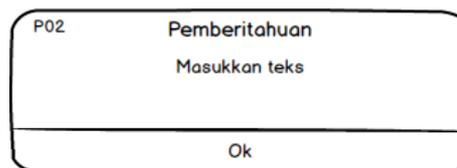
Berikut ini adalah perancangan pesan untuk aplikasi android. Perancangan pesan dapat dilihat dibawah ini :

1. Tangan atau gerakan tangan tidak terdeteksi



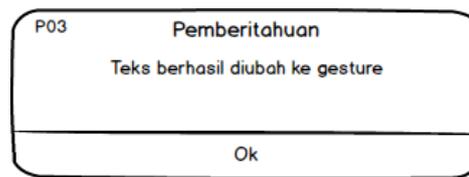
Gambar 3.24. Pesan Tangan atau gerakan tangan tidak terdeteksi

2. Masukkan teks



Gambar 3.25. Pesan Masukkan teks

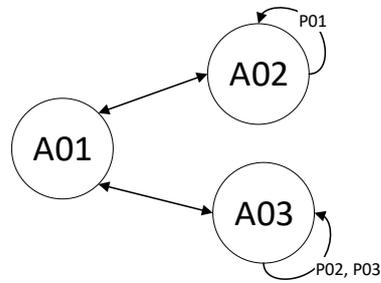
3. Teks berhasil diubah ke *gesture*



Gambar 3.26. Teks berhasil diubah ke *gesture*

3.2.3 Jaringan Semantik

Jaringan Semantik merupakan gambaran yang menggambarkan hubungan antarmuka satu dengan antarmuka lainnya. Berikut adalah jaringan semantik yang dapat dilihat pada Gambar 3.27.



Gambar 3.27. Jaringan Semantik

