



**TUGAS AKHIR - KI091391**

# **Rancang Bangun Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**LUCKY DEWANTARA**  
NRP 5110 100 064

Dosen Pembimbing I  
Dwi Sunaryono, S.Kom., M.Kom.

Dosen Pembimbing II  
Ridho Rahman H., S.Kom., M.Sc.

**JURUSAN TEKNIK INFORMATIKA**  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014



**FINAL PROJECT - KI091391**

# **Design and Implementation of Emergency Call Application with Voice Command in Bahasa Indonesia on Android Based Mobile Devices**

**LUCKY DEWANTARA**  
NRP 5110 100 064

Advisor I  
Dwi Sunaryono, S.Kom., M.Kom.

Advisor II  
Ridho Rahman H., S.Kom., M.Sc.

**INFORMATICS DEPARTMENT**  
Faculty of Information Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2014

# **RANCANG BANGUN APLIKASI PANGGILAN DARURAT DENGAN PERINTAH SUARA BERBAHASA INDONESIA PADA PERANGKAT BERGERAK BERBASIS ANDROID**

Nama Mahasiswa : Lucky Dewantara  
NRP : 5110 100 064  
Jurusan : Teknik Informatika FTIF-ITS  
Dosen Pembimbing I : Dwi Sunaryono, S.Kom., M.Kom.  
Dosen Pembimbing II : Ridho Rahman H., S.Kom., M.Sc.

## **ABSTRAK**

*Teknologi informasi yang berkembang pesat. Serta adanya layanan berbasis lokasi, membuat seseorang dapat mengetahui informasi mengenai tempat dari seluruh dunia hanya dengan perangkat bergerak. Teknologi bahasa pada perangkat bergerak yang dapat mengenali bahasa manusia. Di samping teknologi tersebut, permasalahan akan penanganan pada korban kecelakaan yang lambat dikarenakan kurangnya pengetahuan untuk mendapatkan nomor telepon layanan publik terdekat. Atas dasar itulah, muncul ide untuk menggabungkan teknologi tersebut dengan membangun sebuah aplikasi panggilan darurat dengan perintah suara berbahasa Indonesia pada perangkat bergerak Android.*

*Teknologi pengenalan suara dalam aplikasi ini menggunakan API masukan suara yang disediakan oleh Google pada perangkat Android. Teknologi API masukan suara ini mengubah masukan suara menjadi teks yang selanjutnya diproses untuk mendapatkan perintah panggilan lokasi layanan publik terdekat, perintah panggilan kontak pribadi, dan perintah penunjukan lokasi. Respon dari aplikasi diberikan kepada pengguna dalam bentuk*

*suara dengan menerapkan teknologi pengubah teks-ke-suara yang tersedia pada perangkat bergerak berbasis Android. Informasi mengenai lokasi didapatkan dari sinyal GPS untuk mendeteksi lokasi pengguna. Proses mendapatkan lokasi layanan publik terdekat menggunakan rumus Spherical Law of Cosines yang kemudian didapatkan layanan publik terdekat untuk dilakukan pemanggilan darurat.*

*Pengujian yang dilakukan pada aplikasi ini yaitu dengan melakukan panggilan darurat dan pencarian lokasi. Proses penentuan layanan publik terdekat dari posisi pengguna dapat berjalan baik, terbukti dengan lokasi nyata dari posisi pengguna menuju layanan publik tersebut. Proses pencarian lokasi juga dapat berjalan dengan baik, dapat dilihat dengan posisi penanda pengguna menuju lokasi yang dicari di tampilan menggunakan Google Maps.*

*Kesimpulan yang dapat diambil dari proses pengerjaan Tugas Akhir ini adalah aplikasi ini dapat membantu untuk memudahkan pencarian lokasi layanan publik terdekat guna pertolongan darurat dengan melakukan panggilan darurat pada layanan publik yang telah dipilih pengguna serta pencarian lokasi dengan menampilkan rute menuju lokasi tersebut. Pengembangan lebih lanjut mengenai teknologi pengenalan suara sangat diperlukan untuk melengkapi fitur aplikasi ini, sehingga tidak hanya untuk aplikasi panggilan dan penunjukan lokasi, namun juga dapat memberikan informasi-informasi lain berdasarkan lokasi pengguna.*

***Kata kunci: Android, Aplikasi Panggilan Darurat, Pengenalan Suara, Penunjukan Lokasi.***

# **DESIGN AND IMPLEMENTATION OF EMERGENCY CALL APPLICATION WITH VOICE COMMAND IN BAHASA INDONESIA ON ANDROID BASED MOBILE DEVICES**

Student Name : Lucky Dewantara  
NRP : 5110 100 064  
Major of Department : Informatics Engineering, FTIf-ITS  
Advisor I : Dwi Sunaryono, S.Kom., M.Kom.  
Advisor II : Ridho Rahman H., S.Kom., M.Sc.

## **ABSTRACT**

*Information technology is evolving rapidly. And the existence of location-based services, making someone can find out information about the place from all over the world only with mobile devices. Also language technology that can recognize human languages on mobile devices. Beside these technologies, problems about handling of the accident victims were slow because lack of knowledge to get the phone number of nearby public services. Based on that fact, the idea appears to combine these technologies to build an emergency call application with Indonesian-language voice commands on Android mobile devices.*

*Voice recognition technology in this application using the API voice input that provided by Google on Android devices. The voice input API technology converts the input voice into text which is further processed to get orders call the nearest location of public services, personal contact call command, and command the appointment location. The response of application is given to the user in the form of sound by applying modifier technology text-to-speech that available on Android-based mobile devices.*

*Information about location obtained from GPS signals to detect the user's location. Process of getting the location nearest public services using Spherical Law of Cosines formula which later obtained the nearest public services to do an emergency call.*

*Tests which performed on this application is to make emergency calls and searching location. The process of determining the nearest public service from the user's position can be running well, proven by the real location of the user's position towards the public service. Location search process also can run well, it can be seen with the user position marker to the location searched that displayed using Google Maps.*

*The conclusion that can be drawn from this Final Project process is this application can help to facilitate the search for nearest location public services for emergency rescue by doing emergency call on public services which have been selected by users and search location with showing the route to the location. Further development of the speech recognition technology is needed to complement the features of this application, so not only for summons and appointment location application, but also can provide other information based on the user's location.*

**Keywords:** *Android, Emergency Calls Applications, Voice Recognition, Appointment Location.*

## LEMBAR PENGESAHAN

### RANCANG BANGUN APLIKASI PANGGILAN DARURAT DENGAN PERINTAH SUARA BERBAHASA INDONESIA PADA PERANGKAT BERGERAK BERBASIS ANDROID

#### TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Bidang Studi Rekayasa Perangkat Lunak  
Program Studi S-1 Jurusan Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh :

**LUCKY DEWANTARA**

NRP. 5110 100 064

Disetujui oleh Dosen Pembimbing Tugas Akhir:

1. Dwi Sunaryono, S.Kom., M.Kom. ....  
NIP. 19720528 199702 1004 ..... (Pembimbing I)
2. Ridho Rahman H., S.Kom., M.Sc. ....  
NIP. 051100123 ..... (Pembimbing II)

**SURABAYA**  
**JUNI, 2014**

## **KATA PENGANTAR**

Segala puji dan syukur, kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Rancang Bangun Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android”.

Melalui lembar ini, penulis ingin menyampaikan ucapan terima kasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah SWT atas segala nikmat dan rahmat yang telah diberikan selama ini.
2. Kedua orang tua penulis yang telah mencurahkan kasih sayang, perhatian, dan doa kepada penulis selama ini.
3. Kedua saudara kandung penulis yang telah memberikan dukungan dan doa kepada penulis dalam menuntut ilmu hingga detik ini.
4. Bapak Dwi Sunaryono dan Bapak Ridho Rahman H. selaku dosen pembimbing yang telah memberikan bimbingan, motivasi, dan meluangkan waktu untuk membantu pengerjaan Tugas Akhir ini.
5. Ibu Siti Rochimah selaku dosen wali yang telah memberikan perhatian dan motivasi kepada penulis selama menjadi mahasiswa di lingkungan Teknik Informatika ITS.
6. Bapak dan ibu dosen Teknik Informatika ITS yang telah membina dan memberikan ilmu yang sangat bermanfaat kepada penulis selama menimba ilmu



delapan semester di lingkungan Teknik Informatika ITS.

7. Tommy Anandra Sunardi, Yorindha Falasifah Tirtasari, Artha Pradyta, Dandy Arganto, Ananta Pratama C. P., dan M. Ahmaluddin Zinny yang selalu membuat penulis bersemangat saat mengerjakan Tugas Akhir.
8. Yudha Nugraha, Mochamad Febrian, Irham Dzulkifli, Varistha P. C. A. dan Ilmal Alifriansyah R. yang menemani penulis sebagai teman seperjuangan selama masa perkuliahan.
9. Teman-teman Teknik Informatika ITS terutama angkatan 2010 atas segala dorongan semangat dan ilmu yang diberikan.
10. Pihak-pihak lain yang tidak sempat penulis sebutkan, yang telah membantu kelancaran pengerjaan Tugas Akhir ini.

Penulis berharap Tugas Akhir ini dapat bermanfaat bagi kemajuan dunia IT dan pendidikan di Indonesia. Bagaimanapun juga penulis telah berusaha sebaik mungkin dalam menyusun Tugas Akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2014

**Lucky Dewantara**

## DAFTAR ISI

<b>LEMBAR PENGESAHAN</b> .....	<b>vii</b>
<b>ABSTRAK</b> .....	<b>ix</b>
<b>ABSTRACT</b> .....	<b>xi</b>
<b>KATA PENGANTAR</b> .....	<b>xiii</b>
<b>DAFTAR ISI</b> .....	<b>xv</b>
<b>DAFTAR GAMBAR</b> .....	<b>xxi</b>
<b>DAFTAR TABEL</b> .....	<b>xxvii</b>
<b>DAFTAR PERSAMAAN</b> .....	<b>xxix</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Tujuan.....	2
1.3 Rumusan Permasalahan.....	3
1.4 Batasan Permasalahan.....	3
1.5 Manfaat Tugas Akhir.....	4
1.6 Metodologi.....	4
1.6.1 Melakukan Studi Literatur.....	4
1.6.2 Analisis dan Membuat Perancangan Perangkat Lunak.....	4
1.6.3 Mengubah Suara Menjadi Teks.....	5
1.6.4 Membuat Fungsi untuk Mengambil Perintah dari Teks.....	5
1.6.5 Mendeteksi Perintah Pengguna.....	5
1.6.6 Mendeteksi Posisi Pengguna.....	5
1.6.7 Melakukan Pencarian Informasi Layanan Publik Berdasarkan Lokasi Terdekat.....	6
1.6.8 Melakukan Panggilan Data Kontak dan Layanan Publik.....	6
1.6.9 Membangkitkan Rute Pencarian Lokasi.....	6
1.6.10 Menampilkan Posisi pada Peta.....	6
1.6.11 Menampilkan Rute pada Peta.....	7

1.6.12	Mengubah Teks Menjadi Suara.....	7
1.6.13	Melakukan Pengujian .....	7
1.6.14	Melakukan Evaluasi Pengujian .....	7
1.6.15	Menyusun Buku Tugas Akhir .....	8
1.7	Sistematika Penulisan .....	8
<b>BAB II DASAR TEORI .....</b>		<b>11</b>
2.1	Perintah Suara pada Perangkat Bergerak .....	11
2.2	Sistem Operasi Android .....	11
2.2.1	API Google Maps v2 pada Android .....	12
2.2.2	API Masukan Suara pada Android .....	13
2.2.3	Teks-Ke-Suara pada Android .....	14
2.2.4	GPS pada Android.....	15
2.3	JSON .....	16
2.4	Bahasa Indonesia.....	17
2.5	Spherical Law of Cosines .....	18
<b>BAB III ANALISIS DAN PERANCANGAN .....</b>		<b>19</b>
3.1	Deskripsi Umum Perangkat Lunak .....	19
3.2	Analisis Perangkat Lunak .....	22
3.2.1	Spesifikasi Kebutuhan Perangkat Lunak.....	22
3.2.2	Analisis Aktor.....	23
3.2.3	Analisis Kasus Penggunaan.....	23
3.2.3.1	Kasus Penggunaan Panggilan Kontak Pribadi (UC-001) .....	25
3.2.3.2	Kasus Penggunaan Penunjukan Lokasi (UC- 002) .....	28
3.2.3.3	Kasus Penggunaan Panggilan Kontak Layanan Publik (UC-003).....	30
3.2.3.4	Kasus Penggunaan Pengiriman Pesan Darurat (UC-004) .....	33
3.2.3.5	Kasus Penggunaan Olah Kontak Pribadi (UC- 005) .....	35
3.2.3.6	Kasus Penggunaan Olah Kontak Layanan Publik (UC-006) .....	37
3.3	Perancangan Basis Data .....	39

3.4	Perancangan Perangkat Lunak .....	41
3.4.1	Perancangan Antarmuka .....	41
3.4.1.1	Antarmuka Halaman Pembuka.....	41
3.4.1.2	Antarmuka Halaman Menu Aplikasi.....	42
3.4.1.3	Antarmuka Halaman Komunikasi .....	45
3.4.1.4	Antarmuka Halaman Peta.....	47
3.4.1.5	Antarmuka Halaman Kontak Pribadi .....	49
3.4.1.6	Antarmuka Halaman Kontak Layanan Publik.....	51
3.4.2	Perancangan Proses .....	52
3.4.3	Perancangan Proses .....	52
3.4.3.1	Proses pada Kasus Penggunaan Panggilan Layanan Publik Terdekat .....	53
3.4.3.2	Proses pada Kasus Penggunaan Panggilan Kontak Pribadi .....	56
3.4.3.3	Proses pada Kasus Penggunaan Penunjukan Lokasi.....	59
3.4.3.4	Proses pada Kasus Penggunaan Pengiriman Pesan Darurat .....	63
3.4.3.5	Proses pada Kasus Penggunaan Olah Kontak Pribadi .....	65
3.4.3.6	Proses pada Kasus Penggunaan Olah Kontak Layanan Publik .....	67
<b>BAB IV IMPLEMENTASI.....</b>		<b>69</b>
4.1	Lingkungan Pembangunan Perangkat Lunak .....	69
4.2	Implementasi Antarmuka Pengguna .....	69
4.2.1	Implementasi Antarmuka Halaman Pembuka .....	69
4.2.2	Implementasi Antarmuka Halaman Menu Aplikasi ..	71
4.2.3	Implementasi Antarmuka Halaman Komunikasi .....	72
4.2.4	Implementasi Antarmuka Halaman Peta.....	73
4.2.5	Implementasi Antarmuka Halaman Kontak Pribadi ..	75
4.2.6	Implementasi Antarmuka Halaman Layanan Publik.	76
4.3	Implementasi Proses .....	77

4.3.1	Implementasi Proses pada Kasus Penggunaan Panggilan Layanan Publik Terdekat.....	78
4.3.1.1	Implementasi Proses Deteksi Suara.....	78
4.3.1.2	Implementasi Proses Pengambilan Nama Layanan.....	79
4.3.1.3	Implementasi Proses Deteksi Posisi Pengguna.....	79
4.3.1.4	Implementasi Proses Keberadaan Nama Layanan.....	79
4.3.2	Implementasi Proses pada Kasus Penggunaan Panggilan Kontak Pribadi.....	80
4.3.2.1	Implementasi Proses Deteksi Suara.....	80
4.3.2.2	Implementasi Proses Pengambilan Nama Kontak.....	81
4.3.2.3	Implementasi Proses Pencarian Keberadaan Nama Kontak.....	81
4.3.3	Implementasi Proses pada Kasus Penunjukan Lokasi.....	82
4.3.3.1	Implementasi Proses Deteksi Suara.....	82
4.3.3.2	Implementasi Proses Pencarian Keberadaan Lokasi.....	83
4.3.3.3	Implementasi Proses Deteksi Posisi Pengguna.....	83
4.3.3.4	Implementasi Proses Pembangkitan Rute Terpendek.....	83
4.3.4	Implementasi Proses pada Kasus Pesan Darurat.....	83
4.3.4.1	Implementasi Proses Pengambilan Nomor dan Pesan Darurat.....	84
4.3.4.2	Implementasi Proses Pengambilan Nama Lokasi.....	84
4.3.4.3	Implementasi Proses Pengiriman Pesan Darurat ..	84
4.3.5	Implementasi Proses pada Kasus Olah Kontak Pribadi.....	85
4.3.5.1	Implementasi Proses Tambah Kontak Pribadi.....	85
4.3.5.2	Implementasi Proses Ubah Kontak Pribadi.....	85
4.3.5.3	Implementasi Proses Hapus Kontak Pribadi.....	85

4.3.6	Implementasi Proses pada Kasus Olah Kontak Layanan Publik.....	85
4.3.6.1	Implementasi Proses Tambah Kontak Layanan Publik.....	86
4.3.6.2	Implementasi Proses Ubah Kontak Layanan Publik.....	86
4.3.6.3	Implementasi Proses Hapus Kontak Layanan Publik.....	86
<b>BAB V PENGUJIAN DAN EVALUASI .....</b>		<b>87</b>
5.1	Lingkungan Pelaksanaan Pengujian .....	87
5.2	Dasar Pengujian .....	87
5.3	Pengujian Fungsionalitas .....	88
5.3.1	Pengujian Pengenalan Suara Pengguna.....	88
5.3.2	Pengujian Panggilan Layanan Publik Terdekat.....	95
5.3.3	Pengujian Panggilan Kontak Pribadi.....	97
5.3.4	Pengujian Pembangkitan Rute Menuju Lokasi .....	99
5.3.5	Pengujian Pengiriman Pesan Darurat .....	101
5.3.6	Pengujian Mendapatkan Nomor dan Pesan Darurat	105
5.3.7	Pengujian Tambah Kontak Pribadi .....	106
5.3.8	Pengujian Ubah Kontak Pribadi.....	109
5.3.9	Pengujian Hapus Kontak Pribadi .....	112
5.3.10	Pengujian Tambah Kontak Layanan Publik.....	115
5.3.11	Pengujian Ubah Kontak Layanan Publik .....	118
5.3.12	Pengujian Hapus Kontak Layanan Publik.....	121
5.4	Pengujian Kegunaan .....	124
5.4.1	Kriteria Responden.....	124
5.4.2	Lingkungan Pengujian Kegunaan .....	125
5.4.3	Skenario Pengujian Kegunaan .....	125
5.4.4	Daftar Responden.....	125
5.4.5	Hasil Pengujian Kegunaan .....	127
5.4.5.1	Rekapitulasi Penilaian Antarmuka Pengguna.....	127

5.4.5.2	Rekapitulasi Penilaian Pengenalan dan Penyampaian Suara .....	128
5.4.5.3	Rekapitulasi Penilaian Telepon Kontak Pribadi dan Telepon Layanan Publik Terdekat.....	128
5.4.5.4	Rekapitulasi Penilaian Informasi Penunjukan Lokasi .....	129
5.4.5.5	Rekapitulasi Penilaian Pesan Darurat dan Mode Darurat.....	129
5.5	Evaluasi Pengujian .....	130
5.5.1	Evaluasi Pengujian Fungsionalitas .....	130
5.5.2	Evaluasi Pengujian Kegunaan .....	133
<b>BAB VI PENUTUP.....</b>		<b>135</b>
6.1	Kesimpulan .....	135
6.2	Saran .....	136
<b>DAFTAR PUSTAKA .....</b>		<b>137</b>
<b>LAMPIRAN A – IMPLEMENTASI LAYOUT XML .....</b>		<b>139</b>
<b>LAMPIRAN B – KODE SUMBER.....</b>		<b>151</b>
<b>LAMPIRAN C – HASIL FORMULIR PENILAIAN PENGUJIAN KEGUNAAN.....</b>		<b>187</b>
<b>LAMPIRAN D – FORMULIR PENILAIAN PENGUJIAN KEGUNAAN.....</b>		<b>207</b>
<b>BIODATA PENULIS .....</b>		<b>211</b>

## DAFTAR GAMBAR

Gambar 2.1 Aplikasi yang menerapkan API Google Maps v2 pada Android.....	13
Gambar 3.1 Arsitektur Sistem.....	20
Gambar 3.2 Diagram Kasus Penggunaan.....	24
Gambar 3.3 Diagram Aktivitas Kasus Penggunaan Panggilan Kontak Pribadi.....	27
Gambar 3.4 Diagram Aktivitas Kasus Penggunaan Penunjukan Lokasi.....	30
Gambar 3.5 Diagram Aktivitas Kasus Penggunaan Panggilan Kontak Layanan Publik.....	33
Gambar 3.6 Diagram Aktivitas Kasus Penggunaan Pengiriman Pesan Darurat.....	35
Gambar 3.7 Diagram Aktivitas Kasus Penggunaan Olah Kontak Pribadi.....	37
Gambar 3.8 Diagram Aktivitas Kasus Penggunaan Olah Layanan Publik.....	38
Gambar 3.9 CDM Basis Data Tolong.....	39
Gambar 3.10 PDM Basis Data Tolong.....	40
Gambar 3.11 Rancangan Antarmuka Halaman Pembuka.....	41
Gambar 3.12 Rancangan Antarmuka Halaman Menu Aplikasi.....	43
Gambar 3.13 Rancangan Antarmuka Halaman Komunikasi.....	46
Gambar 3.14 Rancangan Antarmuka Halaman Peta.....	48
Gambar 3.15 Rancangan Antarmuka Halaman Kontak Pribadi.....	50
Gambar 3.16 Rancangan Antarmuka Halaman Kontak Layanan Publik.....	51
Gambar 3.17 Diagram Alir Proses Panggilan Layanan Publik.....	53
Gambar 3.18 Diagram Alir Proses Panggilan Kontak Pribadi.....	57
Gambar 3.21 Diagram Alir Proses Olah Kontak Pribadi.....	65
Gambar 4.1 Implementasi Antarmuka Halaman Pembuka.....	70
Gambar 4.2 Implementasi Antarmuka Halaman Menu Aplikasi.....	71



Gambar 4.3 Implementasi Antarmuka Halaman Komunikasi.....	72
Gambar 4.4 Implementasi Antarmuka Halaman Peta .....	74
Gambar 4.5 Implementasi Antarmuka Halaman Kontak Pribadi	75
Gambar 4.6 Implementasi Antarmuka Halaman Layanan Publik .....	77
Gambar 5.1 Pengujian Pengenalan Suara untuk Panggilan Layanan Publik Terdekat.....	92
Gambar 5.2 Pengujian Pengenalan Suara untuk Panggilan Kontak Pribadi .....	93
Gambar 5.3 Pengujian Pengenalan Suara untuk Nama Lokasi yang Unik .....	94
Gambar 5.4 Pengujian Pengenalan Suara untuk Nama Lokasi yang Umum .....	95
Gambar 5.5 Proses Pengujian Panggilan Layanan Publik Terdekat. (a) Pengenalan Perintah Pengguna, (b) Tampilan Panggilan Langsung .....	97
Gambar 5.6 Proses Pengujian Panggilan Kontak Pribadi. (a) Pengenalan Perintah Pengguna, (b) Tampilan Panggilan Langsung .....	99
Gambar 5.7 Proses Pembangkitan Rute Menuju Kejawen Putih Tambak Surabaya. (a) Pengenalan Perintah Pengguna, (b) Tampilan Rute .....	101
Gambar 5.8 Proses Pengiriman Pesan Darurat. (a) Pengiriman Pesan Darurat, (b) Pesan Darurat yang Diterima.....	103
Gambar 5.9 Proses Pengiriman Mode Darurat. (a) Pengiriman Pesan Mode Darurat, (b) Pesan Mode Darurat yang Diterima. .	104
Gambar 5.10 Proses Pengujian Mendapatkan Nomor dan Pesan Darurat. (a) Halaman Login, (b) Halaman Menu Aplikasi beserta Nomor dan Pesan Darurat .....	106
Gambar 5.11 Proses Pengujian Tambah Kontak Pribadi. (a) Halaman Sebelum Kontak Ditambahkan, (b) Halaman Tambah Kontak .....	108
Gambar 5.12 Halaman Kontak Setelah Kontak Ditambahkan ..	109

Gambar 5.13 Proses Pengujian Ubah Kontak Pribadi. (a) Kontak Setelah Kontak Diperbarui. (b) Halaman Detail Kontak.....	111
Gambar 5.14 Proses Pengujian Ubah Kontak Pribadi. (a) Halaman Edit Kontak. (b) Halaman Kontak Setelah Kontak Diperbarui.....	112
Gambar 5.15 Proses Pengujian Hapus Kontak Pribadi. (a) Kontak Sesebelum Kontak Dihapus. (b) Halaman Detail Kontak. ....	114
Gambar 5.16 Halaman Kontak Setelah Kontak Dihapus. ....	115
Gambar 5.17 Proses Pengujian Tambah Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Ditambahkan. (b) Halaman Tambah Kontak. ....	117
Gambar 5.18 Proses Pengujian Tambah Kontak Layanan Publik. (a) Menambahkan Posisi Layanan. (b) Halaman Kontak Setelah Kontak Ditambahkan.....	118
Gambar 5.19 Proses Pengujian Ubah Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Diubah. (b) Halaman Edit Kontak. ....	120
Gambar 5.20 Halaman Kontak Layanan Publik Setelah Kontak Diperbarui.....	121
Gambar 5.21 Proses Pengujian Hapus Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Dihapus. (b) Halaman Edit Kontak. ....	123
Gambar 5.22 Halaman Kontak Setelah Kontak Dihapus. ....	124
Gambar A.1 Implementasi Antarmuka Halaman Pembuka dalam XML.....	139
Gambar A.2 Implementasi Antarmuka Halaman Menu Aplikasi .....	143
Gambar A.3 Implementasi <i>ListView</i> untuk Komunikasi .....	144
Gambar A.4 Implementasi pembuatan gelembung obrolan untuk Komunikasi .....	144
Gambar A.5 Implementasi <i>ImageButton</i> pada Tombol Bicara .	145

Gambar A.6 Implementasi <i>MapView</i> dengan <i>ApiKey</i> .....	145
Gambar A.7 Implementasi Antarmuka Halaman Kontak.....	147
Gambar A.8 Implementasi <i>ListView</i> Kontak.....	148
Gambar A.9 Implementasi Antarmuka Halaman Layanan Publik. .....	150
Gambar A.10 Implementasi <i>ListView</i> Layanan Publik.....	150
Gambar B.1 Kode Sumber Kelas <i>GetRouteListTask</i> .....	152
Gambar B.2 Kode Sumber Kelas <i>MyPosition</i> .....	156
Gambar B.3 Kode Sumber Penanda pada Peta.....	157
Gambar B.4 Kode Sumber Inisiasi <i>Intent</i> untuk Penerimaan Suara .....	157
Gambar B.5 Kode Sumber Inisiasi Penerimaan Hasil Respon <i>Intent</i> Untuk Menerima Suara .....	158
Gambar B.6 Kode Sumber Pembacaan Teks.....	158
Gambar B.7 Kode Sumber Proses Pengambilan Kata Nama Layanan .....	158
Gambar B.8 Kode Sumber untuk Deteksi Posisi Pengguna .....	159
Gambar B.9 Kode Sumber Proses Pencarian Nama Layanan dan Pemanggilan Layanan 1 Hasil .....	161
Gambar B.10 Kode Sumber <i>Query Service</i> Pencarian Layanan dengan Metode Spherical Law of Cosines .....	162
Gambar B.11 Kode Sumber Proses Pemanggilan Layanan Publik Lebih dari 1 Hasil .....	162
Gambar B.12 Kode Sumber Proses Pengambilan Nama Kontak .....	162
Gambar B.13 Kode Sumber Proses Pencarian Keberadaan Nama Kontak .....	164
Gambar B.14 Kode Sumber <i>Query Service</i> Pencarian Keberadaan Nama Kontak.....	165
Gambar B.15 Kode Sumber untuk Mencari Keberadaan Lokasi .....	166
Gambar B.16 Kode Sumber kelas <i>DirectionJSONParser</i> .....	169
Gambar B.17 Kode Sumber untuk Proses Permintaan Rute .....	174

Gambar B.18 Kode Sumber untuk Proses Pengambilan Nomor dan Pesan Darurat.....	175
Gambar B.19 Kode Sumber untuk Proses Pengambilan Nama Lokasi.....	176
Gambar B.20 Kode Sumber untuk Pengiriman Pesan Darurat .	177
Gambar B.21 Kode Sumber untuk Pengiriman Mode Darurat .	178
Gambar B.22 Kode Sumber untuk Tambah Kontak Pribadi .....	179
Gambar B.23 Kode Sumber untuk Ubah Kontak Pribadi .....	181
Gambar B.24 Kode Sumber untuk Hapus Kontak Pribadi.....	182
Gambar B.25 Kode Sumber untuk Tambah Kontak Layanan Publik. ....	183
Gambar B.26 Kode Sumber untuk Ubah Kontak Layanan Publik. ....	185
Gambar B.27 Kode Sumber untuk Hapus Kontak Layanan Publik. ....	186
Gambar C.1 Hasil Pengujian Pengguna 1 .....	187
Gambar C.2 Hasil Pengujian Pengguna 2 .....	188
Gambar C.3 Hasil Pengujian Pengguna 3 .....	189
Gambar C.4 Hasil Pengujian Pengguna 4 .....	190
Gambar C.5 Hasil Pengujian Pengguna 5 .....	191
Gambar C.6 Hasil Pengujian Pengguna 6 .....	192
Gambar C.7 Hasil Pengujian Pengguna 7 .....	193
Gambar C.8 Hasil Pengujian Pengguna 8 .....	194
Gambar C.9 Hasil Pengujian Pengguna 9 .....	195
Gambar C.10 Hasil Pengujian Pengguna 10 .....	196
Gambar C.11 Hasil Pengujian Pengguna 11 .....	197
Gambar C.12 Hasil Pengujian Pengguna 12 .....	198
Gambar C.13 Hasil Pengujian Pengguna 13 .....	199
Gambar C.14 Hasil Pengujian Pengguna 14 .....	200
Gambar C.15 Hasil Pengujian Pengguna 15 .....	201
Gambar C.16 Hasil Pengujian Pengguna 16 .....	202

Gambar C.17 Hasil Pengujian Pengguna 17 .....	203
Gambar C.18 Hasil Pengujian Pengguna 18 .....	204
Gambar C.19 Hasil Pengujian Pengguna 19 .....	205

## DAFTAR PERSAMAAN

Persamaan (2.1).....	18
----------------------	----

## DAFTAR TABEL

Tabel 3.1 Deskripsi Kasus Penggunaan .....	24
Tabel 3.2 Spesifikasi Kasus Penggunaan Panggilan Kontak Pribadi .....	25
Tabel 3.3 Spesifikasi Kasus Penggunaan Penunjukan Lokasi ....	28
Tabel 3.4 Spesifikasi Kasus Penggunaan Panggilan Kontak Layanan Publik.....	31
Tabel 3.5 Spesifikasi Kasus Penggunaan Pengiriman Pesan Darurat.....	34
Tabel 3.6 Spesifikasi Kasus Penggunaan Olah Kontak Pribadi ..	36
Tabel 3.7 Spesifikasi Kasus Penggunaan Olah Kontak Layanan Publik .....	37
Tabel 3.8 Spesifikasi Basis Data Tolong.....	40
Tabel 5.1 Spesifikasi Lingkungan Pengujian Perangkat Lunak ..	87
Tabel 5.2 Skenario Pengujian Fungsionalitas Pengenalan Suara Pengguna .....	88
Tabel 5.3 Skenario Pengujian Fungsionalitas Panggilan Layanan Publik Terdekat .....	96
Tabel 5.4 Skenario Pengujian Fungsionalitas Pembangkitan Rute Menuju Lokasi.....	100
Tabel 5.5 Skenario Pengujian Fungsionalitas Pengiriman Pesan Darurat.....	102
Tabel 5.6 Skenario Pengujian Fungsionalitas Mendapatkan Nomor dan Pesan Darurat .....	105
Tabel 5.7 Skenario Pengujian Fungsionalitas Tambah Kontak Pribadi .....	107
Tabel 5.8 Skenario Pengujian Fungsionalitas Ubah Kontak Pribadi .....	110
Tabel 5.9 Skenario Pengujian Fungsionalitas Hapus Kontak Pribadi .....	113
Tabel 5.10 Skenario Pengujian Fungsionalitas Tambah Kontak Layanan Publik.....	116

Tabel 5.11 Skenario Pengujian Fungsionalitas Ubah Kontak Pribadi .....	119
Tabel 5.12 Skenario Pengujian Fungsionalitas Hapus Kontak Layanan Publik.....	122
Tabel 5.13 Daftar Responden Pengujian Kegunaan.....	125
Tabel 5.14 Penilaian Antarmuka Pengguna .....	127
Tabel 5.15 Penilaian Pengenalan dan Penyampaian Suara .....	128
Tabel 5.16 Penilaian Telepon Kontak Pribadi dan Telepon Layanan Publik Terdekat.....	128
Tabel 5.17 Penilaian Informasi Penunjukan Lokasi .....	129
Tabel 5.18 Penilaian Pesan Darurat dan Mode Darurat .....	130
Tabel 5.19 Rekapitulasi Akhir Pengujian Kegunaan.....	133



# **BAB I**

## **PENDAHULUAN**

Bab ini menjelaskan mengenai garis besar Tugas Akhir yang meliputi latar belakang, tujuan, rumusan permasalahan dan batasan permasalahan, metodologi pembuatan Tugas Akhir, dan sistematika penulisan yang digunakan dalam pembuatan Tugas Akhir ini.

### **1.1 Latar Belakang**

Teknologi informasi telah berkembang sangat cepat dan semakin modern hampir di seluruh aspek kehidupan manusia membutuhkannya. Kebutuhan manusia akan informasi yang semakin tinggi untuk meningkatkan produktifitasnya sehingga segala urusannya akan dipermudah dan dapat cepat selesai. Salah satu teknologi yang banyak digunakan oleh manusia adalah teknologi perangkat bergerak (*mobile device*) yang terus berkembang.

Informasi merupakan kebutuhan pokok bagi sebagian besar manusia. Dengan menggunakan perangkat bergerak (*mobile device*) manusia akan membutuhkan banyak informasi di sekelilingnya yang mungkin akan berkaitan dengan kebutuhannya. Di samping dapat memberi kemudahan bagi pengguna juga dapat meningkatkan pelayanan layanan publik yang telah ada.

Salah satu teknologi yang dimanfaatkan dalam penunjuk lokasi adalah GPS (*Global Positioning System*), yang merupakan sebuah alat atau sistem yang dapat digunakan untuk menginformasikan lokasi pengguna (secara global) di permukaan bumi yang berbasis satelit. Data dikirim dari satelit berupa sinyal radio dengan data digital.

Aplikasi yang memanfaatkan GPS pada perangkat bergerak guna memberikan informasi rute jalan sudah banyak dikenal dan digunakan oleh masyarakat. Namun dalam perkembangannya manusia akan selalu menuntut kemudahan yang dapat disediakan

oleh aplikasi untuk memudahkan setiap kegiatannya. Salah satunya dengan memanfaatkan perintah suara secara langsung untuk menghubungi nomor telepon penting dan layanan publik yang letaknya dekat dengan pengguna untuk memudahkan pelaporan dan bantuan ketika terjadi tindak kejahatan atau kecelakaan.

Perkembangan sistem operasi Android sebagai salah satu sistem operasi yang sangat populer dan semakin banyak pengguna perangkat berbasis perangkat bergerak Android di pasaran. Dukungan teknologi terbaru dan berbagai fitur yang dapat memudahkan pengguna terutama koneksi internet yang membuat perangkat ini banyak diminati. Sistem operasi Android juga mendukung adanya GPS dan juga representasi Google Maps yang dapat dengan mudah diakses melalui internet dengan perangkat bergerak Android.

Aplikasi panggilan dengan perintah suara secara langsung sudah tersedia, namun yang secara khusus dengan bahasa Indonesia masih belum tersedia. Pengenalan suara (*voice recognition*) yang dimanfaatkan dalam berbagai aplikasi hanya disediakan dalam bahasa Inggris, Mandarin, dan bahasa-bahasa lain selain bahasa Indonesia. Padahal pengguna bahasa Indonesia tergolong banyak, mengingat bahwa total penduduk Indonesia saat ini menempati urutan ke-4 dunia, yakni sejumlah 237.641.326 jiwa [1]. Di samping itu penanganan pada korban kecelakaan yang lambat dikarenakan kurangnya pengetahuan untuk menghubungi nomor penting dan layanan umum yang telah tersedia. Mengingat angka kecelakaan di Surabaya pada tahun 2013 mencapai 1.136 kasus kecelakaan [2].

## 1.2 Tujuan

Tujuan dari Tugas Akhir adalah sebagai berikut.

1. Membangun sistem aplikasi perangkat bergerak Android yang dapat menerjemahkan suara berbahasa Indonesia menjadi sebuah perintah untuk melakukan panggilan

darurat nomor telepon layanan publik terdekat berdasarkan posisi yang diambil dari GPS.

2. Membangun sistem aplikasi perangkat bergerak Android yang dapat menentukan rute terpendek berdasarkan posisi yang diambil dari GPS menuju lokasi yang diinginkan serta dapat menampilkannya ke dalam Google Maps

### **1.3 Rumusan Permasalahan**

Permasalahan yang dibahas dalam Tugas Akhir ini sebagai berikut.

1. Bagaimana menerjemahkan suara berbahasa Indonesia menjadi sebuah perintah untuk melakukan panggilan darurat nomor telepon layanan publik terdekat dan penunjukan lokasi pada perangkat bergerak Android.
2. Bagaimana mendapatkan informasi koordinat GPS dalam pengembangan guna menunjukkan layanan publik terdekat dan rute menuju lokasi tujuan pada pengguna.
3. Bagaimana menentukan rute terpendek berdasarkan Google Maps menuju lokasi tujuan serta dapat ditampilkan ke dalam Google Maps.

### **1.4 Batasan Permasalahan**

Tugas Akhir ini memiliki beberapa batasan, diantaranya sebagai berikut.

1. Suara yang dapat dikenali adalah perintah panggilan darurat nomor telepon layanan publik terdekat dan penunjukan lokasi dengan bahasa Indonesia.
2. Aplikasi dibangun khusus untuk perangkat bergerak dengan sistem operasi Android versi minimum 4.0.
3. Aplikasi hanya dapat dijalankan ketika perangkat bergerak tersambung dengan internet.
4. Ruang lingkup panggilan darurat layanan publik mencakup rumah sakit, kantor polisi, pemadam

kebakaran, dan stasiun radio yang menyiarkan informasi jalan di kota Surabaya

### **1.5 Manfaat Tugas Akhir**

Manfaat dari Tugas Akhir ini adalah sebagai media untuk memudahkan pengguna perangkat bergerak berbasis Android dalam melakukan panggilan darurat nomor telepon penting dan penunjukan lokasi layanan publik dengan cara yang lebih interaktif, yakni melalui perintah suara dengan Bahasa Indonesia.

### **1.6 Metodologi**

Metodologi yang dilakukan dalam pengerjaan Tugas Akhir ini memiliki beberapa tahapan, di antaranya adalah sebagai berikut.

#### **1.6.1 Melakukan Studi Literatur**

Studi literatur dilakukan pengumpulan informasi yang diperlukan untuk perancangan sistem yang meliputi informasi-informasi sebagai berikut.

1. Perencanaan dan pembangunan perangkat lunak
2. Pola perintah dalam Bahasa Indonesia sebagai pendukung pencarian nama layanan publik terdekat dan penunjukan lokasi tujuan dalam teks
3. Pengembangan aplikasi pada perangkat bergerak berbasis Android
4. Studi literatur Penggunaan API Google Speech dan API Google Maps v2.

Informasi-informasi tersebut diperoleh dari beberapa literatur berupa jurnal, maupun buku-buku yang terkait dengan ketiga topik di atas.

#### **1.6.2 Analisis dan Membuat Perancangan Perangkat Lunak**

Pada tahap dilakukan studi literatur dengan melakukan analisa dan perancangan perangkat lunak yang dikembangkan dalam Tugas Akhir ini. Analisa yang dilakukan meliputi analisis

mengenai kasus penggunaan yang terdapat pada perangkat lunak. Tahap analisis dilakukan dengan proses perancangan perangkat lunak yang dibangun. Perancangan meliputi perancangan antarmuka serta perancangan proses yang ada pada perangkat lunak.

### **1.6.3 Mengubah Suara Menjadi Teks**

Tahap ini merupakan tahap awal proses implementasi. Tahap mengubah suara menjadi teks dilakukan sebagai proses pembacaan perintah suara dari pengguna perangkat lunak. Masukan untuk tahap ini berupa suara dari pengguna yang kemudian menghasilkan keluaran berupa teks.

### **1.6.4 Membuat Fungsi untuk Mengambil Perintah dari Teks**

Tahap ini dilakukan setelah dihasilkan teks dari tahap sebelumnya. Teks diidentifikasi sebagai kalimat perintah yang menyatakan perintah panggilan ataupun penunjukan rute menuju suatu lokasi. Tujuan dari tahap ini adalah mengidentifikasi kalimat perintah tersebut apakah kalimat panggilan atau kalimat penunjukan suatu lokasi.

### **1.6.5 Mendeteksi Perintah Pengguna**

Tahap ini melakukan pendekatan perintah yang diucapkan pengguna untuk mengambil kata pada setiap kalimat perintah untuk mengetahui perintah tersebut. Terdapat 2 jenis perintah yang dapat dilakukan yaitu dengan perintah panggilan darurat dan penunjukan rute menuju suatu lokasi

### **1.6.6 Mendeteksi Posisi Pengguna**

Tahap pendeteksian posisi pengguna ini dilakukan untuk mengambil koordinat longitude dan latitude berdasarkan pendeteksian menggunakan GPS. Keluaran dari proses ini adalah berupa koordinat latitude dan longitude dari posisi pengguna.

### **1.6.7 Melakukan Pencarian Informasi Layanan Publik Berdasarkan Lokasi Terdekat**

Pencarian informasi berdasarkan lokasi dilakukan dengan penghitungan jarak antara posisi pengguna dengan informasi posisi layanan publik pada basis data layanan publik. Pencarian dilakukan dengan masukan teks yang digunakan untuk mendapatkan hasil keluaran proses pengambilan nama layanan publik terdekat. Pencarian memungkinkan untuk didapatkan lebih dari satu lokasi yang cocok. Pencarian juga memungkinkan untuk tidak menghasilkan satu lokasi yang cocok dengan nama jenis layanan yang dicari.

### **1.6.8 Melakukan Panggilan Data Kontak dan Layanan Publik**

Tahap ini merupakan tahap dimana teks hasil perintah masukan suara dicocokkan berdasarkan nama kontak ataupun nama layanan, apabila hasil teks perintah memiliki hasil lebih dari 1 maka dilakukan proses pengambilan pilihan oleh pengguna. Proses pengambilan perintah pilihan oleh pengguna dilakukan dengan masukan suara dengan menyebutkan indeks hasil teks perintah sebelumnya. Apabila teks perintah memiliki hasil bernilai hanya 1 maka sistem akan melakukan proses panggilan langsung.

### **1.6.9 Membangkitkan Rute Pencarian Lokasi**

Membangkitkan rute dilakukan dengan pemanggilan API Google Maps v2 berdasarkan dua koordinat yang menjadi asal dan tujuan. Keluaran tahap ini berupa objek rute dari posisi pengguna menuju lokasi yang menjadi tujuan pengguna melakukan pencarian lokasi.

### **1.6.10 Menampilkan Posisi pada Peta**

Tahap untuk menampilkan posisi pada peta dilakukan dengan penggambaran posisi dengan koordinat longitude dan latitude

yang kemudian diproyeksikan terhadap tampilan peta pada layar perangkat bergerak.

#### **1.6.11 Menampilkan Rute pada Peta**

Tahap untuk menampilkan rute pada peta dilakukan dengan mengambil titik-titik yang terdapat pada rute dan dihubungkan dengan garis-garis yang kemudian diproyeksikan terhadap tampilan peta pada layar perangkat bergerak.

#### **1.6.12 Mengubah Teks Menjadi Suara**

Tahap ini merupakan tahap pemanfaatan teknologi teks-ke-suara yang terdapat pada perangkat bergerak Android. Teks yang menjadi masukan merupakan perintah panggilan darurat dan penunjukan lokasi yang selanjutnya respon dari perangkat lunak yang disampaikan kepada pengguna.

#### **1.6.13 Melakukan Pengujian**

Tahap pengujian dilakukan untuk mengetahui kualitas dari perangkat lunak yang telah dibangun. Pengujian dilakukan dengan dasar rumusan permasalahan yang diangkat pada Tugas Akhir ini.

Pengujian yang dilakukan berupa pengujian fungsionalitas dan pengujian kegunaan perangkat lunak secara langsung oleh pengguna. Pengujian fungsionalitas dilakukan dengan model *black box* untuk masing-masing fungsionalitas dari aplikasi ini. Pengujian kemudahan perangkat lunak dilakukan dengan pengujian langsung kepada pengguna perangkat lunak dan mengambil hasil pengujian dari kuisioner yang telah disebarakan kepada pengguna perangkat lunak.

#### **1.6.14 Melakukan Evaluasi Pengujian**

Evaluasi dilakukan terhadap pengujian sebagai hasil akhir dari proses pembangunan perangkat lunak, sehingga bisa diketahui kualitas dan kegunaan dari perangkat lunak ini. Proses evaluasi dilihat dari keberhasilan dari tahap pengujian perangkat lunak.

Hasil dari evaluasi pengujian ini menjadi salah satu bagian dari kesimpulan pengerjaan Tugas Akhir ini.

### **1.6.15 Menyusun Buku Tugas Akhir**

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai perancangan, pembangunan, serta hasil dari pengujian perangkat lunak yang dibuat untuk Tugas Akhir ini ke dalam buku Tugas Akhir. Secara garis besar, buku Tugas Akhir yang nantinya akan dibuat terdiri dari bagian-bagian berikut.

1. Pendahuluan
2. Kajian Pustaka
3. Analisis dan Perancangan
4. Implementasi
5. Pengujian dan Evaluasi
6. Penutup

Untuk penjelasan mengenai bagian-bagian buku Tugas Akhir ini dijelaskan pada subbab Sistematika Penulisan.

## **1.7 Sistematika Penulisan**

Buku Tugas Akhir ini terdiri dari beberapa bab, yang dijelaskan sebagai berikut.

### **Bab I. Pendahuluan**

Bab ini berisi latar belakang masalah, permasalahan, batasan masalah, tujuan Tugas Akhir, manfaat Tugas Akhir, metodologi yang digunakan, dan sistematika penyusunan buku Tugas Akhir.

### **Bab II. Kajian Pustaka**

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.



### **Bab III. Analisis dan Perancangan**

Bab ini membahas analisis perangkat lunak meliputi spesifikasi kebutuhan perangkat lunak, analisis aktor, dan analisis kasus penggunaan. Perancangan perangkat lunak pada bab ini meliputi perancangan antarmuka dan perancangan proses.

### **Bab IV. Implementasi**

Bab ini membahas implementasi rancangan perangkat lunak meliputi lingkungan pembangunan perangkat lunak, implementasi antarmuka pengguna, serta implementasi proses-proses pada masing-masing kasus penggunaan.

### **Bab V. Pengujian dan Evaluasi**

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat hasil keluaran yang dihasilkan oleh aplikasi serta evaluasi untuk mengetahui kemampuan aplikasi.

### **Bab VI. Penutup**

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan serta saran untuk pengembangan aplikasi selanjutnya.

#### **Daftar Pustaka**

Merupakan daftar referensi yang digunakan untuk mengembangkan tugas akhir.

#### **Lampiran**

Merupakan bab tambahan yang berisi daftar istilah yang penting pada aplikasi ini.

## **BAB II**

### **DASAR TEORI**

Pada bab ini membahas mengenai teori-teori yang menjadi dasar dari pengembangan aplikasi untuk Tugas Akhir ini. Dasar teori yang dibahas di dalam bab ini adalah perintah suara pada perangkat bergerak, sistem operasi Android yang meliputi API Google Maps v2, API masukan suara, teks ke suara, GPS, JSON, Bahasa Indonesia dan Spherical Law of Cosines.

#### **2.1 Perintah Suara pada Perangkat Bergerak**

Perintah suara pada perangkat bergerak merupakan sebuah perintah yang diucapkan dari suara manusia untuk menjalankan aktivitas tertentu pada perangkat bergerak. Saat ini beberapa perangkat bergerak yang beredar di pasaran sudah memiliki fitur pembacaan suara. Beberapa di antaranya memanfaatkan fitur pembacaan suara untuk perintah suara melakukan beberapa aktivitas dalam perangkat bergerak.

Salah satu contoh aplikasi yang memanfaatkan perintah suara adalah Siri yang merupakan layanan milik Apple. Aplikasi tersebut memberikan layanan yang dapat memudahkan pengguna dalam menggunakan perangkat telepon pintar dengan bantuan suara yang diterima oleh perangkat bergerak. Sehingga pengguna dapat melakukan operasi pada perangkat bergerak seperti mengirimkan pesan teks, menelepon, membuka aplikasi tertentu, membuat catatan, hingga menunjukkan arah menuju tempat tertentu menggunakan peta. Hanya saja kedua layanan ini tersedia dalam Bahasa Inggris, belum terdapat versi kedua layanan ini dalam bahasa selain Bahasa Inggris [3].

#### **2.2 Sistem Operasi Android**

Sistem yang dibangun dalam Tugas Akhir ini adalah berupa aplikasi pada perangkat bergerak yang menjalankan sistem operasi Android. Android adalah sebuah sistem operasi berbasis Linux untuk *gadget* yang dikembangkan oleh Google. Android

menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak.

Android dikembangkan oleh Google dengan memiliki banyak layanan yang juga didukung oleh Google seperti Gmail, Google Calendar, Google Maps, dan mesin pencarian Google. Android yang bersifat *open source* membuat pengembang perangkat lunak pada perangkat bergerak dapat menambahkan layanan-layanan lain pada perangkat bergerak berbasis Android [4].

Pada proyek Tugas Akhir ini dikembangkan aplikasi dengan memanfaatkan beberapa layanan yang disediakan pada sistem operasi Android, antara lain API Google Maps v2, API masukan suara, teks-ke-suara, serta GPS. Sistem aplikasi yang dibangun dalam Tugas Akhir ini dapat dijalankan pada Android dengan versi 4.0 atau versi di atasnya.

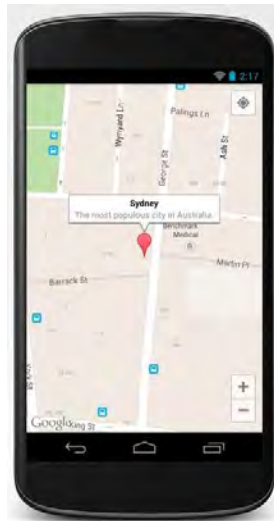
### **2.2.1 API Google Maps v2 pada Android**

API Google Maps v2 merupakan *API* yang dapat diakses melalui javascript agar Google Maps dapat ditampilkan pada klien yang tersambung dengan internet. Google Maps merupakan salah satu layanan gratis yang dimiliki Google untuk peta digital yang menawarkan tampilan peta dan gambar dari satelit. Google Maps v2 merupakan pembaruan dari layanan Google Maps v1 yang mengalami perubahan signifikan pada setiap kelas dan komponennya. Dengan Google Maps API Android, API dapat secara otomatis menangani akses ke Google Maps *server*, *download* data, menampilkan peta, dan respon untuk memetakan gerakan. API yang telah disediakan juga memungkinkan untuk menambahkan *markers*, *poligon*, dan *overlay* ke peta dasar, dan mengubah pandangan pengguna dari peta daerah tertentu.

API Google Maps v2 menyediakan pustaka untuk pembangkitan peta pada aplikasi Android. Pustaka tersebut adalah *com.google.android.gms.maps.SupportMapFragment* yang memiliki kelas-kelas untuk mengunduh dan mendapatkan gambar

peta dan pilihan serta kontrol terhadap gambar peta yang ditampilkan pada aplikasi [5].

Kelas untuk menampilkan peta Google dalam Android adalah *GoogleMap*. Kelas *GoogleMap* dapat menentukan bagian peta Android yang ditampilkan sesuai dengan parameter lintang dan bujur dari posisi yang diinginkan. Kelas ini juga mengontrol perbesaran dan pengecilan secara otomatis gerakan *zoom* dengan menggerakkan peta. Tampilan dari aplikasi yang menerapkan Google Maps API dapat dilihat pada Gambar 2.1.



**Gambar 2.1** Aplikasi yang menerapkan API Google Maps v2 pada Android.

### **2.2.2 API Masukan Suara pada Android**

API masukan suara pada Android berupa sebuah layanan yang disediakan oleh Google yang membantu aplikasi untuk menerima gelombang suara yang terdengar dari sekitar perangkat dan menerjemahkannya ke dalam sebuah teks.

Layanan ini sudah tersedia pada perangkat bergerak berbasis Android. Android yang bersifat *open platform* ini memudahkan

bagi para pemrogram aplikasi dalam mengakses pengenalan suara Google.

Penggunaan API ini dilakukan dengan mengalirkan gelombang suara ke *server* audio Google. *Server* kemudian melakukan proses pengenalan suara yang dilanjutkan dengan pengiriman respon kepada klien berupa kemungkinan-kemungkinan kalimat yang paling mirip dengan suara masukan [6].

API masukan suara saat ini telah mendukung banyak bahasa, termasuk di dalamnya adalah Bahasa Indonesia. Dengan sistem android yang didukung dengan Android SDK, pengembang dapat dengan mudah menggunakan layanan ini, dimulai dengan penerimaan *RecognizerIntent* pada perangkat klien. Setelah diterimanya *RecognizerIntent*, perangkat klien merekam suara yang terdapat di sekitarnya, kemudian *streaming* audio dilakukan dan dikirimkan ke *server* Google. Layanan paket data internet menjadi syarat wajib menggunakan API ini.

Layanan API masukan suara ini sudah banyak digunakan pada perangkat lunak keluaran Google yang sekarang banyak beredar. Di antaranya adalah fitur input *keyboard* pada perangkat bergerak berbasis Android yang dilengkapi dengan pembacaan suara, Google Voice Search, bahkan untuk melakukan perintah-perintah standard di perangkat bergerak juga sudah diterapkan pada perangkat Android Samsung Galaxy S III, Samsung Galaxy Note II, and Samsung Galaxy Stellar dengan aplikasi S Voice yang dapat melakukan perintah menelepon, mengirim SMS, membuka aplikasi tertentu, dan perintah-perintah umum pada perangkat bergerak Android lainnya dengan perintah suara.

### **2.2.3 Teks-Ke-Suara pada Android**

Teks-ke-suara merupakan sebuah layanan pada sebuah perangkat untuk mengeluarkan suara sesuai dengan teks tertentu dengan aturan pelafalan yang telah ditentukan. Dengan kata lain, layanan ini digunakan untuk membacakan sebuah teks yang dapat

diterima oleh pengguna layanan sebagai suara yang sesuai dengan teks.

Layanan teks-ke-suara yang dimiliki Android telah tersedia dalam beberapa bahasa, antara lain Bahasa Inggris, Perancis, Jerman, Italia, dan bahasa Spanyol [7]. Masing-masing bahasa memiliki ketentuan pelafalan yang berbeda-beda sehingga diperlukan spesifikasi di tiap bahasa. Penggunaan layanan teks-ke-suara pada Tugas Akhir ini adalah untuk pembacaan teks perintah suara dari pengguna.

#### **2.2.4 GPS pada Android**

GPS (*Global Positioning System*) adalah sistem radio navigasi dan penentuan posisi menggunakan satelit yang dimiliki dan dikelola oleh Amerika Serikat. Sistem banyak digunakan oleh banyak orang sekaligus dalam segala cuaca, di desain untuk memberikan posisi dan kecepatan tiga dimensi yang teliti dan juga informasi mengenai waktu secara kontinyu di seluruh dunia.

Perangkat GPS menerima sinyal yang ditransmisikan oleh satelit GPS. Dalam menentukan posisi, perangkat GPS membutuhkan paling sedikit 3 satelit untuk penentuan posisi 2 dimensi (lintang dan bujur) dan 4 satelit untuk penentuan posisi 3 dimensi (lintang, bujur, dan ketinggian). Semakin banyak satelit yang diperoleh maka akurasi posisi kita akan semakin tinggi. Untuk mendapatkan sinyal tersebut, perangkat GPS harus berada di ruang terbuka. Apabila perangkat GPS kita berada dalam ruangan atau kanopi yang lebat dan daerah kita dikelilingi oleh gedung tinggi maka sinyal yang diperoleh akan semakin berkurang sehingga akan sukar untuk menentukan posisi dengan tepat atau bahkan tidak dapat menentukan posisi.

Penggunaan GPS pada pengembangan aplikasi Android diatur dalam kelas *Location Manager*. Dengan kelas ini, pengguna Android dapat mengetahui informasi mengenai titik lokasi dari perangkat Android seperti longitude, latitude, serta altitude dari lokasi perangkat Android. Pendeteksian mengenai lokasi ini memanfaatkan *GL Engine*, yang berfungsi untuk mendeteksi

beberapa satelit GPS yang merupakan GPS driver yang telah diprogram. GPS Driver sendiri merupakan sistem perangkat lunak yang menggunakan API tingkat rendah yang berkomunikasi dengan GPS Chip (radio penerima frekuensi yang secara langsung berkomunikasi dengan GPS Satelit), pada tingkat sistem itu dapat terdiri dari file tunggal atau beberapa file yang terletak di `/System/Lib/hw/` atau `/Vendor/Lib/hw/`, tergantung pada versi Android dan platform dari perangkatnya [8].

Pendeteksian lokasi dengan GPS pada Android memberikan informasi lokasi yang akurat. Namun proses ini memiliki kelemahan yaitu hanya dapat bekerja di luar ruangan, konsumsi baterai yang besar, dan respon yang lambat. Sedangkan pendeteksian lokasi dengan Network Location Provider Android dapat bekerja di dalam maupun di luar ruangan, respon yang cepat dan konsumsi baterai yang lebih sedikit. Namun proses ini memberikan informasi lokasi yang kurang akurat.

### 2.3 JSON

JSON adalah singkatan dari *Java Script Object Notation*, yaitu sebuah format untuk pertukaran data. Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh *programmer* keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. JSON terbuat dari dua struktur:

- Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
- Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman modern mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini [9].

## **2.4 Bahasa Indonesia**

Bahasa Indonesia adalah salah satu dari banyak ragam Bahasa Melayu. Bahasa Indonesia biasa digunakan dalam forum publik yang bersifat resmi semacam perguruan-perguruan, media massa, sastra, dan surat-menyurat resmi. Pada praktiknya penutur Bahasa Indonesia mencampuradukkan dengan bahasa-bahasa daerah, sehingga dalam sehari-hari penutur Bahasa Indonesia seringkali menggunakan Bahasa Indonesia versi sehari-hari (kolokial), bukan versi resmi (baku) Bahasa Indonesia. Untuk itu mempelajari Bahasa Indonesia sehari-hari juga tidak lepas dalam proses penelitian mengenai Bahasa Indonesia, terutama dalam proses pengenalan suara berbahasa Indonesia.

Salah satu penelitian mengenai teknologi Bahasa Indonesia dilakukan oleh Departemen Teknik Elektro ITB yang telah menghasilkan teknologi teks-ke-suara Bahasa Indonesia yang pertama pada tahun 2000. Sampai saat ini teknologi teks-ke-suara buatan ITB itu masih dapat dikatakan satu-satunya teknologi teks-to-speech Bahasa Indonesia yang lengkap disertai intonasi Bahasa Indonesia yang cukup baik. Untuk dapat menghasilkan intonasi yang baik, teks-ke-suara tersebut dilengkapi dengan model intonasi Bahasa Indonesia di dalamnya.

Beberapa penelitian dan pengembangan lain yang pernah dilakukan di antaranya pengembangan sistem penerjemah Bahasa Inggris-Indonesia dan Indonesia-Inggris atau bahkan penerjemah multi bahasa. Teknologi ini dapat digunakan untuk berbagai aplikasi, mulai dari aplikasi bisnis, pendidikan, kemanusiaan, bahkan pelestarian budaya. Aplikasi alat bantu baca untuk tuna netra dan alat bantu bicara untuk tuna netra telah dilakukan di



Departemen Teknik Elektro ITB, demikian pula pengembangan prototip telpon untuk orang bisu-tuli dikembangkan bersama-sama dengan Risti Telkom dan ATR-Jepang [10].

## 2.5 Spherical Law of Cosines

Spherical law of cosines adalah teorema yang berkaitan dengan sisi dan sudut segitiga pada permukaan bola yang dianalogikan dengan law of cosines pada bidang datar. Teorema ini digunakan untuk menghitung jarak antara dua titik pada permukaan bumi. Penggunaan Spherical Law of Cosines dikarenakan bentuk permukaan bumi yang berbentuk bola. Data posisi yang digunakan adalah nilai dari longitude dan latitude dari pengguna dan tempat yang dipilih [11].

Rumus dari Spherical Law of Cosines yang digunakan dalam perhitungan longitude dan latitude dapat dilihat pada Persamaan (2.1).

$$d = \text{acos}(\sin(\text{lat1}) \cdot \sin(\text{lat2}) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \cos(\text{lon2} - \text{lon1})) \cdot 6371 \quad (2.1)$$

## **BAB III**

### **ANALISIS DAN PERANCANGAN**

Bab ini membahas tentang analisis dan perancangan perangkat lunak dari Aplikasi panggilan darurat dengan perintah suara berbahasa Indonesia pada perangkat bergerak berbasis Android. Analisis perangkat lunak dilakukan berdasarkan perumusan permasalahan yang ada pada Tugas Akhir ini. Hasil analisis digunakan sebagai dasar dalam perancangan perangkat lunak.

#### **3.1 Deskripsi Umum Perangkat Lunak**

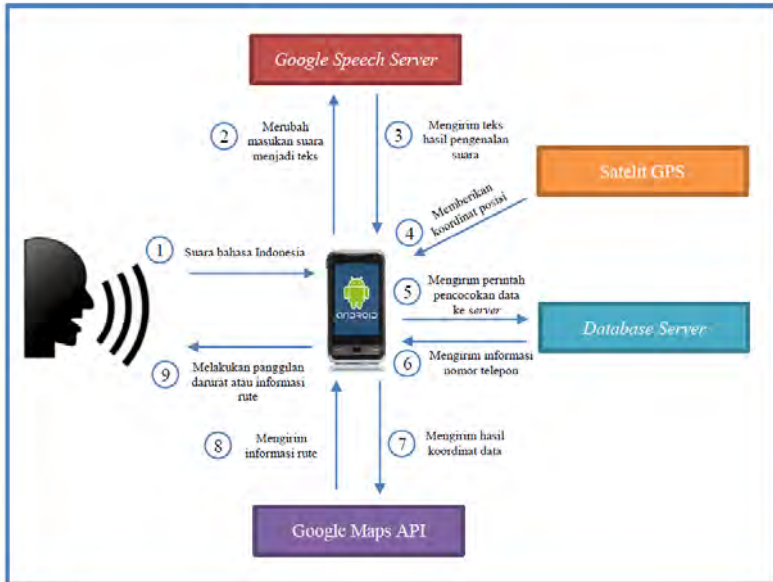
Pada Tugas Akhir ini dibuat sebuah perangkat lunak berupa aplikasi berbasis sistem operasi Android yang memanfaatkan teknologi input suara untuk melakukan pencarian informasi lokasi layanan publik terdekat yang selanjutnya dilakukan pemanggilan darurat. Suara yang diinputkan merupakan suara dalam Bahasa Indonesia yang diucapkan secara langsung pada saat menggunakan aplikasi ini.

Respon yang dikeluarkan oleh aplikasi setelah menerima permintaan oleh pengguna adalah berupa tampilan panggilan langsung dalam layar perangkat bergerak dimana nomor telepon merupakan nomor telepon layanan publik yang diinginkan pengguna beserta instruksi yang dikeluarkan dalam bentuk suara kepada pengguna. Instruksi yang dimaksud adalah instruksi jarak dan waktu tempuh, serta informasi-informasi pendukung lainnya mengenai laporan pencarian lokasi oleh aplikasi.

Aplikasi juga memungkinkan untuk melakukan penunjukan lokasi dengan inputan yang sama yaitu secara langsung pada saat menggunakan aplikasi ini. Respon yang ditampilkan oleh aplikasi ini adalah peta dalam layar perangkat bergerak Android. pada peta ditunjukkan rute lokasi pengguna menuju lokasi yang dicarinya.

Pengenalan suara pengguna dilakukan oleh perangkat Android dengan memanfaatkan API masukan suara pada Android, basis data layanan publik dan tampilan peta pada aplikasi

menggunakan Google Maps. Adapun arsitektur sistem aplikasi dapat digambarkan sebagaimana Gambar 3.1.



**Gambar 3.1 Arsitektur Sistem**

Pada Gambar 3.1 dijelaskan bahwa:

1. Pengguna menyampaikan perintah suara dengan bahasa Indonesia. Sistem menerima suara dari pengguna menggunakan Android Speech Input API.
2. Sistem menerjemahkan suara menjadi sebuah teks dengan memanfaatkan Google's Speech Recognizer.
3. Sistem mendeteksi kata pada teks hasil pengenalan suara menjadi perintah pemanggilan darurat atau penunjukan rute lokasi layanan publik. kalimat perintah dideteksi yang kemudian menghasilkan kesimpulan apakah kalimat tersebut merupakan sebuah perintah untuk menunjukkan permintaan pemanggilan atau penunjukan rute lokasi layanan publik. Jika bukan merupakan

perintah untuk menunjukkan permintaan pemanggilan atau penunjukan rute lokasi layanan publik maka aplikasi akan meminta pengguna untuk mengulangi proses penyampaian suara.

4. Sistem kemudian mengambil informasi koordinat lokasi GPS dari perangkat bergerak Android.
5. Apabila perintah panggilan darurat maka respon berupa data nama dan lokasi pengguna dikirim ke server untuk dilakukan penghitungan dengan rumus Spherical Law of Cosines agar mendapatkan informasi nomor telepon penting dan lokasi layanan publik terdekat.
6. Sistem kemudian mengirim informasi nomor telepon yang telah disimpan di server.
7. Apabila perintah penunjukan rute lokasi maka respon berupa koordinat lokasi GPS dari perangkat bergerak Android dikirimkan pada Google Maps API untuk mendapatkan posisi lokasi dan rute menuju lokasi tujuan.
8. Sistem menentukan rute terpendek dengan menggunakan algoritma Dijkstra yang diterapkan pada Google Maps berdasarkan kedua koordinat lokasi yang telah didapatkan dari GPS.
9. Visualisasi panggilan darurat ditampilkan dengan panggilan telepon langsung sedangkan penunjukan rute lokasi tujuan ditampilkan dengan Google Maps yang memanfaatkan Google Maps API yang kemudian ditampilkan kepada pengguna.

Pada aplikasi ini juga ditambahkan kontak pribadi yang dapat dikelola untuk setiap penggunanya. Kontak pribadi pengguna dapat diakses dengan suara yang diucapkan secara langsung yang kemudian mendapatkan hasil panggilan sesuai dengan nama kontak pribadi yang diinginkan oleh pengguna.

Disamping beberapa fitur yang ada pada aplikasi ini, aplikasi ini juga memiliki fitur pesan darurat yang dapat dikirimkan pada nomor darurat yang telah disimpan pada basis data pengguna.

Untuk melakukan pesan darurat, pengguna dapat melakukan sms satu kali ke nomor darurat dan sms berkali-kali hingga sistem dimatikan oleh pengguna. Isi pesan darurat merupakan data pesan darurat yang sudah disimpan oleh pengguna beserta lokasi pengguna ketika melakukan pengiriman pesan darurat secara lengkap.

### **3.2 Analisis Perangkat Lunak**

Analisis perangkat lunak disusun berdasarkan rumusan permasalahan. Analisis yang dilakukan meliputi spesifikasi kebutuhan perangkat lunak, analisis aktor dan analisis kasus penggunaan.

#### **3.2.1 Spesifikasi Kebutuhan Perangkat Lunak**

Berdasarkan pada deskripsi umum aplikasi, maka terdapat beberapa hal yang harus dipenuhi untuk membangun aplikasi ini yang selanjutnya disebut sebagai spesifikasi kebutuhan perangkat lunak. Spesifikasi kebutuhan dari aplikasi ini adalah sebagai berikut.

- a. Aplikasi ini dapat mengubah masukan suara pengguna menjadi teks.
- b. Aplikasi ini dapat mendeteksi perintah dari kalimat yang diucapkan oleh pengguna.
- c. Aplikasi ini dapat mendeteksi perintah panggilan darurat layanan publik terdekat.
- d. Aplikasi ini dapat mendeteksi perintah panggilan data kontak pribadi.
- e. Aplikasi ini dapat mendeteksi perintah pencarian layanan publik terdekat.
- f. Aplikasi ini dapat mendeteksi perintah penunjukan rute ke suatu tempat dan layanan publik terdekat.
- g. Aplikasi ini dapat mengidentifikasi posisi pengguna.
- h. Aplikasi ini dapat mengidentifikasi nama lokasi pengguna.

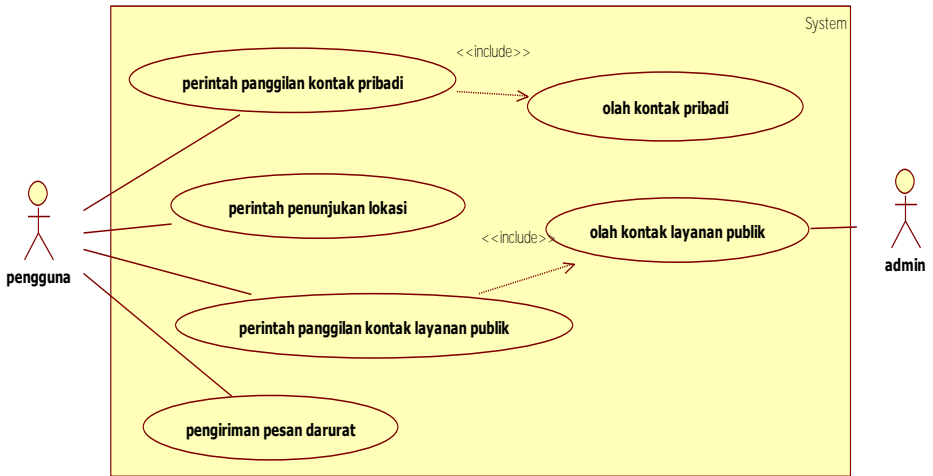
- i. Aplikasi ini dapat mendeteksi koordinat tujuan yang dicari oleh pengguna.
- j. Aplikasi ini dapat melakukan panggilan darurat.
- k. Aplikasi ini dapat melakukan pengiriman pesan darurat.
- l. Aplikasi ini dapat membangkitkan rute dari posisi pengguna menuju lokasi layanan publik terdekat.
- m. Aplikasi ini dapat membangkitkan rute dari posisi pengguna menuju lokasi tujuan.
- n. Aplikasi ini dapat menampilkan posisi pengguna pada peta.
- o. Aplikasi ini dapat menyimpan data kontak pribadi pengguna aplikasi.
- p. Aplikasi ini dapat menyimpan data kontak layanan publik pada pengguna admin.

### **3.2.2 Analisis Aktor**

Aktor pada perangkat lunak ini terdiri dari pengguna admin dan pengguna aplikasi. Pengguna admin adalah orang yang dapat melakukan manajemen data layanan publik. Pengguna aplikasi adalah orang yang menjalankan aplikasi meliputi panggilan darurat layanan publik, panggilan kontak pribadi, penunjukan lokasi, pengiriman pesan darurat dan manajemen data kontak pribadi. Penjelasan lebih lanjut tentang keterlibatan pengguna dengan perangkat lunak akan dijelaskan pada subbab Analisis Kasus Penggunaan.

### **3.2.3 Analisis Kasus Penggunaan**

Setelah proses analisis mengenai spesifikasi kebutuhan perangkat lunak dan analisis aktor, maka dapat dirumuskan analisis kasus penggunaan berdasarkan aktor dan spesifikasi kebutuhan perangkat lunak ini. Kasus Penggunaan pada perangkat lunak ini dijelaskan pada Gambar 3.2.



**Gambar 3.2 Diagram Kasus Penggunaan**

Diagram kasus penggunaan pada Gambar 3.2 menjelaskan bahwa terdapat enam kasus penggunaan pada perangkat lunak ini, yakni perintah panggilan kontak pribadi, perintah penunjukan lokasi, perintah panggilan layanan publik, pengiriman pesan darurat, olah kontak pribadi, dan olah kontak layanan publik. Penjelasan dari masing-masing kasus penggunaan terdapat pada Tabel 3.1 berikut ini.

**Tabel 3.1 Deskripsi Kasus Penggunaan**

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1	UC-001	Perintah Panggilan Kontak Pribadi	Pengguna mendapatkan nomor dan melakukan panggilan pada kontak pribadi melalui perintah suara

2	UC-002	Perintah Penunjukan Lokasi	Pengguna mendapatkan mendapatkan penunjukan lokasi yang diperlihatkan pada peta
3	UC-003	Perintah Panggilan Kontak Layanan Publik	Pengguna mendapatkan nomor dan melakukan panggilan pada kontak layanan publik terdekat melalui perintah suara
4	UC-004	Pengiriman Pesan Darurat	Pengguna mengirim pesan darurat melalui tombol sms darurat atau mode darurat
5	UC-005	Olah Kontak Pribadi	Pengguna melakukan olah kontak pribadi
6	UC-006	Olah Kontak Layanan Publik	Admin melakukan olah kontak layanan publik

### 3.2.3.1 Kasus Penggunaan Panggilan Kontak Pribadi (UC-001)

Pada kasus penggunaan panggilan kontak pribadi, pengguna dapat melakukan panggilan dengan menggunakan masukan suara. Panggilan ditampilkan dengan panggilan langsung kepada masukan nama pengguna yang dipilih. Spesifikasi kasus penggunaan panggilan kontak pribadi dijelaskan pada Tabel 3.2.

**Tabel 3.2 Spesifikasi Kasus Penggunaan Panggilan Kontak Pribadi**

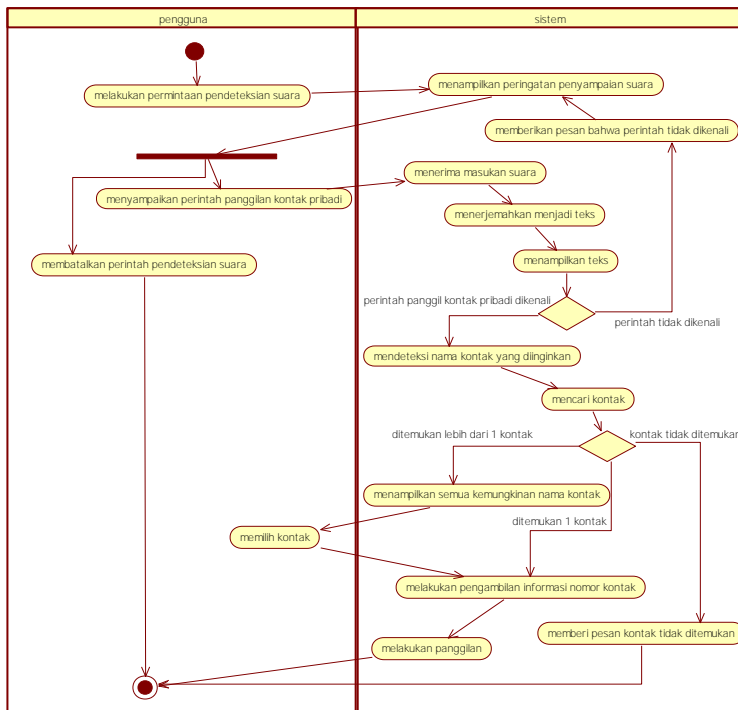
<b>Nama Kasus Penggunaan</b>	<b>Panggilan Kontak Pribadi</b>
Kode	UC-001
Deskripsi	Pengguna melakukan panggilan kontak pribadi yang dimintanya melalui perintah suara.
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada halaman komunikasi
Alur Normal	1. Pengguna melakukan permintaan pendeteksian suara



	<ol style="list-style-type: none"> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia       <ol style="list-style-type: none"> <li>A1. Pengguna membatalkan permintaan pendeteksian suara</li> </ol> </li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks</li> <li>7. Sistem mendeteksi kalimat yang merupakan nama kontak pribadi       <ol style="list-style-type: none"> <li>A2. Sistem tidak mengenali perintah</li> </ol> </li> <li>8. Sistem mencari nama kontak pribadi       <ol style="list-style-type: none"> <li>A3. Sistem tidak menemukan nama kontak yang cocok</li> <li>A4. Sistem menemukan lebih dari 1 kontak</li> </ol> </li> <li>9. Sistem melakukan pengambilan informasi kontak pribadi</li> <li>10. Sistem melakukan panggilan</li> <li>11. Kasus penggunaan panggilan kontak pribadi berakhir</li> </ol>
Alur Alternatif	<ol style="list-style-type: none"> <li>A1. Pengguna membatalkan permintaan pendeteksian suara       <ol style="list-style-type: none"> <li>A1.1. Menuju alur normal nomor 11</li> </ol> </li> <li>A2. Sistem tidak mengenali perintah       <ol style="list-style-type: none"> <li>A2.1. Sistem memberikan peringatan bahwa perintah tidak dikenali</li> <li>A2.2. Menuju alur normal nomor 2</li> </ol> </li> <li>A3. Sistem tidak menemukan nama kontak yang cocok       <ol style="list-style-type: none"> <li>A3.1. Sistem memberikan pesan bahwa nama kontak tidak ditemukan</li> <li>A3.2. Menuju alur normal nomor 11</li> </ol> </li> <li>A4. Sistem menemukan lebih dari 1 nama kontak       <ol style="list-style-type: none"> <li>A4.1. Sistem menampilkan semua kemungkinan nama kontak</li> <li>A4.2. Pengguna memilih salah satu pilihan dengan mengucapkan nomor pilihannya</li> </ol> </li> </ol>

	A4.3. Sistem memroses pilihan pengguna sebagai nama kontak A4.4. Menuju alur normal nomor 9
Kondisi Akhir	Pengguna melakukan panggilan kontak pribadi

Berdasarkan pada spesifikasi kasus penggunaan pemanggilan kontak pribadi pada Tabel 3.2 yang melibatkan aktor dan sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan panggilan kontak pribadi seperti pada Gambar 3.3 berikut.



**Gambar 3.3 Diagram Aktivitas Kasus Penggunaan Panggilan Kontak Pribadi**

### 3.2.3.2 Kasus Penggunaan Penunjukan Lokasi (UC-002)

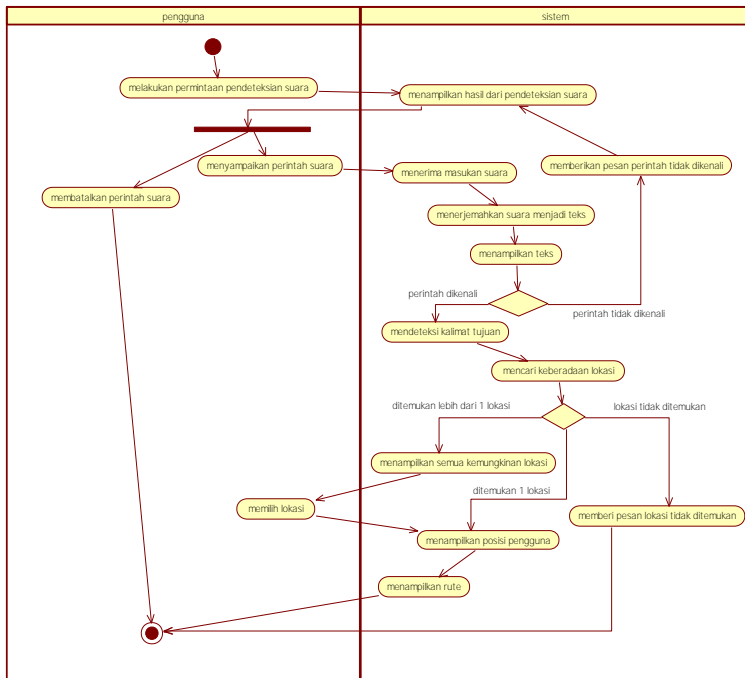
Pada kasus penggunaan penunjukan lokasi, pengguna dapat melakukan penunjukan lokasi dengan menggunakan masukan suara. Penunjukan lokasi ditampilkan dengan Google Map berupa posisi pengguna menuju lokasi yang diinginkan. Spesifikasi kasus penggunaan penunjukan lokasi dijelaskan pada Tabel 3.3.

**Tabel 3.3 Spesifikasi Kasus Penggunaan Penunjukan Lokasi**

Nama Kasus Penggunaan	Penunjukan Lokasi
Kode	UC-002
Deskripsi	Pengguna mendapatkan penunjukan lokasi yang diperlihatkan pada peta.
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada halaman komunikasi.
Alur Normal	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia               <ol style="list-style-type: none"> <li>A1. Pengguna membatalkan permintaan pendeteksian suara</li> </ol> </li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks</li> <li>7. Sistem mendeteksi kalimat tujuan               <ol style="list-style-type: none"> <li>A2. Sistem tidak mengenali perintah</li> </ol> </li> <li>8. Sistem mencari keberadaan lokasi</li> <li>9. Sistem menampilkan posisi pengguna               <ol style="list-style-type: none"> <li>A3. Sistem tidak menemukan nama lokasi</li> <li>A4. Sistem menemukan lebih dari 1 lokasi</li> </ol> </li> <li>10. Sistem menampilkan rute</li> <li>11. Kasus penggunaan penunjukan lokasi berakhir</li> </ol>
Alur Alternatif	A1. Pengguna membatalkan permintaan pendeteksian suara

	<ul style="list-style-type: none"> <li>A1.1. Menuju alur normal nomor 11</li> <li>A2. Sistem tidak mengenali perintah <ul style="list-style-type: none"> <li>A2.1. Sistem memberikan peringatan bahwa perintah tidak dikenali</li> <li>A2.2. Menuju alur normal nomor 2</li> </ul> </li> <li>A3. Sistem tidak menemukan nama lokasi <ul style="list-style-type: none"> <li>A3.1. Sistem memberikan pesan bahwa nama lokasi tidak ditemukan</li> <li>A3.2. Menuju alur normal nomor 11</li> </ul> </li> <li>A4. Sistem menemukan lebih dari 1 nama lokasi <ul style="list-style-type: none"> <li>A4.1. Sistem menampilkan semua kemungkinan nama lokasi</li> <li>A4.2. Pengguna memilih salah satu pilihan dengan mengucapkan nomor pilihannya</li> <li>A4.3. Sistem memroses pilihan pengguna sebagai nama lokasi</li> <li>A4.4. Menuju alur normal nomor 9</li> </ul> </li> </ul>
Kondisi Akhir	Pengguna melihat lokasi posisinya dan rute lokasi pada peta

Berdasarkan pada spesifikasi kasus penggunaan penunjukan lokasi pada Tabel 3.3 yang melibatkan aktor dan sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan penunjukan lokasi seperti pada Gambar 3.4 berikut.



**Gambar 3.4 Diagram Aktivitas Kasus Penggunaan Penunjukan Lokasi**

### 3.2.3.3 Kasus Penggunaan Panggilan Kontak Layanan Publik (UC-003)

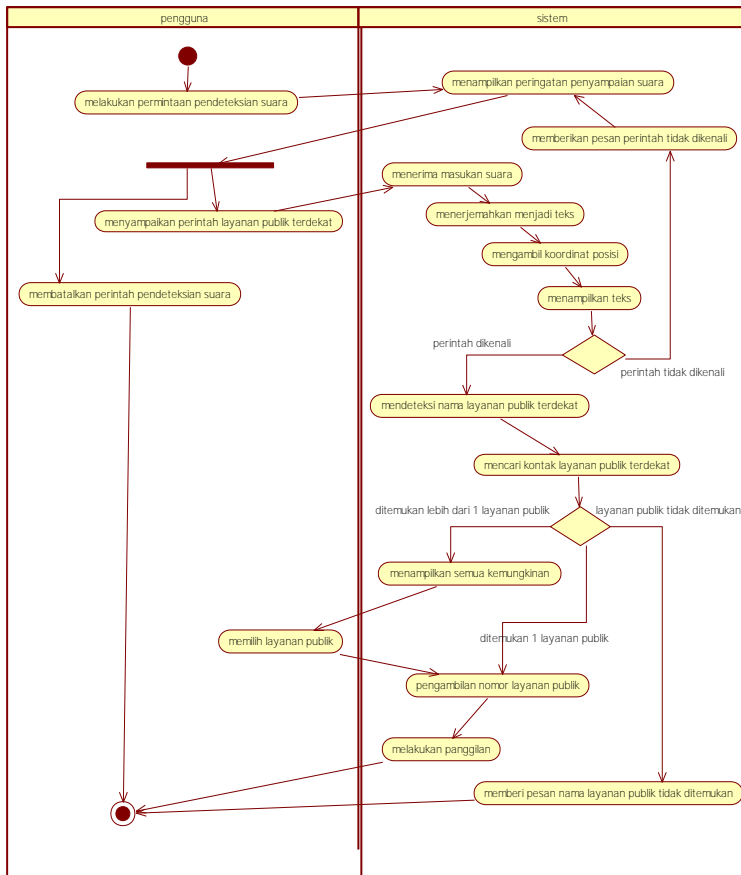
Pada kasus penggunaan panggilan kontak layanan publik, pengguna dapat melakukan panggilan dengan menggunakan masukan suara. Panggilan ditampilkan dengan panggilan langsung kepada masukan nama layanan publik yang dipilih dan memiliki posisi terdekat dengan koordinat posisi pengguna. Spesifikasi kasus penggunaan panggilan kontak layanan publik dijelaskan pada Tabel 3.4.

**Tabel 3.4 Spesifikasi Kasus Penggunaan Panggilan Kontak Layanan Publik**

<b>Nama Kasus Penggunaan</b>	<b>Panggilan Layanan Publik</b>
Kode	UC-003
Deskripsi	Pengguna melakukan panggilan layanan publik terdekat yang dimintanya melalui perintah suara.
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada halaman komunikasi
Alur Normal	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia               <ol style="list-style-type: none"> <li>A1. Pengguna membatalkan permintaan pendeteksian suara</li> </ol> </li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem mengambil koordinat posisi pengguna</li> <li>7. Sistem menampilkan teks</li> <li>8. Sistem mendeteksi kalimat yang merupakan nama layanan publik dan terdekat dengan pengguna               <ol style="list-style-type: none"> <li>A2. Sistem tidak mengenali perintah</li> </ol> </li> <li>9. Sistem mencari nama layanan kontak publik               <ol style="list-style-type: none"> <li>A3. Sistem tidak menemukan nama layanan publik yang cocok</li> <li>A4. Sistem menemukan lebih dari 1 layanan publik</li> </ol> </li> <li>10. Sistem melakukan pengambilan informasi kontak layanan publik</li> <li>11. Sistem melakukan panggilan</li> <li>12. Kasus penggunaan panggilan layanan publik terdekat berakhir</li> </ol>
Alur Alternatif	A1. Pengguna membatalkan permintaan pendeteksian suara

	<p>A1.2. Menuju alur normal nomor 12</p> <p>A2. Sistem tidak mengenali perintah</p> <p>A2.1. Sistem memberikan peringatan bahwa perintah tidak dikenali</p> <p>A2.2. Menuju alur normal nomor 2</p> <p>A3. Sistem tidak menemukan nama layanan publik yang cocok</p> <p>A3.1. Sistem memberikan pesan bahwa nama layanan publik tidak ditemukan</p> <p>A3.2. Menuju alur normal nomor 12</p> <p>A4. Sistem menemukan lebih dari 1 nama layanan publik</p> <p>A4.1. Sistem menampilkan semua kemungkinan nama layanan publik</p> <p>A4.2. Pengguna memilih salah satu pilihan dengan mengucapkan nomor pilihannya</p> <p>A4.3. Sistem memproses pilihan pengguna sebagai nama layanan publik</p> <p>A4.4. Menuju alur normal nomor 10</p>
Kondisi Akhir	Pengguna melakukan panggilan layanan publik terdekat

Berdasarkan pada spesifikasi kasus penggunaan pemanggilan layanan publik pada Tabel 3.4 yang melibatkan aktor dan sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan pemanggilan layanan publik seperti pada Gambar 3.5 berikut.



**Gambar 3.5 DiagramAktivitas Kasus Penggunaan Panggilan Kontak Layanan Publik**

### 3.2.3.4 Kasus Penggunaan Pengiriman Pesan Darurat (UC-004)

Pada kasus penggunaan pengiriman pesan darurat, pengguna dapat melakukan pengiriman pesan darurat dengan memilih tombol sms darurat atau pengiriman dengan rentang waktu yang



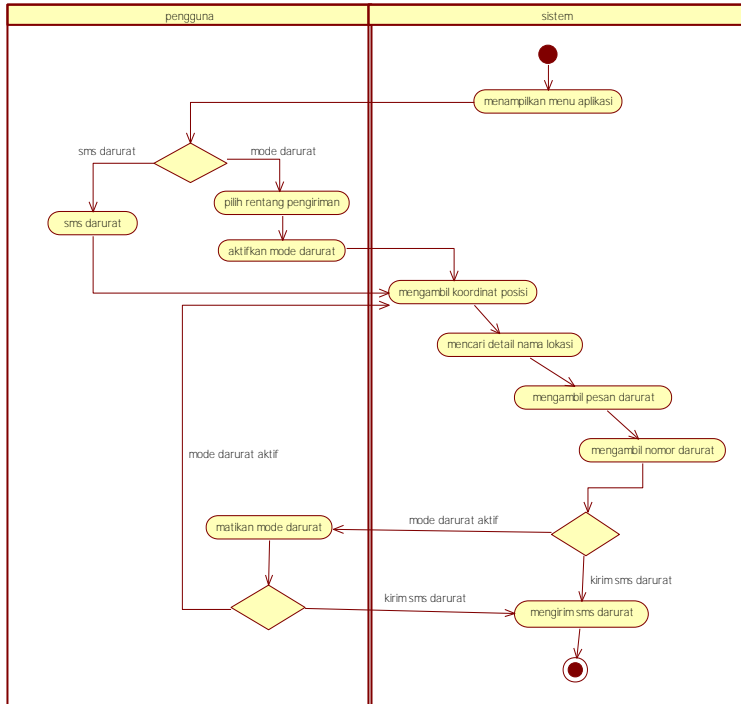
ditentukan pengguna kemudian menekan tombol mode darurat. Pengiriman pesan darurat akan berhenti ketika pengguna menekan tombol hentikan mode darurat. Spesifikasi kasus penggunaan pengiriman pesan darurat dijelaskan pada Tabel 3.5.

**Tabel 3.5 Spesifikasi Kasus Penggunaan Pengiriman Pesan Darurat**

<b>Nama Kasus Penggunaan</b>	<b>Pengiriman Pesan Darurat</b>
Kode	UC-004
Deskripsi	Pengguna melakukan pengiriman pesan darurat dengan memilih sms darurat atau mode darurat
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada halaman menu aplikasi
Alur Normal	<ol style="list-style-type: none"> <li>1. Sistem menampilkan menu aplikasi</li> <li>2. Pengguna memilih sms darurat               <ol style="list-style-type: none"> <li>A1. Pengguna memilih rentang pengiriman</li> <li>A2. Pengguna aktifkan mode darurat</li> <li>A3. Menuju alur nomor 3</li> </ol> </li> <li>3. Sistem mengambil koordinat posisi pengguna</li> <li>4. Sistem mencari detail nama lokasi</li> <li>5. Sistem mengambil pesan darurat pada basis data pengguna</li> <li>6. Sistem mengambil nomor darurat pada basis data pengguna</li> <li>7. Sistem mengirim sms darurat               <ol style="list-style-type: none"> <li>A4. Sistem mendeteksi mode darurat aktif</li> <li>A5. Pengguna mematikan mode darurat</li> <li>A6. Menuju alur nomor 3</li> </ol> </li> <li>8. Kasus penggunaan pengiriman pesan darurat berakhir</li> </ol>
Kondisi Akhir	Sistem melakukan pengiriman pesan darurat

Berdasarkan pada spesifikasi kasus penggunaan pengiriman pesan darurat pada Tabel 3.5 yang melibatkan pengguna dan

sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan pengiriman pesan darurat seperti pada Gambar 3.6 berikut.



**Gambar 3.6 Diagram Aktivitas Kasus Penggunaan Pengiriman Pesan Darurat**

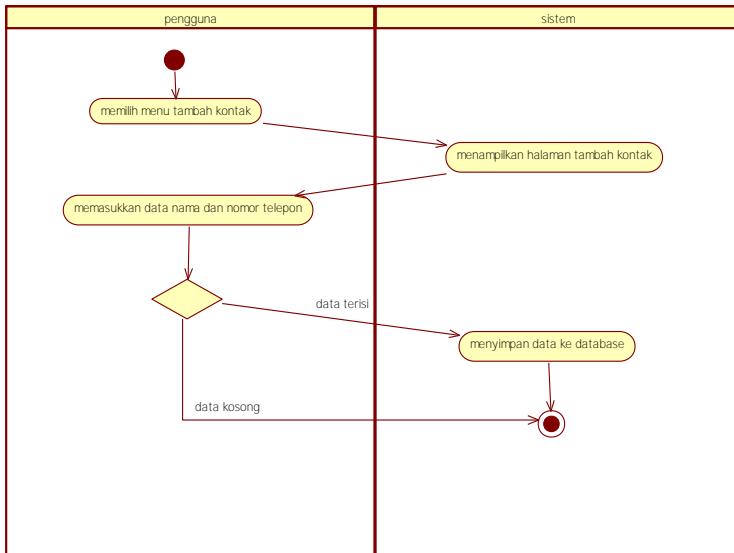
### 3.2.3.5 Kasus Penggunaan Olah Kontak Pribadi (UC-005)

Pada kasus penggunaan olah kontak pribadi, pengguna dapat menambahkan data kontak pribadi, melakukan pembaruan kontak dan menghapus kontak. Spesifikasi kasus penggunaan olah kontak pribadi dijelaskan pada Tabel 3.6.

**Tabel 3.6 Spesifikasi Kasus Penggunaan Olah Kontak Pribadi**

<b>Nama Kasus Penggunaan</b>	<b>Olah Kontak Pribadi</b>
Kode	UC-005
Deskripsi	Pengguna melakukan olah kontak pribadi.
Aktor	Pengguna
Kondisi Awal	Pengguna berada pada halaman kontak pribadi
Alur Normal	<ol style="list-style-type: none"> <li>1. Pengguna memilih menu tambah kontak</li> <li>2. Sistem menampilkan halaman tambah kontak</li> <li>3. Pengguna memasukkan data nama dan nomor telepon</li> <li>4. Sistem menyimpan data kontak pribadi</li> <li>5. Kasus penggunaan olah kontak pribadi berakhir</li> </ol>
Kondisi Akhir	Sistem menyimpan data kontak pribadi pada basis data kontak

Berdasarkan pada spesifikasi kasus penggunaan olah kontak pribadi pada Tabel 3.6 yang melibatkan aktor dan sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan olah kontak pribadi seperti pada Gambar 3.7 berikut.



**Gambar 3.7 Diagram Aktivitas Kasus Penggunaan Olah Kontak Pribadi**

### 3.2.3.6 Kasus Penggunaan Olah Kontak Layanan Publik (UC-006)

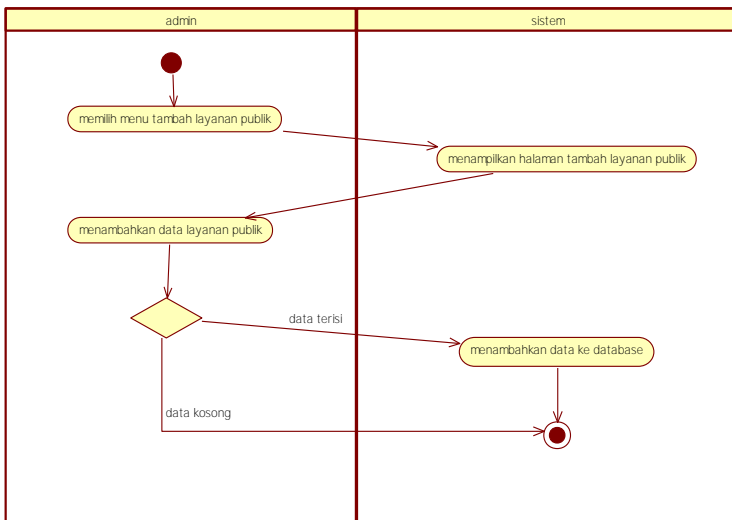
Pada kasus penggunaan olah kontak layanan publik, pengguna dapat menambahkan data layanan publik, melakukan pembaruan layanan publik dan menghapus layanan publik. Spesifikasi kasus penggunaan olah kontak layanan publik dijelaskan pada Tabel 3.7.

**Tabel 3.7 Spesifikasi Kasus Penggunaan Olah Kontak Layanan Publik**

Nama Kasus Penggunaan	Olah Kontak Layanan Publik
Kode	UC-006
Deskripsi	Admin melakukan olah kontak layanan publik.

Aktor	Admin
Kondisi Awal	Pengguna berada pada halaman admin
Alur Normal	<ol style="list-style-type: none"> <li>1. Admin memilih menu tambah layanan publik</li> <li>2. Sistem menampilkan halaman tambah layanan publik</li> <li>3. Admin memasukkan data layanan publik</li> <li>4. Sistem menambahkan data ke basis data layanan publik</li> <li>5. Kasus penggunaan olah kontak pribadi berakhir</li> </ol>
Kondisi Akhir	Sistem menyimpan data layanan publik pada basis data layanan publik

Berdasarkan pada spesifikasi kasus penggunaan olah layanan publik pada Tabel 3.7 yang melibatkan aktor dan sistem, maka dapat digambarkan diagram aktivitas untuk kasus penggunaan olah layanan publik seperti pada Gambar 3.8 berikut.

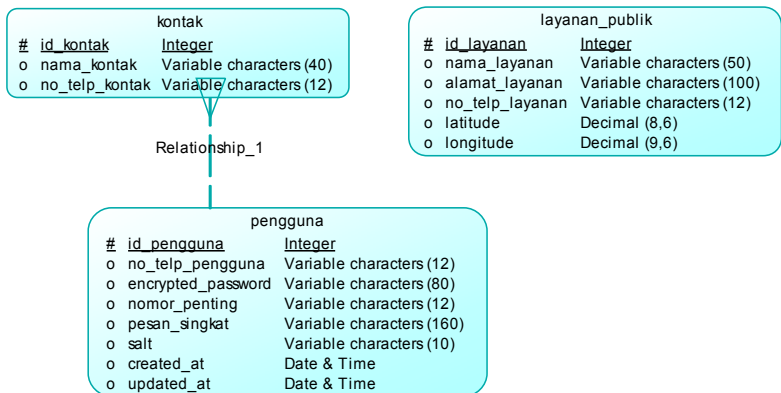


**Gambar 3.8 Diagram Aktivitas Kasus Penggunaan Olah Layanan Publik**

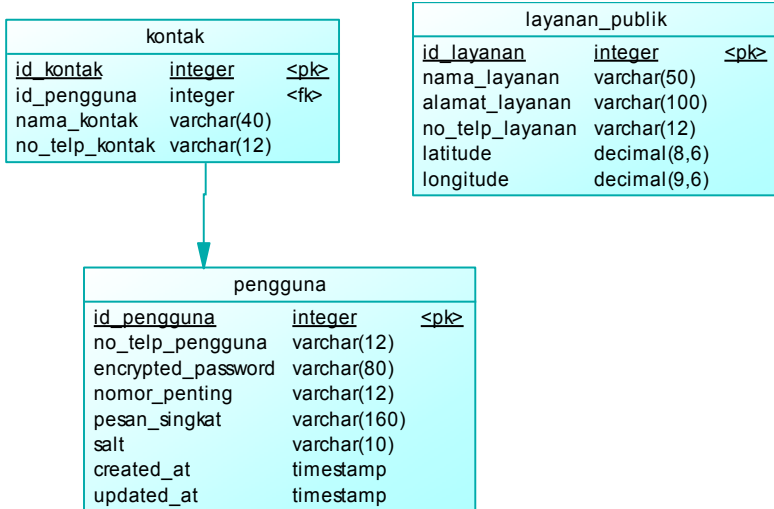
### 3.3 Perancangan Basis Data

Basis data merupakan sekumpulan informasi yang disimpan di dalam server secara sistematis sehingga dapat diperiksa menggunakan suatu data terformat untuk memperoleh informasi dari basis data tersebut [JSON].

Ilustrasi perancangan basis data digambarkan menggunakan CDM (*Conceptual Data Model*) pada Gambar 3.9 dan PDM (*Physical Data Model*) seperti pada Gambar 3.10. Spesifikasi basis data sistem dijelaskan pada Tabel 3.8.



Gambar 3.9 CDM Basis Data Tolong



Gambar 3.10 PDM Basis Data Tolong

Tabel 3.8 Spesifikasi Basis Data Tolong

No	Tabel	Atribut	Tipe Data	Fungsi
1.	Pengguna	id_pengguna	Int (11)	Menyimpan data pengguna
		no_telp_pengguna	Varchar (12)	
		encrypted_password	Varchar (80)	
		nomor_penting	Varchar (12)	
		pesan_singkat	Varchar (160)	
		salt	Varchar (10)	
		created_at	Datetime	
		updated_at	Datetime	
2.	Kontak	id kontak	Int (11)	Menyimpan data kontak pengguna
		nama kontak	Varchar (40)	
		no_telp_kontak	Varchar (12)	
3.	Layanan Publik	id layanan	Int (11)	Menyimpan data kontak layanan pengguna
		nama layanan	Varchar (50)	
		alamat layanan	Varchar (100)	
		no_telp layanan	Varchar (12)	
		latitude	Decimal (8,6)	
		longitude	Decimal (9,6)	

### 3.4 Perancangan Perangkat Lunak

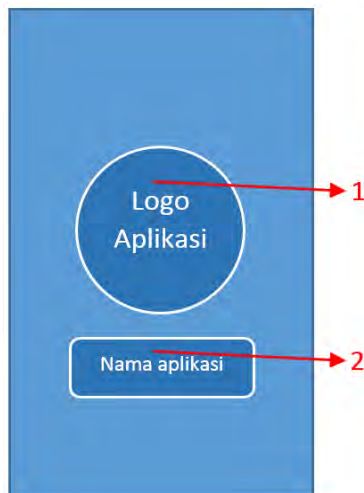
Perancangan perangkat lunak merupakan tahap lanjutan dari analisis perangkat lunak. Tahap ini merupakan penjelasan mengenai perancangan antarmuka pengguna, perancangan diagram kelas, serta perancangan proses.

#### 3.4.1 Perancangan Antarmuka

Antarmuka pengguna merupakan bagian perangkat lunak yang berhubungan langsung dengan pengguna sebagai media komunikasi antara perangkat lunak dengan pengguna. Pada bagian ini dijelaskan mengenai perancangan antarmuka pada aplikasi panggilan darurat dengan perintah suara berbahasa Indonesia pada perangkat bergerak berbasis Android.

##### 3.4.1.1 Antarmuka Halaman Pembuka

Antarmuka halaman pembuka merupakan antarmuka yang ditampilkan ketika aplikasi pertama kali dijalankan Rancangan antarmuka halaman pembuka dapat dilihat pada Gambar 3.11.



Gambar 3.11 Rancangan Antarmuka Halaman Pembuka



Pada rancangan antarmuka halaman pembuka yang terlihat pada Gambar 3.11, tampilan layar belakang berwarna biru. Pada antarmuka ini terdapat bagian-bagian antarmuka sebagai berikut.

### **1. Logo Aplikasi**

Logo merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman pembuka dengan nomor 1.

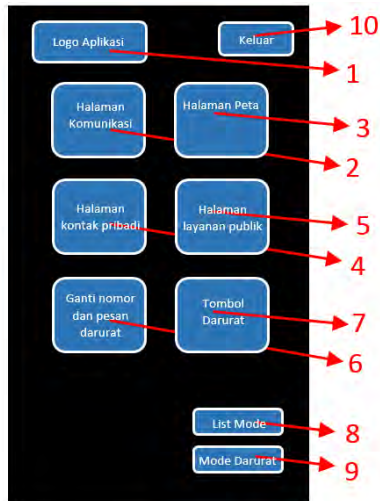
### **2. Nama Aplikasi**

Nama aplikasi berada pada posisi di bawah logo aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman pembuka dengan nomor 2.

#### **3.4.1.2 Antarmuka Halaman Menu Aplikasi**

Antarmuka halaman menu aplikasi merupakan antarmuka yang ditampilkan ketika pengguna telah mendaftar dan merupakan antarmuka awal pada aplikasi. Pada halaman menu aplikasi ditampilkan logo aplikasi, pilihan halaman komunikasi, pilihan halaman kontak pribadi, pilihan halaman peta, pilihan halaman layanan publik, tombol edit pesan darurat, tombol sms darurat, list rentang mode, tombol mode darurat, dan tombol keluar.

Tampilan layar belakang dipilih dengan warna hitam agar dapat memperjelas pengguna aplikasi untuk dapat mempergunakan tiap fungsi pada halaman menu aplikasi dengan mudah dan sederhana. Tampilan halaman antarmuka awal aplikasi dapat dilihat pada Gambar 3.12.



**Gambar 3.12 Rancangan Antarmuka Halaman Menu Aplikasi**

Pada rancangan antarmuka halaman menu aplikasi yang terlihat pada Gambar 3.12, tampilan layar belakang berwarna hitam. Pada antarmuka ini terdapat bagian-bagian antarmuka sebagai berikut.

### **1. Logo Aplikasi**

Logo aplikasi merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 1.

### **2. Menu Komunikasi**

Menu komunikasi merupakan menu yang digunakan untuk menuju halaman komunikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 2.

### **3. Menu Peta**

Menu peta merupakan menu yang digunakan untuk menuju halaman peta. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 3.

### **4. Menu Kontak Pribadi**

Menu kontak pribadi merupakan menu yang digunakan untuk menuju halaman kontak pribadi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 4.

### **5. Menu Layanan Publik**

Menu layanan publik merupakan menu yang digunakan untuk menuju halaman layanan publik. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 5.

### **6. Menu Ganti Nomor Darurat dan Pesan Darurat**

Menu ganti nomor darurat dan pesan darurat merupakan menu yang digunakan untuk melakukan perubahan data nomor darurat dan pesan darurat. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 6.

### **7. Tombol SMS Darurat**

Tombol sms darurat merupakan tombol yang digunakan untuk melakukan pengiriman sms darurat. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 7.

### **8. List Rentang Mode Darurat**

List rentang mode darurat merupakan list yang digunakan untuk menentukan rentang waktu pengiriman

sms pada mode darurat sesuai dengan pilihan pengguna. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 8.

### **9. Tombol Mode Darurat**

Tombol mode darurat merupakan tombol yang digunakan untuk mengaktifkan mode darurat. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 9.

### **10. Tombol Keluar**

Tombol keluar merupakan tombol yang digunakan untuk keluar dari aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman menu aplikasi dengan nomor 10.

#### **3.4.1.3 Antarmuka Halaman Komunikasi**

Antarmuka halaman komunikasi digunakan untuk menampilkan komunikasi antara perintah suara yang diberikan oleh pengguna serta respon dari aplikasi. Pada halaman komunikasi ditampilkan percakapan antara pengguna dan sistem dengan posisi kalimat komunikasi pengguna berada di sisi kanan dengan latar biru sedangkan kalimat komunikasi sistem di sebelah kiri dengan latar abu-abu. Rancangan antarmuka halaman komunikasi dapat dilihat pada Gambar 3.13.



**Gambar 3.13 Rancangan Antarmuka Halaman Komunikasi**

Pada rancangan antarmuka halaman komunikasi yang terlihat pada Gambar 3.13, terdapat bagian-bagian antarmuka sebagai berikut.

### **1. Logo Aplikasi**

Logo aplikasi merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman komunikasi dengan nomor 1.

### **2. Tombol Bicara**

Tombol ini digunakan untuk memulai permintaan pengenalan suara oleh pengguna. Tombol ditekan pertama kali untuk mengenali permintaan pengguna. Tombol ini juga ditekan ketika pengguna ingin memberikan konfirmasi kepada aplikasi untuk pemilihan

nomor apabila hasil yang ditemukan lebih dari satu hasil yang cocok. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman komunikasi dengan nomor 2.

### **3. Teks Perintah Suara Pengguna**

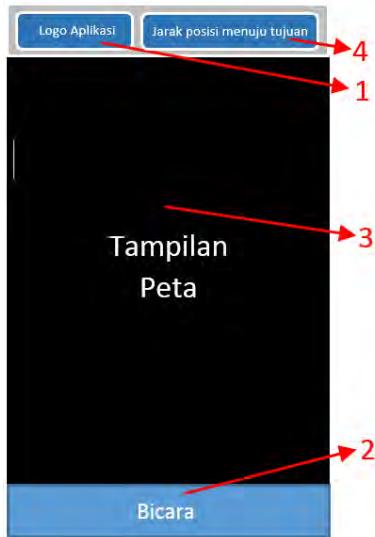
Teks perintah suara pengguna merupakan teks hasil pengenalan suara yang dilakukan aplikasi terhadap masukan suara yang diberikan oleh pengguna. Daftar teks akan ditambahkan sesaat setelah pengguna memberikan masukan suara. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman komunikasi dengan nomor 3.

### **4. Teks Respon Aplikasi**

Teks respon aplikasi merupakan teks respon yang dibuat dari aplikasi terhadap perintah suara yang diterima aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman komunikasi dengan nomor 4.

#### **3.4.1.4 Antarmuka Halaman Peta**

Antarmuka halaman peta digunakan untuk menampilkan peta beserta rute, posisi pengguna, penanda lokasi tujuan, serta informasi jarak menuju lokasi tujuan. Rancangan antarmuka halaman peta dapat dilihat pada Gambar 3.14.



**Gambar 3.14 Rancangan Antarmuka Halaman Peta**

Pada rancangan antarmuka halaman peta yang terlihat pada Gambar 3.14, terdapat bagian-bagian antarmuka sebagai berikut.

**1. Logo Aplikasi**

Logo aplikasi merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman peta dengan nomor 1.

**2. Tombol Bicara**

Tombol ini digunakan untuk memulai permintaan pengenalan suara oleh pengguna. Tombol ditekan pertama kali untuk mengenali permintaan pengguna. Tombol ini juga ditekan ketika pengguna ingin

memberikan konfirmasi kepada aplikasi untuk pemilihan nomor apabila hasil yang ditemukan lebih dari satu hasil yang cocok. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman peta dengan nomor 2.

### **3. Peta**

Peta merupakan bagian yang ditampilkan paling besar di halaman ini. Di dalam peta tergambar pula posisi pengguna serta lokasi tujuan. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman peta dengan nomor 3.

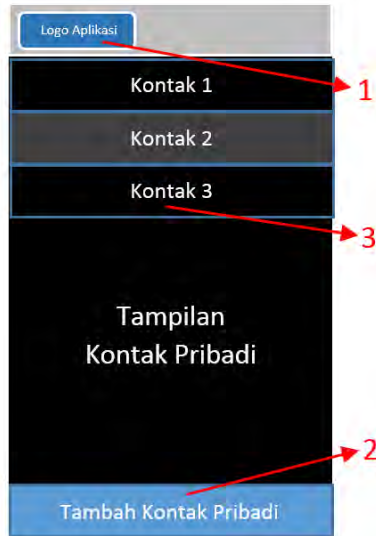
### **4. Jarak dan Waktu Sisa**

Bagian ini menampilkan jarak dari tempat pengguna berada hingga tujuan akhir dan waktu. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman peta dengan nomor 4.

#### **3.4.1.5 Antarmuka Halaman Kontak Pribadi**

Antarmuka kontak pribadi digunakan untuk menampilkan data kontak pribadi yang sudah disimpan oleh pengguna. Data kontak yang ditampilkan meliputi nama kontak pribadi dan nomor kontak pribadi. Pada halaman ini data kontak pribadi diurutkan dengan *ascending*. Rancangan antarmuka halaman kontak pribadi dapat dilihat pada Gambar 3.15.





**Gambar 3.15 Rancangan Antarmuka Halaman Kontak Pribadi**

Pada rancangan antarmuka halaman kontak pribadi yang terlihat pada Gambar 3.15, terdapat bagian-bagian antarmuka sebagai berikut.

**1. Logo Aplikasi**

Logo aplikasi merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak pribadi dengan nomor 1.

**2. Tombol Tambah Kontak Pribadi**

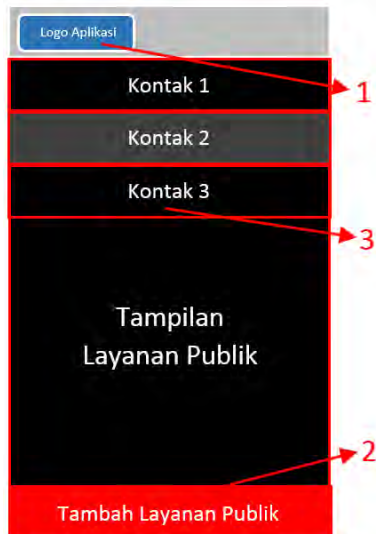
Tombol ini digunakan untuk melakukan penambahan kontak pribadi pengguna. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak pribadi dengan nomor 2.

### 3. Baris Kontak Pribadi

Baris kontak pribadi merupakan bagian yang ditampilkan merupakan data kontak pribadi yang telah disimpan oleh pengguna. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak pribadi dengan nomor 3.

#### 3.4.1.6 Antarmuka Halaman Kontak Layanan Publik

Antarmuka kontak layanan publik digunakan untuk menampilkan data kontak layanan publik yang sudah disimpan oleh admin. Data kontak yang ditampilkan meliputi nama layanan publik, alamat layanan publik, nomor telepon layanan publik, dan lokasi layanan publik. Rancangan antarmuka halaman kontak layanan publik dapat dilihat pada Gambar 3.16.



Gambar 3.16 Rancangan Antarmuka Halaman Kontak Layanan Publik

Pada rancangan antarmuka halaman kontak layanan publik yang terlihat pada Gambar 3.16, terdapat bagian-bagian antarmuka sebagai berikut.

### **1. Logo Aplikasi**

Logo aplikasi merupakan logo identitas aplikasi. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak pribadi dengan nomor 1.

### **2. Tombol Tambah Kontak Layanan Publik**

Tombol ini digunakan untuk melakukan penambahan kontak layanan publik oleh admin. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak pribadi dengan nomor 2.

### **3. Baris Kontak Layanan Publik**

Baris kontak layanan publik merupakan bagian yang ditampilkan merupakan data kontak layanan publik yang telah disimpan oleh admin. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman kontak layanan publik dengan nomor 3.

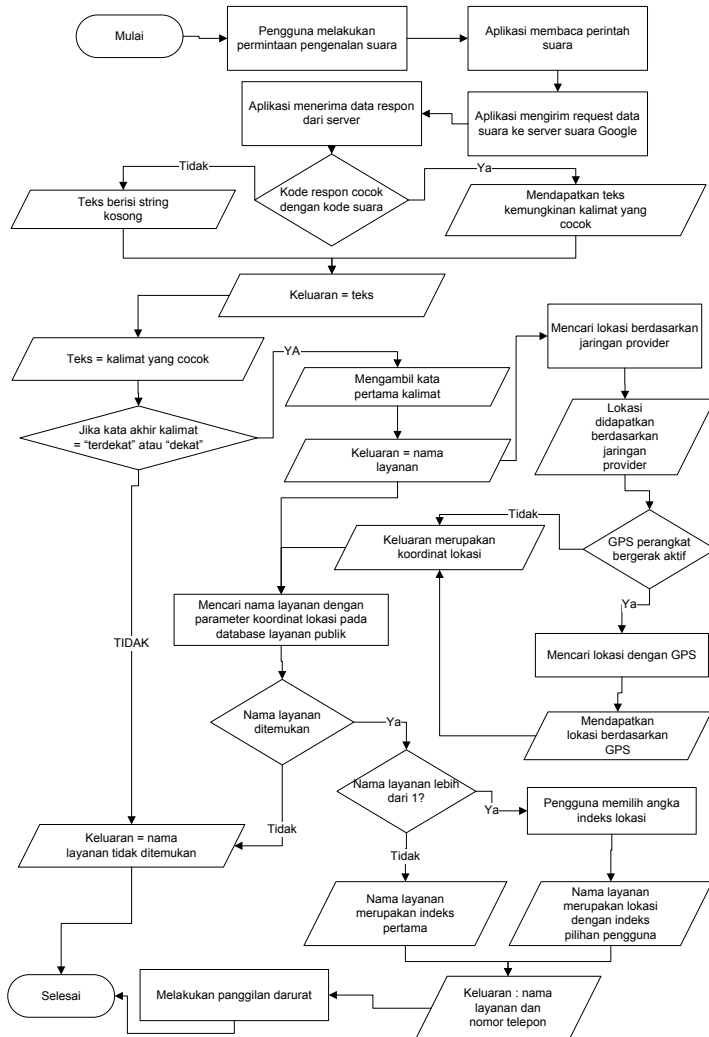
## **3.4.2 Perancangan Proses**

Subbab ini menjelaskan tentang perancangan proses-proses yang ada pada perangkat lunak. Berikut ini adalah rancangan dari proses-proses pada perangkat lunak yang dikembangkan dalam Tugas Akhir ini.

## **3.4.3 Perancangan Proses**

Subbab ini menjelaskan tentang perancangan proses-proses yang ada pada perangkat lunak. Berikut ini adalah rancangan dari proses-proses pada perangkat lunak yang dikembangkan dalam Tugas Akhir ini.

### 3.4.3.1 Proses pada Kasus Penggunaan Panggilan Layanan Publik Terdekat



Gambar 3.17 Diagram Alir Proses Panggilan Layanan Publik

Merujuk pada spesifikasi kasus penggunaan panggilan layanan publik terdekat, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses deteksi suara, proses pengambilan nama layanan, deteksi posisi pengguna, dan pencarian nama layanan publik. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses panggilan layanan publik yang ditunjukkan oleh Gambar 3.17.

#### **3.4.3.1.1 Proses Deteksi Suara**

Proses deteksi suara merupakan proses pengenalan suara yang diterima aplikasi hingga menjadi teks. Proses deteksi suara pada aplikasi ini memanfaatkan masukan suara pada perangkat bergerak berbasis Android dan pengenalan suara Google sebagai pemroses suaranya.

Proses deteksi suara ini dimulai dengan permintaan pengguna kepada aplikasi untuk melakukan pengenalan suara. Aplikasi kemudian melakukan pembacaan masukan suara. Masukan dari proses ini merupakan suara yang ditangkap di sekitar perangkat bergerak saat aplikasi melakukan proses pembacaan suara yang ditandai dengan munculnya tampilan pembacaan suara oleh aplikasi.

Aplikasi ini kemudian mengirim permintaan berupa data suara ke *server* suara Google. Di *server* ini data suara diproses oleh pengenalan suara Google. Kemudian aplikasi menangkap respon dari *server* suara Google yang kemudian diambil teks yang paling cocok jika kode respon sesuai dengan kode permintaan yang dikirim untuk membaca suara.

#### **3.4.3.1.2 Proses Pengambilan Nama Layanan**

Proses pengambilan nama layanan merupakan lanjutan dari proses deteksi suara. Keluaran proses deteksi suara berupa teks diolah dalam proses pengambilan nama layanan ini menjadi sebuah perintah yang menentukan aksi yang akan dilakukan oleh aplikasi. Perintah dalam kasus penggunaan ini adalah perintah

untuk mendapatkan nama layanan tertentu. Tujuan dari proses ini adalah mendapatkan nama layanan.

#### **3.4.3.1.3 Proses Deteksi Posisi Pengguna**

Perangkat lunak memiliki kemampuan untuk mendeteksi posisi pengguna. Proses deteksi posisi pengguna dilakukan dengan menggunakan GPS dan jaringan dari *provider*. Proses deteksi posisi ini dilakukan untuk mendeteksi posisi awal pengguna untuk melakukan perjalanan.

Proses deteksi posisi dilakukan dengan dua tahap. Tahap pertama, aplikasi mendeteksi posisi pengguna dengan pencarian posisi perangkat bergerak berdasarkan jaringan *provider*. Tahap selanjutnya dilakukan pencarian posisi perangkat bergerak berdasarkan GPS jika pada perangkat bergerak mengaktifkan fitur GPS. Jika GPS aktif, maka koordinat lokasi pengguna diambil dari hasil pendeteksian posisi berdasarkan GPS. Jika GPS tidak aktif, aplikasi mendapatkan koordinat lokasi pengguna berdasarkan pendeteksian posisi perangkat bergerak menggunakan jaringan *provider*.

Pada tahap pendeteksian berdasarkan jaringan *provider* sebenarnya lokasi telah diperoleh namun kurang akurat. Tahap kedua merupakan pendeteksian posisi yang lebih akurat dengan menggunakan GPS. Namun pengidentifikasian dengan GPS memiliki kelemahan yaitu sulitnya GPS untuk bekerja di dalam ruangan[11].

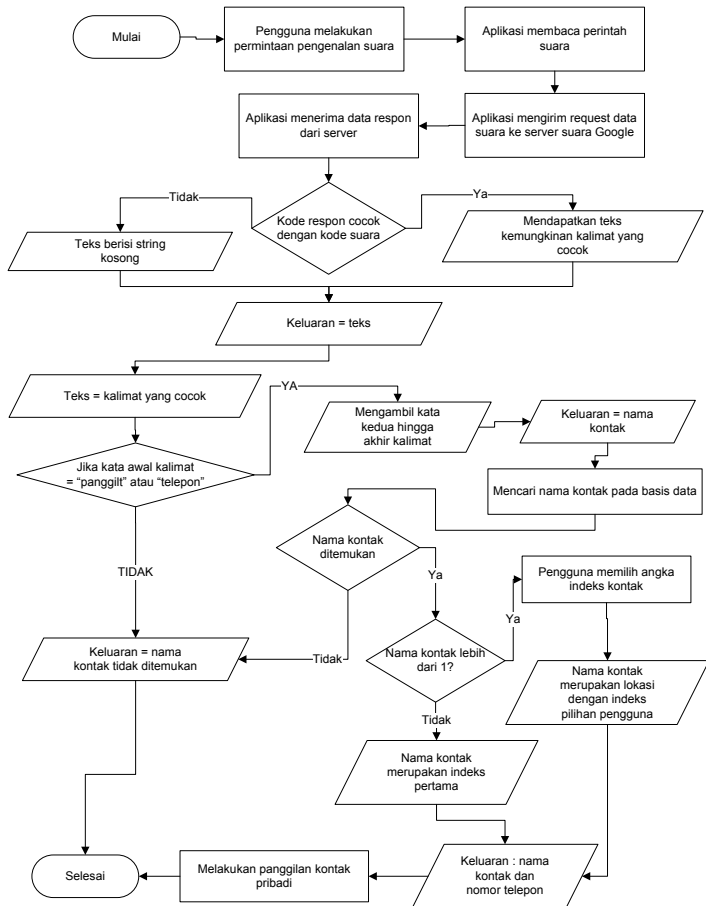
#### **3.4.3.1.4 Proses Pencarian Keberadaan Nama Layanan**

Proses pencarian keberadaan nama layanan dilakukan dengan melakukan pengecekan pada data basis data layanan publik. Proses dimulai dengan menginisiasikan nama layanan dengan nilai keluaran dari proses sebelumnya, yakni proses pengambilan nama layanan serta koordinat lokasi pada proses sebelumnya. Alur dilanjutkan dengan proses pencarian informasi nomor layanan publik pada basis data layanan publik berdasarkan nama layanan yang diinisialisasikan.

Jika hasil pencarian pada basis data layanan publik menghasilkan lebih dari satu nama layanan, maka diambil kontak dengan indeks yang dipilih oleh pengguna. Jika ditemukan tepat satu nama layanan, maka layanan tersebut yang diambil sebagai nama layanan yang dicari oleh pengguna. selanjutnya dilakukan proses pemanggilan darurat pada nama dan nomor layanan yang didapatkan.

#### **3.4.3.2 Proses pada Kasus Penggunaan Panggilan Kontak Pribadi**

Merujuk pada spesifikasi kasus penggunaan panggilan kontak pribadi, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses deteksi suara, proses pengambilan nama kontak dan pencarian nama kontak. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses panggilan kontak pribadi yang ditunjukkan oleh Gambar 3.18.



**Gambar 3.18 Diagram Alir Proses Panggilan Kontak Pribadi**



#### **3.4.3.2.1 Proses Deteksi Suara**

Proses deteksi suara merupakan proses pengenalan suara yang diterima aplikasi hingga menjadi teks. Proses deteksi suara pada aplikasi ini memanfaatkan masukan suara pada perangkat bergerak berbasis Android dan pengenal suara Google sebagai pemroses suaranya.

Proses deteksi suara ini dimulai dengan permintaan pengguna kepada aplikasi untuk melakukan pengenalan suara. Aplikasi kemudian melakukan pembacaan masukan suara. Masukan dari proses ini merupakan suara yang ditangkap di sekitar perangkat bergerak saat aplikasi melakukan proses pembacaan suara yang ditandai dengan munculnya tampilan pembacaan suara oleh aplikasi.

Aplikasi ini kemudian mengirim permintaan berupa data suara ke server suara Google. Di server ini data suara diproses oleh pengenal suara Google. Kemudian aplikasi menangkap respon dari server suara Google yang kemudian diambil teks yang paling cocok jika kode respon sesuai dengan kode permintaan yang dikirim untuk membaca suara.

#### **3.4.3.2.2 Proses Pengambilan Nama Kontak**

Proses pengambilan nama kontak merupakan lanjutan dari proses deteksi suara. Keluaran proses deteksi suara berupa teks diolah dalam proses pengambilan nama kontak ini menjadi sebuah perintah yang menentukan aksi yang akan dilakukan oleh aplikasi. Perintah dalam kasus penggunaan ini adalah perintah untuk mendapatkan nama kontak tertentu. Tujuan dari proses ini adalah mendapatkan nama kontak.

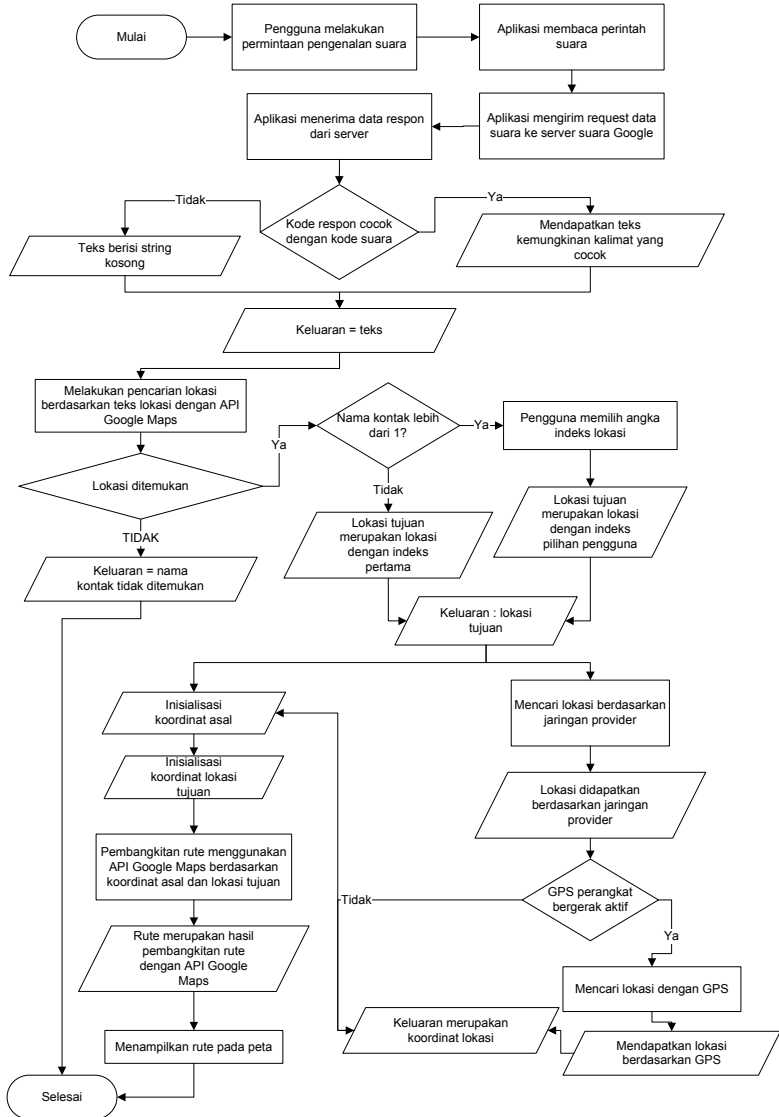
### **3.4.3.2.3 Proses Pencarian Keberadaan Nama Kontak**

Proses pencarian keberadaan nama kontak dilakukan dengan melakukan pengecekan pada data basis data kontak. Proses dimulai dengan menginisiasikan nama kontak dengan nilai keluaran dari proses sebelumnya, yakni proses pengambilan nama layanan. Alur dilanjutkan dengan proses pencarian informasi nomor telepon pada basis data kontak berdasarkan nama kontak yang diinisialisasikan.

Jika hasil pencarian pada basis data kontak menghasilkan lebih dari satu nama kontak, maka diambil kontak dengan indeks yang dipilih oleh pengguna. Jika ditemukan tepat satu nama kontak, maka nama kontak tersebut yang diambil sebagai nama kontak yang dicari oleh pengguna. selanjutnya dilakukan proses pemanggilan darurat pada nama dan nomor kontak yang didapatkan.

### **3.4.3.3 Proses pada Kasus Penggunaan Penunjukan Lokasi**

Merujuk pada spesifikasi kasus penggunaan panggilan kontak pribadi, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses deteksi suara, proses pengambilan nama lokasi dan pencarian nama kontak. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses penunjukan lokasi yang ditunjukkan oleh Gambar 3.19.



Gambar 3.19 Diagram Alir Proses Penunjukan Lokasi

#### **3.4.3.3.1 Proses Deteksi Suara**

Proses deteksi suara merupakan proses pengenalan suara yang diterima aplikasi hingga menjadi teks. Proses deteksi suara pada aplikasi ini memanfaatkan masukan suara pada perangkat bergerak berbasis Android dan pengenalan suara Google sebagai pemroses suaranya.

Proses deteksi suara ini dimulai dengan permintaan pengguna kepada aplikasi untuk melakukan pengenalan suara. Aplikasi kemudian melakukan pembacaan masukan suara. Masukan dari proses ini merupakan suara yang ditangkap di sekitar perangkat bergerak saat aplikasi melakukan proses pembacaan suara yang ditandai dengan munculnya tampilan pembacaan suara oleh aplikasi.

Aplikasi ini kemudian mengirim permintaan berupa data suara ke server suara Google. Di server ini data suara diproses oleh pengenalan suara Google. Kemudian aplikasi menangkap respon dari server suara Google yang kemudian diambil teks yang paling cocok jika kode respon sesuai dengan kode permintaan yang dikirim untuk membaca suara.

#### **3.4.3.3.2 Proses Pencarian Keberadaan Lokasi**

Proses pencarian keberadaan lokasi dilakukan dengan melakukan pengecekan pada data Google Maps. Proses dimulai dengan menginisiasikan nama tempat dengan nilai keluaran dari proses sebelumnya, yakni proses pengambilan suara. Alur dilanjutkan dengan proses pencarian lokasi dengan API Google Maps berdasarkan nama tempat yang diinisialisasikan.

Jika hasil pencarian dengan API Google Maps menghasilkan lebih dari satu tempat, maka diambil tempat dengan indeks yang dipilih oleh pengguna. Jika ditemukan tepat satu lokasi, maka lokasi tersebut yang diambil sebagai lokasi yang dicari oleh pengguna.

#### **3.4.3.3.3 Proses Deteksi Posisi Pengguna**

Perangkat lunak memiliki kemampuan untuk mendeteksi posisi pengguna. Proses deteksi posisi pengguna dilakukan dengan menggunakan GPS dan jaringan dari *provider*. Proses deteksi posisi ini dilakukan untuk mendeteksi posisi pengguna.

Proses deteksi posisi dilakukan dengan dua tahap. Tahap pertama, aplikasi mendeteksi posisi pengguna dengan pencarian posisi perangkat bergerak berdasarkan jaringan *provider*. Tahap selanjutnya dilakukan pencarian posisi perangkat bergerak berdasarkan GPS jika pada perangkat bergerak mengaktifkan fitur GPS. Jika GPS aktif, maka koordinat lokasi pengguna diambil dari hasil pendeteksian posisi berdasarkan GPS. Jika GPS tidak aktif, aplikasi mendapatkan koordinat lokasi pengguna berdasarkan pendeteksian posisi perangkat bergerak menggunakan jaringan *provider*.

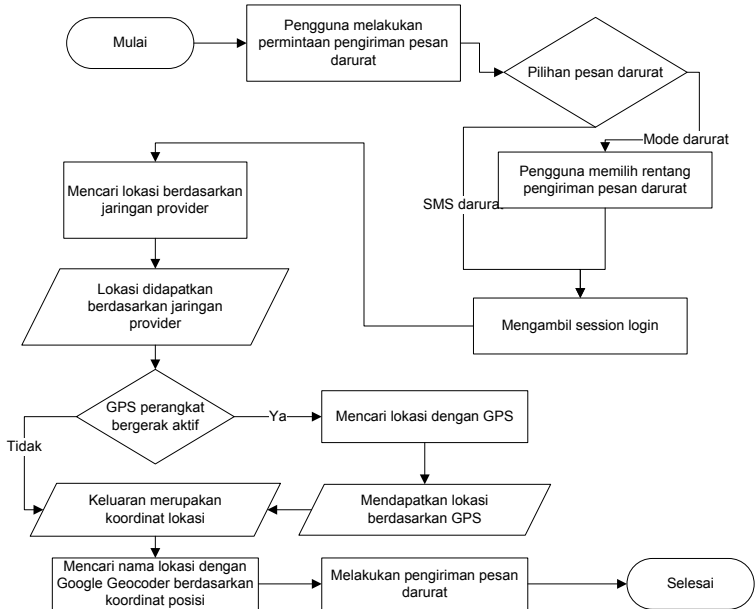
Pada tahap pendeteksian berdasarkan jaringan *provider* sebenarnya lokasi telah diperoleh namun kurang akurat. Tahap kedua merupakan pendeteksian posisi yang lebih akurat dengan menggunakan GPS. Namun pengidentifikasian dengan GPS memiliki kelemahan yaitu sulitnya GPS untuk bekerja di dalam ruangan.

#### **3.4.3.3.4 Proses Pembangkitan Rute Terpendek**

Pembangkitan rute terpendek dilakukan setelah didapatkan koordinat posisi pengguna dan koordinat lokasi tujuan. Proses pembangkitan rute terpendek ini memanfaatkan fitur pembangkitan rute dari Google Maps API berdasarkan dua koordinat yang didapatkan.

Proses pembangkitan rute dimulai dari inisialisasi koordinat asal dan lokasi tujuan. Kedua koordinat tersebut digunakan untuk memanggil permintaan layanan dari fitur pembangkitan rute pada API Google Maps. Pembangkitan rute dilakukan sehingga menghasilkan rute sebagai nilai kembalian dari API Google Maps.

### 3.4.3.4 Proses pada Kasus Penggunaan Pengiriman Pesan Darurat



Gambar 3.20 Diagram Alir Proses Pengiriman Pesan Darurat

Merujuk pada spesifikasi kasus penggunaan pesan darurat dan mode darurat, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses pengambilan nomor dan pesan darurat, proses pengambilan nama lokasi. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses pengiriman pesan darurat dan mode darurat yang ditunjukkan oleh Gambar 3.20.

#### 3.4.3.4.1 Proses Pengambilan Nomor dan Pesan Darurat

Proses pengambilan nomor dan pesan darurat dimulai dengan inputan pengguna dengan memilih sms darurat atau mode darurat selanjutnya proses mengambil id\_pengguna dari *session login*.

kemudian sistem mengambil pada basis data pengguna dengan nomor dan pesan darurat berdasarkan id pengguna.

#### **3.4.3.4.2 Proses Pengambilan Nama Lokasi**

Proses deteksi posisi pengguna dilakukan dengan menggunakan GPS dan jaringan dari *provider*. Proses deteksi posisi dan nama lokasi ini dilakukan untuk mendeteksi posisi pengguna.

Proses deteksi posisi dilakukan dengan dua tahap. Tahap pertama, aplikasi mendeteksi posisi pengguna dengan pencarian posisi perangkat bergerak berdasarkan jaringan *provider*. Tahap selanjutnya dilakukan pencarian posisi perangkat bergerak berdasarkan GPS jika pada perangkat bergerak mengaktifkan fitur GPS. Jika GPS aktif, maka koordinat lokasi pengguna diambil dari hasil pendeteksian posisi berdasarkan GPS. Jika GPS tidak aktif, aplikasi mendapatkan koordinat lokasi pengguna berdasarkan pendeteksian posisi perangkat bergerak menggunakan jaringan *provider*.

Pada tahap pendeteksian berdasarkan jaringan *provider* sebenarnya lokasi telah diperoleh namun kurang akurat. Tahap kedua merupakan pendeteksian posisi yang lebih akurat dengan menggunakan GPS. Namun pengidentifikasian dengan GPS memiliki kelemahan yaitu sulitnya GPS untuk bekerja di dalam ruangan.

Selanjutnya pada tahap pencarian lokasi dengan Geocoder berdasarkan posisi yang didapatkan dari GPS dengan memberikan parameter nama alamat, nama kelurahan, nama kecamatan, nama kota, dan kode pos. Selanjutnya dilakukan pengiriman sms darurat atau pengiriman sms mode darurat kepada nomor darurat.

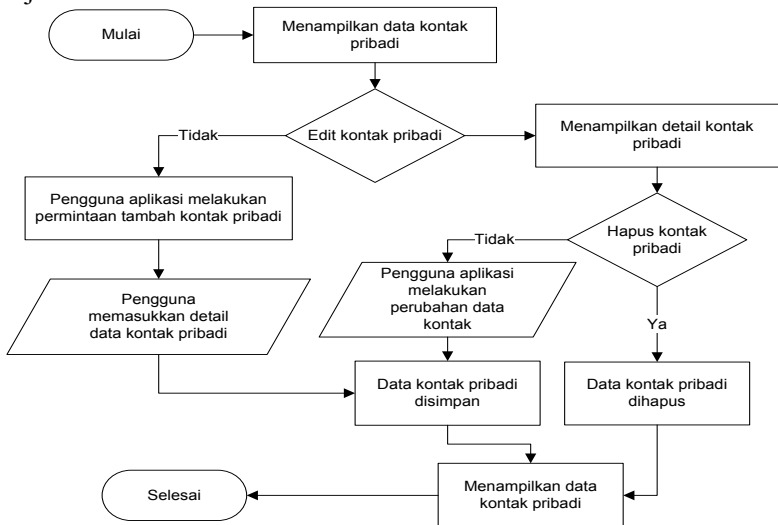
#### **3.4.3.4.3 Proses Pengiriman Pesan Darurat**

Proses pengiriman pesan darurat dilakukan melalui masukan dari pengguna yang selanjutnya di proses sesuai pilihan pengguna. Terdapat 2 jenis pengiriman pesan darurat yaitu dengan pengiriman sms darurat dan mode darurat. Pengiriman

sms darurat merupakan pengiriman untuk sekali pengiriman yang berisi pesan darurat dan nama lokasi pengguna yang membutuhkan pertolongan. Sedangkan mode darurat merupakan pengiriman pesan darurat dengan berkali-kali hingga pengguna memamatkannya. Mode darurat berisi pesan darurat, kalimat yang berarti mode darurat aktif, dan nama lokasi pengguna yang membutuhkan pertolongan.

### 3.4.3.5 Proses pada Kasus Penggunaan Olah Kontak Pribadi

Merujuk pada spesifikasi kasus penggunaan olah kontak pribadi, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses tambah kontak pribadi, edit kontak pribadi, dan hapus kontak pribadi. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses olah kontak pribadi yang ditunjukkan oleh Gambar 3.21.



Gambar 3.21 Diagram Alir Proses Olah Kontak Pribadi



### **3.4.3.5.1 Proses Tambah Kontak Pribadi**

Proses tambah kontak pribadi merupakan proses dimana pengguna aplikasi melakukan penambahan data kontak pribadi meliputi nama kontak pribadi dan nomor telepon kontak pribadi. Proses penambahan kontak pribadi ini dilakukan agar pada proses penggunaan panggilan layanan kontak pribadi dapat diproses oleh pengguna.

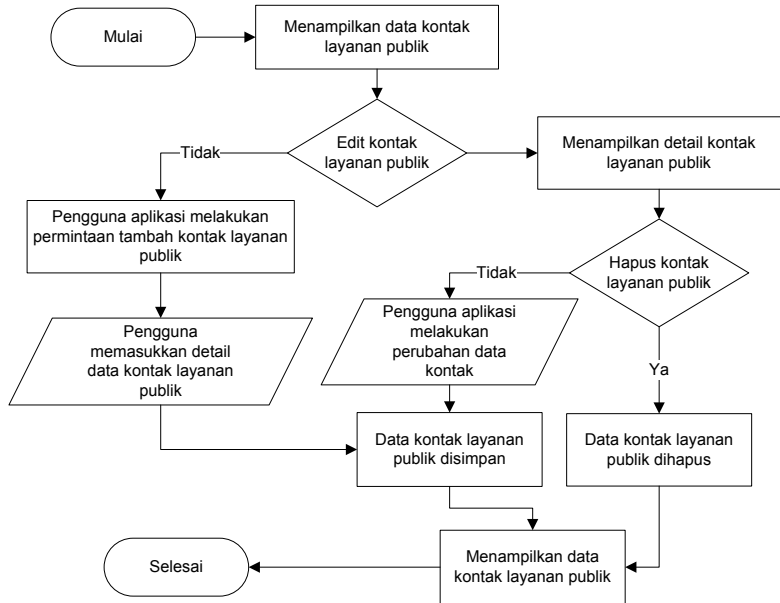
### **3.4.3.5.2 Proses Edit Kontak Pribadi**

Proses edit kontak pribadi merupakan proses dimana pengguna aplikasi melakukan perubahan data kontak pribadi meliputi nama kontak pribadi dan nomor telepon kontak pribadi. Perubahan data kontak pribadi dilakukan apabila pengguna aplikasi merasa terdapat perubahan data nomor telepon pribadi, sehingga pengguna tidak perlu untuk melakukan penambahan data kontak untuk data nama kontak pribadi yang sudah ada pada kontak pribadi pengguna.

### **3.4.3.5.3 Proses Hapus Kontak Pribadi**

Proses hapus kontak pribadi merupakan proses dimana pengguna aplikasi melakukan penghapusan data pribadi untuk kontak pribadi secara menyeluruh meliputi nama kontak pribadi dan nomor kontak pribadi. Penghapusan kontak pribadi dilakukan apabila data kontak pribadi yang telah tersimpan sudah tidak sesuai dengan data yang sebenarnya.

### 3.4.3.6 Proses pada Kasus Penggunaan Olah Kontak Layanan Publik



**Gambar 3.22 Diagram Alir Proses Olah Kontak Layanan Publik**

Merujuk pada spesifikasi kasus penggunaan olah kontak layanan publik, terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses tambah layanan publik, edit layanan publik, dan hapus layanan publik. Proses pada kasus penggunaan ini dijelaskan pada diagram alir proses olah kontak layanan publik yang ditunjukkan oleh Gambar 3.22.

#### 3.4.3.6.1 Proses Tambah Kontak Layanan Publik

Proses tambah kontak layanan publik merupakan proses dimana pengguna admin melakukan penambahan data layanan publik meliputi nama layanan publik, alamat layanan publik, nomor telepon layanan publik, dan data lokasi layanan publik.

Proses penambahan layanan publik ini dilakukan agar pada proses penggunaan panggilan layanan publik terdekat dapat diproses oleh pengguna aplikasi.

#### **3.4.3.6.2 Proses Ubah Kontak Layanan Publik**

Proses edit kontak layanan publik merupakan proses dimana pengguna admin melakukan perubahan data layanan publik meliputi nama layanan publik, alamat layanan publik, nomor telepon layanan publik, dan data lokasi layanan publik. Perubahan data kontak kontak layanan publik dilakukan apabila pengguna admin merasa terdapat perubahan data alamat layanan publik, nomor telepon layanan publik, dan lokasi layanan publik, sehingga pengguna tidak perlu untuk melakukan penambahan data kontak untuk data nama kontak layanan publik yang sudah ada pada kontak layanan publik admin.

#### **3.4.3.6.3 Proses Hapus Kontak Layanan Publik**

Proses hapus kontak layanan publik merupakan proses dimana pengguna admin melakukan penghapusan data layanan publik secara menyeluruh meliputi nama layanan publik, alamat layanan publik, nomor telepon layanan publik, dan data lokasi layanan publik. Penghapusan layanan publik dilakukan apabila data layanan publik yang telah tersimpan sudah tidak sesuai dengan data yang sebenarnya.

## **BAB IV IMPLEMENTASI**

Pada bab ini dijelaskan mengenai implementasi dari perancangan perangkat lunak. Implementasi yang dijelaskan meliputi lingkungan pembangunan perangkat lunak, implementasi antarmuka pengguna, dan implementasi proses-proses yang terjadi pada masing-masing kasus penggunaan pada perangkat lunak.

### **4.1 Lingkungan Pembangunan Perangkat Lunak**

Perangkat keras yang digunakan dalam pembangunan perangkat lunak ini berupa laptop dengan spesifikasi processor Intel Core i5-3337M @ 1,8 GHz (4CPUs), memori 4096 MB dan media penyimpanan 750 GB. Perangkat lunak pada laptop yang digunakan dalam pembangunan perangkat lunak antara lain:

- Sistem Operasi Windows 8 Pro 64-bit
- StarUML 5.0.2.1570
- Android Developer Tools v22.3.0-887826

Selain itu digunakan pula perangkat bergerak Sony Xperia M dengan sistem operasi Android versi 4.3 (*Ice Cream Sandwich*) dan CPU Dual Core 1 GHz.

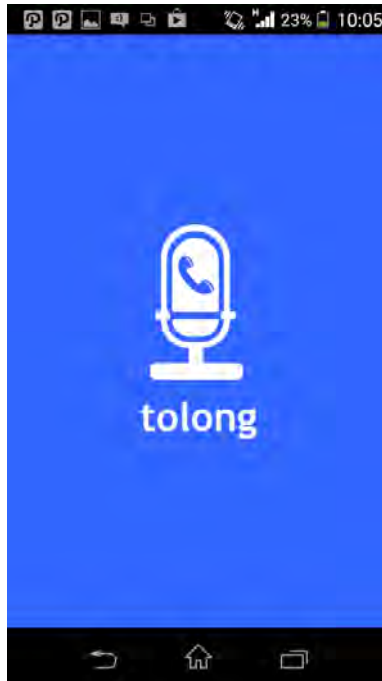
### **4.2 Implementasi Antarmuka Pengguna**

Subbab ini membahas mengenai implementasi antarmuka pengguna yang meliputi implementasi antarmuka pembuka, implementasi antarmuka menu aplikasi, implementasi antarmuka komunikasi, dan implementasi antarmuka peta.

#### **4.2.1 Implementasi Antarmuka Halaman Pembuka**

Antarmuka komunikasi diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam berkas *layout* pada berkas

proyek Android dengan nama *splashscreen.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.1.



**Gambar 4.1 Implementasi Antarmuka Halaman Pembuka**

Antarmuka hanya terdiri dari komponen *ImageView* yang digunakan untuk menampilkan gambar logo, sudah termasuk dengan nama aplikasi. Implementasi antarmuka halaman pembuka dalam bentuk *XML* dapat dilihat pada Gambar A.1 Lampiran A.

#### 4.2.2 Implementasi Antarmuka Halaman Menu Aplikasi

Antarmuka komunikasi diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam berkas *layout* pada berkas proyek Android dengan nama *menu\_aplikasi.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.2.



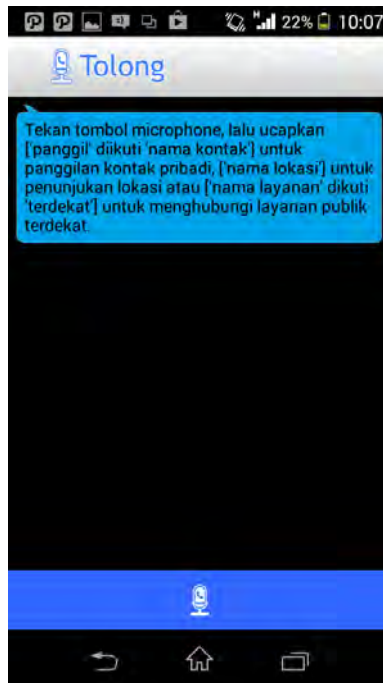
Gambar 4.2 Implementasi Antarmuka Halaman Menu Aplikasi

Tampilan halaman menu aplikasi terdiri dari beberapa komponen *Imagebutton* yang digunakan untuk melakukan

perpindahan halaman serta *ImageView* yang menampilkan logo aplikasi. Implementasi antarmuka halaman menu aplikasi dalam bentuk *XML* dapat dilihat pada Gambar A.2 Lampiran A

#### 4.2.3 Implementasi Antarmuka Halaman Komunikasi

Antarmuka komunikasi diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam berkas layout pada berkas proyek Android dengan nama *activity\_komunikasi.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.3.



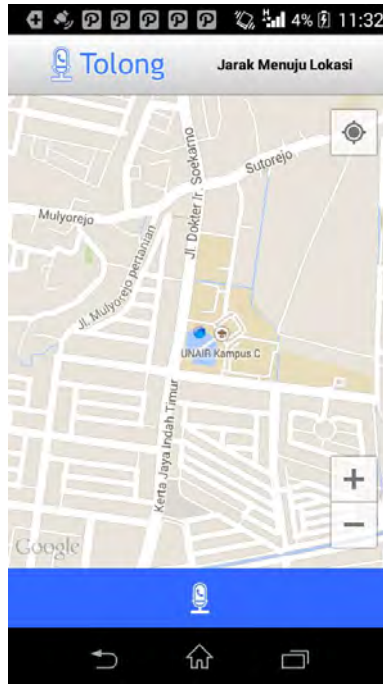
Gambar 4.3 Implementasi Antarmuka Halaman Komunikasi

Tampilan komunikasi antara pengguna dengan respon sistem disusun atas *ListView*. Tampilan *ListView* biasa dimodifikasi dengan tambahan gambar serta penghilangan pemisah antar isi *ListView*. Bagian pemisah antar isi *ListView* diwakili oleh atribut *android:dividerHeight* dan *android:divider*. Untuk menghilangkan pemisah antar *ListView*. Bagian tersebut disusun dengan sintaks *XML* seperti pada Gambar A.3 Lampiran A. Untuk membuat model gelembung obrolan pada antarmuka ini disusun dengan sintaks *XML* seperti pada Gambar A.4 Lampiran A. Sedangkan tombol bicara dan tombol peta diimplementasikan menggunakan *ImageButton*. Implementasi tombol bicara dalam sintaks *XML* diperlihatkan pada Gambar A.5 Lampiran A.

#### **4.2.4 Implementasi Antarmuka Halaman Peta**

Antarmuka halaman peta diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam *folder* layout pada berkas proyek Android dengan nama *activity\_peta.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.4.





**Gambar 4.4 Implementasi Antarmuka Halaman Peta**

Tampilan peta ini memanfaatkan peta dari Google dengan memanfaatkan *fragment* yang merupakan kelas bawaan dari Google Maps v2. Implementasi dari *fragment* dengan sintaks *XML* digambarkan pada Gambar A.6 Lampiran A.

Pada peta juga terdapat lapisan berupa rute, penanda asal, dan penanda tujuan. Penanda asal dan penanda tujuan menggunakan kelas *GetRouteListTask* yang digunakan untuk menampilkan penanda lapisan di peta, sedangkan untuk penanda posisi pengguna digunakan kelas *MyPosition*. Kode sumber kelas *GetRouteListTask* dan kelas *MyPosition* dijelaskan pada Gambar

B.1 Lampiran B dan Gambar B.2 Lampiran B. Sedangkan kode sumber untuk menampilkan penanda asal dan tujuan pada peta dapat dilihat pada Gambar B.3 Lampiran B.

#### 4.2.5 Implementasi Antarmuka Halaman Kontak Pribadi

Antarmuka halaman peta diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam *folder* layout pada berkas proyek Android dengan nama *kontak.xml* dan *list\_item\_kontak.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.5.



Gambar 4.5 Implementasi Antarmuka Halaman Kontak Pribadi

Tampilan kontak pribadi merupakan gabungan antara 2 layout dimana ada *list\_item\_kontak.xml* dan *kontak.xml*. *kontak.xml* merupakan tampilan dasar untuk tampilan kontak pribadi sedangkan *list\_item\_kontak.xml* berisi *ListView* data kontak pribadi. Implementasi dari kontak.xml digambarkan pada Gambar A.7 Lampiran A sedangkan *list\_item\_kontak.xml* digambarkan pada Gambar A.8 Lampiran A.

#### **4.2.6 Implementasi Antarmuka Halaman Layanan Publik**

Antarmuka halaman layanan publik diimplementasikan dalam dokumen berbentuk *XML* yang tersimpan dalam *folder* layout pada berkas proyek Android dengan nama *layananpublik\_admin.xml* dan *list\_layananpublik\_admin.xml*. Tampilan grafis dari antarmuka ini dijelaskan pada Gambar 4.6.

Tampilan kontak layanan publik merupakan gabungan antara 2 layout dimana ada *list\_item\_kontak.xml* dan *kontak.xml*. *layananpublik\_admin.xml* merupakan tampilan dasar untuk tampilan kontak pribadi sedangkan *list\_layananpublik.xml* berisi *ListView* data kontak layanan publik. Implementasi dari kontak.xml digambarkan pada Gambar A.9 Lampiran A sedangkan *list\_item\_kontak.xml* digambarkan pada Gambar A.10 Lampiran A.



**Gambar 4.6 Implementasi Antarmuka Halaman Layanan Publik**

### **4.3 Implementasi Proses**

Bagian ini menjelaskan tentang implementasi proses-proses pada perangkat lunak yang dikembangkan dalam Tugas Akhir ini. Implementasi proses dilakukan berdasarkan pada perancangan proses yang telah dijelaskan pada bab sebelumnya. Penjelasan mengenai implementasi proses pada aplikasi panggilan darurat dengan perintah suara berbahasa Indonesia pada perangkat bergerak berbasis Android dibagi menjadi enam bagian umum, yakni perintah panggilan kontak pribadi, perintah penunjukan

lokasi, perintah panggilan layanan publik, pengiriman pesan darurat, olah kontak pribadi, dan olah kontak layanan publik.

#### **4.3.1 Implementasi Proses pada Kasus Penggunaan Panggilan Layanan Publik Terdekat**

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus penggunaan perintah panggilan layanan publik terdekat. Proses-proses tersebut meliputi proses deteksi suara, proses pengambilan nama layanan, deteksi posisi pengguna, dan pencarian nama layanan publik.

##### **4.3.1.1 Implementasi Proses Deteksi Suara**

Proses deteksi suara ditangani oleh objek *RecognizerIntent*. *RecognizerIntent* kemudian membaca data suara pengguna lalu mengirimkannya ke *server* suara Google. Di sinilah data suara diproses oleh pengenalan suara Google. Setiap kali *Activity* menerima respon dari luar, *Activity* mencocokkan dengan kode-kode permintaan yang dikirimkannya ke *server*. Jika didapatkan kode respon yang cocok dengan kode permintaan, maka teks diambil dari data yang paling cocok.

Proses dimulai dengan inisiasi objek *Intent* dari *RecognizerIntent* disusul dengan penerimaan suara oleh *Intent* ini. Kode sumber untuk implementasi inisiasi objek *Intent* dari *RecognizerIntent* dan penerimaan suara dijelaskan pada Gambar B.4 Lampiran B.

Ketika sudah didapatkan hasil respon dari *server* suara Google, maka *Intent* ini akan keluar dan memberikan nilai hasil yang dapat diperoleh dari fungsi *onActivityResult()* pada kelas *Activity* yang memanggil *Intent*. Fungsi ini mendapatkan semua hasil yang dikirimkan dari pemanggilan *Intent* yang dijalankannya saat *Intent* tersebut sudah menyelesaikan tugasnya. Setelah didapatkan hasil dari *Intent*, dilanjutkan dengan proses pengecekan kode respon. Jika kode sesuai dengan kode

permintaan pembacaan suara, maka dilanjutkan dengan pembacaan teks dari data respon yang dikirim.

Implementasi dalam penerimaan hasil respon *Intent* terlihat pada kode sumber pada Gambar B.5 Lampiran B, dan implementasi untuk pembacaan teks yang paling cocok dengan masukan suara dijelaskan oleh kode sumber pada Gambar B.6 Lampiran B.

#### **4.3.1.2 Implementasi Proses Pengambilan Nama Layanan**

Pengenalan perintah untuk kasus penggunaan panggilan layanan publik dilakukan dengan pencarian kata terakhir seperti yang telah dijelaskan pada tahap perencanaan proses pengambilan nama layanan. Proses dari pengambilan ini menghasilkan sebuah *String* yang menunjukkan nama layanan publik. Implementasi proses ini dijabarkan dalam kode sumber dengan Gambar B.7 Lampiran B.

#### **4.3.1.3 Implementasi Proses Deteksi Posisi Pengguna**

Implementasi dari proses identifikasi lokasi melibatkan kelas *MyPosition*. Kelas *MyPosition* dipanggil untuk mendapatkan posisi pengguna saat melakukan permintaan yang koordinat posisinya diambil guna proses pencarian lokasi layanan terdekat. Implementasi dari proses ini ada pada Gambar B.8 Lampiran B.

#### **4.3.1.4 Implementasi Proses Keberadaan Nama Layanan**

Proses pencarian keberadaan nama layanan diimplementasikan dengan pemanggilan pada basis data layanan publik dengan menyertakan nama layanan yang dicari beserta posisi pengguna. Implementasi dari proses pembangkitan rute terpendek diperlihatkan pada Gambar B.9 Lampiran B. Proses pencarian nama layanan terdekat diimplementasikan *query service* dengan rumus *Spherical Law of Cosines* untuk mendapatkan posisi layanan terdekat. Implementasi dijelaskan pada pemaparan kode sumber pada Gambar B.10 Lampiran B. Setelah mendapatkan

nama layanan terdekat, pengguna diberikan pilihan apabila hasil yang didapat lebih dari 1 dan langsung melakukan panggilan apabila hanya 1 hasil. Implementasi proses pemilihan layanan oleh pengguna diimplementasikan pada Gambar B11 Lampiran B.

### 4.3.2 Implementasi Proses pada Kasus Penggunaan Panggilan Kontak Pribadi

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus penggunaan panggilan kontak pribadi. Proses-proses tersebut meliputi proses deteksi suara, proses pengambilan nama kontak dan pencarian nama kontak.

#### 4.3.2.1 Implementasi Proses Deteksi Suara

Proses deteksi suara ditangani oleh objek *RecognizerIntent*. *RecognizerIntent* kemudian membaca data suara pengguna lalu mengirimkannya ke *server* suara Google. Di sinilah data suara diproses oleh pengenalan suara Google. Setiap kali *Activity* menerima respon dari luar, *Activity* mencocokkan dengan kode-kode permintaan yang dikirimkannya ke *server*. Jika didapatkan kode respon yang cocok dengan kode permintaan, maka teks diambil dari data yang paling cocok.

Proses dimulai dengan inisiasi objek *Intent* dari *RecognizerIntent* disusul dengan penerimaan suara oleh *Intent* ini. Kode sumber untuk implementasi inisiasi objek *Intent* dari *RecognizerIntent* dan penerimaan suara dijelaskan pada Gambar B.4 Lampiran B.

Ketika sudah didapatkan hasil respon dari *server* suara Google, maka *Intent* ini akan keluar dan memberikan nilai hasil yang dapat diperoleh dari fungsi *onActivityResult()* pada kelas *Activity* yang memanggil *Intent*. Fungsi ini mendapatkan semua hasil yang dikirimkan dari pemanggilan *Intent* yang dijalankannya saat *Intent* tersebut sudah menyelesaikan tugasnya. Setelah didapatkan hasil dari *Intent*, dilanjutkan dengan proses

pengecekan kode respon. Jika kode sesuai dengan kode permintaan pembacaan suara, maka dilanjutkan dengan pembacaan teks dari data respon yang dikirim.

Implementasi dalam penerimaan hasil respon *Intent* terlihat pada kode sumber pada Gambar B.5 Lampiran B, dan implementasi untuk pembacaan teks yang paling cocok dengan masukan suara dijelaskan oleh kode sumber pada Gambar B.6 Lampiran B.

#### **4.3.2.2 Implementasi Proses Pengambilan Nama Kontak**

Proses pengambilan nama kontak dilakukan dengan pencarian kata depan panggil atau telepon seperti yang telah dijelaskan pada tahap perencanaan proses pengambilan nama kontak. Proses dari pengambilan kata tujuan ini menghasilkan sebuah *String* yang menunjukkan nama kontak. Implementasi proses ini dijabarkan dalam kode sumber dengan Gambar B.12 Lampiran B.

#### **4.3.2.3 Implementasi Proses Pencarian Keberadaan Nama Kontak**

Proses pencarian keberadaan nama kontak diimplementasikan dengan pemanggilan pada basis data kontak dengan menyertakan nama kontak yang dicari. Implementasi dari proses pencarian keberadaan nama kontak diperlihatkan pada Gambar B.13 Lampiran B. Selanjutnya proses pencarian nama kontak dengan query untuk mendapatkan nama kontak yang sesuai dengan permintaan pengguna. Implementasi dijelaskan pada pemaparan kode sumber pada Gambar B.14 Lampiran B.

Setelah mendapatkan nama layanan terdekat, pengguna diberikan pilihan apabila hasil yang didapat lebih dari 1 dan langsung melakukan panggilan apabila hanya 1 hasil. Implementasi proses pemilihan layanan oleh pengguna diimplementasikan pada Gambar B.11 Lampiran B



### 4.3.3 Implementasi Proses pada Kasus Penunjukan Lokasi

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus Penunjukan Lokasi. Proses-proses tersebut meliputi proses deteksi suara, proses pencarian keberadaan lokasi, proses deteksi posisi pengguna dan proses pembangkitan rute terpendek.

#### 4.3.3.1 Implementasi Proses Deteksi Suara

Proses deteksi suara ditangani oleh objek *RecognizerIntent*. *RecognizerIntent* kemudian membaca data suara pengguna lalu mengirimkannya ke *server* suara Google. Di sinilah data suara diproses oleh pengenalan suara Google. Setiap kali *Activity* menerima respon dari luar, *Activity* mencocokkan dengan kode-kode permintaan yang dikirimkannya ke *server*. Jika didapatkan kode respon yang cocok dengan kode permintaan, maka teks diambil dari data yang paling cocok.

Proses dimulai dengan inisiasi objek *Intent* dari *RecognizerIntent* disusul dengan penerimaan suara oleh *Intent* ini. Kode sumber untuk implementasi inisiasi objek *Intent* dari *RecognizerIntent* dan penerimaan suara dijelaskan pada Gambar B.4 Lampiran B.

Ketika sudah didapatkan hasil respon dari *server* suara Google, maka *Intent* ini akan keluar dan memberikan nilai hasil yang dapat diperoleh dari fungsi *onActivityResult()* pada kelas *Activity* yang memanggil *Intent*. Fungsi ini mendapatkan semua hasil yang dikirimkan dari pemanggilan *Intent* yang dijalankannya saat *Intent* tersebut sudah menyelesaikan tugasnya. Setelah didapatkan hasil dari *Intent*, dilanjutkan dengan proses pengecekan kode respon. Jika kode sesuai dengan kode permintaan pembacaan suara, maka dilanjutkan dengan pembacaan teks dari data respon yang dikirim.

Implementasi dalam penerimaan hasil respon *Intent* terlihat pada kode sumber pada Gambar B.5 Lampiran B, dan implementasi untuk pembacaan teks yang paling cocok dengan

masukannya dijelaskan oleh kode sumber pada Gambar B.6 Lampiran B.

#### **4.3.3.2 Implementasi Proses Pencarian Keberadaan Lokasi**

Implementasi proses pencarian keberadaan lokasi dilakukan dengan melakukan pengecekan pada data Google Maps. Proses dimulai dengan menginisiasikan nama tempat dengan nilai keluaran dari proses sebelumnya, yakni proses pengambilan suara. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.15 Lampiran B.

#### **4.3.3.3 Implementasi Proses Deteksi Posisi Pengguna**

Implementasi dari proses identifikasi lokasi melibatkan kelas MyPosition. Kelas MyPosition dipanggil untuk mendapatkan posisi pengguna saat melakukan permintaan yang koordinat posisinya diambil guna proses pencarian lokasi. Implementasi dari proses ini ada pada Gambar B.8 Lampiran B.

#### **4.3.3.4 Implementasi Proses Pembangkitan Rute Terpendek**

Implementasi dari proses pembangkitan rute terpendek diimplementasikan dalam kelas DirectionJSONParser. Kelas DirectionJSONParser dipanggil untuk membaca objek JSON. Implementasi dari proses ini ada pada Gambar B.16 Lampiran B. Proses mengartikan objek JSON yang telah didapat dari API Google Maps v2 untuk melakukan permintaan rute dijelaskan dalam kode sumber Gambar B.17 Lampiran B.

#### **4.3.4 Implementasi Proses pada Kasus Pesan Darurat**

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus pesan darurat. Proses-proses tersebut meliputi proses pengambilan nomor dan pesan darurat pada session login, proses pengambilan nama lokasi, dan proses pengiriman pesan darurat.

#### **4.3.4.1 Implementasi Proses Pengambilan Nomor dan Pesan Darurat**

Implementasi proses pengambilan nomor dan pesan darurat dilakukan dengan melakukan mengambil data pengguna dari session login yang kemudian ditampilkan pada menu aplikasi. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.18 Lampiran B.

#### **4.3.4.2 Implementasi Proses Pengambilan Nama Lokasi**

Implementasi proses pengambilan nama lokasi dilakukan dengan melakukan mengambil data GPS dari jaringan *network* dan GPS. Pengambilan posisi kemudian diproses menggunakan *Geocoder* untuk mendapatkan nama lokasi detail pengguna. Hasil dari nama lokasi digunakan sebagai data lokasi pengguna pada pengiriman pesan darurat dan mode darurat. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.19 Lampiran B.

#### **4.3.4.3 Implementasi Proses Pengiriman Pesan Darurat**

Implementasi proses pengiriman pesan dilakukan dengan dua jenis pengiriman pesan. Yang pertama dengan pengiriman pesan darurat dan yang kedua dengan mode darurat yang dapat mengirimkan pesan dengan berkali-kali hingga pengguna mematikan tombol mode darurat. Pada pesan darurat berisi pesan darurat yang sudah disimpan pengguna beserta lokasi pengguna sedangkan isi mode darurat berisi pesan darurat yang sudah disimpan pengguna, kalimat yang menyatakan mode darurat, dan nama lokasi. Kode sumber untuk proses pengiriman pesan darurat terdapat pada Gambar B.20 Lampiran B. Sedangkan Kode sumber untuk proses pengiriman pesan darurat terdapat pada Gambar B.21 Lampiran B.

### **4.3.5 Implementasi Proses pada Kasus Olah Kontak Pribadi**

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus olah kontak pribadi. Proses-proses tersebut meliputi proses tambah kontak pribadi, ubah kontak pribadi, dan hapus kontak pribadi.

#### **4.3.5.1 Implementasi Proses Tambah Kontak Pribadi**

Implementasi proses tambah kontak pribadi dilakukan dengan memasukkan data kontak pribadi yang berisi nama kontak pribadi dan nomor kontak pribadi. Kode sumber untuk proses tambah kontak pribadi terdapat pada Gambar B.22 Lampiran B.

#### **4.3.5.2 Implementasi Proses Ubah Kontak Pribadi**

Implementasi proses ubah kontak pribadi dilakukan dengan perubahan pada data kontak yang ada di basis data untuk dilakukan perubahan nomor kontak pribadi. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.23 Lampiran B.

#### **4.3.5.3 Implementasi Proses Hapus Kontak Pribadi**

Implementasi proses hapus kontak pribadi dilakukan dengan penghapusan data kontak pribadi meliputi nama kontak pribadi dan nomor telepon kontak pribadi. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.24 Lampiran B.

### **4.3.6 Implementasi Proses pada Kasus Olah Kontak Layanan Publik**

Bagian ini menjelaskan tentang implementasi proses-proses yang terdapat dalam kasus olah layanan publik. Proses-proses tersebut meliputi proses tambah kontak layanan publik, ubah kontak layanan publik, dan hapus kontak layanan publik.

#### **4.3.6.1 Implementasi Proses Tambah Kontak Layanan Publik**

Implementasi proses tambah kontak layanan publik dilakukan dengan memasukkan data kontak layanan publik yang berisi nama kontak layanan publik, alamat kontak layanan publik, nomor telepon layanan publik, dan data lokasi layanan publik. Kode sumber untuk proses tambah kontak layanan publik terdapat pada Gambar B.25 Lampiran B.

#### **4.3.6.2 Implementasi Proses Ubah Kontak Layanan Publik**

Implementasi proses ubah kontak layanan publik dilakukan dengan perubahan pada data kontak yang ada di basis data untuk dilakukan perubahan alamat kontak layanan publik, nomor kontak layanan publik, dan data lokasi layanan publik. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.26 Lampiran B.

#### **4.3.6.3 Implementasi Proses Hapus Kontak Layanan Publik**

Implementasi proses hapus kontak layanan publik dilakukan dengan penghapusan data kontak meliputi nama kontak layanan publik, alamat kontak layanan publik, nomor kontak layanan publik, dan data lokasi kontak layanan publik. Kode sumber untuk proses pencarian keberadaan lokasi terdapat pada Gambar B.27 Lampiran B.

## **BAB V**

### **PENGUJIAN DAN EVALUASI**

Bab ini membahas tentang pengujian dan evaluasi dari perangkat lunak pada Tugas Akhir ini. Pengujian dilakukan pada fungsionalitas perangkat lunak dan pengujian kemudahan perangkat lunak oleh beberapa pengguna. Pembahasan tentang pengujian dan evaluasi meliputi lingkungan pelaksanaan pengujian, dasar pengujian, pengujian fungsionalitas, pengujian kegunaan, dan evaluasi pengujian.

#### **5.1 Lingkungan Pelaksanaan Pengujian**

Lingkungan pengujian merupakan perangkat-perangkat yang dilibatkan dalam proses pengujian. Lingkungan pengujian ini menggunakan perangkat keras berupa perangkat bergerak berbasis Android yang terhubung dengan internet dan memiliki fasilitas GPS. Spesifikasi lingkungan pengujian dijelaskan pada Tabel 5.1.

**Tabel 5.1 Spesifikasi Lingkungan Pengujian Perangkat Lunak**

Perangkat Bergerak Android	
Processor	CPU Dual-core 1.2 GHz
RAM	1GB
Sistem Operasi	Android Jelly Bean 4.3

#### **5.2 Dasar Pengujian**

Pengujian yang dilakukan berupa pengujian fungsionalitas dan pengujian kegunaan perangkat lunak secara langsung oleh pengguna. Pengujian fungsionalitas dilakukan dengan model *black box* untuk masing-masing fungsionalitas dari aplikasi ini.

Pengujian ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap kebutuhan benar-benar diimplementasikan dan bekerja seperti yang diharapkan.

Pengujian kemudahan perangkat lunak dilakukan dengan pengujian langsung kepada pengguna perangkat lunak. Hasil pengujian diambil dari kuisioner yang telah disebarkan kepada pengguna perangkat lunak, sebagai titik acuan dari keberhasilan kegunaan perangkat lunak.

### **5.3 Pengujian Fungsionalitas**

Subbab ini menjelaskan tentang skenario pengujian fungsionalitas perangkat lunak ada Tugas Akhir ini. Pengujian yang dilakukan meliputi pengujian pengenalan suara pengguna, pengujian pendeteksian tujuan, pengujian pendeteksian posisi pengguna, pengujian pembangkitan rute menuju lokasi, dan pengujian penggantian rute.

#### **5.3.1 Pengujian Pengenalan Suara Pengguna**

Bagian ini menjelaskan tentang pengujian terhadap fungsionalitas pengenalan suara pengguna. Skenario pengujian fungsionalitas ini terdapat pada Tabel 5.2. Pengujian dilakukan dengan beberapa macam masukan suara yang berbeda-beda. Pengujian pertama dilakukan dengan melakukan panggilan nama jenis layanan publik terdekat. Pengujian kedua dengan melakukan panggilan nama kontak pribadi. Pengujian ketiga dengan memasukkan nama tempat yang unik, dimana tidak ada kemungkinan tidak terjadi pencarian dengan hasil lebih dari satu. Pengujian keempat dilakukan dengan pencarian nama tempat yang umum yang memungkinkan dapat dihasilkan lebih dari satu lokasi yang cocok.

**Tabel 5.2 Skenario Pengujian Fungsionalitas Pengenalan Suara Pengguna**

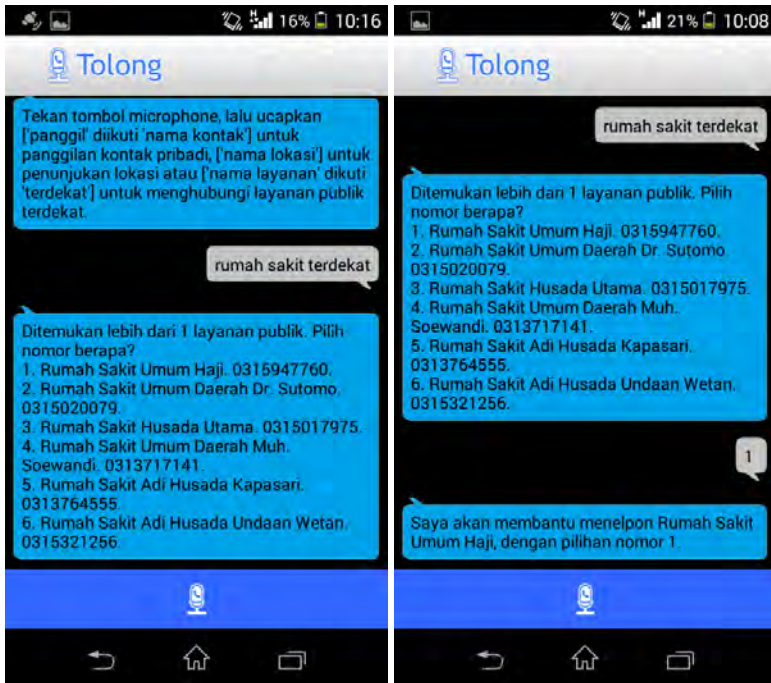
Kode Pengujian		TC-FR-01		
Tujuan Pengujian		Melakukan pengujian untuk mengenali suara pengguna		
Kondisi Awal		Pengguna berada Jalan Dr. Ir. Soekarno Surabaya dan pengguna mengakses halaman komunikasi		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
<b>Panggilan nama layanan publik terdekat</b>				
Masukan suara : "rumah sakit terdekat"  Nomor pilihan : "1"	1. Menekan tombol suara 2. Mengucapkan masukan suara 3. Mengucapkan nomor pilihan setelah ditampilkan pilihan nama layanan publik	Muncul teks "rumah sakit terdekat" dan respon teks yang isinya "Ditemukan lebih dari 1 layanan publik, Pilih nomor berapa?" serta tulisan nomor konfirmasi dan respon "Saya akan memanggil Rumah Sakit Umum Haji dengan pilihan nomor 1 "	Muncul teks "rumah sakit terdekat" dan respon teks yang isinya "Ditemukan lebih dari 1 layanan publik, Pilih nomor berapa?" serta tulisan nomor konfirmasi dan respon "Saya akan memanggil Rumah Sakit Umum Haji dengan pilihan nomor 1 "	Proses pengenalan suara berhasil
<b>Panggilan nama kontak pribadi</b>				



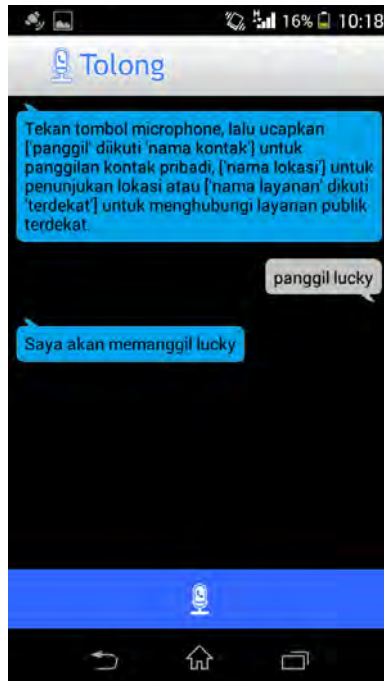
Masukan suara : “panggil lucky”	1. Menekan tombol suara 2. Mengucapkan masukan suara	Muncul teks “panggil lucky” dan “Saya akan memanggil lucky”	Muncul teks “panggil lucky” dan “Saya akan memanggil lucky”	Proses pengenalan suara berhasil
Nama lokasi yang diucapkan bersifat unik				
Masukan suara : “kejawan putih tambak surabaya”	1. Menekan tombol suara 2. Mengucapkan masukan suara	Muncul teks “kejawan putih tambak surabaya” dan “Saya akan membantu Anda ke Jalan Kejawan Putih Tambak”	Muncul teks “kejawan putih tambak surabaya” dan “Saya akan membantu Anda ke Jalan Kejawan Putih Tambak”	Proses pengenalan suara berhasil
Nama lokasi yang diucapkan bersifat umum				
Masukan suara : “manyar surabaya”  Nomor pilihan : “2”	1. Menekan tombol suara 2. Mengucapkan masukan suara 3. Mengucapkan nomor pilihan setelah ditampilkan pilihan lokasi	Muncul teks “manyar surabaya” dan respon teks yang isinya “Ditemukan lebih dari 1 lokasi, Pilih nomor berapa?” serta tulisan nomor konfirmasi	Muncul teks “manyar surabaya” dan respon teks yang isinya “Ditemukan lebih dari 1 lokasi, Pilih nomor berapa?” serta tulisan nomor	Proses pengenalan suara berhasil

		dan respon “Saya akan membantu Anda ke Manyar Sabrangan dengan pilihan nomor 2 ”	konfirmasi dan respon “Saya akan membantu Anda ke Manyar Sabrangan dengan pilihan nomor 2 ”	
Kondisi Akhir	Pengguna berada pada halaman komunikasi			

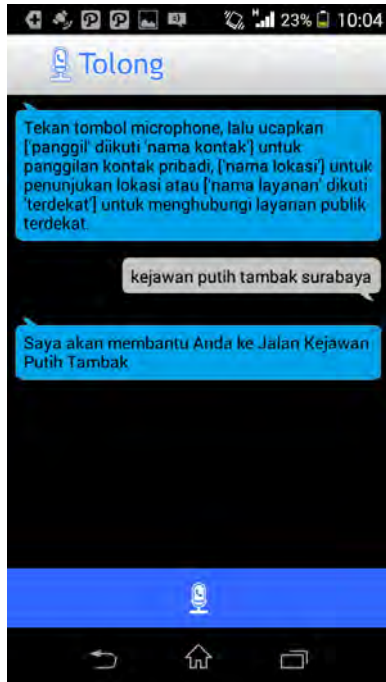
Gambar 5.1 menunjukkan pengujian pengenalan suara perintah panggilan dengan ketentuan nama layanan publik terdekat. Gambar 5.2 menunjukkan pengujian pengenalan suara perintah panggilan dengan ketentuan nama kontak pribadi. Gambar 5.3 menunjukkan potongan gambar dari pengujian pengenalan suara dengan ketentuan nama tempat yang diucapkan bersifat unik. Gambar 5.4 menunjukkan pengujian pengenalan suara dengan ketentuan nama tempat yang diucapkan bersifat umum.



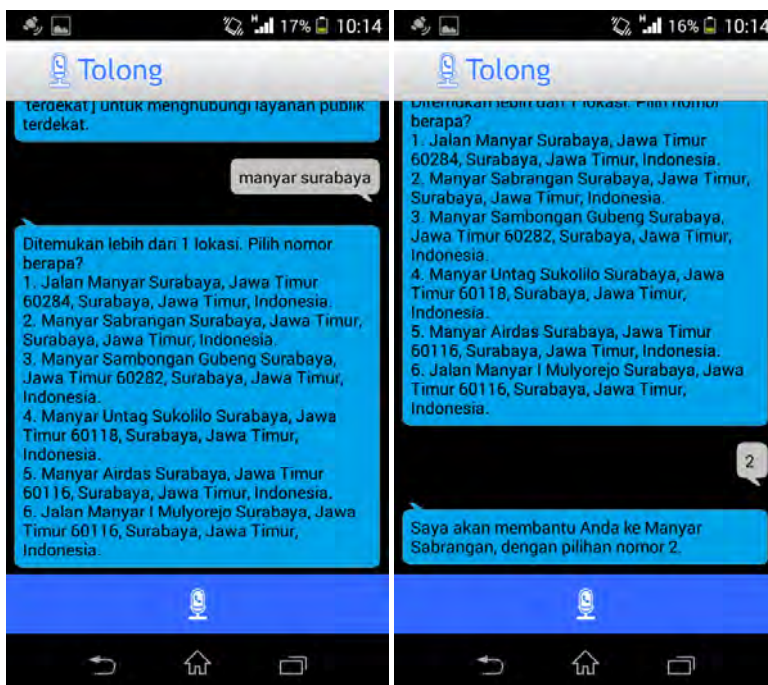
Gambar 5.1 Pengujian Pengenalan Suara untuk Panggilan Layanan Publik Terdekat



**Gambar 5.2** Pengujian Pengenalan Suara untuk Panggilan Kontak Pribadi



**Gambar 5.3** Pengujian Pengenalan Suara untuk Nama Lokasi yang Unik



Gambar 5.4 Pengujian Pengenalan Suara untuk Nama Lokasi yang Umum

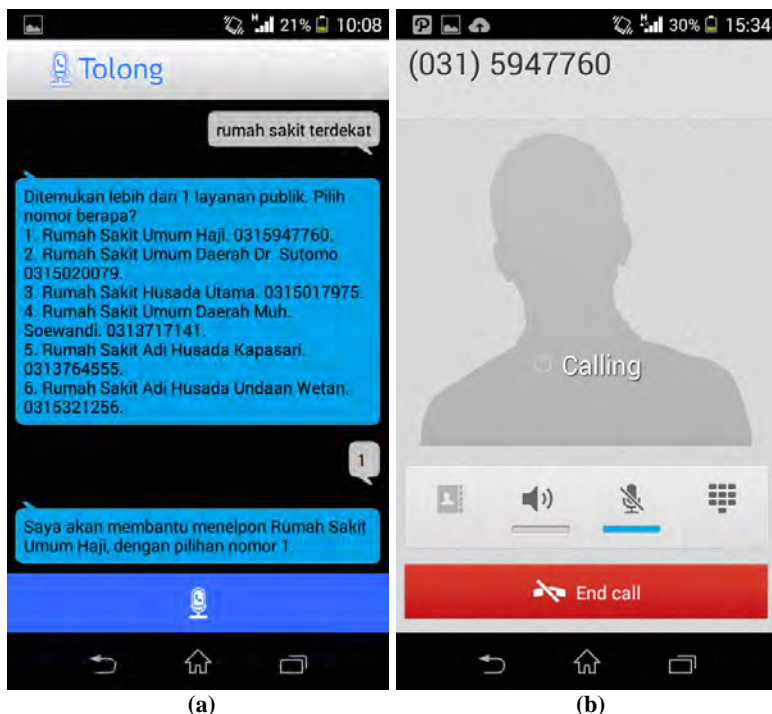
### 5.3.2 Pengujian Panggilan Layanan Publik Terdekat

Pengujian panggilan layanan publik terdekat dilakukan untuk menguji fungsionalitas panggilan dari kalimat yang diucapkan pengguna serta pencarian data layanan publik pada basis data layanan publik berdasarkan posisi terdekat pengguna. Skenario dari pengujian ini terdapat pada Tabel 5.3 berikut.

**Tabel 5.3 Skenario Pengujian Fungsionalitas Panggilan Layanan Publik Terdekat**

Kode Pengujian		TC-FR-02		
Tujuan Pengujian		Melakukan pengujian untuk melakukan panggilan layanan publik terdekat		
Kondisi Awal		Pengguna berada di Jalan Dr. Ir. Soekarno Surabaya dan mengakses halaman komunikasi		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Masukan suara : “rumah sakit terdekat”  Nomor pilihan : “1”	1. Menekan tombol suara 2. Mengucapkan masukan suara 3. Mengucapkan nomor pilihan setelah ditampilkan pilihan nama layanan publik	Panggilan rumah sakit dengan pilihan nomor 1 dan detail pilihan merupakan rumah sakit dengan jarak terdekat	Panggilan rumah sakit dengan pilihan nomor 1 Rumah Sakit Umum Haji dan nomor telepon 0315947760	Proses panggilan layanan publik terdekat berhasil
Kondisi Akhir	Pengguna melakukan panggilan langsung			

Gambar 5.5 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas panggilan layanan publik berdasarkan Tabel 5.3.



(a) (b)  
**Gambar 5.5 Proses Pengujian Panggilan Layanan Publik Terdekat.**  
 (a) Pengenalan Perintah Pengguna, (b) Tampilan Panggilan Langsung

### 5.3.3 Pengujian Panggilan Kontak Pribadi

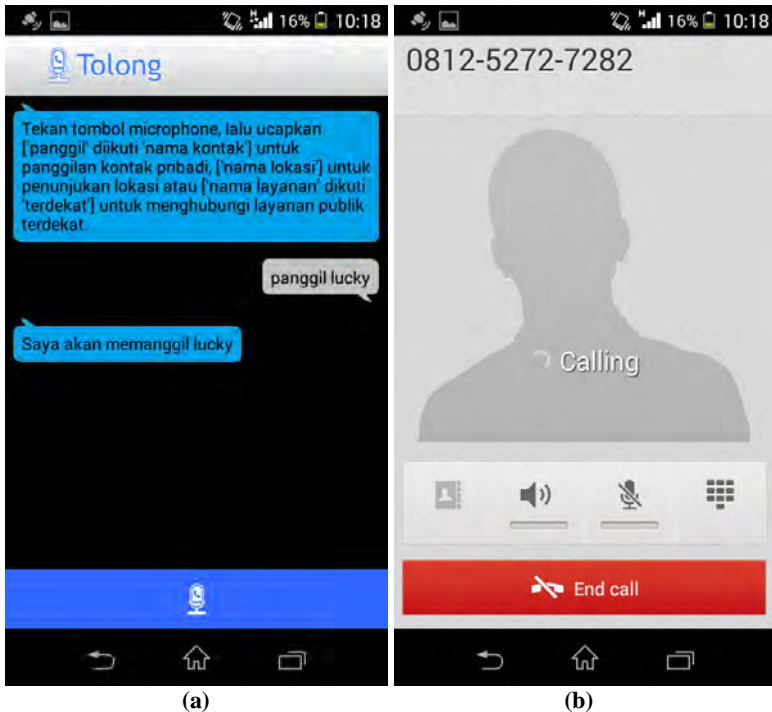
Pengujian panggilan kontak pribadi dilakukan untuk menguji fungsionalitas panggilan dari kalimat yang diucapkan pengguna serta pencarian data kontak pada basis data kontak. Skenario dari pengujian ini terdapat pada Tabel 5.4 berikut.



**Tabel 5.4 Skenario Pengujian Fungsionalitas Panggilan Kontak Pribadi**

Kode Pengujian		TC-FR-03		
Tujuan Pengujian		Melakukan pengujian untuk melakukan panggilan kontak pribadi		
Kondisi Awal		Pengguna mengakses halaman komunikasi		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Masukan suara : "panggil suci"	1. Menekan tombol suara 2. Mengucapkan masukan suara	Panggilan kontak pengguna dengan nama kontak suci.	Panggilan kontak dengan nama suci dan nomor telepon 0313729591	Proses panggilan kontak pribadi berhasil
Kondisi Akhir	Pengguna berada pada halaman komunikasi			

Gambar 5.6 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas pendeteksian tujuan berdasarkan Tabel 5.4.



(a) (b)  
**Gambar 5.6** Proses Pengujian Panggilan Kontak Pribadi.  
 (a) Pengenalan Perintah Pengguna, (b) Tampilan Panggilan Langsung

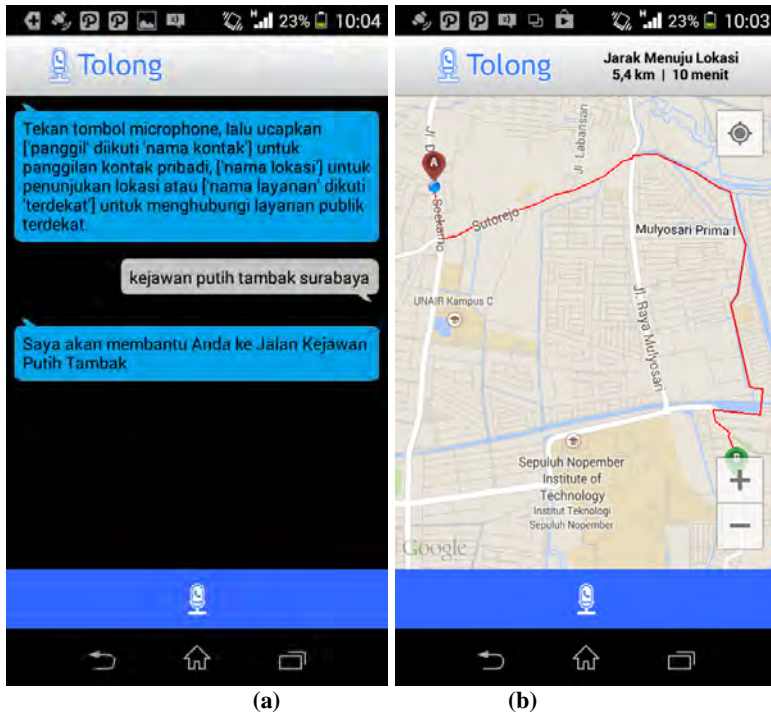
### 5.3.4 Pengujian Pembangkitan Rute Menuju Lokasi

Bagian ini menjelaskan tentang tantangan pengujian terhadap fungsionalitas pembangkitan rute menuju lokasi. Proses pengujian pembangkitan rute ini digunakan untuk menguji keberhasilan membangkitkan rute yang benar untuk menuju lokasi serta keberhasilan menampilkannya pada peta. Proses ini dimulai dengan proses pengenalan suara. Skenario pengujian fungsionalitas ini terdapat pada Tabel 5.5.

**Tabel 5.5 Skenario Pengujian Fungsionalitas Pembangkitan Rute Menuju Lokasi**

Kode Pengujian		TC-FR-04		
Tujuan Pengujian		Melakukan pengujian untuk membangkitkan rute menuju lokasi		
Kondisi Awal		Pengguna berada di Jalan Dr. Ir. Soekarno Surabaya dan mengakses halaman komunikasi		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Masukan suara : “kejawan putih tambak surabaya”	1. Menekan tombol suara 2. Mengucapkan masukan suara	Ditampilkan rute pada peta dari posisi pengguna menuju Jalan Kejawan Putih Tambak Surabaya.	Ditampilkan rute pada peta dari posisi pengguna menuju Jalan Kejawan Putih Tambak Surabaya.	Proses pembangkitan rute berhasil
Kondisi Akhir	Pengguna berada pada halaman peta			

Gambar 5.7 berikut merupakan potongan gambar pengujian fungsionalitas pembangkitan rute menuju lokasi Ketintang Baru Tengah setelah mengikuti langkah-langkah pada skenario pengujian fungsionalitas pembangkitan rute menuju lokasi. Rute ditunjukkan oleh penanda berwarna hijau, yang dimulai dari Jalan Kedung Cowek yang ditunjukkan oleh penanda huruf A dan diakhiri di Ketintang Baru Tengah yang ditandai oleh huruf B.



Gambar 5.7 Proses Pembangkitan Rute Menuju Kejawan Putih Tambak Surabaya. (a) Pengenalan Perintah Pengguna, (b) Tampilan Rute

### 5.3.5 Pengujian Pengiriman Pesan Darurat

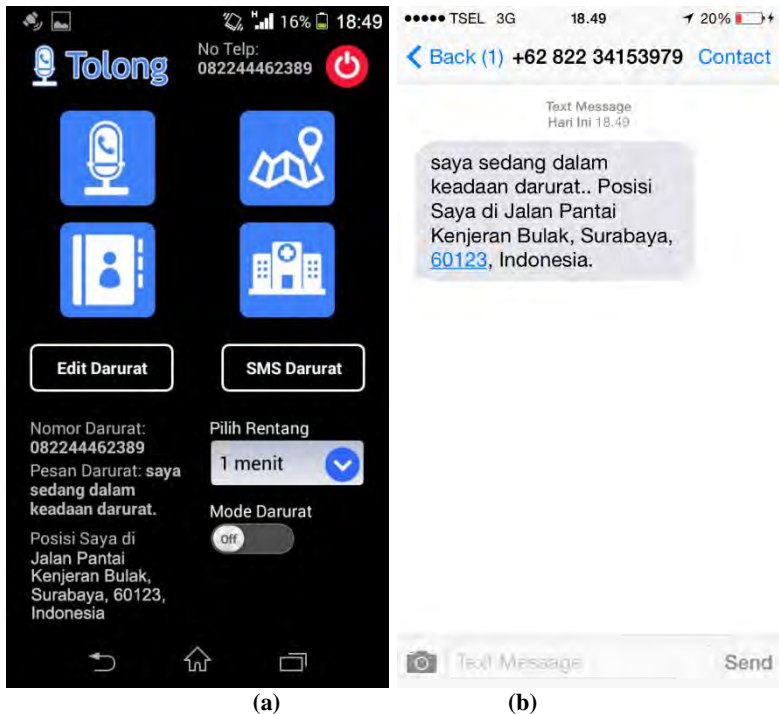
Bagian ini menjelaskan tentang tantangan pengujian terhadap fungsionalitas pengiriman pesan darurat dan mode darurat. Proses pengujian pengiriman pesan darurat dan mode darurat ini digunakan untuk menguji keberhasilan pengiriman pesan darurat beserta lokasi pengguna yang selalu berpindah tempat serta mengirimkan pesan kepada nomor darurat. Proses ini dimulai dengan proses pemilihan pengiriman pesan darurat oleh pengguna. Skenario pengujian fungsionalitas ini terdapat pada Tabel 5.6.

**Tabel 5.6 Skenario Pengujian Fungsionalitas Pengiriman Pesan Darurat**

Kode Pengujian		TC-FR-05		
Tujuan Pengujian		Melakukan pengujian untuk pengiriman pesan darurat dan mode darurat		
Kondisi Awal		Pegguna berada di Jalan Pantai Kenjeran Surabaya menuju Jalan Kertajaya Surabaya dan melakukan pengiriman pesan darurat		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Masukan tombol : sms darurat Masukan tombol pilihan : ya	1. Menekan tombol “SMS darurat” 2. Tombol pilihan “Ya”	Nomor darurat menerima pesan darurat yang berisi pesan darurat dan lokasi pengguna	Nomor darurat menerima pesan darurat yang berisi pesan darurat dan lokasi pengguna	Proses pengiriman pesan darurat berhasil
Masukan rentang : 1 menit Mengaktifkan tombol : mode darurat	1. Memilih rentang pengiriman “1 menit” 2. Mengaktifkan tombol “Mode Darurat”	Nomor darurat menerima pesan darurat yang berisi pesan darurat, keterangan mode darurat aktif, dan lokasi pengguna secara berkala tiap 1 menit.	Nomor darurat menerima pesan darurat yang berisi pesan darurat, keterangan mode darurat aktif, dan lokasi pengguna secara berkala tiap	Proses mode darurat berhasil

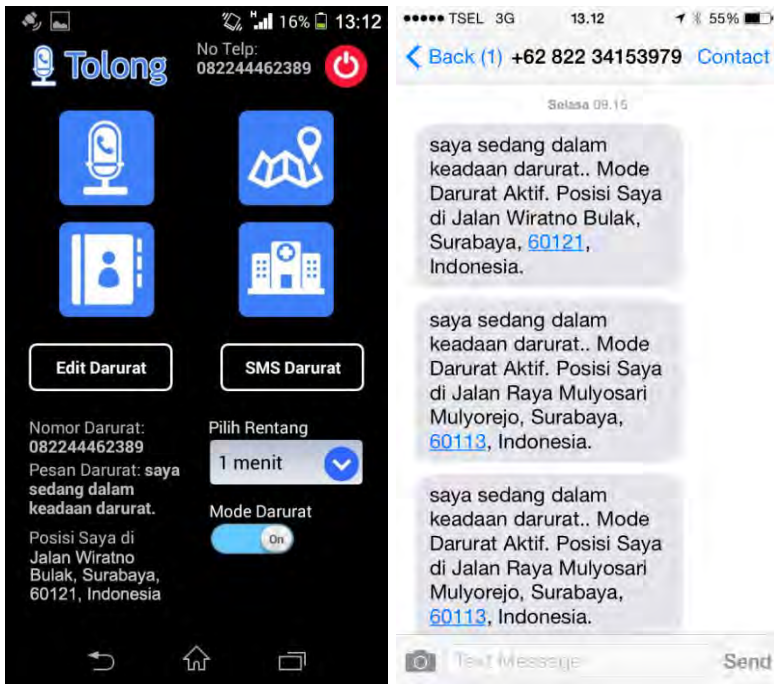
			1 menit.	
Kondisi Akhir	Pengguna berhasil mengirimkan pesan darurat			

Gambar 5.8 berikut merupakan potongan gambar pengujian fungsionalitas pengiriman pesan darurat dan pesan darurat yang diterima oleh nomor darurat dari pengguna.



**Gambar 5.8** Proses Pengiriman Pesan Darurat. (a) Pengiriman Pesan Darurat, (b) Pesan Darurat yang Diterima.

Gambar 5.9 berikut merupakan potongan gambar pengujian fungsionalitas pengiriman pesan mode darurat dengan mode rentang pengiriman 1 menit dan pesan mode darurat yang diterima oleh nomor darurat dari pengguna. Pada pengiriman pesan mode darurat, pengguna juga melakukan perpindahan lokasi dari Jalan Wiratno Bulak Surabaya menuju ke Jalan Raya Mulyosari Surabaya.



**Gambar 5.9** Proses Pengiriman Mode Darurat. (a) Pengiriman Pesan Mode Darurat, (b) Pesan Mode Darurat yang Diterima.

### 5.3.6 Pengujian Mendapatkan Nomor dan Pesan Darurat

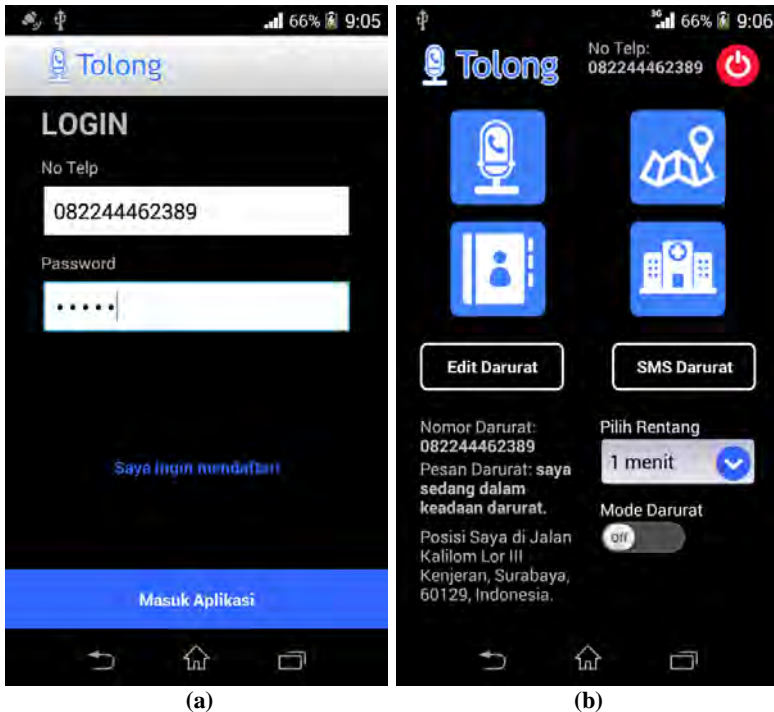
Pengujian mendapatkan nomor dan pesan darurat dilakukan untuk menguji fungsionalitas mendapatkan data pesan darurat yang diambil dari data *session login* pengguna. Data yang diambil dari *session login* pengguna meliputi nomor telepon pengguna, nomor darurat pengguna, dan pesan darurat pengguna. Skenario dari pengujian ini terdapat pada Tabel 5.7.

**Tabel 5.7 Skenario Pengujian Fungsionalitas Mendapatkan Nomor dan Pesan Darurat**

Kode Pengujian		TC-FR-06		
Tujuan Pengujian		Melakukan pengujian untuk mendapatkan nomor dan pesan darurat		
Kondisi Awal		Pengguna melakukan login dan masuk pada halaman menu aplikasi		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Masukan login : No_telp: "082244462389" Password : "12345"	1. Menekan tombol masuk aplikasi	Masuk pada halaman menu aplikasi dan mendapatkan nomor darurat dan pesan darurat	Masuk pada halaman menu aplikasi dan mendapatkan nomor darurat dan pesan darurat	Proses panggilan layanan publik terdekat berhasil
Kondisi Akhir	Pengguna berada pada halaman menu aplikasi dan mendapatkan nomor dan pesan darurat.			

Gambar 5.10 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas mendapatkan nomor dan pesan darurat berdasarkan Tabel 5.7.





Gambar 5.10 Proses Pengujian Mendapatkan Nomor dan Pesan Darurat. (a) Halaman Login, (b) Halaman Menu Aplikasi beserta Nomor dan Pesan Darurat

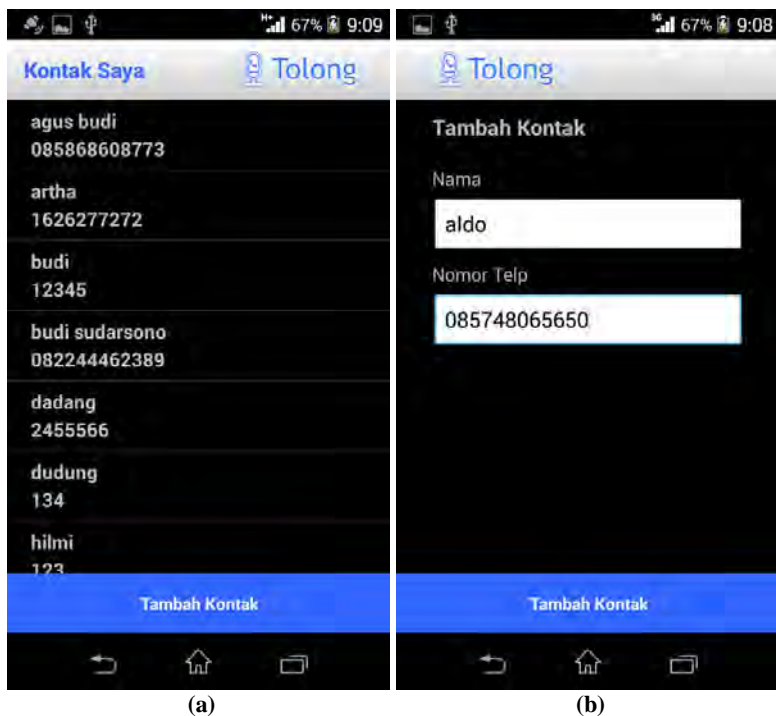
### 5.3.7 Pengujian Tambah Kontak Pribadi

Pengujian tambah kontak pribadi dilakukan untuk menguji fungsionalitas tambah kontak pribadi sebagai data yang akan di proses dalam panggilan kontak pribadi. Data kontak pribadi yang ditambahkan meliputi nama kontak dan nomor telepon kontak. Skenario dari pengujian ini terdapat pada Tabel 5.8.

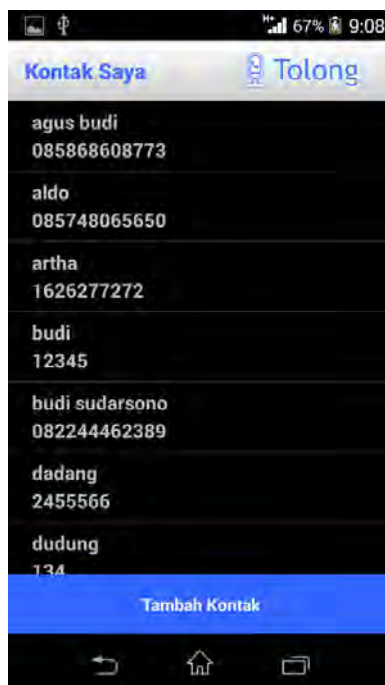
**Tabel 5.8 Skenario Pengujian Fungsionalitas Tambah Kontak Pribadi**

Kode Pengujian		TC-FR-07		
Tujuan Pengujian		Melakukan pengujian untuk menambah kontak pribadi		
Kondisi Awal		Pengguna melakukan penambahan kontak pribadi pada halaman tambah kontak		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Memasukkan Nama Kontak: "aldo" Nomor Telp: "08574806565 0"  Menekan tombol "tambah kontak"	1. Menekan tombol tambah kontak	Data kontak pribadi disimpan pada basis data dan dapat ditampilkan pada halaman kontak.	Data kontak pribadi disimpan pada basis data dan dapat ditampilkan pada halaman kontak.	Proses tambah kontak pribadi berhasil
Kondisi Akhir	Pengguna berada pada halaman kontak dan kontak telah ditambahkan.			

Gambar 5.11 dan Gambar 5.12 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas tambah kontak pribadi berdasarkan Tabel 5.8.



Gambar 5.11 Proses Pengujian Tambah Kontak Pribadi. (a) Halaman Sebelum Kontak Ditambahkan, (b) Halaman Tambah Kontak.



**Gambar 5.12** Halaman Kontak Setelah Kontak Ditambahkan

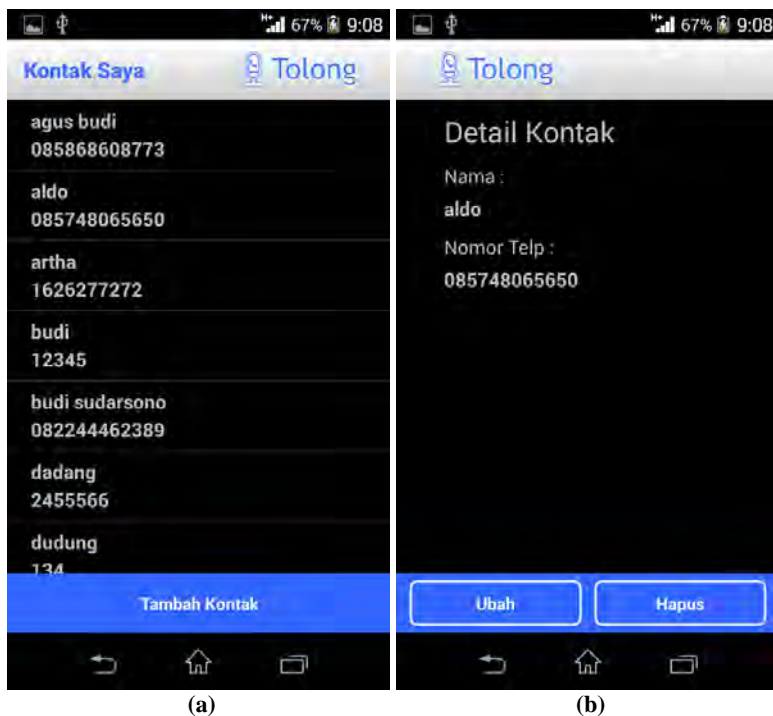
### **5.3.8 Pengujian Ubah Kontak Pribadi**

Pengujian ubah kontak pribadi dilakukan untuk menguji fungsionalitas ubah kontak pribadi sebagai data yang akan di proses dalam panggilan kontak pribadi. Data kontak pribadi dilakukan perubahan apabila terdapat perubahan nama kontak pribadi dan nomor telepon kontak pribadi. Skenario dari pengujian ini terdapat pada Tabel 5.8 berikut.

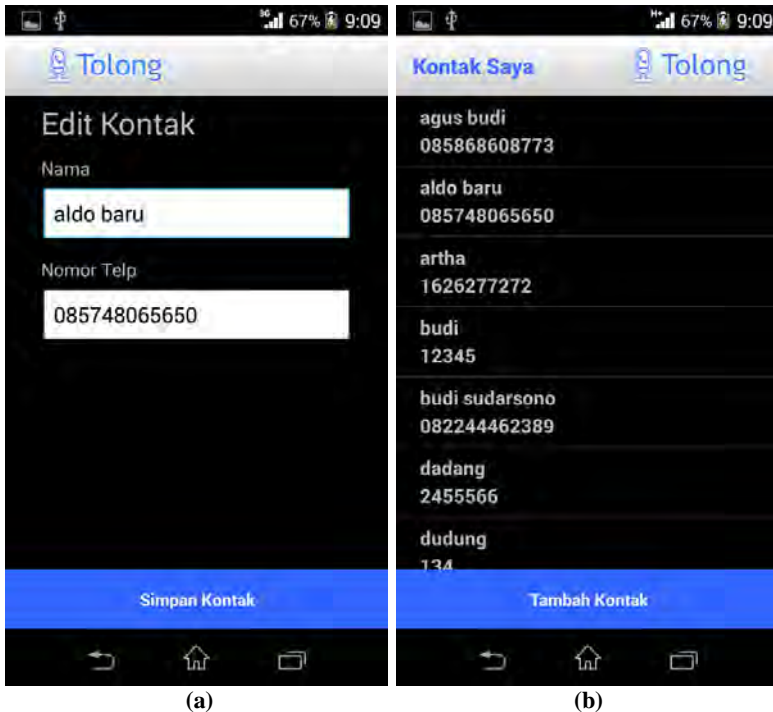
**Tabel 5.4 Skenario Pengujian Fungsionalitas Ubah Kontak Pribadi**

Kode Pengujian		TC-FR-08		
Tujuan Pengujian		Melakukan pengujian untuk mengubah kontak pribadi		
Kondisi Awal		Pengguna melakukan perubahan kontak pribadi pada halaman ubah kontak		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Memasukkan Nama Kontak: "aldo baru" Nomor Telp: "08574806565 0"  Menekan tombol "simpan kontak"	1. Menekan tombol simpan kontak	Data kontak pribadi yang disimpan pada basis data telah dirubah dengan data baru dan dapat ditampilkan pada halaman kontak.	Data kontak pribadi yang disimpan pada basis data telah dirubah dengan data baru dan dapat ditampilkan pada halaman kontak.	Proses ubah kontak pribadi berhasil
Kondisi Akhir	Pengguna berada pada halaman kontak dan kontak telah diubah.			

Gambar 5.13 dan Gambar 5.14 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas ubah kontak pribadi berdasarkan Tabel 5.8.



Gambar 5.13 Proses Pengujian Ubah Kontak Pribadi. (a) Kontak Setelah Kontak Diperbarui. (b) Halaman Detail Kontak.



(a) (b)  
**Gambar 5.14** Proses Pengujian Ubah Kontak Pribadi. (a) Halaman Edit Kontak. (b) Halaman Kontak Setelah Kontak Diperbarui

### 5.3.9 Pengujian Hapus Kontak Pribadi

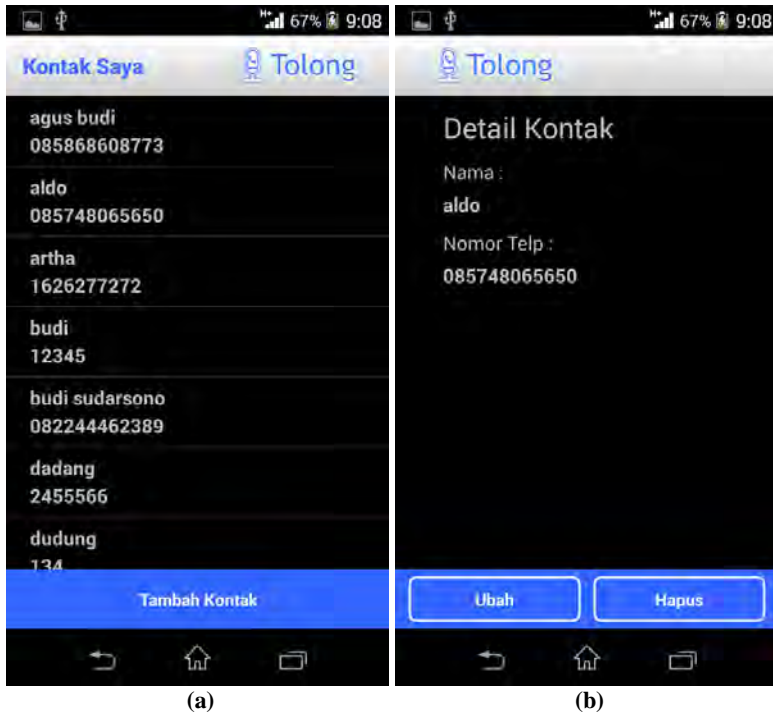
Pengujian hapus kontak pribadi dilakukan untuk menguji fungsionalitas hapus kontak pribadi sebagai data yang dihilangkan dalam basis data. Data kontak pribadi dilakukan penghapusan apabila sudah tidak dipergunakan kembali. Skenario dari pengujian ini terdapat pada Tabel 5.9 berikut.

**Tabel 5.5 Skenario Pengujian Fungsionalitas Hapus Kontak Pribadi**

Kode Pengujian		TC-FR-09		
Tujuan Pengujian		Melakukan pengujian untuk menghapus kontak pribadi		
Kondisi Awal		Pengguna melakukan hapus kontak pribadi pada halaman detail kontak		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Menekan tombol “hapus”	1. Menekan tombol hapus	Data kontak pribadi yang disimpan pada basis data telah dihapus dan kemudian menampilkan data kontak pribadi	Data kontak pribadi yang disimpan pada basis data telah dihapus dan kemudian menampilkan data kontak pribadi	Proses hapus kontak pribadi berhasil
Kondisi Akhir	Pengguna berada pada halaman kontak dan kontak telah dihapus.			

Gambar 5.15 dan Gambar 5.16 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas hapus kontak pribadi berdasarkan Tabel 5.9.





(a) (b)  
Gambar 5.15 Proses Pengujian Hapus Kontak Pribadi. (a) Kontak Sesebelum Kontak Dihapus. (b) Halaman Detail Kontak.



**Gambar 5.16 Halaman Kontak Setelah Kontak Dihapus.**

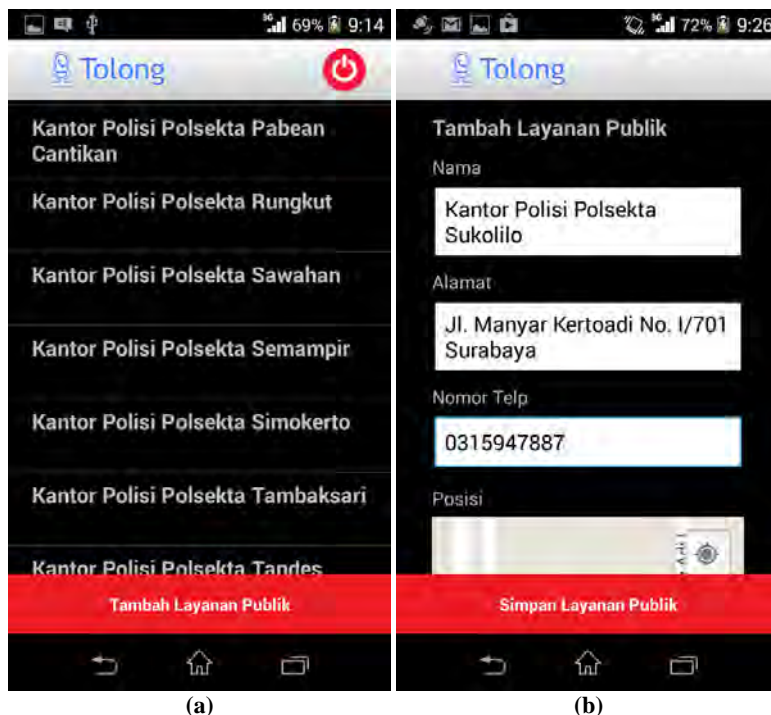
### **5.3.10 Pengujian Tambah Kontak Layanan Publik**

Pengujian tambah kontak pribadi dilakukan oleh pengguna admin untuk menguji fungsionalitas tambah kontak layanan publik sebagai data yang akan di proses dalam panggilan layanan publik pada pengguna aplikasi. Data kontak layanan publik yang ditambahkan meliputi nama kontak, alamat kontak, nomor telepon kontak dan lokasi layanan publik. Skenario dari pengujian ini terdapat pada Tabel 5.10 berikut.

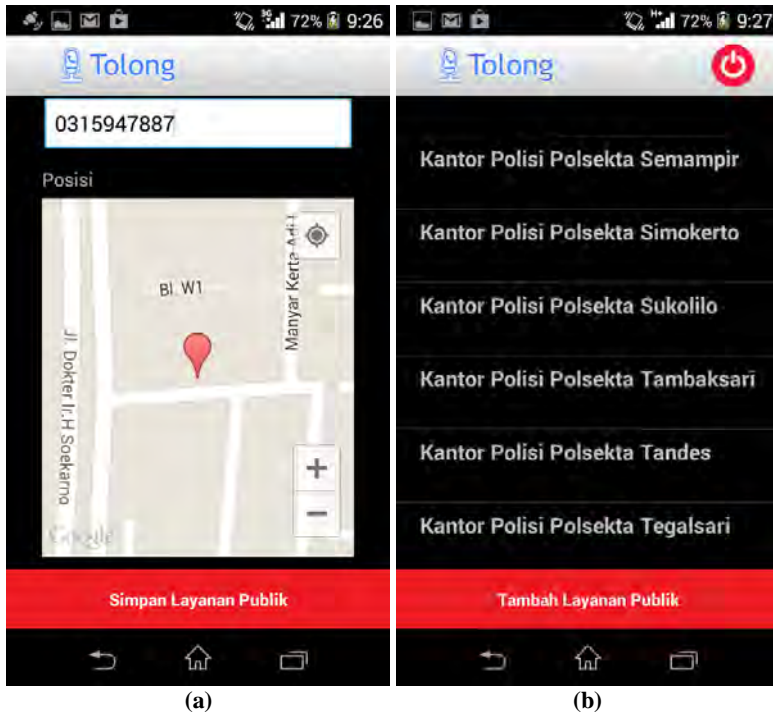
**Tabel 5.6 Skenario Pengujian Fungsionalitas Tambah Kontak Layanan Publik**

Kode Pengujian		TC-FR-10		
Tujuan Pengujian		Melakukan pengujian untuk menambah kontak layanan publik		
Kondisi Awal		Admin melakukan penambahan kontak layanan publik pada halaman tambah layanan publik		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
<p>Memasukkan Nama: “Kantor Polisi Polsekta Sukolilo” Alamat: “Jl. Manyar Kertoadi No. I/701 Surabaya” Nomor Telp: ”0315947887” Dan lokasi layanan publik</p> <p>Menekan tombol “Simpan layanan publik”</p>	1. Menekan tombol simpan layanan publik	Data kontak layanan publik disimpan pada basis data dan dapat ditampilkan pada halaman layanan publik.	Data kontak layanan publik disimpan pada basis data dan dapat ditampilkan pada halaman layanan publik.	Proses tambah kontak layanan publik berhasil
Kondisi Akhir	Pengguna berada pada halaman layanan publik dan kontak telah ditambahkan.			

Gambar 5.17 dan Gambar 5.18 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas tambah kontak pribadi berdasarkan Tabel 5.10.



**Gambar 5.17** Proses Pengujian Tambah Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Ditambahkan. (b) Halaman Tambah Kontak.



(a) Menambahkan Posisi Layanan. (b) Halaman Kontak Setelah Kontak Ditambahkan.

**Gambar 5.18 Proses Pengujian Tambah Kontak Layanan Publik.**

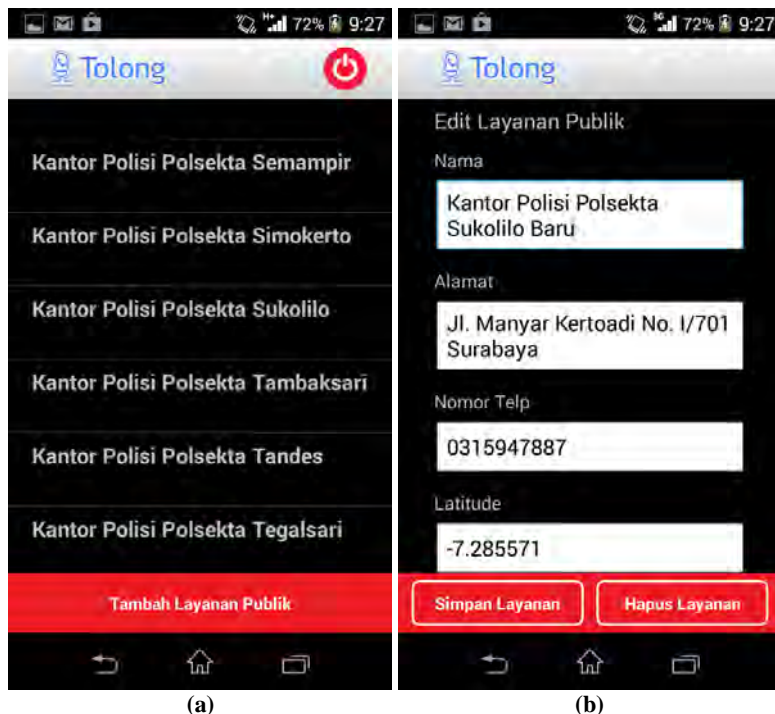
### 5.3.11 Pengujian Ubah Kontak Layanan Publik

Pengujian ubah kontak layanan publik dilakukan untuk menguji fungsionalitas ubah kontak layanan publik sebagai data yang akan di proses dalam panggilan layanan publik. Data kontak layanan publik dilakukan perubahan apabila terdapat perubahan nama layanan publik, alamat layanan publik, nomor telepon layanan publik, dan lokasi layanan publik. Skenario dari pengujian ini terdapat pada Tabel 5.11 berikut.

**Tabel 5.7 Skenario Pengujian Fungsionalitas Ubah Kontak Pribadi**

Kode Pengujian		TC-FR-11		
Tujuan Pengujian		Melakukan pengujian untuk mengubah kontak pribadi		
Kondisi Awal		Pengguna melakukan perubahan kontak pribadi pada halaman ubah kontak		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Memasukkan Nama: “Kantor Polisi Polsekta Sukolilo Baru” Alamat: “Jl. Manyar Kertoadi No. I/701 Surabaya” Nomor Telp: ”0315947887” Dan lokasi layanan publik  Menekan tombol “Simpan layanan publik”	1. Menekan tombol simpan layanan publik	Data kontak layanan publik yang disimpan pada basis data telah dirubah dengan data baru dan dapat ditampilkan pada halaman layanan publik.	Data kontak layanan publik yang disimpan pada basis data telah dirubah dengan data baru dan dapat ditampilkan pada halaman layanan publik.	Proses ubah kontak layanan publik berhasil
Kondisi Akhir	Pengguna berada pada halaman layanan publik dan kontak telah diubah.			

Gambar 5.19 dan Gambar 5.20 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas ubah kontak pribadi berdasarkan Tabel 5.11.



**Gambar 5.19** Proses Pengujian Ubah Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Diubah. (b) Halaman Edit Kontak.



**Gambar 5.20 Halaman Kontak Layanan Publik Setelah Kontak Diperbarui.**

### **5.3.12 Pengujian Hapus Kontak Layanan Publik**

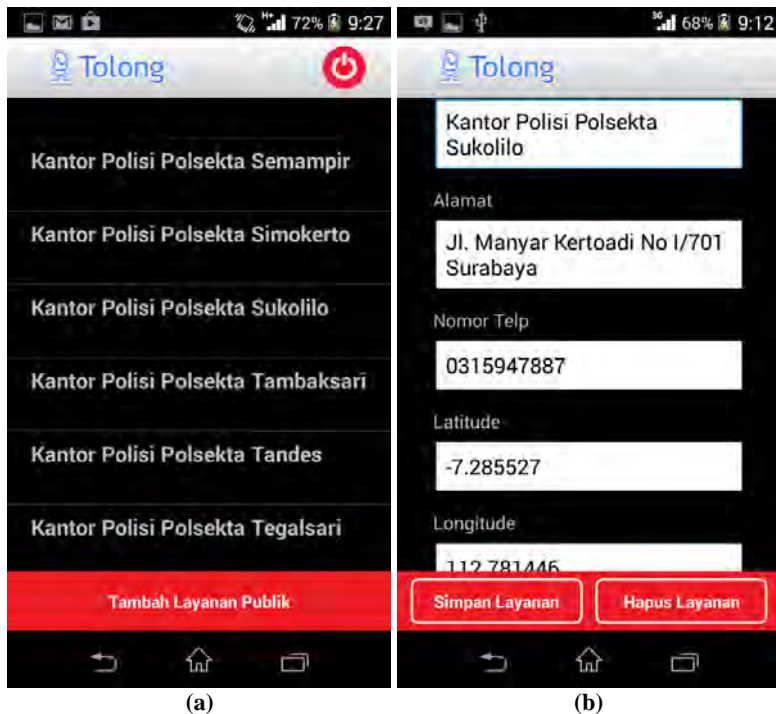
Pengujian hapus kontak layanan publik dilakukan untuk menguji fungsionalitas hapus kontak layanan publik sebagai data yang dihilangkan dalam basis data. Data kontak layanan publik dilakukan penghapusan apabila sudah tidak dipergunakan kembali. Skenario dari pengujian ini terdapat pada Tabel 5.12 berikut.



**Tabel 5.8 Skenario Pengujian Fungsionalitas Hapus Kontak Layanan Publik**

Kode Pengujian		TC-FR-12		
Tujuan Pengujian		Melakukan pengujian untuk menghapus kontak layanan publik		
Kondisi Awal		Pegguna melakukan hapus kontak layanan publik pada halaman detail kontak		
Data Input	Prosedur Pengujian	Hasil yang Diharapkan	Hasil yang Diperoleh	Kesimpulan
Menekan tombol “hapus layanan”	1. Menekan tombol hapus layanan	Data kontak layanan publik yang disimpan pada basis data telah dihapus dan kemudian menampilkan data kontak layanan publik	Data kontak layanan publik yang disimpan pada basis data telah dihapus dan kemudian menampilkan data kontak layanan publik	Proses hapus kontak layanan publik berhasil
Kondisi Akhir	Pegguna berada pada halaman layanan publik dan kontak telah dihapus.			

Gambar 5.21 dan Gambar 5.22 berikut menunjukkan proses pengujian yang dilakukan sesuai pada skenario pengujian fungsionalitas hapus kontak layanan publik berdasarkan Tabel 5.12.



Gambar 5.21 Proses Pengujian Hapus Kontak Layanan Publik. (a) Halaman Kontak Sebelum Kontak Dihapus. (b) Halaman Edit Kontak.



Gambar 5.22 Halaman Kontak Setelah Kontak Dihapus.

## 5.4 Pengujian Kegunaan

Selain melakukan pengujian fungsionalitas dengan metode *black box*, juga dilakukan pengujian kegunaan untuk menilai kegunaan perangkat lunak secara langsung kepada pengguna. Pengujian kegunaan dilakukan untuk mengetahui penilaian dan tanggapan dari pengguna terhadap sejumlah aspek dari perangkat lunak ini.

### 5.4.1 Kriteria Responden

Pengujian kegunaan dilakukan dengan melibatkan beberapa pengguna dari aplikasi ini. Para pengguna yang terlibat dalam

pengujian ini selanjutnya disebut sebagai responden. Para responden diberikan keleluasaan untuk menjalankan aplikasi pada perangkat bergerak Android.

#### 5.4.2 Lingkungan Pengujian Kegunaan

Masing-masing responden menjalankan perangkat lunak pada perangkat bergerak berbasis Android. Perangkat bergerak Android yang digunakan oleh responden terhubung dengan jaringan internet dengan versi minimum Android 4.0.

#### 5.4.3 Skenario Pengujian Kegunaan

Dalam melakukan pengujian kegunaan aplikasi, responden diminta untuk menjalankan aplikasi dan melakukan permintaan melakukan pengujian semua fitur yang ada pada aplikasin ini. Dalam memberikan penilaian terhadap aplikasi, responden mengisi formulir penilaian yang telah disediakan untuk pengujian ini. Formulir penilaian memiliki beberapa aspek penilaian secara kualitatif. Formulir penilaian pengujian kegunaan aplikasi yang digunakan adalah sebagaimana pada Lampiran D tentang kuisioner pengujian kegunaan.

#### 5.4.4 Daftar Responden

Terdapat 19 orang yang menjadi responden dalam pengujian kegunaan aplikasi. Daftar responden beserta keterangan perangkat bergerak yang digunakannya serta lokasi ketika menggunakan aplikasi terdapat pada Tabel 5.13.

**Tabel 5.9 Daftar Responden Pengujian Kegunaan**

No	Nama	Tipe Perangkat Bergerak	Versi Android	Lokasi
1	Hafiz Muftir	Samsung P6800	4.1.3	Surabaya
2	Bapak Ricky Wahyu	Sony Xperia C	4.2.2	Kertajaya, Surabaya

3	Kirana Risnu	Lenovo K900	4.2	Surabaya
4	Sabrina Amalia	Sony Xperia Miro	4.0	Rungkut Tengah, Surabaya
5	Helma Fiverio	Samsung Galaxy Tab 3	4.3	Surabaya
6	Amelia Puspasari	Sony Xperia M	4.3.0	Rungkut Harapan, Surabaya
7	Nadia Fajrin	Samsung Note 3	4.3	Rungkut Asri Tengah, Surabaya
8	Bapak Pariyun	Samsung Note 2	4.1.2	Kertajaya, Surabaya
9	Ibu Erina Prasetya	Sony Xperia Tipo	4.0	Kertajaya, Surabaya
10	Moniqua Darma	Hisense A10	4.2.0	Surabaya
11	Yunita Praba Wati	Samsung Galaxy Tab 2	4.1.3	Surabaya
12	Ahmad Alan Ananta	Sony Xperia Tipo ST2i	4.0.1	Surabaya
13	Intan Permata Sari	Samsung P3100	4.1	Surabaya
14	Ibu Jessica Jeje	Samsung Galaxy Tab 7 plus	4.1.3	Surabaya
15	Alditha Cahyani	Sony Xperia C	4.2.2	Kertajaya, Surabaya
16	Deddy Baadillah	Samsung P6200	4.1.3	Surabaya
17	Devi Permatasari	Galaxy Tab 7 plus	4.1.3	Surabaya

18	Harining Tyas Dian	Galaxy Tab 7 plus	4.1.3	Surabaya
19	Syah Rizal	Sony Xperia M	4.3	Surabaya

### 5.4.5 Hasil Pengujian Kegunaan

Pada penilain pengujian kegunaan, terdapat empat pilihan nilai untuk setiap pertanyaan yaitu tidak baik, kurang baik, cukup baik, dan sangat baik. Pengujian dibagi menjadi lima aspek yaitu antarmuka pengguna, pengenalan dan penyampaian suara, telepon kontak pribadi dan telepon layanan terdekat, informasi penunjukan lokasi, pesan darurat dan mode darurat. Berikut ini adalah rekapitulasi penilaian dari hasil pengujian kegunaan.

#### 5.4.5.1 Rekapitulasi Penilaian Antarmuka Pengguna

Antarmuka pengguna merupakan salah satu aspek yang dinilai pada pengujian kegunaan aplikasi ini. Rekapitulasi penilaian pengujian terhadap antarmuka pengguna terdapat pada Tabel 5.14.

**Tabel 5.10 Penilaian Antarmuka Pengguna**

No.	Penilaian				Rata-rata
	Tidak	Kurang	Cukup	Sangat	
1	0	0	14	5	3,26
2	0	0	10	9	3,47
3	0	1	10	8	3,36
<b>Nilai Akhir</b>					<b>3,36</b>

Tabel 5.14 di atas menunjukkan bahwa antarmuka pengguna dari perangkat lunak memiliki nilai rata-rata 3,36. Nilai ini menandakan bahwa aspek ini sudah lebih dari cukup baik.

#### 5.4.5.2 Rekapitulasi Penilaian Pengenalan dan Penyampaian Suara

Pengujian kegunaan aplikasi juga menilai keberhasilan pengenalan dan penyampaian suara berdasarkan kondisi yang dialami pengguna. Rekapitulasi penilaian pengujian terhadap penilaian pengenalan dan penyampaian suara terdapat pada Tabel 5.15.

**Tabel 5.11 Penilaian Pengenalan dan Penyampaian Suara**

No.	Penilaian				Rata-rata
	Tidak	Kurang	Cukup	Sangat	
1	0	1	12	6	3,26
2	0	2	9	8	3,31
3	0	1	12	6	3,26
<b>Nilai Akhir</b>					<b>3,28</b>

Berdasarkan pada Tabel 5.15, pengujian kegunaan pada aspek penilaian pengenalan dan penyampaian suara memiliki nilai rata-rata 3,28 yang berarti aspek ini sudah lebih dari cukup baik.

#### 5.4.5.3 Rekapitulasi Penilaian Telepon Kontak Pribadi dan Telepon Layanan Publik Terdekat

Telepon kontak pribadi dan telepon layanan publik terdekat merupakan salah satu aspek yang dinilai pada pengujian kegunaan. Rekapitulasi penilaian telepon kontak pribadi dan telepon layanan publik terdekat terdapat pada Tabel 5.16.

**Tabel 5.12 Penilaian Telepon Kontak Pribadi dan Telepon Layanan Publik Terdekat**

No.	Penilaian				Rata-rata
	Tidak	Kurang	Cukup	Sangat	
1	0	0	8	11	3,57
2	0	1	10	8	3,36
3	0	1	12	6	3,26
<b>Nilai Akhir</b>					<b>3,40</b>

Sesuai dengan rekapitulasi penilaian pada Tabel 5.16, dapat diketahui bahwa layanan telepon kontak pribadi dan telepon layanan publik terdekat yang diberikan oleh aplikasi memiliki nilai rata-rata 3,40 yang berarti aspek ini sudah lebih dari cukup baik.

#### 5.4.5.4 Rekapitulasi Penilaian Informasi Penunjukan Lokasi

Informasi penunjukan lokasi merupakan salah satu aspek yang dinilai pada pengujian kegunaan. Rekapitulasi penilaian informasi penunjukan lokasi terdapat pada Tabel 5.17.

**Tabel 5.13 Penilaian Informasi Penunjukan Lokasi**

No.	Penilaian				Rata-rata
	Tidak	Kurang	Cukup	Sangat	
1	0	0	11	8	3,42
2	0	1	8	10	3,47
3	0	0	8	11	3,57
<b>Nilai Akhir</b>					<b>3,49</b>

Tabel 5.17 menunjukkan bahwa pengujian kegunaan untuk aspek penilaian informasi penunjukan lokasi mencapai angka yang lebih dari cukup, yaitu 3,49.

#### 5.4.5.5 Rekapitulasi Penilaian Pesan Darurat dan Mode Darurat

Pesan darurat dan mode darurat merupakan salah satu aspek yang dinilai pada pengujian kegunaan. Rekapitulasi penilaian layanan pesan darurat dan mode darurat terdapat pada Tabel 5.18.



**Tabel 5.14 Penilaian Pesan Darurat dan Mode Darurat**

No.	Penilaian				Rata-rata
	Tidak	Kurang	Cukup	Sangat	
1	0	1	7	11	3,52
2	0	2	8	9	3,36
3	0	1	10	8	3,36
<b>Nilai Akhir</b>					<b>3,42</b>

Tabel 5.18 menunjukkan bahwa pengujian kegunaan untuk aspek penilaian layanan pesan darurat dan mode darurat mencapai angka yang lebih dari cukup, yaitu 3,42.

## 5.5 Evaluasi Pengujian

Berdasarkan hasil pengujian fungsionalitas dan pengujian kegunaan, maka dapat dilakukan evaluasi sebagai berikut.

### 5.5.1 Evaluasi Pengujian Fungsionalitas

Pengujian fungsionalitas yang telah dilakukan memberikan hasil yang sesuai dengan skenario yang telah direncanakan. Evaluasi pengujian untuk masing-masing fungsional aplikasi ini dijelaskan sebagai berikut.

1. Pengujian pengenalan suara berjalan sesuai dengan yang diharapkan, baik untuk pengenalan perintah panggilan layanan publik terdekat, lokasi layanan terdekat, panggilan kontak pribadi, penunjukan lokasi dengan nama tujuan yang unik dan penunjukan lokasi dengan nama tujuan yang umum. Kondisi ini diperlihatkan pada pengujian TC-FR-01 yang memberikan informasi bahwa proses pengenalan suara berjalan dengan benar.
2. Pengujian panggilan layanan publik terdekat berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-02 yang memberikan keluaran

- berupa nama layanan publik terdekat sesuai lokasi layanan publik terdekat dan dapat langsung melakukan panggilan darurat.
3. Pengujian panggilan kontak pribadi berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-03 yang memberikan keluaran berupa nama masukan pengguna yang selanjutnya dapat melakukan panggilan langsung sesuai perintah pengguna.
  4. Pengujian pembangkitan rute menuju lokasi berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-04 yang memberikan keluaran berupa informasi bahwa proses pembangkitan rute berjalan dengan benar.
  5. Pengujian pengiriman pesan darurat berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-05 yang memberikan keluaran pada pemilik nomor darurat berupa pesan darurat yang berisi pesan darurat pada pengguna beserta nama lokasi pengguna yang dapat berjalan dengan benar.
  6. Pengujian mendapatkan nomor dan pesan darurat berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-06 yang memberikan keluaran berupa nomor darurat dan pesan darurat pengguna yang dapat berjalan dengan benar.
  7. Pengujian tambah kontak pribadi berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-07 yang memberikan keluaran berupa kontak pribadi bertambah sesuai data masukan dan dapat berjalan dengan benar.

8. Pengujian edit kontak pribadi berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-08 yang memberikan keluaran berupa data kontak pribadi yang disimpan di basis data telah berubah menjadi data kontak pribadi yang baru dan dapat berjalan dengan benar.
9. Pengujian hapus kontak pribadi berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-09 yang memberikan keluaran berupa data kontak pribadi yang disimpan di basis data telah dihapus dan dapat berjalan dengan benar.
10. Pengujian tambah kontak layanan publik berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-10 yang memberikan keluaran berupa kontak layanan publik bertambah sesuai data masukan dan dapat berjalan dengan benar.
11. Pengujian edit kontak layanan publik berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-11 yang memberikan keluaran berupa data kontak layanan publik yang disimpan di basis data telah berubah menjadi data kontak layanan publik yang baru dan dapat berjalan dengan benar.
12. Pengujian hapus kontak layanan publik berjalan sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian TC-FR-12 yang memberikan keluaran berupa data kontak layanan publik yang disimpan di basis data telah dihapus dan dapat berjalan dengan benar.

Semua skenario pengujian telah berhasil dilaksanakan dan memberikan hasil yang sesuai dengan harapan. Kondisi tersebut

memberikan kesimpulan bahwa secara fungsionalitas aplikasi ini telah berjalan seperti yang diharapkan.

### 5.5.2 Evaluasi Pengujian Kegunaan

Untuk mengevaluasi pengujian kegunaan perangkat lunak, maka dilakukan rekapitulasi akhir. Rekapitulasi akhir menghasilkan nilai persentase terhadap aspek-aspek yang dinilai dalam pengujian kegunaan. Rekapitulasi akhir terdapat pada Tabel 5.19. Formulir penilaian pengujian kegunaan aplikasi yang digunakan adalah sebagaimana pada Lampiran C tentang kuisioner pengujian kegunaan.

**Tabel 5.15 Rekapitulasi Akhir Pengujian Kegunaan.**

No.	Aspek Pengujian	Nilai Akhir	Persentase
1.	Antarmuka Pengguna	3,36	84 %
2.	Pengenalan dan Penyampaian Suara	3,28	82 %
3.	Telepon Kontak Pribadi dan Layanan Publik Terdekat	3,40	85 %
4.	Informasi Penunjukan Lokasi	3,49	87,25 %
5.	Pesan Darurat dan Mode Darurat	3,42	85,5 %

Dari Tabel 5.19 dapat dilihat bahwa aspek-aspek yang dinilai menghasilkan nilai-nilai persentase. Penilaian terhadap antarmuka pengguna menghasilkan persentase sebesar 84 %, penilaian terhadap pengenalan dan penyampaian suara menghasilkan persentase sebesar 82 %, penilaian terhadap telepon kontak pribadi dan layanan publik terdekat menghasilkan persentase sebesar 85 %, penilaian terhadap informasi penunjukan lokasi menghasilkan persentase sebesar 87,25 %, dan penilaian terhadap pesan darurat dan mode darurat menghasilkan persentase sebesar 85,5 %. Keempat aspek menunjukkan persentase nilai yang lebih dari 75 %, sehingga dapat disimpulkan bahwa dari segi kegunaan,

perangkat lunak yang dibuat pada Tugas Akhir ini memberikan hasil yang lebih dari cukup baik.

## DAFTAR PUSTAKA

- [1] PTKPT, “Jumlah Penduduk (Seluruhnya) di Masing-masing Negara,” [Online]. Available: [http://statistik.ptkpt.net/\\_a.php?\\_a=penduduk\\_usia&info1=3..](http://statistik.ptkpt.net/_a.php?_a=penduduk_usia&info1=3..) [Diakses 25 Februari 2014].
- [2] Badan Intelijen Republik Indonesia, “Kecelakaan Lalu Lintas Menjadi Pembunuh Terbesar Ketiga,” [Online]. Available: <http://www.bin.go.id/awas/detil/197/4/21/03/2013/%20Kecelakaan-lalu-lintas-menjadi-pembunuh-terbesar-ketiga>. [Diakses 10 Februari 2014].
- [3] Paseban Portal, “Perbandingan Tingkat Akurasi Fitur Perintah Suara Siri dan S Voice,” [Online]. Available: <http://portal.paseban.com/review/9768/perintah-suara-siri-dan-s-voice>. [Diakses 20 April 2014].
- [4] Safaat, Nazruddin, Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android, Bandung: Informatika Bandung, 2012.
- [5] Google, “Google Maps Android API v2,” [Online]. Available: <https://developers.google.com/maps/documentation/android/>. [Diakses 24 April 2014].
- [6] Gruenstein, Alex, “Speech Input API For Android,” [Online]. Available: [http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/www.google.co.jp/ja/jp/events/developerday/2010/tokyo/pdf/tt1-gruenstein.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/www.google.co.jp/ja/jp/events/developerday/2010/tokyo/pdf/tt1-gruenstein.pdf). [Diakses 20 April 2014].
- [7] Android Developers Blog, “An Introduction to Text-To-Speech in Android,” [Online]. Available: <http://android-developers.blogspot.com/2009/09/introduction-to-text-to-speech-in.html>. [Diakses 24 April 2014].
- [8] Rooting Android, “How GPS is Structured in Android,” [Online]. Available: <http://rootingandroid.net/how-gps-is-structured-in-android/>. [Diakses 20 April 2014].
- [9] JSON, “Pengenalan JSON,” [Online]. Available: <http://json.org/json-id.html>. [Diakses 10 Februari 2014].

- [10] Arman, Ary Akhmad, “Teknologi Pemrosesan Bahasa Alami sebagai Teknologi Kunci untuk Meningkatkan Cara Interaksi antara Manusia dengan Mesin,” [Online]. Available: [http://www.itb.ac.id/focus/focus\\_file/Pidato%20Ilmiah%20pada%20Sidang%20Terbuka%20PMB%202004.pdf](http://www.itb.ac.id/focus/focus_file/Pidato%20Ilmiah%20pada%20Sidang%20Terbuka%20PMB%202004.pdf). [Diakses 20 April 2014].
- [11] Dian Wahyu Sambodo, Nur, Sunaryono, Dwi, Sarwosri, Perancangan dan Pembangunan Perangkat Lunak Jejaring Sosial Berbasis Lokasi untuk Wisata Kuliner pada Platform Android, Surabaya: Institut Teknologi Sepuluh Nopember, 2013.
- [12] Math UNL, “SphericalLawOfCosines,” [Online]. Available: <http://www.math.unl.edu/~shartke2/teaching/2011m896/SphericalLawOfCosines.pdf>. [Diakses 22 Juli 2014].

## BIODATA PENULIS



Lucky Dewantara, lahir di Bojonegoro pada tanggal 11 Februari 1992, merupakan anak pertama dari tiga bersaudara. Penulis telah menempuh pendidikan mulai TK Tunas Sejati (1996-1998), SDN Tanah Kali Kedinding V/579 (1998-2004), SMP YP 17 Surabaya (2004-2007), SMAN 19 Surabaya (2007-2010), dan terakhir sebagai mahasiswa Teknik Informatika ITS (2010-2014). Selama masa perkuliahan di Teknik Informatika ITS, penulis aktif dalam kegiatan keorganisasian, di antaranya Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) dan Badan Eksekutif Mahasiswa Fakultas Teknologi Informasi (BEM-FTIf) ITS. Dalam menyelesaikan pendidikan sarjana, penulis mengambil bidang minat Rekayasa Perangkat Lunak (RPL). Penulis dapat dihubungi melalui email di [luckydewantara.064@gmail.com](mailto:luckydewantara.064@gmail.com).



## LAMPIRAN A – IMPLEMENTASI LAYOUT XML

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ff3366ff"
    android:gravity="center|center_horizontal"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/header_button_timeline"
        android:layout_width="700dp"
        android:layout_height="404dp"
        android:layout_weight="0.06"
        android:background="@null"
        android:src="@drawable/Logo2" />
/RelativeLayout>
```

Gambar A.1 Implementasi Antarmuka Halaman Pembuka dalam XML.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    style="@style/Page" >

    <LinearLayout
        android:id="@+id/page_main_layout_content"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/page main layout header"
    >

        <RelativeLayout
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:orientation="vertical"
            android:background="@android:color/black">

            <ImageView
                android:id="@+id/header_button_timeline"
                android:layout_width="150dp"
                android:layout_height="wrap_content"
```

```

        android:layout_alignParentLeft="true"
        android:paddingRight="10dp"
        android:background="@null"
        android:src="@drawable/atastolong" />

<TextView
    android:id="@+id/lblNoTelP"
    android:layout_width="150dp"
    android:layout_height="50dp"
    android:textAlignment="center"
    android:layout_marginTop="20dp"

    android:layout_alignParentRight="true"
    android:layout_marginRight="120dp"
    android:text="Medium Text"
    android:textAppearance="?android:attr/textAppearanceMedium"
/>

<Button
    android:id="@+id/menuLogout"
    android:layout_width="100dp"
    android:layout_height="50dp"
    android:layout_marginTop="20dp"
    android:layout_alignParentRight="true"
    android:layout_marginRight="10dp"
    android:onClick="keKeLuar"
    android:text="Logout" />

<ImageButton
    android:id="@+id/menuBicara"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_marginTop="120dp"
    android:layout_marginLeft="90dp"
    android:layout_weight="0.06"
    android:background="@null"
    android:onClick="keKomunikasi"
    android:src="@drawable/btn11" />

<ImageButton
    android:id="@+id/menuPeta"
    android:layout_width="200dp"
    android:layout_height="200dp"

```

```

android:layout_marginLeft="310dp"
android:layout_marginTop="120dp"
android:onClick="kePeta"
android:background="@null"
android:src="@drawable/btn22"/>

```

#### <ImageButton

```

android:id="@+id/menuKontak"
android:layout_width="200dp"
android:layout_height="200dp"
android:layout_marginTop="320dp"
android:layout_marginLeft="90dp"
android:background="@null"
android:onClick="keALLKontak"
android:src="@drawable/btn33" />

```

#### <ImageButton

```

android:id="@+id/menuPublik"
android:layout_width="200dp"
android:layout_height="200dp"
android:layout_marginTop="320dp"
android:layout_marginLeft="310dp"
android:onClick="kePublik"
android:background="@null"
android:src="@drawable/btn44"/>

```

#### <Button

```

android:id="@+id/menuEditPesanDarurat"
android:layout_width="150dp"
android:layout_height="50dp"
android:layout_marginTop="550dp"
android:layout_marginLeft="120dp"
android:onClick="keEditPesanDarurat"
android:text="Edit Darurat" />

```

#### <Button

```

android:id="@+id/menuSMS"
android:layout_width="150dp"
android:layout_height="50dp"
android:layout_marginTop="550dp"
android:layout_marginLeft="330dp"
android:onClick="keSMS"

```

```

        android:text="SMS Darurat" />

        <TextView
            android:id="@+id/LblNomorDarurat"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentLeft="true"
            android:layout_marginTop="670dp"
            android:layout_marginLeft="50dp"
            android:text="Medium Text"
            android:textAppearance="?android:attr/textAppearanceMedium"
        />

        <TextView
            android:id="@+id/LblPesanDarurat"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentLeft="true"
            android:layout_marginLeft="50dp"
            android:layout_marginTop="720dp"
            android:text="Medium Text"
            android:textAppearance="?android:attr/textAppearanceMedium"
        />

        <TextView
            android:id="@+id/LblRentangSpinner"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:layout_alignParentRight="true"
            android:layout_marginRight="60dp"
            android:layout_marginTop="720dp"
            android:text="Pilih Rentang"
            android:textColor="@android:color/white"
            android:textAppearance="?android:attr/textAppearanceMedium"
        />

        <Spinner
            android:id="@+id/spinner_rentang"
            android:layout_width="200dp"
            android:layout_height="50dp"
            android:layout_alignParentRight="true"
            android:layout_marginRight="50dp"
            android:layout_marginTop="750dp"/>

        <TextView

```

```

        android:id="@+id/lblModeDarurat"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_marginRight="150dp"
        android:layout_marginTop="820dp"
        android:text="Mode Darurat"
        android:textColor="@android:color/white"
        android:textAppearance="?android:attr/textAppearanceMedium"
    />

    <Switch
        android:id="@+id/modeDarurat"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_marginRight="30dp"
        android:layout_marginTop="820dp"
        android:checked="false"
        android:textOff="Off"
        android:textOn="On" />

    <TextView
        android:id="@+id/address"
        android:layout_width="220dp"
        android:layout_height="100dp"
        android:layout_alignParentLeft="true"
        android:layout_marginLeft="50dp"
        android:layout_marginTop="770dp"
        android:text=""
        android:textAppearance="?android:attr/textAppearanceMedium"
    />
</RelativeLayout>
</LinearLayout>
</RelativeLayout>

```

**Gambar A.2 Implementasi Antarmuka Halaman Menu Aplikasi.**

```

<ListView
    android:id="@+id/chatView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/form"

```

```

        android:layout_alignParentTop="true"
        android:dividerHeight="0dp"
        android:divider="@null">

</ListView>

```

**Gambar A.3 Implementasi *ListView* untuk Komunikasi.**

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" >

    <LinearLayout
        android:id="@+id/wrapper"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/comment"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_margin="5dip"
            android:background="@drawable/bubble_blue"
            android:paddingLeft="10dip"
            android:text="Hello bubbles!"

            android:textColor="@android:color/primary_text_light" />
        </LinearLayout>
    </LinearLayout>

```

**Gambar A.4 Implementasi pembuatan gelembung obrolan untuk Komunikasi.**

```

<ImageButton
    android:id="@+id/header_button_more"
    style="@style/ButtonNavigation"
    android:onClick="speak"
    android:src="@drawable/ic_btn_speak"
    android:background="@null"/>

```

**Gambar A.5 Implementasi *ImageButton* pada Tombol Bicara.**

```

<fragment
    android:id="@+id/map"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    class="com.google.android.gms.maps.SupportMapFragment"
/>

```

**Gambar A.6 Implementasi *MapView* dengan *ApiKey*.**

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    style="@style/Page"
    android:id="@+id/page_main_all">
    <LinearLayout
        android:id="@+id/page_main_layout_header"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:orientation="vertical" >
        <RelativeLayout style="@style/Header" >
            <ImageView
                android:id="@+id/header_button_timeline"
                android:layout_width="150dp"
                android:layout_height="wrap_content"
                android:layout_alignParentRight="true"
                android:layout_centerVertical="true"
                android:layout_marginRight="-10dp"
                android:background="@null"
                android:src="@drawable/atastolong" />

            <TextView
                android:id="@+id/header_bar_title"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@null"
        android:layout_centerVertical="true"
        android:layout_marginLeft="10dp"
        android:textColor="@color/background_page"
        android:textSize="@dimen/text_size_large"
        android:textStyle="bold"
        android:text="Kontak Saya" />
    </RelativeLayout>
    <ImageView style="@style/BarSeparator.Horizontal.Grey"
/>
</LinearLayout>
<LinearLayout
    android:id="@+id/page_main_layout_content"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_above="@+id/page_main_layout_footer"
    android:layout_below="@+id/page_main_layout_header" >

    <RelativeLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="vertical"
        android:background="@android:color/black"
    >
        <ListView
            android:id="@android:id/list"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentLeft="true" >
        </ListView>
    </RelativeLayout>
</LinearLayout>
<LinearLayout
    android:id="@+id/page_main_layout_footer"
    style="@style/Footer"
    android:layout_alignParentBottom="true"
    android:background="@null">

    <Button
        android:id="@+id/btnAddNewKontak"

```



```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"
        android:background="@null"
        android:textColor="@android:color/white"
        android:onClick="keNewKontak"
        android:textStyle="bold"
        android:text="Tambah Kontak" />
    </LinearLayout>
</RelativeLayout>

```

**Gambar A.7 Implementasi Antarmuka Halaman Kontak.**

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView
        android:id="@+id/id_kontak"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone" />

    <TextView
        android:id="@+id/nama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="6dip"
        android:paddingLeft="20dip"
        android:textSize="@dimen/text_size_medium"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/no_telp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="1dip"
        android:paddingLeft="20dip"
        android:paddingBottom="6dip"
        android:textSize="@dimen/text_size_medium"

```

```

        android:textStyle="bold" />
</LinearLayout>

```

**Gambar A.8 Implementasi *ListView* Kontak.**

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
style="@style/PageAdmin"
android:id="@+id/page_main_all">
    <LinearLayout
        android:id="@+id/page_main_layout_header"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:orientation="vertical" >
        <RelativeLayout style="@style/Header" >
            <ImageView
                android:id="@+id/header_button_timeline"
                android:layout_width="150sp"
                android:layout_height="wrap_content"
                android:layout_centerVertical="true"
                android:layout_marginLeft="10sp"
                android:background="@null"
                android:src="@drawable/atastoLong" />
            <ImageButton
                android:id="@+id/menuLogout"
                android:layout_width="50sp"
                android:layout_height="50sp"
                android:layout_alignParentRight="true"
                android:layout_marginRight="10dp"
                android:layout_gravity="center_vertical"
                android:onClick="keKeluarPrivate"
                android:background="@null"
                android:src="@drawable/btn_quit" />
            <TextView
                android:id="@+id/header_bar_title"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_centerInParent="true"
                android:background="@null"
                android:textColor="@color/background_page"

```

```

        android:textSize="@dimen/text_size_extra_Large"
        android:textStyle="bold" />
    </RelativeLayout>
    <ImageView style="@style/BarSeparator.Horizontal.Grey"
/>
</LinearLayout>

<LinearLayout
    android:id="@+id/page_main_layout_content"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_above="@+id/page_main_layout_footer"
    android:layout_below="@+id/page_main_layout_header" >

    <RelativeLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="vertical"
        android:background="@android:color/black">
        <ListView
            android:id="@android:id/list"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentLeft="true" >
        </ListView>
    </RelativeLayout>
</LinearLayout>

<LinearLayout
    android:id="@+id/page_main_layout_footer"
    style="@style/Footer"
    android:layout_alignParentBottom="true"
    android:background="@null" >

    <Button
        android:id="@+id/btnAddNewPrivateKontak"
        android:layout_width="100dp"
        android:layout_height="match_parent"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:paddingLeft="8dp"
        android:paddingRight="8dp"
        android:background="@null"

```

```

        android:textColor="@android:color/white"
        android:textStyle="bold"
        android:layout_weight="1"
        android:onClick="keTambahKontakPublik"
        android:text="Tambah Layanan Publik" />
    </LinearLayout>
</RelativeLayout>

```

**Gambar A.9 Implementasi Antarmuka Halaman Layanan Publik.**

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <TextView
        android:id="@+id/id_private_kontak"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone" />

    <TextView
        android:id="@+id/nama_private"
        android:layout_width="wrap_content"
        android:layout_height="60sp"
        android:paddingTop="6dip"
        android:paddingLeft="20dip"
        android:paddingBottom="6dip"
        android:textSize="@dimen/text_size_large"
        android:textStyle="bold" />

</LinearLayout>

```

**Gambar A.10 Implementasi *ListView* Layanan Publik.**

## LAMPIRAN B – KODE SUMBER

```
public class GetRouteListTask extends AsyncTask<Void, Void,
Void> {

    private final Context mContext;
    GmapV2Direction mGMDirection = new GmapV2Direction();
    LatLng fromPosition;
    LatLng toPosition;
    List<LatLng> mPointList;
    private ProgressDialog dialog;
    private int mDirectionMode;
    DirectionReceivedListener mListener;

    public GetRouteListTask(Context context, LatLng
        fromPosition, LatLng toPosition, int
        mDirectionMode, DirectionReceivedListener
        mListener) {
        this.mContext = context;
        this.fromPosition = fromPosition;
        this.toPosition = toPosition;
        this.mDirectionMode = mDirectionMode;
        this.mListener = mListener;
    }

    @Override
    protected Void doInBackground(Void... params) {
        mGMDirection.setParams(fromPosition, toPosition,
mDirectionMode);
        mPointList = mGMDirection.getPointList(this.mContext);
        return null;
    }

    @Override
    protected void onPostExecute(Void result) {
        if (dialog.isShowing()) {
            dialog.dismiss();
        }

        if (mPointList != null) {
            mListener.OnDirectionListReceived(mPointList);
        } else {
```

```

        Toast.makeText(this.mContext, "Error downloading
direction!",
        Toast.LENGTH_LONG).show();
    }
}

@Override
protected void onPreExecute() {
    ConnectivityManager conMgr = (ConnectivityManager)
mContext.getSystemService(
    Context.CONNECTIVITY_SERVICE);
    if (conMgr.getActiveNetworkInfo() != null
        && conMgr.getActiveNetworkInfo().isAvailable()
        &&
conMgr.getActiveNetworkInfo().isConnectedOrConnecting()) {

        dialog = new ProgressDialog(mContext);
        dialog.setMessage("Downloading directions...");
        dialog.show();
    } else {
        this.cancel(true);
        Toast.makeText(mContext, "Not connected to
internet!",
        Toast.LENGTH_LONG).show();
    }
}

@Override
protected void onCancelled() {
    super.onCancelled();
}
}

```

Gambar B.1 Kode Sumber Kelas *GetRouteListTask*.

```

public class MyPosition extends Service implements
LocationListener {

    private final Context mContext;
    boolean isGPSEnabled = false;
    boolean isNetworkEnabled = false;
    boolean canGetLocation = false;

```

```

    Location location; // location
    double latitude; // latitude
    double longitude; // longitude
    private static final long
MIN_DISTANCE_CHANGE_FOR_UPDATES = 10; // 10 meters
    private static final long MIN_TIME_BW_UPDATES = 1000
* 60 * 1; // 1 minute

    protected LocationManager locationManager;
    public MyPosition(Context context) {
        this.mContext = context;
        getLocation();
    }

    public Location getLocation() {
        try {
            locationManager = (LocationManager) mContext
                .getSystemService(LOCATION_SERVICE);
            // getting GPS status
            isGPSEnabled = locationManager
                .isProviderEnabled(LocationManager.GPS_PROVIDER);
            isNetworkEnabled = locationManager
                .isProviderEnabled(LocationManager.NETWORK_PROVIDER);

            if (!isGPSEnabled && !isNetworkEnabled) {
                // no network provider is enabled
            } else {
                this.canGetLocation = true;
                // First get location from Network Provider
                if (isNetworkEnabled) {
                    locationManager.requestLocationUpdates(
                        LocationManager.NETWORK_PROVIDER,
                        MIN_TIME_BW_UPDATES,
                        MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                    Log.d("Network", "Network");
                    if (locationManager != null) {
                        location = locationManager
                            .getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
                        if (location != null) {
                            latitude = location.getLatitude();
                            longitude = location.getLongitude();
                        }
                    }
                }
            }
        }
    }

```

```

    }
}
// if GPS Enabled get lat/long using GPS Services
if (isGPSEnabled) {
    if (location == null) {
        locationManager.requestLocationUpdates(
            locationManager.GPS_PROVIDER,
            MIN_TIME_BW_UPDATES,
            MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
        Log.d("GPS Enabled", "GPS Enabled");
        if (locationManager != null) {
            location = locationManager
                .getLastKnownLocation(locationManager.GPS_PROVIDER);
            if (location != null) {
                latitude = location.getLatitude();
                longitude = location.getLongitude();
            }
        }
    }
}
}
}
}
} catch (Exception e) {
    e.printStackTrace();
}
return location;
}

public void stopUsingGPS(){
    if(locationManager != null){
        locationManager.removeUpdates(MyPosition.this);
    }
}

public double getLatitude(){
    if(location != null){
        latitude = location.getLatitude();
    }
    return latitude;
}

public double getLongitude(){
    if(location != null){

```



```

        longitude = location.getLongitude();
    }
    return longitude;
}

public boolean canGetLocation() {
    return this.canGetLocation;
}

public void showSettingsAlert(){
    AlertDialog.Builder alertDialog = new
AlertDialog.Builder(mContext);
    alertDialog.setTitle("GPS is settings");
    alertDialog.setMessage("GPS is not enabled. Do
you want to go to settings menu?");
    alertDialog.setPositiveButton("Settings", new
DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog,int
which) {
            Intent intent = new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
            mContext.startActivity(intent);
        }
    });

    alertDialog.setNegativeButton("Cancel", new
DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int
which) {
            dialog.cancel();
        }
    });
    alertDialog.show();
}

@Override
public void onLocationChanged(Location location) {
}

@Override
public void onProviderDisabled(String provider) {
}

```

```

@Override
public void onProviderEnabled(String provider) {
}

@Override
public void onStatusChanged(String provider, int
status, Bundle extras) {
}

@Override
public IBinder onBind(Intent arg0) {
    return null;
}
}

```

Gambar B.2 Kode Sumber Kelas MyPosition.

```

MyPosition gps = new MyPosition(this);
    if(gps.canGetLocation()){
        asal = new LatLng
(gps.getLatitude(),gps.getLongitude());
    }
    namaAsal = "Posisi Anda";
    tujuan = pTujuan;
    namaTujuan = nTujuan;
    markerPoints = new ArrayList<LatLng>();
    MarkerOptions options = new MarkerOptions();

    options.position(asal);
    LatLngBounds bounds =
this.peta.getProjection().getVisibleRegion().latLngBo
unds;

//penanda posisi tujuan
    MarkerOptions mDestination = new
MarkerOptions()
    .position(tujuan)
    .icon(BitmapDescriptorFactory.fromResource(R.drawable
    .bmarker));

```

```

        peta.addMarker(mDestination);

//penanda posisi asal
        MarkerOptions mStart = new MarkerOptions()
        .position(asal)
        .icon(BitmapDescriptorFactory.fromResource(R.drawable
        .amarker));
        peta.addMarker(mStart);

```

Gambar B.3 Kode Sumber Penanda pada Peta.

```

public void bacaSuara(String pesan, int kodeTanya){
    Intent intent = new
    Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
        RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_PREFERENCE,
    "id"); //menerima suara berbahasa Indonesia
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, "id");
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, pesan);
    startActivityForResult(intent, kodeTanya);
}

```

Gambar B.4 Kode Sumber Inisiasi *Intent* untuk Penerimaan Suara.

```

@Override
protected void onActivityResult(int requestCode, int
resultCode, Intent data)
{
    loading.dismiss();
    if (requestCode == KODE_BACA_SUARA && resultCode ==
RESULT_OK){
        terimaSuara(data);
    }
    else if (requestCode == KODE_KONFIRMASI_TEMPAT &&
resultCode == RESULT_OK) {
        terimaNomorKonfirmasi(data);
    }
    else if(requestCode == KODE_TTS){
        if (resultCode ==
TextToSpeech.Engine.CHECK_VOICE_DATA_PASS) {

```

```

        suara = new TextToSpeech(this, this);
    }
    else {
        Intent pasang_TTS = new Intent();
        pasang_TTS.setAction(TextToSpeech.Engine.ACTION_INSTALL_TTS
_DATA);
        startActivity(pasang_TTS);
    }
}
super.onActivityResult(requestCode, resultCode, data);
}

```

**Gambar B.5 Kode Sumber Inisiasi Penerimaan Hasil Respon *Intent* Untuk Menerima Suara.**

```

private String getKalimat(Intent data){
    ArrayList<String> matches =
data.getStringArrayListExtra(RecognizerIntent.EXTRA_R
ESULTS);
    String statement;
    if(!matches.isEmpty()){
        statement = matches.get(0);
    }
    else statement = "";
    return statement;
}

```

**Gambar B.6 Kode Sumber Pembacaan Teks.**

```

else if (arrPerintah[maksArray-1].equals("dekat") ||
arrPerintah[maksArray-1].equals("terdekat")){

        String kp = arrPerintah[0];

```

**Gambar B.7 Kode Sumber Proses Pengambilan Kata Nama Layanan.**

```
String latNow = null;
String lngNow = null;
MyPosition gps = new MyPosition(this);
    if(gps.canGetLocation()){
        latNow = String.valueOf(gps.getLatitude());
        lngNow = String.valueOf(gps.getLongitude());
    }
}
```

**Gambar B.8 Kode Sumber untuk Deteksi Posisi Pengguna.**

```
String kp = arrPerintah[0];
int success;
try {
    List<NameValuePair> params = new
ArrayList<NameValuePair>();
    params.add(new BasicNameValuePair("nama", kp));
    params.add(new BasicNameValuePair("latitude",
latNow));
    params.add(new BasicNameValuePair("longitude",
lngNow));

    JSONObject json =
    jsonParser.makeHttpRequest(url_select_call_distance, "GET",
params);

    Log.d("Call Distance Details", json.toString());

    success = json.getInt(TAG_SUCCESS);
    if(success == 1){
        call_kontak = json.getJSONArray(TAG_KONTAK);
        if (call_kontak.length()==1){
            JSONObject s = call_kontak.getJSONObject(0);
            String nama = s.getString(TAG_NAMA);
            String no_telp = s.getString(TAG_NO_TELP);
            String distance = s.getString(TAG_DISTANCE);
            ucapRespon("Saya akan memanggil"+nama,
true);

            while(suara.isSpeaking()){
                Intent callIntent = new
Intent(Intent.ACTION_CALL);

                callIntent.setData(Uri.parse("tel:"+no_telp));
```

```

        startActivity(callIntent);
    }
    else {
        String responAsli = "Ditemukan lebih dari 1 layanan
publik. Pilih nomor berapa?";
        String respon = responAsli;
        for (int h = 0; h < call_kontak.length(); h++) {
            JSONObject s = call_kontak.getJSONObject(h);

            String nama = s.getString(TAG_NAMA);
            String no_telp = s.getString(TAG_NO_TELP);
            String distance = s.getString(TAG_DISTANCE);

            HashMap<String, String> map = new HashMap<String,
String>();
            map.put(TAG_NAMA, nama);
            map.put(TAG_NO_TELP, no_telp);
            map.put(TAG_DISTANCE, distance);

            panggilanList.add(map);
            System.out.println(nama);
            System.out.println(no_telp);
            System.out.println(distance);
        }

        int listData = panggilanList.size();
        for(int j=0;j<listData;j++)
        {
            respon += ( "\n" + (j+1) + ". " +
panggilanList.get(j).get(TAG_NAMA) + ". " +
panggilanList.get(j).get(TAG_NO_TELP) + ".");
        }

        ucapRespon(respon, false);
        bicara(PengolahKalimat.ubahNomorJadiKata(responAsli));
        isKonfirmasi = true;
    }
}

else{
    ucapRespon("Data layanan publik tidak ditemukan.",
true);
}

```

```

}
} catch (JSONException e) {
    ucapRespon("Data layanan publik tidak
ditemukan.", true);
    e.printStackTrace();
}
return;

```

**Gambar B.9 Kode Sumber Proses Pencarian Nama Layanan dan Pemanggilan Layanan 1 Hasil.**

```

<?php

$response = array();
require_once __DIR__ . '/db_connect.php';

$nama = $_GET["nama"];
$latitude = $_GET["latitude"];
$longitude = $_GET["longitude"];

$db = new DB_CONNECT();

$result = mysql_query("SELECT nama, no_telp, ( 6371 * acos(
cos( radians('$latitude') ) * cos( radians( latitude ) ) *
cos( radians( longitude ) - radians('$longitude') ) + sin(
radians('$latitude') ) * sin( radians( latitude ) ) ) ) AS
distance FROM private_kontak WHERE nama LIKE '%$nama%'
HAVING distance < 20 ORDER BY distance LIMIT 0 , 10");

if (mysql_num_rows($result) > 0) {
    $response["kontak"] = array();
    while ($row = mysql_fetch_array($result)) {
        $kontak = array();
        $kontak["nama"] = $row["nama"];
        $kontak["no_telp"] = $row["no_telp"];
        $kontak["distance"] = $row["distance"];

        array_push($response["kontak"], $kontak);
    }
    $response["success"] = 1;
    echo json_encode($response);
}

```

```

} else {
    $response["success"] = 0;
    $response["message"] = "No kontak found";
    echo json_encode($response);
}
?>

```

**Gambar B.10 Kode Sumber *Query Service* Pencarian Layanan dengan Metode Spherical Law of Cosines.**

```

else if (panggilanList!=null && n>0 &&
n<=panggilanList.size()){
    isKonfirmasi = false;
    n--;
    String respon = "Saya akan membantu menelpon
"+panggilanList.get(n).get(TAG_NAMA) + ", dengan pilihan
nomor " + angka + ".";
    ucapRespon(respon, true);
    while(suara.isSpeaking()){
        Intent callIntent = new
Intent(Intent.ACTION_CALL);

        callIntent.setData(Uri.parse("tel:"+panggilanList.get(n)
.get(TAG_NO_TELP)));
        startActivity(callIntent);
    }
}

```

**Gambar B.11 Kode Sumber Proses Pemanggilan Layanan Publik Lebih dari 1 Hasil.**

```

else if (arrPerintah[0].equals("panggil") ||
arrPerintah[0].equals("telepon") ||
arrPerintah[0].equals("telpon")){

    String db = arrPerintah[1];
}

```

**Gambar B.12 Kode Sumber Proses Pengambilan Nama Kontak.**



```

int success;
try {
    List<NameValuePair> params = new
ArrayList<NameValuePair>();
    params.add(new BasicNameValuePair("nama", db));
    params.add(new BasicNameValuePair("id_user", no_telp));

    JSONObject json =
jsonParser.makeHttpRequest(url_select_call, "GET", params);

    Log.d("Call Details", json.toString());
    success = json.getInt(TAG_SUCCESS);
    if(success == 1){
        call_kontak = json.getJSONArray(TAG_KONTAK);
        if (call_kontak.length()==1){
            JSONObject s = call_kontak.getJSONObject(0);

            String nama = s.getString(TAG_NAMA);
            String no_telp = s.getString(TAG_NO_TELP);
            ucapRespon("Saya akan memanggil"+nama, true);
            while(suara.isSpeaking()){
                Intent callIntent = new
Intent(Intent.ACTION_CALL);

                callIntent.setData(Uri.parse("tel:"+no_telp));
                startActivity(callIntent);
            }
            else {
                String responAsli = "Ditemukan lebih dari 1
kontak. Pilih nomor berapa?";
                String respon = responAsli;
                for (int h = 0; h < call_kontak.length(); h++)
                {
                    JSONObject s = call_kontak.getJSONObject(h);

                    String nama = s.getString(TAG_NAMA);
                    String no_telp = s.getString(TAG_NO_TELP);

                    HashMap<String, String> map = new
HashMap<String, String>();

```

```

        map.put(TAG_NAMA, nama);
        map.put(TAG_NO_TELP, no_telp);

        panggilanList.add(map);
        System.out.println(nama);
        System.out.println(no_telp);
    }
    Int listData = panggilanList.size();
    for(int j=0;j<listData;j++)
    {
        respon += ( "\n" + (j+1) + ". " +
panggilanList.get(j).get(TAG_NAMA) + ". " +
panggilanList.get(j).get(TAG_NO_TELP) + ". " );
    }
    ucapRespon(respon, false);

    bicara(PengolahKalimat.ubahNomorJadiKata(responAsli));
    isKonfirmasi = true;
    }
}
else{
    ucapRespon("Data kontak tidak ditemukan.", true);
}
} catch (JSONException e) {
    ucapRespon("Data kontak tidak ditemukan.", true);
    e.printStackTrace();
}
}
return;

```

**Gambar B.13 Kode Sumber Proses Pencarian Keberadaan Nama Kontak.**

```

<?php
$response = array();
require_once __DIR__ . '/db_connect.php';
$nama = $_GET["nama"];
$id_user = $_GET["id_user"];

$db = new DB_CONNECT();
$result = mysql_query("SELECT id_kontak, nama, no_telp,
id_user='$id_user' FROM kontak WHERE nama LIKE '%$nama%'")
or die(mysql_error());

```

```

if (mysql_num_rows($result) > 0) {
    $response["kontak"] = array();
    while ($row = mysql_fetch_array($result)) {
        $kontak = array();
        $kontak["id_kontak"] = $row["id_kontak"];
        $kontak["nama"] = $row["nama"];
        $kontak["no_telp"] = $row["no_telp"];
        $kontak["id_user"] = $row["id_user"];
        array_push($response["kontak"], $kontak);
    }
    $response["success"] = 1;
    echo json_encode($response);
} else {
    $response["success"] = 0;
    $response["message"] = "No kontak found";
    echo json_encode($response);
}
?>

```

**Gambar B.14 Kode Sumber *Query Service* Pencarian Keberadaan Nama Kontak.**

```

else if(!ntujuan.equals("")){
    try {
        Locale loc = new Locale("in");
        Geocoder geocoder = new Geocoder(this, loc);
        List <Address> addresses =
geocoder.getFromLocationName(ntujuan, 5);
        if (addresses != null && addresses.size()>0 ) {
            int size = addresses.size();
            if(size==1){
                Address address = addresses.get(0);
                tampilkanPesan("Saya akan membantu
Anda ke "+address.getAddressLine(0));
                inisialisasiNilai();
                peta.clear();
                LatLng dest = new
LatLng(address.getLatitude(),address.getLongitude());
                gantiTujuan(dest,
address.getAddressLine(0));
            }
            //kalo lebih dari 1?

```

```

        else{
            String responAsli = "Ditemukan lebih dari
1 lokasi. Pilih nomor berapa?";
            String respon = "Pilih nomor berapa?";
            int i=0;
            for(;i<size; i++){
                Address address = addresses.get(i);
                String addr=alamatLengkap(address);
                respon += ( "\n" + (i+1) + ". " + addr
+ "." );
            }
            hanyaPesan(respon);

            bicara(PengolahKalimat.ubahNomorJadiKata(responAsli));
            banyakAddress = addresses;
            isKonfirmasi = true;
        }
    }
    else
        tampilkanPesan("Maaf, saya tidak menemukan
lokasi dengan nama " +ntujuan +".");

    } catch (IOException e) {
        tampilkanPesan("Sepertinya koneksi saya sedang
buruk. Silakan coba lagi.");
        e.printStackTrace();
    }
}
}

```

**Gambar B.15 Kode Sumber untuk Mencari Keberadaan Lokasi.**

```

public class DirectionsJSONParser {

    // menerima JSONObject kembalian latitude and longitude
    public List<List<HashMap<String, String>>>
parse(JSONObject jobject)
    {
        List<List<HashMap<String, String>>> routes = new
ArrayList<List<HashMap<String, String>>>();
        JSONArray jRoutes = null;
        JSONArray jLegs = null;
        JSONArray jSteps = null;
    }
}

```

```

JSONObject jDistance = null;
JSONObject jDuration = null;

try
{
jRoutes = jobject.getJSONArray("routes");
for (int i = 0; i < jRoutes.length(); i++)
{
jLegs = ((JSONObject)
jRoutes.get(i)).getJSONArray("legs");
List<HashMap<String, String>> path = new
ArrayList<HashMap<String, String>>();
for (int j = 0; j < jLegs.length(); j++)
{
//dapat jarak dari data JSON
jDistance = ((JSONObject)
jLegs.get(j)).getJSONObject("distance");
HashMap<String, String> hmDistance = new HashMap<String,
String>();
hmDistance.put("distance", jDistance.getString("text"));

//dapat waktu dari data JSON
jDuration = ((JSONObject)
jLegs.get(j)).getJSONObject("duration");
HashMap<String, String> hmDuration = new HashMap<String,
String>();
hmDuration.put("duration", jDuration.getString("text"));

path.add(hmDistance);
path.add(hmDuration);

jsteps = ((JSONObject) jLegs.get(j)).getJSONArray("steps");

for (int k = 0; k < jsteps.length(); k++)
{
String polyline = "";
polyline = (String) ((JSONObject)
((JSONObject)jsteps.get(k)).get("polyline")).get("points");
List<LatLng> list = this.decodePoly(polyline);

for (int l = 0; l < list.size(); l++)
{

```

```

    HashMap<String, String> hm = new HashMap<String,
String>();
    hm.put("lat", Double.toString((list.get(1)).latitude));
    hm.put("lng", Double.toString((list.get(1)).longitude));
    path.add(hm);
    }
    }
    }
    routes.add(path);
    }

    } catch (JSONException e)
    {
    e.printStackTrace();
    } catch (Exception e)
    {
    }

    return routes;
    }

    private List<LatLng> decodePoly(String encoded)
    {

    List<LatLng> poly = new ArrayList<LatLng>();
    int index = 0, len = encoded.length();
    int lat = 0, lng = 0;

    while (index < len)
    {
    int b, shift = 0, result = 0;
    do
    {
    b = encoded.charAt(index++) - 63;
    result |= (b & 0x1f) << shift;
    shift += 5;
    } while (b >= 0x20);
    int dlat = ((result & 1) != 0 ? ~(result >> 1) : (result
>> 1));
    lat += dlat;

    shift = 0;

```

```

    result = 0;
    do
    {
        b = encoded.charAt(index++) - 63;
        result |= (b & 0x1f) << shift;
        shift += 5;
    } while (b >= 0x20);
    int dlng = ((result & 1) != 0 ? ~(result >> 1) : (result
    >> 1));
    lng += dlng;

    LatLng p = new LatLng(((lat / 1E5)), ((lng / 1E5)));
    poly.add(p);
    }

    return poly;
}
}

```

**Gambar B.16** Kode Sumber kelas DirectionJSONParser

```

private String getDirectionUrl(LatLng origin,LatLng dest){
    String str_origin =
    "origin="+origin.latitude+","+origin.longitude;
    String str_dest =
    "destination="+dest.latitude+","+dest.longitude;

    String sensor =
    "sensor=true&mode=driving&language=id&alternatives=true";

    String parameters = str_origin+"&" +str_dest+"&" +sensor;

    String output = "json";

    String url =
    "https://maps.googleapis.com/maps/api/directions/" +output+"
    ?"+parameters;

    return url;
}
/** A method to download json data from url */
private String downloadUrl(String strUrl) throws

```

```

IOException
{
    String data = "";
    InputStream iStream = null;
    HttpURLConnection urlConnection = null;
    try
    {
        URL url = new URL(strUrl);
        urlConnection = (HttpURLConnection) url.openConnection();
        urlConnection.connect();

        // membaca data dari url
        iStream = urlConnection.getInputStream();
        BufferedReader br = new BufferedReader(new
InputStreamReader(iStream));

        StringBuffer sb = new StringBuffer();
        String line = "";
        while ((line = br.readLine()) != null)
        {
            sb.append(line);
        }
        data = sb.toString();
        br.close();
    } catch (Exception e)
    {
        Log.d("Exception while downloading url", e.toString());
    } finally
    {
        {
            iStream.close();
            urlConnection.disconnect();
        }
        return data;
    }
}

private class DownloadTask extends AsyncTask<String,
Void, String>
{
    @Override
    protected String doInBackground(String... url)
    {
        String data = "";
    }
}

```



```

        try
        {
            data = PetaActivity.this.downloadUrl(url[0]);
        } catch (Exception e)
        {
            Log.d("Background Task", e.toString());
        }
        return data;
    }

    @Override
    protected void onPostExecute(String result)
    {
        super.onPostExecute(result);
        ParserTask parserTask = new ParserTask();
        parserTask.execute(result);
    }
}

private class ParserTask extends AsyncTask<String,
Integer, List<List<HashMap<String, String>>>>
{
    @Override
    protected List<List<HashMap<String, String>>>
doInBackground(String... jsonData)
    {
        JSONObject jObject;
        List<List<HashMap<String, String>>> routes =
null;

        try
        {
            jObject = new JSONObject(jsonData[0]);
            DirectionsJSONParser parser = new
DirectionsJSONParser();

            routes = parser.parse(jObject);
        } catch (Exception e)
        {
            e.printStackTrace();
        }
        return routes;
    }
}

```

```

    }

    @Override
    protected void
onPostExecute(List<List<HashMap<String, String>>> result)
    {
        ArrayList<LatLng> points = null;
        PolylineOptions lineOptions = null;
        MarkerOptions markerOptions = new
MarkerOptions();
        String distance = "";
        String duration = "";

        if (result.size() < 1)
        {

Toast.makeText(PetaActivity.this.getContext(), "No
Points", Toast.LENGTH_SHORT).show();
            return;
        }

        for (int i = 0; i < result.size(); i++)
        {
            points = new ArrayList<LatLng>();
            lineOptions = new PolylineOptions();
            List<HashMap<String, String>> path =
result.get(i);

            for (int j = 0; j < path.size(); j++)
            {
                HashMap<String, String> point =
path.get(j);

                if (j == 0)
                { // jarak
                    distance = point.get("distance");
                    continue;
                } else if (j == 1)
                { // waktu
                    duration = point.get("duration");
                    continue;
                }
                double lat =

```

```

Double.parseDouble(point.get("lat"));
        double lng =
Double.parseDouble(point.get("lng"));
        LatLng position = new LatLng(lat, lng);
        points.add(position);
    }
    System.out.println("Hallo");

    // gambar garis rute
    lineOptions.addAll(points);
    lineOptions.width(2);
    lineOptions.color(Color.RED);
}

    TextView sisa = (TextView)
findViewById(R.id.judul_sisa_perjalanan);
    String sisaInfo = "";
    sisaInfo += "Jarak Menuju Lokasi Tujuan ";
    sisaInfo += "\n" + distance + " | ";
    sisaInfo += duration;
    sisa.setText(sisaInfo);

    // gambar rute hingga rute ke-i
    PetaActivity.this.peta.addPolyline(lineOptions);
    CameraPosition mCPFrom = new
CameraPosition.Builder()
        .target(asal).zoom(15.5f).bearing(0).tilt(25)
        .build();

    final CameraPosition mCPTo = new
CameraPosition.Builder()
        .target(tujuan).zoom(15.5f).bearing(0)
        .tilt(50).build();

changeCamera(CameraUpdateFactory.newCameraPosition(mCPFrom)
,
    new CancelableCallback() {
        @Override
        public void onFinish() {
            changeCamera(CameraUpdateFactory.newCameraPosition(mCPTo

```

```

),new CancelableCallback() {

    @Override
    public void onFinish() {

        LatLngBounds bounds = new
LatLngBounds.Builder()
        .include(asal)
        .include(tujuan)
        .build();

        changeCamera(CameraUpdateFactory.newLatLngBounds(bounds,
50),null, false);
    }

    Override
    public void onCancel() {
    }
}, false);
}

@Override
public void onCancel() {
}
}, true);
}

}

private void changeCamera(CameraUpdate update,
CancelableCallback callback,
boolean instant) {

    if (instant) {
        peta.animateCamera(update, 1, callback);
    } else {
        peta.animateCamera(update, 4000, callback);
    }
}
}

```

**Gambar B.17 Kode Sumber untuk Proses Permintaan Rute.**

```

session.checkLogin();

recordArray = new HashMap<String, String>();

HashMap<String, String> user = session.getUserDetails();

no_telpR = user.get(SessionManagerLogin.KEY_NO_TELP);

nomor_darurat =
user.get(SessionManagerLogin.KEY_NOMOR_PENTING);

pesan_darurat =
user.get(SessionManagerLogin.KEY_PESAN_SINGKAT);

lblNoTelp.setText(Html.fromHtml("No Telp: \n<b>" + no_telpR
+ "</b>"));
    lblNomorDarurat.setText(Html.fromHtml("Nomor
Darurat: \n<b>" + nomor_darurat + "</b>"));
    lblPesanDarurat.setText(Html.fromHtml("Pesan
Darurat: \n<b>" + pesan_darurat + "</b>"));

```

**Gambar B.18 Kode Sumber untuk Proses Pengambilan Nomor dan Pesan Darurat.**

```

location_manager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
LocationListener listner = new MyLocationListner();
location_manager.requestLocationUpdates(LocationManager.NET
WORK_PROVIDER, 1000 * 60 * 5, 100, listner);
location_manager.requestLocationUpdates(LocationManager.GPS
_PROVIDER, 1000 * 60 * 5, 100, listner);

public class MyLocationListner implements LocationListener
{
    @Override
    public void onLocationChanged(Location location) {
        getLatitude = "" + location.getLatitude();
        getLongitude = "" + location.getLongitude();
        Geocoder gcd = new
Geocoder(getApplicationContext(),Locale.getDefault());
        List<Address> addresses;

```

```

try {
    addresses =
gcd.getFromLocation(location.getLatitude(),
                    location.getLongitude(), 1);

    if (addresses.size() > 0) {
        String s = "Posisi Saya di " +
addresses.get(0).getAddressLine(0) + " "
        + addresses.get(0).getLocality() + ", "
        + addresses.get(0).getSubAdminArea() + ", "
        + addresses.get(0).getPostalCode() + ", "
        + addresses.get(0).getCountryName()+ ".";

        address.setText(s);
        alamatLengkap = s;
    }
} catch (IOException e) {
    e.printStackTrace();
}
}

```

Gambar B.19 Kode Sumber untuk Proses Pengambilan Nama Lokasi.

```

//Pesan darurat
protected void sendSMSMessage() {
    Log.i("Send SMS", "");

    String phoneNo = nomor_darurat;
    String message = pesan_darurat + ". " + alamatLengkap;

    try {
        SmsManager smsManager = SmsManager.getDefault();
        smsManager.sendTextMessage(phoneNo, null, message,
null, null);
        Toast.makeText(getApplicationContext(), "SMS
terkirim.", Toast.LENGTH_LONG).show();
    }
    catch (Exception e) {
        Toast.makeText(getApplicationContext(), "SMS gagal,
tolong coba lagi.", Toast.LENGTH_LONG).show();
        e.printStackTrace();
    }
}

```

```

    }
    System.out.println(phoneNo);
    System.out.println(message);
}

```

Gambar B.20 Kode Sumber untuk Pengiriman Pesan Darurat.

```

modeDarurat.setOnCheckedChangeListener(new
OnCheckedChangeListener() {

    @Override
    public void onCheckedChanged(CompoundButton
buttonView,boolean isChecked) {
        if (isChecked) {
            Toast.makeText(getApplicationContext(), "Mode darurat
diaktifkan",Toast.LENGTH_LONG).show();
            Toast.makeText(MenuAplikasi.this,"Pilihan rentang : "
+ "\n" + String.valueOf(spinner_rentang.getSelectedItem())
,Toast.LENGTH_LONG).show();
            int rentang = 0;
            String nilai =
String.valueOf(spinner_rentang.getSelectedItem());
            if (nilai == "1 menit"){rentang = 100000;}
            if (nilai == "5 menit"){rentang = 500000;}
            if (nilai == "15 menit"){rentang = 1500000;}
            if (nilai == "30 menit"){rentang = 3000000;}
            if (nilai == "60 menit"){rentang = 6000000;}

            String phoneNumber = nomor_darurat;
            String message = pesan_darurat + ". Mode Darurat Aktif."
+ " " + alamatLengkap;
            recordArray.put(phoneNumber, message);
            fireAlarm(rentang);
        } else {
            Toast.makeText(getApplicationContext(), "Mode darurat
tidak aktif",
            Toast.LENGTH_LONG).show();
            CancelAlarm();
        }
    }
});

```

```

public void fireAlarm(int rentang) {
    Intent intent = new Intent(context, AlarmReceiver.class);
    intent.setAction("hallo");
    PendingIntent pendingIntent =
        PendingIntent.getBroadcast(MenuAplikasi.this, 0,
            intent, PendingIntent.FLAG_CANCEL_CURRENT);
    Calendar calendar = Calendar.getInstance();
    calendar.setTimeInMillis(System.currentTimeMillis());
    AlarmManager alarm = (AlarmManager) MenuAplikasi.this
        .getSystemService(Context.ALARM_SERVICE);
        alarm.cancel(pendingIntent);
        alarm.setRepeating(AlarmManager.RTC_WAKEUP,
            calendar.getTimeInMillis(),
            rentang, pendingIntent);
    }

public void CancelAlarm()
    {
    Intent intent = new Intent(context, AlarmReceiver.class);
    intent.setAction("hallo");
    PendingIntent pendingIntent =
        PendingIntent.getBroadcast(MenuAplikasi.this, 0, intent,
            PendingIntent.FLAG_CANCEL_CURRENT);
    Calendar calendar = Calendar.getInstance();

    calendar.setTimeInMillis(System.currentTimeMillis());
    AlarmManager alarm = (AlarmManager) MenuAplikasi.this
        .getSystemService(Context.ALARM_SERVICE);
        alarm.cancel(pendingIntent);

    }

```

**Gambar B.21 Kode Sumber untuk Pengiriman Mode Darurat.**

```

class CreateNewKontak extends AsyncTask<String, String,
String> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = new
        ProgressDialog(AddKontakActivity.this);

```



```

        pDialog.setMessage("Creating Kontak..");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(true);
        pDialog.show();
    }
    protected String doInBackground(String... args) {
        String nama = inputNama.getText().toString();
        String no_telp =
inputNo_Telp.getText().toString();
        List<NameValuePair> params = new
ArrayList<NameValuePair>();
        params.add(new BasicNameValuePair("nama_kontak",
nama));
        params.add(new BasicNameValuePair("no_telp_kontak",
no_telp));
        params.add(new BasicNameValuePair("id_pengguna",
no_telpR));
        JSONObject json =
jsonParser.makeHttpRequest(url_create_kontak, "POST",
params);
        Log.d("Create Response", json.toString());
        try {
            int success = json.getInt(TAG_SUCCESS);
            if (success == 1) {
                Intent i = new Intent(getApplicationContext(),
AllKontakView.class);
                startActivity(i);
                finish();
            } else {
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
        return null;
    }
    protected void onPostExecute(String file_url) {
        pDialog.dismiss();
    }
}
}
}

```

Gambar B.22 Kode Sumber untuk Tambah Kontak Pribadi.

```

class SaveKontakDetails extends AsyncTask<String, String,
String> {

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = new
ProgressDialog(EditKontakActivity.this);
        progressDialog.setMessage("Saving kontak ...");
        progressDialog.setIndeterminate(false);
        progressDialog.setCancelable(true);
        progressDialog.show();
    }

    protected String doInBackground(String... args) {
        String nama = txtNama.getText().toString();
        String no_telp = txtNo_Telp.getText().toString();
        List<NameValuePair> params = new
ArrayList<NameValuePair>();
        params.add(new BasicNameValuePair(TAG_ID_KONTAK,
id_kontakedit));
        params.add(new BasicNameValuePair(TAG_NAMA_KONTAK,
nama));
        params.add(new BasicNameValuePair(TAG_NO_TELP_KONTAK,
no_telp));
        JSONObject json =
jsonParser.makeHttpRequest(url_update_kontak, "POST",
params);
        try {
            int success = json.getInt(TAG_SUCCESS);
            if (success == 1) {
                Intent kembali = new
Intent(getApplicationContext(), AllKontakView.class);
                startActivity(kembali);
                finish();
            } else {
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
        return null;
    }
}

```

```

}

protected void onPostExecute(String file_url) {
    progressDialog.dismiss();
}
}
}

```

**Gambar B.23 Kode Sumber untuk Ubah Kontak Pribadi.**

```

class DeleteKontak extends AsyncTask<String, String,
String> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = new
ProgressDialog(DetailKontakActivity.this);
        progressDialog.setMessage("Deleting Kontak...");
        progressDialog.setIndeterminate(false);
        progressDialog.setCancelable(true);
        progressDialog.show();
    }

    protected String doInBackground(String... args) {
        int success;
        try {
            List<NameValuePair> params = new
ArrayList<NameValuePair>();
            params.add(new
BasicNameValuePair("id_kontak", id_kontakdetail));
            JSONObject json =
jsonParser.makeHttpRequest(url_delete_kontak, "POST",
params);

            Log.d("Delete Kontak", json.toString());
            success = json.getInt(TAG_SUCCESS);
            if (success == 1) {
                Intent i = getIntent();
                setResult(100, i);
                Intent kembali = new
Intent(getApplicationContext(), AllKontakView.class);
                startActivity(kembali);
            }
        } catch (Exception e) {
            success = 0;
        }
    }
}

```

```

        finish();
    }else{
        alert.showAlertDialog(getApplicationContext(),
        "Koneksi erorr", "Sepertinya koneksi Anda sedang buruk",
        false);
    }
    } catch (JSONException e) {
        e.printStackTrace();
    }
    return null;
}

protected void onPostExecute(String file_url) {
    pDialog.dismiss();
}
}
}

```

**Gambar B.24 Kode Sumber untuk Hapus Kontak Pribadi.**

```

class CreateNewPrivateKontak extends AsyncTask<String,
String, String> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog = new
        ProgressDialog(AddLayananPublikAdmin.this);
        pDialog.setMessage("Creating Kontak..");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(true);
        pDialog.show();
    }

    protected String doInBackground(String... args) {
        System.out.println(posisi.latitude);
        String nama =
        inputPrivateNama.getText().toString();
        String alamat =
        inputPrivateAlamat.getText().toString();
        String no_telp =
        inputPrivateNo_Telp.getText().toString();
        String latitude =
        Double.toString(posisi.latitude);
    }
}

```

```

        String longitude =
Double.toString(posisi.longitude);

        List<NameValuePair> params = new
ArrayList<NameValuePair>();
        params.add(new
BasicNameValuePair("nama_layanan", nama));
        params.add(new
BasicNameValuePair("alamat_layanan", alamat));
        params.add(new
BasicNameValuePair("no_telp_layanan", no_telp));
        params.add(new BasicNameValuePair("latitude",
latitude));
        params.add(new BasicNameValuePair("longitude",
longitude));
        JSONObject json =
jsonParser.makeHttpRequest(url_create_private_kontak,"POST"
, params);
        Log.d("Create Response", json.toString());
        try {
            int success = json.getInt(TAG_SUCCESS);
            if (success == 1) {
                Intent i = new
Intent(getApplicationContext(),
AllLayananPublikAdmin.class);
                startActivity(i);
                finish();
            } else {

            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
        return null;
    }
    protected void onPostExecute(String file_url) {
        pDialog.dismiss();
    }
}
}
}

```

Gambar B.25 Kode Sumber untuk Tambah Kontak Layanan Publik.

```

class SavePrivateKontakDetails extends AsyncTask<String,
String, String> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog = new
ProgressDialog(EditLayananPublikAdmin.this);
        pDialog.setMessage("Menyimpan layanan
publik...");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(true);
        pDialog.show();
    }
    protected String doInBackground(String... args) {
        String nama_layanan =
txtNamaPrivate.getText().toString();
        String alamat_layanan =
txtAlamatPrivate.getText().toString();
        String no_telp_layanan =
txtNo_TelpPrivate.getText().toString();
        String latitude =
txtLatitudePrivate.getText().toString();
        String longitude =
txtLongitudePrivate.getText().toString();
        List<NameValuePair> params = new
ArrayList<NameValuePair>();
        params.add(new
BasicNameValuePair(TAG_ID_LAYANAN, id_layanan_publik));
        params.add(new
BasicNameValuePair(TAG_NAMA_LAYANAN, nama_layanan));
        params.add(new
BasicNameValuePair(TAG_ALAMAT_LAYANAN, alamat_layanan));
        params.add(new
BasicNameValuePair(TAG_NO_TELP_LAYANAN, no_telp_layanan));
        params.add(new
BasicNameValuePair(TAG_LATITUDE_PRIVATE, latitude));
        params.add(new
BasicNameValuePair(TAG_LONGITUDE_PRIVATE, longitude));
        JSONObject json =
jsonParser.makeHttpRequest(url_update_private_kontak, "POST"
, params);
        try {

```

```

        int success = json.getInt(TAG_SUCCESS);
        if (success == 1) {
            Intent i = getIntent();
            setResult(100, i);
            Intent kembali = new
Intent(getApplicationContext(),AllLayananPublikAdmin.class)
;
            startActivity(kembali);
            finish();
        } else {

        }
    } catch (JSONException e) {
        e.printStackTrace();
    }
    return null;
}

protected void onPostExecute(String file_url) {
    progressDialog.dismiss();
}
}

```

**Gambar B.26 Kode Sumber untuk Ubah Kontak Layanan Publik.**

```

class DeletePrivateKontak extends AsyncTask<String, String,
String> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = new
ProgressDialog(EditLayananPublikAdmin.this);
        progressDialog.setMessage("Menghapus layanan
publik...");
        progressDialog.setIndeterminate(false);
        progressDialog.setCancelable(true);
        progressDialog.show();
    }
    protected String doInBackground(String... args) {
        int success;
        try {

```

```

        List<NameValuePair> params = new
ArrayList<NameValuePair>();
        params.add(new
BasicNameValuePair("id_layanan", id_layanan_public));
        JSONObject json =
jsonParser.makeHttpRequest(url_delete_private_kontak,
"POST", params);
        Log.d("Delete Private Kontak",
json.toString());
        success = json.getInt(TAG_SUCCESS);
        if (success == 1) {
            Intent i = getIntent();
            setResult(100, i);
            Intent kembali = new
Intent(getApplicationContext(),AllLayananPublikAdmin.class)
;
                startActivity(kembali);
                finish();
            }
            else {
                alert.showAlertDialog(getApplicationContext(),
                "Koneksi error", "Sepertinya koneksi Anda sedang buruk",
                false);
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }

        return null;
    }
    protected void onPostExecute(String file_url) {
        pDialog.dismiss();
    }
}
}

```

Gambar B.27 Kode Sumber untuk Hapus Kontak Layanan Publik.



## LAMPIRAN C – HASIL FORMULIR PENILAIAN PENGUJIAN KEGUNAAN

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : .....

Nama : .....

**Lingkungan Pengujian**

Tipe Ponsel : .....

Versi Android : .....

Lokasi Pengguna : .....

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**Gambar C.1 Hasil Pengujian Pengguna 1.**

**Kuesioner Pengujian Kegunaan  
Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat  
Bergerak Berbasis Android**

**Data Responden**

Tanggal : 20 Juli 2024  
 Nama : Kiky Wahyu

**Lingkungan Pengujian**

Tipe Ponsel : SM (10.1)  
 Versi Android : 12  
 Lokasi Pengguna : Kertajaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak  Kurang c. Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang c. Cukup  Sangat

**Gambar C.2 Hasil Pengujian Pengguna 2.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 14-07-2024  
 Nama : .....

**Lingkungan Pengujian**

Tipe Ponsel : .....

Versi Android : .....

Lokasi Pengguna : .....

**A. Antarmuka Pengguna**

- Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah antarmuka aplikasi ini mudah dipahami?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah memiliki tata letak yang baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**B. Pengenalan dan Penyampaian Suara**

- Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah pengucapan respon aplikasi sudah baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**C. Telepon Kostak Pribadi dan Telepon Layanan Publik**

- Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**D. Informasi Penunjukan Lokasi**

- Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**E. Pesan Darurat dan Mode Darurat**

- Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah konten pesan darurat pada aplikasi ini sudah lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**Gambar C.3 Hasil Pengujian Pengguna 3.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal .....  
 Nama .....

**Lingkungan Pengujian**

Tipe Ponsel .....  
 Versi Android .....  
 Lokasi Pengguna .....

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak  Kurang c. Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.4 Hasil Pengujian Pengguna 4.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbahasa Android**

**Data Responden**

Tanggal : 11/07/2014  
 Nama : Yohana F. Kusuma

**Lingkungan Pengujian**

Tipe Ponsel : Samsung Galaxy Tab 3  
 Versi Android : 4.2.2  
 Lokasi Pengguna : Depok, Jawa Barat

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang c. Cukup  Sangat

**Gambar C.5 Hasil Pengujian Pengguna 5.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 19 Juli 2014  
 Nama : Andia Ruzmah

**Lingkungan Pengujian**

Uji Pengguna M  
 Tipe Ponsel : LG  
 Versi Android : 4.2.2  
 Lokasi Pengguna : Bandung, Hutan, Atabuga

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak  Kurang c. Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terdekat untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang  Cukup d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak  Kurang c. Cukup d. Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.6 Hasil Pengujian Pengguna 6.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 30 Juli 2024  
 Nama : Nodia Rajad

**Lingkungan Pengujian**

Tipe Ponsel : Samsung Note 3  
 Versi Android : 4.2  
 Lokasi Pengguna : Pungkit Acri Tengah Surabaya

**A. Antarmuka Pengguna**

- Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah antarmuka aplikasi ini mudah dipahami?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah memiliki tata letak yang baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**B. Pengenalan dan Penyampaian Suara**

- Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah pengucapan respon aplikasi sudah baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

- Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**D. Informasi Penunjukan Lokasi**

- Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**E. Pesan Darurat dan Mode Darurat**

- Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah konten pesan darurat pada aplikasi ini sudah lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**Gambar C.7 Hasil Pengujian Pengguna 7.**

**Kuesioner Pengujian Kegunaan**  
**Apakah Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 20/11/2019  
 Nama : FERTAJA VA

**Lingkungan Pengujian**

Tipe Ponsel : HONOR  
 Versi Android : 7.0  
 Lokasi Pengguna : KERTAJAYA

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup  Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang  Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang  Cukup d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.8 Hasil Pengujian Pengguna 8.**



**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 10 Juli 2019  
 Nama : Rizka Prasetya

**Lingkungan Pengujian**

Tipe Ponsel : Sony Xperia Tipe  
 Versi Android : 4.4  
 Lokasi Pengguna : Kerinci Kutabaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c. Cukup  d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c. Cukup  d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak b. Kurang c. Cukup  d. Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang c. Cukup  d. Sangat

**Gambar C.9 Hasil Pengujian Pengguna 9.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 27 Juli 2021  
 Nama : Rizki Nur Hafidha

**Lingkungan Pengujian**

Tipe Ponsel : Xiaomi A10  
 Versi Android : 11.0  
 Lokasi Pengguna : Surabaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang c. Cukup  Sangat

**Gambar C.10 Hasil Pengujian Pengguna 10.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : .....

Nama : .....

**Lingkungan Pengujian**

Tipe Ponsel : .....

Versi Android : .....

Lokasi Pengguna : .....

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
c. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
d. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
c. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
b. Tidak b. Kurang c. Cukup  Sangat

**Gambar C.11 Hasil Pengujian Pengguna 11.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 21.03.2019  
 Nama : Ahmad Alimudin

**Lingkungan Pengujian**

Tipe Ponsel : Samsung I9692  
 Versi Android : 7.0  
 Lokasi Pengguna : Surabaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang  Cukup d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang  Cukup d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.12 Hasil Pengujian Pengguna 12.**

**Kuesioner Pengujian Kegunaan  
Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat  
Bergerak Berbasis Android**

**Data Responden**

Tanggal : .....  
 Nama : .....

**Lingkungan Pengujian**

Tipe Ponsel : .....  
 Versi Android : .....  
 Lokasi Pengguna : .....

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.13 Hasil Pengujian Pengguna 13.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 18 Juli 2024  
 Nama : Ahsanul

**Lingkungan Pengujian**

Tipe Ponsel : Samsung  
 Versi Android : 11.0  
 Lokasi Pengguna : Surabaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang  Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang  Cukup d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.14 Hasil Pengujian Pengguna 14.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 14.05.2024  
 Nama : M.A.T.A. C.A.S.T.A

**Lingkungan Pengujian**

Tipe Ponsel : SAMSUNG GALAXY S23 FE  
 Versi Android : 14  
 Lokasi Pengguna : K.S.R.T.A.J.A

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c. Cukup  Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 a. Tidak b. Kurang c. Cukup  Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 a. Tidak b. Kurang c. Cukup  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat
6. Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 a. Tidak b. Kurang c. Cukup  Sangat

**Gambar C.15 Hasil Pengujian Pengguna 15.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : .....  
 Nama : .....

**Lingkungan Pengujian**

Tipe Ponsel : .....  
 Versi Android : .....  
 Lokasi Pengguna : .....

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang c.  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c. Cukup d.  Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c. Cukup d.  Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c.  Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c. Cukup d.  Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c. Cukup d.  Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang c. Cukup d.  Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c.  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang c.  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c.  Cukup d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang c. Cukup d.  Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang c.  Cukup d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang c.  Cukup d. Sangat

**Gambar C.16 Hasil Pengujian Pengguna 16.**



**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat**  
**Bergerak Berbasis Android**

**Data Responden**

Tanggal : 21 Juli 2019  
 Nama : Muryanti

**Lingkungan Pengujian**

Tipe Ponsel : Galaxy S6  
 Versi Android : 4.2  
 Lokasi Pengguna : Surabaya

**A. Antarmuka Pengguna**

- Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah antarmuka aplikasi ini mudah dipahami?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah memiliki tata letak yang baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**B. Pengenalan dan Penyampaian Suara**

- Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah pengucapan respon aplikasi sudah baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

- Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**D. Informasi Penunjukan Lokasi**

- Apakah aplikasi ini sudah menampilkan rute terdekat untuk menuju lokasi pilihan Anda?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**E. Pesan Darurat dan Mode Darurat**

- Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Apakah konten pesan darurat pada aplikasi ini sudah lengkap?
  - Tidak
  - Kurang
  - Cukup
  - Sangat
- Pada mode darurat, apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?
  - Tidak
  - Kurang
  - Cukup
  - Sangat

**Gambar C.17 Hasil Pengujian Pengguna 17.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 27 Juli 2024  
 Nama : Nurhidayah Diah

**Lingkungan Pengujian**

Tipe Ponsel : Samsung S23  
 Versi Android : 14  
 Lokasi Pengguna : Surabaya

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang c.  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang c.  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak b. Kurang c. Cukup d.  Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang c.  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak c.  Kurang c. Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang c.  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang c.  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang c.  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup d.  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang c. Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak c.  Kurang c. Cukup d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b.  Kurang c. Cukup d. Sangat

**Gambar C.18 Hasil Pengujian Pengguna 18.**

**Kuesioner Pengujian Kegunaan**  
**Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android**

**Data Responden**

Tanggal : 2019 - 2019  
 Nama : Cahya R. A.

**Lingkungan Pengujian**

Tipe Posel : 2019  
 Versi Android : 8.1  
 Lokasi Pengguna :

**A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?  
 a. Tidak  Kurang c. Cukup d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?  
 a. Tidak b. Kurang  Cukup d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?  
 a. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?  
 a. Tidak b. Kurang  Cukup d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?  
 c. Tidak b. Kurang  Cukup d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?  
 b. Tidak b. Kurang  Cukup d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?  
 a. Tidak b. Kurang c. Cukup  Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
 d. Tidak b. Kurang  Cukup d. Sangat
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
 c. Tidak b. Kurang  Cukup d. Sangat
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
 b. Tidak b. Kurang  Cukup d. Sangat

**Gambar C.19 Hasil Pengujian Pengguna 19.**

*[Halaman ini sengaja dikosongkan]*

## LAMPIRAN D – FORMULIR PENILAIAN PENGUJIAN KEGUNAAN

### Kuesioner Pengujian Kegunaan Aplikasi Panggilan Darurat dengan Perintah Suara Berbahasa Indonesia pada Perangkat Bergerak Berbasis Android

#### Data Responden

Tanggal : .....

Nama : .....

#### Lingkungan Pengujian

Tipe Ponsel : .....

Versi Android : .....

Lokasi Penggunaan : .....

#### **A. Antarmuka Pengguna**

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?  
a. Tidak    b. Kurang    c. Cukup    d. Sangat
2. Apakah antarmuka aplikasi ini mudah dipahami?  
b. Tidak    b. Kurang    c. Cukup    d. Sangat
3. Apakah aplikasi ini sudah memiliki tata letak yang baik?  
a. Tidak    b. Kurang    c. Cukup    d. Sangat

#### **B. Pengenalan dan Penyampaian Suara**

1. Apakah aplikasi ini sudah dapat mengenali suara anda dengan baik?  
a. Tidak    b. Kurang    c. Cukup    d. Sangat

2. Apakah aplikasi ini memberikan respon yang sesuai dengan perintah yang Anda maksud?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah pengucapan respon aplikasi sudah baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**C. Telepon Kontak Pribadi dan Telepon Layanan Publik**

1. Apakah aplikasi ini melakukan panggilan kontak pribadi akurat sesuai perintah Anda?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah aplikasi ini memberikan informasi yang akurat tentang posisi layanan publik terdekat?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah aplikasi ini memberikan informasi layanan publik yang lengkap?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**D. Informasi Penunjukan Lokasi**

1. Apakah aplikasi ini sudah menampilkan rute terpendek untuk menuju lokasi pilihan Anda?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
2. Apakah aplikasi ini sudah dapat menunjukkan lokasi pilihan Anda dengan baik?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat
3. Apakah rute yang disarankan oleh aplikasi sudah sesuai dengan keadaan sebenarnya?
  - a. Tidak
  - b. Kurang
  - c. Cukup
  - d. Sangat

**E. Pesan Darurat dan Mode Darurat**

4. Apakah aplikasi ini sudah menampilkan alamat posisi Anda secara akurat?  
d. Tidak    b. Kurang    c. Cukup    d. Sangat
  
5. Apakah konten pesan darurat pada aplikasi ini sudah lengkap?  
c. Tidak    b. Kurang    c. Cukup    d. Sangat
  
6. Pada mode darurat, Apakah konten pesan posisi Anda menunjukkan perubahan ketika posisi Anda telah berpindah tempat?  
b. Tidak    b. Kurang    c. Cukup    d. Sangat