APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID

SKRIPSI

Disusun Oleh:

Danan Ekatama Khusna

1018157

MILIK
PERPUSTAKAAN
ITN MALANG

PROGRAM STUDI TEKNIK INFORMATIKA S-1

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

2014

MILIK
PERPUSTAKAAN
UTM MALANG

LEMBAR PERSETUJUAN DAN PENGESAHAN

APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID

SKRIPSI

Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna mencapai Gelar Sarjana Teknik Informatika Strata Satu (S-1)

Disusun Oleh :
Danan Ekatama Khusna
1018157

Diperiksa dan Disetujui,

Dosen Pembimbing I	Dosen Pembimbing II
Ir. Yusuf Ismail Nakhoda, MT	**Febriana Santi W, SKom, MKom**
NIP. Y. 1018800189	**NIP.P. 1031000425**

Ketua Program Studi Teknik Informatika S-1

Joseph Dedy Irawan,ST, MT
NIP. 197404162005031002

PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG

INSTITUTE TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PPROGRAM STUDI TEKNIK INFORMATIKA S-1

BERITA ACARA UJIAN SKRIPSI

Nama : Danan Ekatama Khusna
NIM : 1018157
Jurusan : Teknik Informatika S-1
Judul : Aplikasi Pengenalan Rambu Lalulintas Berbasis Android

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Kamis
Tanggal : 21 Agustus 2014
Tempat : Ruang Lab Robotik Teknik Informatika S-1
Nilai : (A)

Panitia Ujian Skripsi :

Ketua Penguji

Joseph Dedy Irawan, ST, MT
NIP. 197404162005011002

Anggota Penguji :

Penguji I	**Penguji II**
Ali Mahmudi, BEng, Phd	**Sonny Prasetio, ST, MT**
NIP.P. 1031000429	**NIP.P. 1031000433**

MOTO DAN PERSEMBAHAN

Keberhasilan tidak datang secara tiba-tiba, tapi karena usaha dan kerjakeras.

Pengalaman tidak dapat dibeli dengan uang.

Belajarlah dari pengalaman yang telah kita dapat.

Manfaatkanlah waktu dengan sebaik-baiknya.

PERSEMBAHAN

- Ayah dan Ibu tercinta yang selalu mendukung keberhasilanku, serta memberikan do'a cinta dan kasih sayang, pikiran tenaga, waktu, serta materi yang tidak bisa terbalas sekalipun dengan nyawaku.
- Bapak ibu dosen yang selama ini telah memberikan ilmu yang sangat berguna.
- Teman-teman seperjuangan yang tidak bisa disebutkan satu persatu.

KATA PENGANTAR

Puji syukur kehadirat Allah yang maha kuasa, karena telah memberikan rahmat dan hidayah-Nya sehingga penyusun dapat menyelesaikan skripsi dengan judul APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID sesuai dengan waktu yang ditentukan.

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada penyusunan skripsi ini kami mengucapkan banyak terima kasih sebesar-besarnya kepada :

1. Bapak Sutomo dan Ibu Sun Rofi'ah, yang merupakan kedua orang tua dan pendukung utama dari segi moril maupun materiil.
2. Ir. Soeparno Djiwo, MT, selaku Rektor Institut Teknologi Nasional Malang.
3. Ir. Anang Subardi, MT, selaku Dekan Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.
4. Joseph Dedy Irawan, ST, MT, selaku Ketua Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
5. Sonny Prasetio, ST, MT, selaku Sekertaris Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
6. Ir. Yusuf Ismail Nakhoda, MT, selaku Dosen Pembimbing I, yang selalu memberikan masukkan.
7. Febriana Santi W, Skom, Mkom, selaku Dosen Pembimbing II dan sekaligus Kepala Lab Animasi dan Game.
8. Semua dosen Program Studi Teknik Informatika yang telah membantu dalam penulisan dan masukan.
9. Semua teman seperjuangan yang telah membantu dalam terselesaikannya skripsi ini.

Penyusun menyadari bahwa skripsi masih jauh dari sempurna, oleh karena itu penyusun mengharapkan kritik dan saran dari pembaca, Semoga skripsi ini bisa bermanfaat bagi pembaca.

Malang, 20 Agustus 2014

Penyusun

DAFTAR ISI

HALAMAN JUDUL i
LEMBAR PERSETUJUAN DAN PENGESAHAN ii
BERITA ACARA iii
MOTO DAN PERSEMBAHAN iv
KATA PENGANTAR v
DAFTAR ISI vii
DAFTAR GAMBAR ix
DAFTAR TABEL x
SURAT PERNYATAAN xi
ABSTRAK xii

BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan 2
1.4 Batasan Masalah 2
1.5 Metode Penelitian 2
1.6 Sistematika Penulisan 3

BAB II LANDASAN TEORI 5
2.1 Rambu Lalulintas 5
2.1.1 Rambu Peringatan 6
2.1.2 Rambu Larangan 10
2.1.3 Rambu Perintah 13
2.1.4 Rambu Petunjuk 15
2.1.5 Rambu Tambahan 19
2.2 Undang-undang Lalulintas 20
2.3 *Autocomplete* 22
2.4 *Sistem Operasi Android* 23
2.4.1 *Platform* Masa Depan 24
2.4.2 *The Dalvik Virtual Machine* (DVM) 26
2.4.3 Android SDK (*SoftwareDevelopment Kit*) 27
2.4.4 ADT (*Android Development Tools*) 28
2.4.5 Arsitektur Android 29
2.4.6 Fundamental Aplikasi 31
2.4.7 Versi Android 32
2.5 *Eclipse* 35

BAB III ANALISIS DAN PERANCANGAN SISTEM 36
3.1 Analisa Kebutuhan 36
3.1.1 Analisa Kebutuhan Fungsional 36
3.1.2 Analisa Kebutuhan Non Fungsional 37
3.2 Perancangan Aplikasi 38
3.2.1 Perancangan Sistem 38
3.2.2 Perancangan *Flowchart* Aplikasi 39

3.2.2 Perancangan *Flowchart* Kuis 40
3.2.2 Perancangan *Struktur Navigasi* Aplikasi 41
3.3 Perancangan Halaman Aplikasi 43
3.3.1 Halaman Pembuka (*splash*) 43
3.3.2 Halaman Menu Utama 44
3.3.3 Halaman Menu Tentang Rambu 45
3.3.4 Halaman Menu Jenis Rambu 45
3.3.5 Halaman Menu Undang-undang Lalulintas 48
3.3.6 Halaman Menu Kuis 49

BAB IV IMPLEMENTASI DAN PENGUJIAN 51
4.1 Implementasi Sistem 51
4.2 Implementasi User Interface 51
4.2.1 Tampilan Menu *Splash Screen* 52
4.2.2 Tampilan Menu Utama 52
4.2.3 Tampilan Menu Tentang Rambu 53
4.2.4 Tampilan Menu Jenis Rambu 54
4.2.5 Tampilan Menu Rambu Larangan 55
4.2.6 Tampilan Menu Rambu Peringatan 56
4.2.7 Tampilan Menu Rambu Perintah 57
4.2.8 Tampilan Menu Rambu Petunjuk 58
4.2.9 Tampilan Menu Rambu Tambahan 59
4.2.10 Tampilan Menu Undang-undang Lalulintas 60
4.2.11 Tampilan Menu Kuis 61
4.3 Pengujian 62
4.3.1 Pengujian Fungsional Sistem 62
4.3.2 Pengujian Spesifikasi *Hardware* 64
4.3.3 Pengujian *Operating System Android* 64
4.3.3 Pengujian *User* 64

BAB V PENUTUP 66
5.1 Kesimpulan 66
5.2 Saran 66

DAFTAR PUSTAKA 67
LAMPIRAN 68

DAFTAR GAMBAR

Gambar 2.1 Rambu Peringatan 7
Gambar 2.2 Rambu Larangan 11
Gambar 2.3 Rambu Perintah 14
Gambar 2.4 Rambu Petunjuk 16
Gambar 2.5 Rambu Tambahan 19
Gambar 2.6 Cara Kerja *Autocomplete* 23
Gambar 2.7 Arsitektur Android 31
Gambar 3.1 Blok diagram 38
Gambar 3.2 *Flowchart* Aplikasi 49
Gambar 3.3 *Flowchart* Kuis 41
Gambar 3.4 Struktur Navigasi *User* 42
Gambar 3.5 Desain Halaman Pembuka (*splash*) 44
Gambar 3.6 Desain Halaman Menu Utama 44
Gambar 3.7 Desain Halaman Menu Tentang Rambu 45
Gambar 3.8 Desain Halaman Menu Jenis Rambu 46
Gambar 3.9 Desain Halaman Menu Rambu Larangan 46
Gambar 3.10 Desain Halaman Menu Rambu Peringatan 47
Gambar 3.11 Desain Halaman Menu Rambu Perintah 47
Gambar 3.12 Desain Halaman Menu Rambu Petunjuk 48
Gambar 3.13 Desain Halaman Menu Rambu Tambahan 48
Gambar 3.14 Desain Halaman Menu Undang-undang Lalu Lintas 49
Gambar 3.15 Desain Halaman Kuis 49
Gambar 3.16 Desain Halaman Nilai Hasil 50
Gambar 4.1 Tampilan *SplashScreen* 52
Gambar 4.2 Tampilan Menu Utama 53
Gambar 4.3 Tampilan *About* 53
Gambar 4.4 Tampilan Menu Tentang Rambu 54
Gambar 4.5 Tampilan Menu Jenis Rambu 54
Gambar 4.6 Tampilan Rambu Larangan 55
Gambar 4.7 Keterangan Rambu Larangan 55
Gambar 4.8 Tampilan Rambu Peringatan 56
Gambar 4.9 Tampilan Rambu Peringatan 56
Gambar 4.10 Tampilan Rambu Perintah 57
Gambar 4.11 Keterangan Rambu Perintah 57
Gambar 4.12 Tapilan Rambu Petunjuk 58
Gambar 4.13 Keterangan Rambu Petunjuk 58
Gambar 4.14 Tampilan Rambu Tambahan 59
Gambar 4.15 Keterangan Rambu Tambahan 59
Gambar 4.16 Keterangan Undang-undang Lalulintas 60
Gambar 4.17 Gambar Keterangan Undang-undang 60
Gambar 4.18 Keterangan Hasil Pencarian Pasal 61
Gambar 4.19 Gambar Tampilan Soal 61
Gambar 4.20 Nilai Hasil dari Soal 62

DAFTAR TABEL

Tabel 2.1 Keterangan Rambu Peringatan 8
Tabel 2.2 Jarak Penempatan Rambu Peringatan 10
Tabel 2.3 Keterangan Rambu Larangan 11
Tabel 2.4 Keterangan Rambu Perintah 14
Tabel 2.5 Keterangan Rambu Petunjuk 16
Tabel 2.6 Keterangan Rambu Tambahan 19
Tabel 4.1 Tabel Pengujian Fungsional Sistem Aplikasi Pengenalan Rambu Laluintas Dalam Sistem *Android* 63
Tabel 4.2 Hasil Pengujian sistem kepada pengguna *(user)*. 65

LEMBAR KEASLIAN

PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama	: Danan Ekatama Khusna
Nim	: 10.18.157
Program Studi	: Teknik Informatika S-1
Fakultas	: Teknologi Industri

Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul:

"APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID"

Adalah Skripsi saya sendiri bukan duplikat serta mengutip atau menyadur seluruhnya karya orang lain kecuali dari sumber aslinya.

Malang, 20 Agustus 2014

Yang Membuat Pernyataan

Danan Ekatama Khusna

APLIKASI PENGENALAN RAMBU LALULINTAS BERBASISANDROID

Danan Ekatama Khusna (1018157)
Program Studi Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Jl.Raya Karanglo Km. 2 Tasikmadu-Malang
Email : ekatama51@gmail.com

Dosen Pembimbing : I. Ir. Yusuf Ismail Nakhoda, MT
II. Febriana Santi W, SKom, MKom

Abstrak

Rambu lalu lintas merupakan salah satu dari perlengkapan jalan, berupa lambang, huruf, angka, kalimat atau perpaduan diantaranya sebagai peringatan, larangan, perintah atau petunjuk bagi pemakai jalan. Perkembangan teknologi komunikasi masa kini berkembang sangat pesat dan mampu menjadikan informasi sangat mudah didapatkan. Teknologi mobile yang berbasis multimedia mampu menjadi solusi mengatasi masalah tentang kurangnya pengetahuan anak-anak terhadap rambu rambu lalu lintas.

Aplikasi pengenalan rambu lalulintas adalah aplikasi yang berfungsi sebagai pengenalan mengenai berbagai macam rambu lalu lintas yang ada di jalan, terutama sebagai pembelajaran kepada anak agar lebih mengetahui tentang gambar, simbol, atau tulisan peringatan yang ada di jalan raya. Aplikasi pengenalah rambu lalulintas dibuat dari tahap pengumpulan data, merancang tampilan, lalumelakukan pengkodean programpada Eclipse, selanjutnya dilakukan kompilasi dan implementasi pada ponsel Android.

Pada aplikasi pengenalan rambu lalulintas berbasis android ini juga disediakan beberapa fitur seperti menampilkan suara, pencarian pasal undang-undang lalulintas dan kuis, untuk kuis menggunakan pengacakan soal agar soal-soal yang muncul dapat berubah. Hasil pengujian aplikasi ini meliputi pengujian kinerja aplikasi mencapai 100 %. Hasil pengujian aplikasi ini meliputi pengujian kinerja aplikasi beserta layoutnya, pengujian operating system, dan pengujian user. Untuk pengujian user menunjukkan bahwa dari 20 user yang memilih baik sebanyak 86 %, dan yang memilih cukup sebanyak 14 %, dan 0 % untuk yang memiih kurang. Aplikasi pengenalan rambu lalulintas berbasis android ini dapat digunakan kedalam system operasi Android antara versi Frozen Yogurt/Froyo (2.2) hingga versi Kitkat (4.4). sedangkan ukuran file .APK alikasi Rambu Lalulintas ini sebesar 9,67 MB.

***Kata kunci** : Pembelajaran, Android, Rambu lalulintas*

BAB I
PENDAHULUAN

1.1. Latar Belakang

Saat ini perangkat handphone sudah memiliki banyak fungsi, tidak hanya untuk melakukan telekomunikasi atau sms. Namun handphone saat ini bisa mencari informasi di internet, bermain game, atau dapat juga dijadikan sebagai sarana multimedia, edukasi atau pembelajaran. Dengan berkembangnya teknologi saat ini khususnya pembelajaran pada anak-anak, banyak metode-metode yang sudah digunakan untuk menarik rasa suka dan minat anak-anak agar tetap mau belajar, adapun beberapa contoh seperti game-game mini untuk belajar matematika, mewarnai, pengetahuan umum dan lain sebagainya.

Rambu lalu lintas adalah salah satu dari perlengkapan jalan, berupa lambang, huruf, angka, kalimat atau perpaduan diantaranya sebagai peringatan, larangan, perintah atau petunjuk bagi pemakai jalan. Rambu lalu lintas yang sering kita lihat di jalan raya atau jalan tol berfungsi sebagai pengatur jalannya lalu lintas agar lalu lintas berjalan lancar. Pada jalan tol terdapat batasan-batasan toleransi kecepatan kendaraan maksimum yang bertujuan agar pengendara melaju pada batas maksimum yang ditentukan, sehingga pengguna jalan tidak semaunya sendiri ketika berkendara di jalan raya.

Untuk menanggulangi masalah tersebut masyarakat harus mentaati dan mengetahui tanda dari rambu lalu lintas yang ada. Saat ini masyarakat mengenai rambu lalu lintas bisa dibilang sangatlah minim atau sedikit sekali yang mengetahui rambu lalu lintas, di karenakan kurangnya sarana informasi atau kurangnya akan keingintahuan mengenal rambu lalulintas.

Aplikasi ini dibuat agar dapat membantu para pengguna memberikan pengetahuan mengenai berbagai macam rambu lalu lintas yang ada di jalan, terutama sebagai pembelajaran kepada anak agar lebih mengetahui tentang gambar, simbol, atau tulisan peringatan yang ada di jalan, dan mudah untuk di pahami karena menggunakan fasilitas telepon seluler, jadi bisa di pelajari kapan dan dimana saja.

1.2. Rumusan Masalah

Berdasarkan pada latar belakang diatas, maka didapat rumusan masalah yang akan dibahas yaitu bagaimana membuat apliikasi yang dapat digunakan sebagai pengenalan tentang rambu lalu lintas, maupun undang-undang rambu lalu lintas kepada anak yang berusia antara 4 sampai 7 tahun.

1.3. Tujuan

Adapun yang menjadi tujuan dalam penyusunan skripsi ini adalah membangun sebuah apliikasi yang digunakan sebagai media pengenalan atau pembelajaran tentang rambu lalu lintas berbasis *Android* menggunakan *Eclipse.*

1.4. Batasan Masalah

Agar pembahasan dalam skripsi ini tidak meluas, maka ditentukan beberapa batasan masalah diantaranya, yaitu :

1. Aplikasi dibuat dengan menggunakan bahasa pemrograman *java.*
2. Apliikasi ini hanya menampiikan gambar dan suara penjelasan dari rambu lalu lintas beserta beberapa soal latihan.
3. Sasaran usia pengguna 4 sampai 7 tahun.
4. Rambu lalulintas terdiri dari 5 jenis rambu, yaitu rambu larangan, rambu peringatan, rambu perintah, rambu petunjuk, dan rambu tambahan.

1.5. Metode Penelitian

Pada pembuatan Skripsi ini terdapat beberapa tahapan yang dilakukan yaitu tahap studi literatur, pengumpulan data, analisa dan desain sistem, pembangunan aplikasi, pengujian, dan kesimpulan.

1. Studi Literatur

   Penelitian ini dilakukan dengan mempelajari jurnal, buku-buku baik dari perpustakaan, internet maupun milik pribadi.
2. Analisa Kebutuhan

a. Fungsional

b. Non fungsional

3. Perancangan

   Pada tahap ini dilakukan perancangan berdasarkan analisis yang telah di peroleh untuk diimplementasikan ke dalam aplikasi dalam bentuk layout dan tool-tool yang diperlukan untuk mendukung perancangan aplikasi.

4. Implementasi

   Pada tahap ini dilakukan proses implementasi sistem berdasarkan hasil perancangan. Proses implementasi berhubungan dengan proses pengkodingan program berdasarkan bahasa pemrograman yang telah ditentukan sebelumya.

5. Pengujian

   Pada tahap ini dilakukan proses pengujian berdasarkan hasil dari proses implementasi, pengujian aplikasi ini dilakukan untuk mengetahui apakah aplikasi berjalan sesuai yang diharapkan atau tidak.

1.6. Sistematika Penulisan

Dalam menyusun skripsi ini menggunakan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metode penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Berisi landasan teori dari beberapa literatur yang berhubungan dengan topik skripsi ini.

BAB III ANALISIS DAN PERANCANGAN SISTEM

Berisi uraian mengenai perancangan aplikasi disertai dengan struktur menu, flowchart dan layout perancangan aplikasi.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Berisi mengenai implementasi dengan menjabarkan langkah-langkah yang disetai *screenshot* pembuatan aplikasi dan melakukan pengujian aplikasi.

BAB V PENUTUP

Berisi kesimpulan dan saran. Kesimpulan didapat dari ulasan data – data penelitian, menyimpulkan bukti – bukti yang dapat diperoleh dari hasil pengujian dan akhirnya menarik intisari apakah hasil yang didapat (dikerjakan), layak untuk digunkan (diimplementasikan).

BAB II
LANDASAN TEORI

2.1 Rambu Lalu lintas

Menurut faisal afandi rambu lalu lintas jalan merupakan alat pengendali lalu lintas untuk menyampaikan informasi berupa peringatan, larangan, perintah, atau petunjuk dengan tujuan untuk menjaga keamanan, ketertiban, kelancaran dan kenyamanan bagi pengguna jalan. Terdapat 3 (tiga) bentuk informasi yang masing – masing memiliki arti dan makna tersendiri, antara lain :

1. Makna perintah dan larangan merupakan informasi yang harus dipatuhi dan ditaati.
2. Peringatan merupakan informasi terhadap suatu kondisi di lingkungan lalu lintas, seperti suatu kondisi jalan yang rawan, bahaya dll.
3. Petunjuk merupakan informasi lokasi, tempet, jarak, fasilitas – fasilitas dll.

Keberadaan salah satu alat pelengkap jalan tersebut adalah penting dan harus dipasang dengan memenuhi unsur – unsur yaitu :

1. Dibuat berdasarkan spek dan desain teknis yang telah ditetapkan serta memenuhi suatu kebutuhan tertentu.
2. Dipasang dengan benar dan tidak terhalang sehingga dapat dengan jelas terlihat dan terbaca oleh para pengguna jalan
3. Menyampaikan suatu informasi yang jelas dan sederhana.
4. Informasi perintah dan larangannya harus dipatuhi oleh pengguna jalan.

Terdapat beberapa jenis lalu lintas sesuai kebutuhan dan fungsinya, antara lain :

1. Perintah, yaitu bentuk pengaturan yang jelas dan tegas tanpa ada arti dan interpretasi lain kecuali yang disampaikan oleh rambu larangan tersebut. Karena sifatnya perintah, maka tidak benar jika ada interpretasi lain. Semua perintah rambu tersebut harus dipatuhi oleh pengguna jalan.
2. Larangan, yaitu bentuk pengaturan yang jelas dan tegas. Sama halnya dengan rambu perintah, rambu larangan dalam ruang lalu lintas jalan, seperti dilarang parker, diiarang berhenti, dll.
3. Peringatan, menunjukkan kemungkinan adanya bahaya atau hal – hal yag harus diwaspadai oleh pengguna jalan.

4. Petunjuk, merupakan rambu panduan bagi para pengguna jalan, seperti rambu petunjuk arah, jurusan, jarak, kondisi, lokasi, tempat beribadah, rumah makan, dll.
5. Rambu papan tambahan, adalah papan yang dipasang dibawah daun rambu yang memberikan penjelasan lebih lanjut dari suatu rambu.

Untuk memenuhi kebutuhan tersebut, pertimbangan-pertimbangan yang harus diperhatikan dalam perencanaan dan pemasangan rambu adalah

1. Keseragaman bentuk dan ukuran

   Keseragaman dalam alat kontrol lalu lintas memudahkan tugas pengemudi untuk mengenal, memahami dan memberikan respon. Konsistensi dalam penerapan bentuk dan ukuran rambu akan menghasilkan konsistensi persepsi dan respon pengemudi.
2. Desain rambu

   Warna, bentuk, ukuran, dan tingkat retroreflesi yang memenuhi standar akan menarik perhatian pengguna jalan, mudah dipahami dan memberikan waktu yang cukup bagi pengemudi dalam memberikan respon.
3. Lokasi rambu

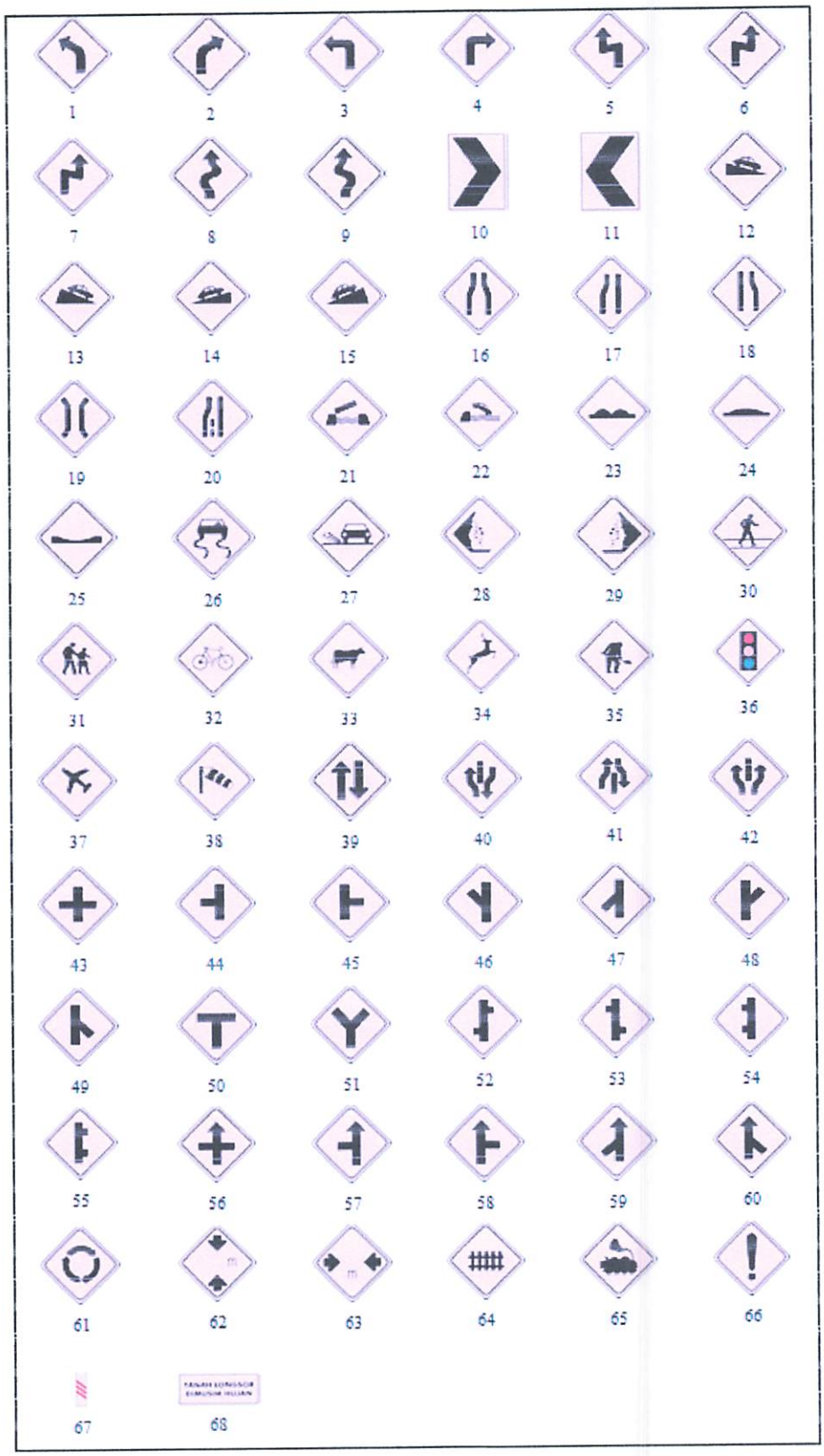
   Lokasi rambu berhubungan dengan pengemudi sehingga pengemudi yang berjalan dengan kecepatan normal dapat memiliki waktu yang cukup dalam memberikan respon.
4. Operasi rambu

   Rambu yang benar pada lokasi yang tepat harus memenuhi kebutuhan lalu lintas dan diperlukan pelayanan yang konsisten dengan memasang rambu yang sesuai kebutuhan.
5. Pemeliharaan rambu

   Pemeliharaan rambu diperlukan agar rambu tetap berfungsi baik.

2.1.1. Rambu Peringatan

Rambu peringatan digunakan untuk memberi peringatan kemungkinan ada bahaya atau tempat berbahaya di depan pengguna jalan. Warna dasar rambu peringatan berwarna kuning dengan lambang atau tulisan berwarna hitam. Gambar-gambar dari rambu peringatan ditunjukkan oleh gambar 2.1.



Gambar 2.1 Rambu Peringatan

Tabel 2.1. Keterangan Rambu Peringatan

No	Keterangan
1	Tikungan kekiri
2	Tikungan ke kanan
3	Tikungan tajam kekiri
4	Tikungan tajam ke kanan
5	Tikungan ganda, tikungan pertama ke kiri
6	Tikungan ganda, tikungan pertama ke kanan
7	Tikungan, tikungan pertama ke kanan
8	Banyak tikungan atau urutan beberapa tikungan, tikungan pertama ke kiri
9	Banyak tikungan atau urutan beberapa tikungan, tikungan pertama ke kanan
10	Pengarah tikungan ke kanan
11	Pengarah tikungan ke kiri
12	Turunan
13	Turunan curam
14	Tanjakan
15	Tanjakan terjal
16	Penyempitan di kiri dan kanan jalan
17	Penyempitan di kiri jalan
18	Penyempitan di kanan jalan
19	Jembatan atau Penyempitan di jembatan
20	Pengurangan lajur kiri
21	Jembatan angkat
22	Jalan menuju tepian air, tepian jurang
23	Jalan tidak datar, bergelombang bergelombang atau berbukit-bukit
24	Jalan cembung atau jembatan cembung
25	Jalan cekung
26	Jalan licin
27	Jalan kerikil lepas
28	Longsoran tanah atau batu yang berjatuhan disebelah kiri jalan
29	Longsoran tanah atau batu yang berjatuhan disebelah kanan jalan

30	Penyeberangan orang
31	Banyak anak-anak
32	Banyak orang bersepeda dan sering menyeberang jalan
33	Banyak satwa jinak yang sering menyeberang jalan
34	Banyak satwa liar dan sering menyeberang jalan
35	Ada pekerja jalan
36	Lampu pengatur lalu lintas
37	Lintasan pesawat terbang
38	Ada Angin dari samping
39	Lalu lintas dua arah
40	Awal bangunan pemisah untuk lalu lintas dua arah
41	Akhir bangunan pemisah untuk lalu lintas dua arah
42	Awal bangunan pemisah untuk lalu lintas dua arah
43	Persimpang empat
44	Persimpang tiga sisi kiri
45	Persimpang tiga sisi kanan
46	Persimpang tiga serong ke kiri
47	Persimpang tiga serong kiri
48	Persimpang tiga serong ke kanan
49	Persimpang tiga serong kanan
50	Persimpang tiga type T
51	Persimpang tiga type Y
52	Persimpang ganda kanan kiri
53	Persimpang ganda kanan kiri
54	Persimpang ganda kiri
55	Persimpang ganda kanan
56	Persimpang tiga dengan prioritas
57	Persimpang tiga sisi kiri dengan prioritas
58	Persimpang tiga sisi kanan dengan prioritas
59	Persimpang tiga serong kiri dengan prioritas
60	Persimpang tiga serong kanan dengan prioritas
61	Persimpang bundaran dengan perioritas

62	Tinggi ruang bebas ... M
63	Lebar ruang bebas ... M
64	Persilang datar dengan lintasan kereta api berpintu
65	Persilangan datar dengan lintasan kereta api tanpa pintu
66	Hati-hati
67	Rambu tambahan menyatakan jarak 450 m
68	Peringatan tanah longsor di musim hujan

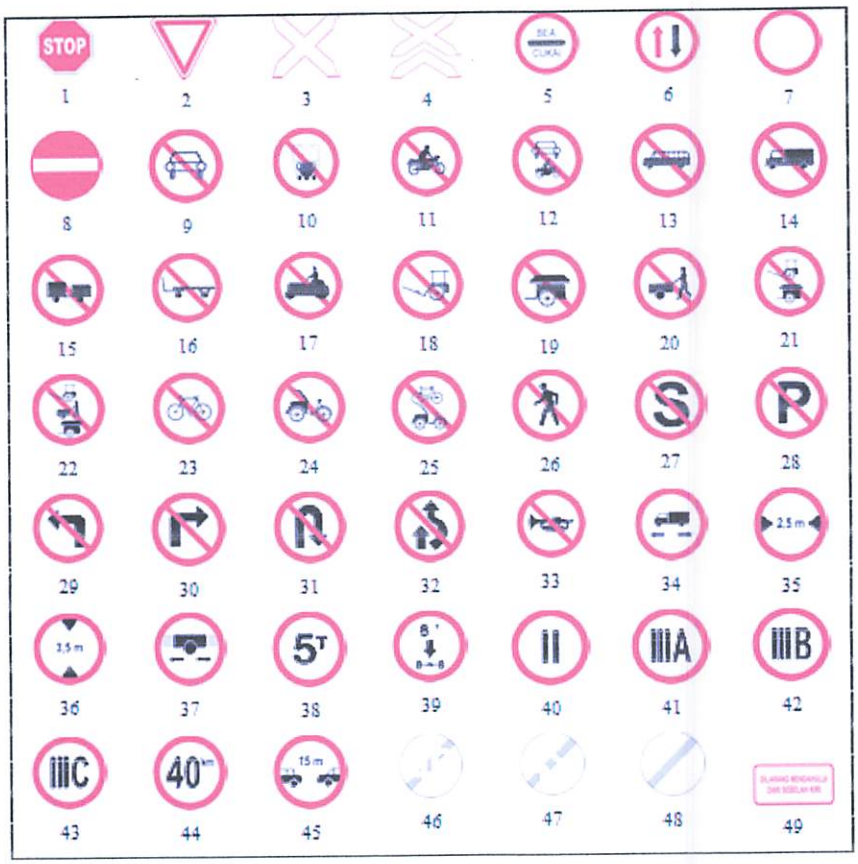
Pada gambar 2.1 menunjukkan macam-macam gambar dari rambu peringatan. Bentuk rambu peringatan adalah bujur sangkar dan persegi panjang, untuk penempatan rambu peringatan ditempatkan pada sisi jalan sebelum tempat atau bagian bagian jalan yang berbahaya dengan jarak sesuai dengan table 2.2.

Table 2.2. Jarak Penempatan Rambu Peringatan

Kecepatan Rencana (km/jam)	**Jarak Minimum (x)**
> 100	180 m
81 - 100	100 m
61 - 80	80 m
< 60	50 m

2.1.2. Rambu Larangan

Rambu larangan menunjukkan perbuatan yang dilarang dilakukan oleh pemakai jalan. Warna dasar rambu larangan berwarna putih dan lambang atau tulisan berwarna hitam atau merah. . Untuk penempatan rambu larangan, rambu larangan ditempatkan pada sisi jalan pada awal bagian jalan dimulainya rambu larangan dan pada bagian jalan berakhirnya rambu larangan.



Gambar 2.2 Rambu Larangan

Tabel 2.3. Keterangan Rambu Larangan

No	Keterangan
1	Larangan berjalan terus, wajib berhenti sesaat dan meneruskan perjalanaan setelah mendapat kepastian aman meneruskan dari lalu lintas arah lainnya
2	Berjalan terus apabila mengakibatkan rintangan/habatan/gangguan bagi lalu lintas dari arah lain yang wajib didahulukan
3	Larangan masuk
4	Larangan bagi semua kendaraan, dilarang masuk
5	Wajib berhenti
6	Rambu larangan berjalan terus, wajib berhenti sesaat sebelum bagian jalan tertentu dan meneruskan perjalanan setelah mendahulukan kendaran yang datang dari arah depan secara bersamaan
7	Larangan masuk bagi semua kendaraan bermotor maupun tidak

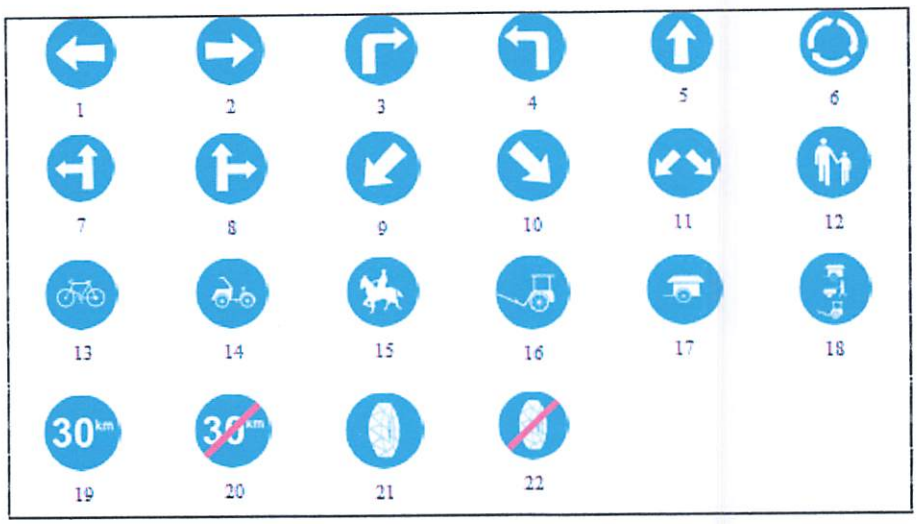
	bermotor dari kedua arah
8	Larangan bagi semua kendaraan, dilarang masuk
9	Rambu larangan masuk bagi kendaraan bermotor roda empat
10	Rambu larangan masuk bagi kendaraan bermotor roda tiga
11	Rambu larangan masuk bagi kendaraan bermotor roda dua
12	Rambu larangan masuk bagi kendaraan bermotor roda 2 dan roda 4
13	Bus dilarang masuk
14	Mobil barang dilarang masuk
15	Kendaraan gandeng dilarang masuk
16	Kendaraan temple dilarang masuk
17	Kendaraan khusus dilarang masuk
18	Delman dan sejenisnya dilarang masuk
19	Rambu larangan masuk bagi gerobak dan sejenisnya
20	Rambu larangan masuk bagi gerobak dorong dan sejenisnya
21	Rambu larangan masuk bagi gerobak dan dokar
22	Rambu larangan masuk bagi semua kendaraan tidak bermotor
23	Scpcda dilarang masuk
24	Becak dilarang masuk
25	Sepeda dan becak dilarang masuk
26	Pejalan kaki dilarang masuk
27	Dilarang berhenti
28	Dilarang parkir
29	Dilarang belok kea rah kiri
30	Dilarang belok kea rah kanan
31	Dilarang putar balik arah
32	Dilarang mendahului kendaraan lain
33	Dilarang membunyikan isyarat suara
34	Dilarang masuk jika panjang lebih dari 2 meter
35	Dilarang masuk jika lebar lebih dari 2,5 meter
36	Dilarang masuk jika tinggi lebih dari 3,5 meter
37	Dilarang masuk jika panjang lebih dari 2 meter
38	Kendaraan seluruh bobotnya termasuk muatannya melebihi 5 ton

	dilarang masuk
39	Kendaraan yag bobotnya pada sumbu melebihi 8 ton dilarang masuk
40	Larangan masuk bagi kendaraan dengan muatan sumbu terberat lebih besar dari 10 ton atau ukuran lebar tidak melebihi 2500 milimeter atau ukuran panjang tidak melebihi 18.000 milimeter
41	Larangan masuk bagi kendaraan dengan muatan sumbu terberat lebih besar dari 8 ton atau ukuran lebar tidak melebihi 2500 milimeter atau ukuran panjang tidak melebihi 18.000 milimeter
42	Kendaraan dengan sumbu terberat lebih besar dari 8 ton atau ukuran lebar tidak melebihi 2.500 milimeter atau ukuran panjang tidak melebihi 12.000 milimeter
43	Sumbu terberat tidak melebihi besar dari 8 ton atau ukuran lebar tidak melebihi 2.100 milimeter atau ukuran panjang tidak melebihi 9.000 milimeter
44	Kecepatam maksimmal 40 km/jam
45	Jarak kendaraan 15 meter
46	Batas akhir kecepatan maksima 40 m/jam
47	Batas akhir larangan mendahului kendaraan
48	Batas akhir scmua larangan
49	Larangan mendahului dari sebelah kiri

Pada gambar 2.2 menunjukkan macam-macam gambar dari rambu larangan. Bentuk rambu larangan ada 5, yaitu segi delapan sama sisi, segitiga sama sisi dengan titik sudutnya dibulatkan, silang dengan ujung-ujungnya diruncingkan, lingkaran, persegi panjang.

2.1.3. Rambu Perintah

Rambu perintah menyatakan perintah yang wajib dilakukan oleh pemakai jalan. Rambu perintah berbentuk lingkaran berwarna biru dan lambang atas tulisan berwarna putih serta merah untuk garis serong sebagai batas akhir perintah.



Gambar 2.3 Rambu Perintah

Tabel 2.4. Keterangan Rambu Perintah

No	Keterangan
1	Wajib mengikuti arah ke kiri
2	Wajib mengikuti arah ke kanan
3	Wajib mengikuti arah yang ditunjuk (belok kanan)
4	Wajib mengikuti arah yang ditunjuk (belok kiri)
5	Wajib jalan lurus kedepan
6	Wajib mengikuti arah yang ditentukan pada bundaran
7	Wajib mengikuti salah satu arah yang ditunjukan
8	Wajib mengikuti salah satu arah yang ditunjukan
9	Lajur atau bagian jalan yang wajib dilewati
10	Lajur atau bagian jalan yang wajib dilewati
11	Wajib melewati salah satu jalur yang ditunjuk
12	Wajb untuk pejalan kaki
13	Wajib untuk sepeda
14	Wajib untuk becak
15	Wajib untuk pengendara kuda
16	Wajib untuk dokar
17	Wajib untuk pedati
18	Wajib untuk pedati, gerobak dorong, dan dokar

19	Perintah kecepatan minimum
20	Batas akhir kecepatan minimum
21	Wajib memakai rantai ban
22	Batas akhir wajib memakai rantai ban

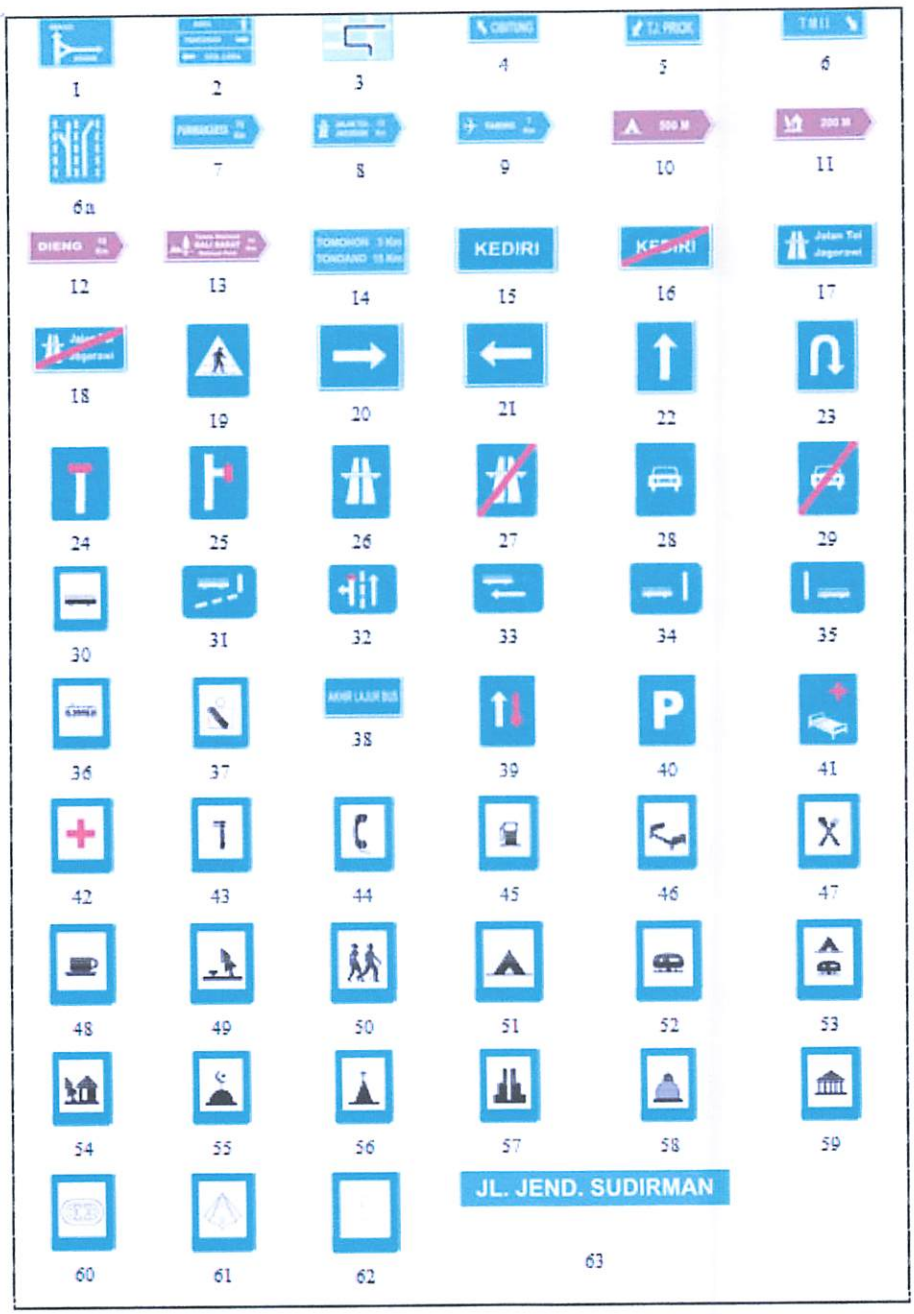
Pada gambar 2.3 menunjukkan macam-macam gambar dari rambu perintah. Rambu perintah berbentuk lingkaran, untuk penempatn rambu perintah, rambu perintah ditempatkan pada sisi jalan pada bagian awal lajur atau bagian jalan yang wajib dilewati dan pada batas akhir berlakunya rambu perintah.

2.1.4. Rambu Petunjuk

Rambu petunjuk jurusan adalah bagian dari rambu petunjuk yang menyediakan informasi kepada pengemudi tentang tujuan dan fasilitas-fasilitas sepanjang jalan.

Rambu pendahulu petunjuk jurusan sangat penting dalam keselamatan jalan. Pengemudi yang belum mengenal tujuannya sangat bergantung kepada rambu pendahulu petunjuk jurusan. Rambu pendahulu petunjuk jurusan yang baik harus jelas dan mudah dipahami dan memberi informasi kepada pengemudi dalam memilih jalan.

a) Rambu pendahulu petunjuk jurusan, rambu petunjuk jurusan dan rambu penegas jurusan yang menyatakan petunjuk arah untuk mencapai tujuan antara lain kota, daerah/wilayah serta rambu yang menyatakan nama jalan dinyatakan dengan warna dasar hijau dengan lambang dan atau tulisan warna putih.
b) Rambu petujuk jurusan menggunakan huruf capital pada huruf pertama, dan selanjutnya menggunakan huruf kecil dan atau seluruhnya menggunakan huruf capital dan atau huruf kecil.
c) Khusus rambu petunjuk jurusan kawasan objek wisata dinyatakan dengan warna dasar coklat dengan lambang dan atau tulisan warna putih.
d) Rambu petunjuk yang menyatakan tempat fasilitas umum, batas wilayah suatu daerah, situasi jalan, dan rambu berupa kata-kata serta tempat khusus dinyatakan dengan warna dasar biru.



Gambar 2.4 Rambu Petunjuk

Tabel 2.5. Keterangan Rambu Petunjuk

No	Keterangan
1	Rambu pendahulu petunjuk jurusan pada persimpangan di depan
2	Rambu petunjuk jurusan yang menunjukkan arah daerah
3	Rambu petunjuk arah daerah yang harus ditempuh
4	Rambu petunjuk jurusan yang menyatakan arah untuk mencapai suatu

	tempat keluar dari jalan tol
5	Rambu petunjuk jurusan yang menyatakan lajur kiri yang harus dilewati untuk jurusan yang dituju
6	Rambu petunjuk jurusan yang menyatakan lajur kanan yang harus dilewati untuk jurusan yang dituju
6a	Rambu petunjuk jurusan yang menyatakan adanya pilihan lajur sesuai jurusan pada persimpangan
7	Rambu petunjuk jurusan purwakarta dengan jarak 70 Km
8	Rambu petunjuk jurusan menuju jalan tol jogorawi
9	Rambu petunjuk jurusan ke pelabuhan udara
10	Rambu petunjuk jurusan untuk kea rah perkemahan 500 meter
11	Rambu petunjuk jurusan untuk wisata kearah pesanggrahan pemuda
12	Rambu petunjuk jurusan daerah wisata dieng dengan jarak 10 Km
13	Rambu petunjuk jurusan ke daerah taman nasional
14	Jalan ini menuju ke Tomohon 3 Km dan ke Tondano 15 Km
15	Masuk batas wilayah kediri
16	Keluar batas wilayah kediri
17	Awal batas wilayah jalan tol jogorawi
18	Akhir batas wilayah jalan tol jogorawi
19	Tempat penyeberangan orang
20	Jalan satu arah kanan
21	Jalan satu arah kiri
22	Jalan satu arah lurus
23	Rambu petunjuk tempat berbalik arah
24	Jalan buntu
25	Jalan buntu arah kanan
26	Jalan tol
27	Batas akhir jalan tol
28	Khusus kendaraan bermotor roda 4
29	Batas akhir jalan khusus kendaraan bermotor roda 4
30	Tempat pemberhentian bus
31	Awal lajur bus

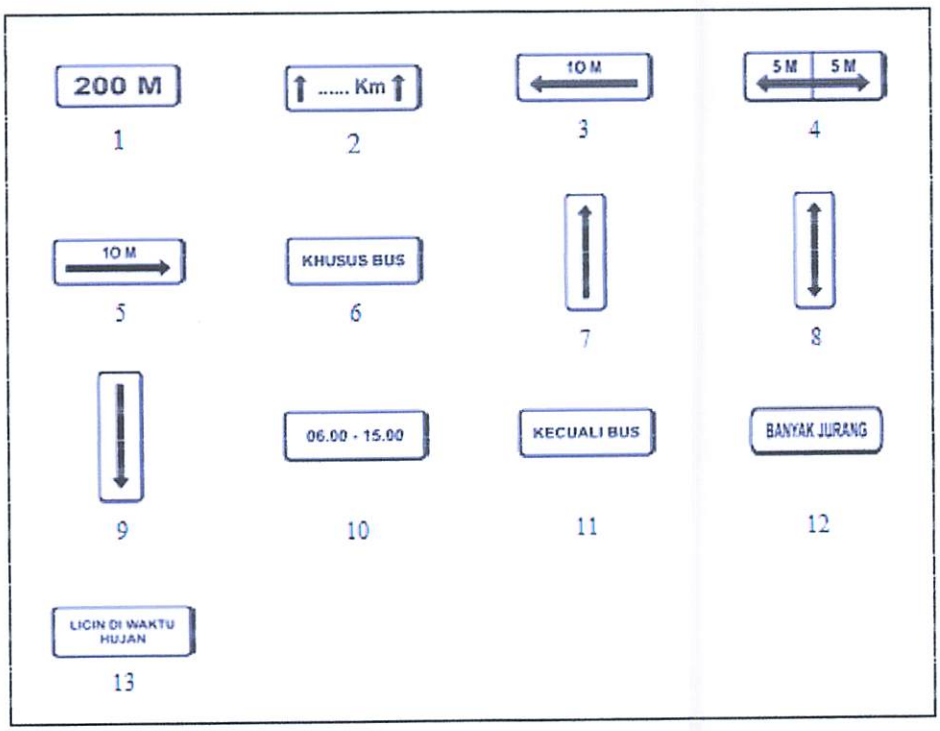
32	Bagian lajur yang dapat digunakan lalu lintas lainnya
33	Rambu yang menjelaskan bahwa akan memasuki jalan yang mempunyai lajur khusus bus
34	Lajur bus searah dengan arah lalu lintas
35	Lajur bus berlawanan dengan arah lalu lintas
36	Akhir lajur bus
37	Tempat pemberhentian kendaraan dengan lintas tetap
38	Memasuki daerah penggunaan sabuk pengaman
49	Prioritas lalu lintas dari arah depan
40	Tempat parkir
41	Rumah sakit
42	Balai pertolongan pertama
43	Bengkel perbaikan kendaraan
44	Telepon umum
45	Pompa bahan bakar
46	Hotel dan motel
47	Rumah makan
48	Kedai kopi
49	Tempat wisata
50	Tempat berjalan kaki
51	Tempat berkemah
52	Tempat kereta kemah
53	Tempat berkemah dan kereta kemah
54	Pesanggrahan pemuda
55	Masjid / Rumah ibadah umat islam
56	Gereja / Rumah ibadah umat kristen
57	Rumah ibadat hindu
58	Museum
59	Stadion / Lapangan terbuka
61	Lapangan gantole
62	Gelora / Bangsal olahraga
63	Jalan Jendral Sudirman

Pada gambar 2.4 menunjukkan macam-macam gambar dari rambu petunjuk, rambu petunjuk berbentuk persegi empat maupun persegi panjang, rambu petunjuk ditempatkan atau dipasang pada awal jalan sebelum lokasi yang ditunjuk dan jarak jarak menuju lokasi dan akhir berlakunya rambu sesuai dengan yang dinyatakan dalam rambu tersebut

2.1.5. Rambu Tambahan

a. Papan tambahan digunakan untuk memuat keterangan yang diperlukan untuk menyatakan hanya berlaku untuk sewaktu-waktu tertentu ataupun perihal lain sebagai hasil manajemen dan rekayasa lalu lintas.

b. Papan tambahan ditempatkan dengan jarak 5 sentimeter sampai dengan 10 sentimeter dari sisi terbawah daun rambu dengan ketentuan lebar papan tambahan secara vertical tidak melebihi sisi daun rambu.



Gambar 2.5 Rambu Tambahan

Tabel 2.6. Keterangan Rambu Tambahan

No	Keterangan
1	Jarak dari rambu sampai awal bagian jalan yang berbahaya atau awal daerah dimana peraturan berlaku, 200 meter

2	Panjang bagian jalan yang berbahaya atau panjang daerah dimana peraturan berlaku, sebagaimana contoh dimaksud
3	Berlakunya rambu sesuai arah panah ke kiri 10 meter
4	Berlakunya rambu sesuai arah panah ke kiri dan ke kanan 5 meter
5	Berlakunya rambu sesuai arah panah ke kanan 10 meter
6	Jalan khusus untuk bis
7	Awal berlakunya rambu sesuai arah panah lalu lintas ke depan
8	Berlakunya rambu sesuai arah panah lalu lintas ke depan dan ke belakang
9	Akhir berlakunya rambu sesuai arah panah
10	Berlakunya rambu dari pukul 06.00 – 15.00
11	Berlakunya rambu bagi semua kendaraan kecuai bus
12	Papan tambahan untuk penegasan jenis bahaya banyaknya jurang
13	Papan tambahan untuk menegaskan jenis bahaya jalan licin waktu hujan

Pada gambar 2.5 menunjukkan macam-macam gambar dari rambu tambahan. Untuk persyaratan rambu tambahan sebegai berikut :

1) Papan tambahan menggunakan warna dasar putih dengan tulisan dan bingkai berwarna hitam.
2) Papan tambahan tidak boleh menyatakan suatu keterangan yang tidak berkaitan dengan rambunya sendiri.
3) Pesan yang termuat dalam papan tambahan harus bersifat khusus, singkat, jelas dan mudah serta cepat dimengerti oleh pengguna jalan.
4) Ukuran perbandingan papan tambahan antara panjang dan lebar adalah 1 (satu) berbanding 2 (dua).

2.2 Undang-undang lalu lintas

Menurut Undang – Undang republik Indonesia No 22 Tahun 2009 tentang lalu lintas dan angkutan jalan.

Menimbang :

a. Bahahwa Lalu Lintas dan Angkutan Jalan mempunyai peran strategis dalam mendukung pembangunan dan integrasi nasional sebagai bagian dari upaya

memajukan kesejahteraan umum sebagaimana diamanatkan oleh Undang-Undang Dasar Negara Republik Indonesia Tahun 1945.

b. bahwa Lalu Lintas dan Angkutan Jalan sebagai bagian dari sistem transportasi nasional harus dikembangkan potensi dan perannya untuk mewujudkan keamanan, keselamatan, ketertiban, dan kelancaran berlalu lintas dan Angkutan Jalan dalam rangka mendukung pembangunan ekonomi dan pengembangan wilayah.

c. bahwa perkembangan lingkungan strategis nasional dan internasional menuntut penyelenggaraan Lalu Lintas dan Angkutan Jalan yang sesuai dengan perkembangan ilmu pengetahuan dan teknologi, otonomi daerah, serta akuntabilitas penyelenggaraan negara.

d. bahwa Undang-Undang Nomor 14 Tahun 1992 tentang Lalu Lintas dan Angkutan Jalan sudah tidak sesuai lagi dengan kondisi, perubahan lingkungan strategis, dan kebutuhan penyelenggaraan Lalu Lintas dan Angkutan Jalan saat ini sehingga perlu diganti dengan undang-undang yang baru.

e. bahwa berdasarkan pertimbangan sebagaimana dimaksud dalam huruf a, huruf b, huruf c, dan huruf d perlu membentuk Undang-Undang tentang Lalu Lintas dan Angkutan Jalan.

Penjelasan atas undang-undang republik Indonesia nomor 22 tahun 2009 tentang lalu lintas dan angkutan jalan, sebagai berikut :

Berkat rahmat Tuhan Yang Maha Esa, Negara Kesatuan Republik Indonesia telah dianugerahi sebagai negara kepulauan yang terdiri atas beribu pulau, terletak memanjang di garis khatulistiwa, serta di antara dua benua dan dua samudera, mempunyai posisi dan peranan yang sangat penting dan strategis untuk mendukung pembangunan ekonomi, pemantapan integrasi nasional guna memperkukuh ketahanan nasional, serta menciptakan ketertiban dunia dan kehidupan berbangsa dan bernegara dalam rangka memajukan kesejahteraan umum sebagaimana diamanatkan oleh Undang-Undang Dasar Negara Republik Indonesia Tahun 1945.

Lalu Lintas dan Angkutan Jalan mempunyai peran strategis dalam mendukung pembangunan dan integrasi nasional, sebagai bagian dari upaya memajukan kesejahteraan umum sebagaimana diamanatkan oleh Undang-Undang Dasar Negara Republik Indonesia Tahun 1945. Sebagai bagian dari sistem

transportasi nasional, Lalu Lintas dan Angkutan Jalan harus dikembangkan potensi dan perannya untuk mewujudkan keamanan, kesejahteraan, ketertiban berlalu lintas dan Angkutan Jalan dalam rangka, mendukung pembangunan ekonomi dan pengembangan ilmu pengetahuan dan teknologi, otonomi daerah, serta akuntabilitas penyelenggaraan negara.

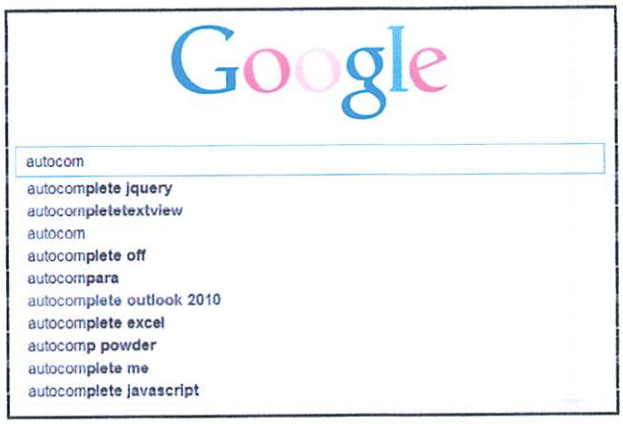
Dalam Undang-Undang ini, pembinaan bidang Lalu Lintas dan Angkutan Jalan dilaksanakan secara bersama-sama oleh semua instansi terkait (*stakeholders*) sebagai berikut:

1. Urusan pemerintahan di bidang prasarana Jalan, oleh kementerian yang bertanggung jawab di bidang jalan.
2. Urusan pemerintahan di bidang sarana dan prasarana lalu lintas dan angkutan jalan, oleh kementerian yang bertanggung jawab di bidang sarana dan Prasarana Lalu Lintas dan Angkutan Jalan.
3. Urusan pemerintahan di bidang pengembangan industri Lalu Lintas dan Angkutan Jalan, oleh kementerian yang bertanggung jawab di bidang industri.
4. Urusan pemerintahan di bidang pengembangan teknologi Lalu Lintas dan Angkutan Jalan, oleh kementerian yang bertanggung jawab di bidang teknologi.
5. Urusan pemerintahan di bidang registrasi dan identifikasi Kendaraan Bermotor dan Pengemudi, Penegakan Hukum, Operasional Manajemen dan Rekayasa Lalu Lintas, serta pendidikan berlalu lintas oleh Kepolisian Negara Republik Indonesia.

2.3 *Autocomplete*

Menurut Kevin Levandi Temotius (2011) a*utocomplete* merupakan metode pencarian yang dimiliki *google*. Metode ini juga ditanamkan pada *android* sebagai metode pencarian kata. Pada pembentukan fitur *autocomplete* atau *autocompletion*, struktur data yang diperlukan adalah *array* yang berisi *reserved words* dan nama fungsi dari berbagai bahasa pemrograman yang didukung oleh *text editor* atau *Integrated Development Environment (IDE)* serta daftar kata apa aja yang telah ditulis oleh pengguna. *Array* yang digunakan tersebut sudah terurut secara alfabetik.

Ketika pengguna mengetikkan karakter atau huruf pertama, fitur *autocomplete* atau *autocompletion* akan bekerja dengan mencari kata yang berada dalam *array* yang karakter pertamanya sama dengan karakter yang diketikkan oleh pengguna. Setelah semua elemen *array* sudah diperiksa, elemen *array* yang bersesuaian dengan karakter pertama yang diketikkan oleh pengguna ditampilkan kedalam *menu list*. Proses pemrediksian berulang ketika pengguna mengetikkan karakter kedua, namun *array* yang diperiksa adalah *array* hasil dari pemeriksaan sebelumnya. Adapun salah - satu jenis pencarian yang menggunakan metode *autocomplete* ditunjukkan pada gambar 2.4.



Gambar 2.6 Cara Kerja *Autocomplete*

2.4 Sistem Operasi Android

Menurut Nazruddin (2012), Android sebuah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalanya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat piranti lunak untuk ponsel/ *smartphone*. Kemudian untuk mengembangkan Android dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan open *platform* perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

Sekitar September 2007 Google mengenalkan *Nexus One*, salah satu *smartphone* yang menggunakan Android sebagai sistem operasinya. Telephone seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Thosiba Corp, dan Vodafone Group Plc. Seiring pembentukan *Open Handset Alliance*, OHA mengumumkan produk perdana mereka, Android, perangkat *mobile* yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaharuan berupa perbaikan *bug* dan penambahan fitur baru.

Pada masa saat ini sebagian besar *vendor-vendor smartphone* sudah memproduksi *smartphone* berbasis Android, *vendor-vendor* itu antara lain HTC, Motorola, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, LG, Acer, Philips, T-Mobile, Nexian, IMO, Asus, dan masih banyak lagi *vendor smartphone* di dunia yang mempunyai Android. Hal ini, karena Android itu sistem operasi yang *open source* sehingga bebas didistribusikan dan dipakai oleh *vendor* manapun.

Tidak hanya menjadi sistem operasi di *smartphone*, saat ini Android menjadi pesaing utama dari Apple pada sistem operasi Table PC. Pesatnya pertumbuhan Android selain factor yang disebutkan di atas adalah karena Android itu sendiri adalah *platform* yang sangat lengkap baik itu sistem operasinya, aplikasi, dan tool pengembangan, market aplikasi Android serta dukungan yang sangat tinggi dari komunitas *Open Source* di dunia, sehingga Android terus berkembang pesat baik teknologi maupun dari segi jumlah *device* yang ada di dunia.

2.4.1. Platform Masa Depan

Android dipuji sebagai "*platform mobile* pertama yang Lengkap, Terbuka, dan Bebas" (Nazruddin, 2012).

a. Lengkap (*Complete Platform*): Para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform* Android. Android merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.
b. Terbuka (*Open Source Platform*): *Platform* Android disediaka melalui lisensi *open source.* Pengembang dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux Kernel 2.6.
c. *Free* (*Free Platform*): Android adalah *platform/* aplikasi yang bebas untuk *develop.* Tidak ada lisensi atau biaya royalti untuk dikembangkan pada *platform* Android. Tidak ada biaya keanggotaan diperlukan. Tidak diperlukan biaya pengujian. Tidak ada kontrak yang diperlukan. Aplikasi untuk Android dapat didistribusikan dan diperdagangkan dalam bentuk apapun.

Android merupakan generasi baru *platform mobile, platform* yang meberikan pengembangan untuk melakukan pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari Android dilisensikan di bawah GNU, General Public Lisensi Versi 2 (GPL_{V2}), yang sering dikenal dengan istilah "*copyleft*" lisensi di mana setiap pebaikan pihak ketiga harus terus jatuh di bawah terms. Android didistribusikan di bawah Lisensi Apache *Software* (ASL/Apache2), yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen *handset* khususnya) dapat memilih untuk meningkatkan *platform* tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebalikmya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apa pun yang mereka inginkan. Pengembang aplikasi Android diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apa pun yang mereka inginkan.

Pengembang memiliki beberapa pilihan ketika membuat aplikasi yang berbasis Android. Sebagian besar pengembang menggunakan *Eclipse* yang tersedia secara bebas untuk merancang dan mengambangkan aplikasi Android. *Eclipse* adalah IDE yang paling popular untuk pengembangan Android, karena memiliki Android *plug-in* yang tersedia untuk mefasilitasi pengembangan Android. Selain itu, *Eclipse* juga mendapat dukungan langsung dari Google untuk

manjadi IDE pengembangan aplikasi Android, ini terbukti dengan adanya penambahan *plugins* untuk *eclipse* untuk membuat *project* Android di mana *source software* langsung dari situs resminya Google. Akan tetapi, hal di atas tidak menutup kemungkinan untuk menggunakan IDE yang lain seperti Netbeans untuk melakukan pengembangan Android.

Aplikasi Android dapat dikembangkan pada sistem operasi berikut (Nazruddin, 2012):

a. Windows XP Vista/ Seven.
b. Mac OS X (Mac OS X 10.4.8 atau lebih baru).
c. Linux.

2.4.2. *The Dalvik Virtual Machine* (DVM)

Menurut Nazruddin (2012), Salah satu elemen kunci dari Android adalah *Dalvik Virtual Machine* (DVM). Android berjalan di dalam *Dalvik Virtual Machine* (DVM) bukan di *Java Virtual Machine* (JVM), sebenarnya banyak persamaannya dengan *Java Virtual Machine* (VM) seperti Java ME (*Java Mobile Edition*), tetapi Android menggunakan *Virtual Machine* sendiri yang menurut saya dikustomisasi dan dirancang untuk memastikan bahwa beberapa *feature-feature* berjalan lebih efisien pada perangkat *mobile.*

Dalvik Virtual Machine (DVM) adalah "register bases" sementara *Java Virtual Machine* (JVM) adalah "stack based", DVM didesain dan ditulis dan ditulis oleh Dan Bornsten dan beberapa *engineers* Google lainnya. Jadi, bisa kita katakan " *Dalvik equals(Java) == False".* D *Dalvik Virtual Machine* menggunakan karnel Linux untuk menangani fungsionalitas tingkat rendah termasuk keamanaan, *threading*, dan proses serta manajemen memori. Ini memungkinkan kita untuk menulis Aplikasi C/ C+ sama halnya seperti pada OS Linux kebanyakan. Meskipun dalam kenyataannya kita harus banyak memahami arsitektur dan proses sistem karnel Linux yang digunakan dalam Android tersebut.

Semua *hardware* yang berbasis Android dijalankan dengan menggunakan *Virtual Machine* untuk eksekusi aplikasi, pengembang tidak perlu khawatir tentang implementasi perangkat keras tertentu. *Dalvik Virtual Machine* mengeksekusi *executable file*, sebuah format yang dioptimalkan untuk

memastikan memori yang digunakan kecil. *The executable file* diciptakan dengan mengubah kelas bahasa java dan kompilasi menggunakan *tools* yang disediakan dalam SDK Android (Nazruddin, 2012).

2.4.3. Android SDK (*Software Development Kit*)

Menurut Nazruddin (2012), Android SDK adalah *tools* API (*Applications Programing Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh Google. Saat ini disediakan Android SDK (*Software Development Kit)* sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi-netral, Android member Anda kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *Handphone/ Smartphone*. Beberapa fitur-fitur Android yang paling penting adalah (Nazruddin, 2012):

1) *Framework* aplikasi yang mendukung penggantian komponen dan *reusable.*
2) Mesin *Virtual Dalvik* dioptimalkan untuk perangkat *mobile.*
3) *Integrated browser* berdasarkan *engine open source WebKit.*
4) Grafis yang dioptimalkan dan didukung *libraries* grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional akselerasi *hardware*).
5) SQLite untuk penyimpanan data.
6) *Media Support* yang mendukung audio, video, dan gambar (MPG4, H.264. MP3, AAC, AMR, JPG, PNG, GIF), GSM *Telephony* (tergantung *hardware*).
7) *Bluetooth,* EDGE, 3G, dan WiFi (tergantung *hardware*).
8) Kamera, GPS, kompas, dan *accelerometer* (tergantung *hardware*).
9) Lingkungan *Development* yang lengkap dan kaya termasuk perangkat *emulator, tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuj IDE *Eclipse*.

Untuk *source* SDK Android ini dapat dilihat dan diunduh langsung di situs resmi pengembang SDK Android di *http://www.developer.android.com* atau SDK Android juga terdapat dalam CD yang di-*include*-kan dalam buku ini baik versi

windows maupun versi Linux, karena SDK Android sifatnya gratis serta bebas didistribusikan (Nazruddin, 2012).

2.4.4. ADT (*Android Development Tools*)

Menurut Nazruddin (2012), *Android Development Tools* (ADT) adalah plugin yang didesain untuk IDE *Eclipse* yang memberikan kita kemudahan dalam mengembangkan aplikasi Android dengan menggunkan IDE *Eclipse*. Dengan menggunkan ADT untuk *Eclipse* akan memudahkan kita dalam membuat aplikasi *project* Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainya, begitu juga kita dapat melakukan *running* aplikasi menggunkan Android SDK melalui *eclipse*. Dengan ADT juga kita dapat melakukan pembuatan *package* Android (.apk) yang digunakan untuk distribusi aplikasi Android yang kita rancang.

Mengembangkan aplikasi Android dengan menggunaka ADT di *eclipse* sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi Android. Berikut adalah versi ADT untuk *eclipse* yang sudah dirilis (Nazruddin, 2012):

a. ADT 12.0.0 (July 2011)
b. ADT 11.0.0 (June 2011)
c. ADT 10.0.1 (March 2011)
d. ADT 10.0.0 (February 2011)
e. ADT 9.0.0 (January 2011)
f. ADT 8.0.1 (December 2010)
g. ADT 8.0.0 (December 2010)
h. ADT 0.9.9 (September 2010)
i. ADT 0.9.8 (September 2010
j. ADT 0.9.7 (May 2010)
k. ADT 0.9.6 (March 2010)
l. ADT 0.9.5 (December 2009)
m. ADT 0.9.4 (October 2009)

Semakin tinggi *platform* Android yang kita gunakan, dianjurkan menggunkan ADT yang lebih terbaru, karena biasanya munculnya *platform* baru didikuti oleh munculnya versi ADT yang terbaru. Untuk melakukan instalasi ADT di-*eclipse*

dapat dilakukan secara *on-line* maupun *offline*. Untuk *download* ADT ini dapat dilakukan di *http://developer.Android.com/sdk/eclipe-adt.html* (Nazruddin, 2012).

2.4.5. Arsitektur Android

Secara garis besar Arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut (Nazruddin, 2012):

a. *Applications* dan *Widgets*

Applications dan *Widgets* ini adalah *layer* di mana kita berbuhangan dengan aplikasi saja, di mana kita berhubungan dengan aplikasi saja, di mana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di *layer* terdapat aplikasi inti termasuk klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain. Semua aplikasi ditulis menggunkan bahasa pemrograman Java.

b. *Applications Frameworks*

Android adalah "*Open Development Platform*" yaitu Android menawarkan kepada pengembang dan member kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resource*, menajalnkan *service background*, mengatur *alarm*, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju API *framework* seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunkan kembali komponen yang sudah digunakan (*reuse*).

Sehingga bisa kita simpulkan *Applications Frameworks* ini adalah *layer* dimana para pembuat aplikasi melakukan pengembangan/ pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada *layer* inilah aplikasi dapat dirancangndan dibuat, seperti *content-providers* yang berupa sms dan panggilan telepon.Komponen-komponen yang termasuk dalam *Applications Frameworks* adalah sebagai berikut (Nazruddin, 2012):

1) *Views*
2) *Content Provider*
3) *Resource Manager*
4) *Notification Manager*

5) *Activity Manager*

c. *Libraries*

Libraries ini adalah *layer* di mana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan di atas kernel. *Layer* ini meliputi berbagai *library* C/C++ inti seperti Libc dan SSL, serta (Nazruddin, 2012):

1) *Libraries* media untuk pemutaran media audio dan video
2) *Libraries* untuk manajemen tampilan
3) *Libraries Graphics* mencakup SGL, dan OpenGL untuk grafis 2D dan 3D
4) *Libraries* SQLite untuk dukugan *database.*
5) *Libraries* SSL dan *WebKit* terintegrasi dengan *web browser* dan *security*
6) *Libraries* LiveWebcore mencakup mencakup modern *web browser* dengan *engine embedded web view*
7) *Libraries* 3D yang mencakup implementasi OpenGL ES 1.0 API's

d. *Android Run Time*

Layer yang membuat aplikasi Android dapat dijalankan di mana dalam prosesnya menggunakan implementasi Linux. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam Android *Run Time* dibagi menjadi dua bagian yaitu:

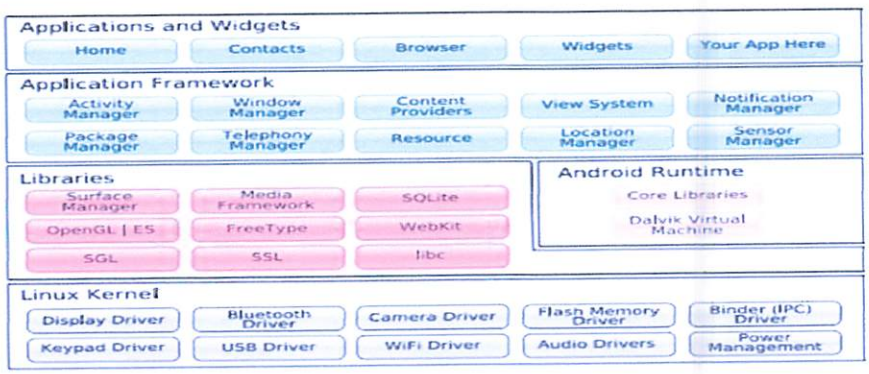
Core Libraries: Aplikasi Android dibangun dalam bahasa Java, sementara Dalvik sebagai virtual mesinnya bukan *Virtual Machine Java,* sehingga diperlukan *libraries* yang berfungsi untuk menterjemahkan bahasa Java/C yang ditangani oleh *Core Libraries.*

Dalvik Virtual Machine: Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, di mana merupakan pengembangan yang mampu membuat Linux kernel untuk melakukan *threading* dan manajemen tingkat rendah.

e. Linux *Kernel*

Linux *kernel* adalah *layer* di mana inti dari *operating* sistem dari Android itu berada. Berisi *file-file system* yang mengatur sistem *processing, memory, resource, drivers,* dan sistem-sistem operasi Android lainnya. Linux *kernel* yang digunakan Android adalah Linux *kernel release* 2.6. Jika Anda ingin mempelajari

Linux *kernel* ini, Anda dapat mengunduh Linux *kernel* secara gratis di situs *www.kernel.org*.



Gambar 2.7 Arsitektur Android (Nazruddin, 2012)

2.4.6. Fundamental Aplikasi

Menurut Nazruddin (2012), Aplikasi Android ditulis dalam bahasa pemrograman Java. Kode Java dikompilasi bersama dengan data *file resource* yang dibutuhkan oleh aplikasi, di mana prosesnya di*package* oleh *tools* yang dinamakan "apt tools" ke dalam paket Android sehingga menghasilkan *file* dengan ekstensi apk. *File* apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat di install di perangkat *mobile.* Ada enam jenis komponen pada aplikasi Android yaitu (Nazruddin, 2012):

a. *Activities*

Suatu *activity* akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi Android bisa jadi hanya memiliki satu *activity*, tetapi umumnya aplikasi memiliki banyak *activity* tergantung pada tujuan aplikasi dan desain dari apikasi tersebut. Satu *activity* biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada *user.* Untuk pndah dari satu *activity* ke *activity* lain kita dapat melakukan dengan satu *even,* misalnya *click* tombol, memilih opsi atau menggunakan triggers tertentu. Secara hirarki sebuah *windows activity* dinyatakan dengan *method* Activity.setContentView(). ContentView adalah objek yang berada pada *root* hirarki.

b. *Service*

Service tidak memiliki *Graphic User Inteface* (GUI), tetapi *service* berjalan secara *background*, sebagai contoh dalam memainkan musik, *service* mungkin

memainkan musik atau mengambil data dari jaringan, tetapi setiap *service* berada pada setiap induknya. Misalnya *media player* sedang memutar lagu dari *list* yang ada. Aplikasi ini akan memiliki dua atau lebih *activity* yang memungkinkan *user* untuk memilih lagu misalnya, atau menulis sms sambil *player* sedang berjalan. Untuk menjaga musik tetap di jalankan, *activity player* dapat menjalankan *service. Service* dijalankan pada *thread* utama dari proses aplikasi.

c. *Broadcast Receiver*

Broadcast receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh *broadcast* seperti notifikasi zona waktu berubah, baterai *low*, gambar telah selesai diambil oleh *camera*, atau pengubahan referensi bahasa yang digunakan. Aplikasi juga dapat menginisiasi *broadcast* misalnya memberikan informasi pada aplikasi lain bahwa ada data yang telah diunduh ke perangkat dan siap untuk digunakan. *Broadcast receiver* tidak memiliki *user interface* (UI), tetapi memiliki sebuah *activity* untuk merespon informasi yang mereka terima, atau mungkin menggunakan *Notificasi Manager* untuk meberitahu kepada pengguna, seperti lampu latar atau *vibrating* (getaran) perangkat, dan lain sebagainya.

d. *Content Provider*

Content Provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan di dalam *file* sistem *database* SQLite. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta (*Map*), atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider.*

2.4.7. Versi Android

Telepon pertama yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua *vendor* seluler di dunia menggunakan Android sebagai *operating system.* Adapun versi-versi Android yang pernah dirilis adalah sebagai berikut (Nazruddin, 2012):

1) Android versi 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1 Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam, alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

2) Android versi 1.5 *(Cupcake)*

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, meng-*upload video* ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan *Bluetooth* A2DP, kemampuan terhubung secara otomatis ke *headset Bluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan sistem.

3) Android versi 1.6 (*Donut*)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik disbanding sebelumnya, menggunakan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untu foto yang akan dihapus; kamera, *camcorder* dan galeri yang diintegrasikan, CDMA/EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine,* kemampuan *dial* kontak, teknolofi *text to change speech* (tidak tersedia pada semua ponsel), pengadaan resolusi VWGA.

4) Android versi 2.0/2.1 (*Eclair*)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/ 2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan *hardware,* peningkatan Google Maps 3.1.2, perubahan UI dengan *browser* baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, *digital Zoom*, dan *Bluetooth* 2.1.

5) Android versi 2. 2 (*Froyo: Frozen Yoghurt*)

Pada bulan mei 2010 Android versi 2.2 Rev 1 diluncurkan. Android inilah yang sekarang sangat banyak beredar di pasaran, salah satunya adalah dipakai si Samsung FX tab yang sudah beredar di pasaran. Fitur yang tersedia di Android versi ini sudah kompleks di antaranya adalah (Nazruddin, 2012):

a) Kerangka aplikasi memungkinkan pengguna dan penghapusan komponen yang tersedia.

b) *Dalvik Virtual Machine* dioptimalkan untuk perangkat *mobile.*

c) Grafik: grafik di 2D dan grafis 3D berdasarkan *libraries* OpenGL.

d) SQLite: untuk penyimpanan data.

e) Mendukung media: *audio, video,* dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).

f) GSM *Bluetooth,* EDGE, 3G, dan WiFi (*hardware independent*).

g) Kamera, *Global Positioning System* (GPS), kompas, dan *accelerometer* (tergantung *hardware*).

6) Android versi 2.3 (*Gingerbread*)

Android versi 2.3 diluncurkan pada Desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan seperti berikut:

a) *SIP-based VoIP*

b) *Near Field Communications* (NFC)

c) *Gyroscope* dan sensor

d) *Multiple cameras support*

e) *Mixable audio effects*

f) *Download manager*

7) Android versi 3.0 (*Honeycomb*)

Dirilis Februari 2011 sebagai Android 3.0 revisi 1 serta Android versi 3.0 revision 2 telah dirilis pada Juli 2011.

8) Android versi 3.1

Dirilis Mei 2011, sedangkan Android 3.1 revisi 2 juga dirilis Mei 2011, serta Android 3.1 revision 3 dirilis pada Juli 2011.

9) Android versi 3.2

Dirilis Juli 2011.

10) Android versi 4.0

Dirilis November 2011.

Android versi 3.0 ke atas adalah generasi *platform* yang digunakan untuk tablet pc. Sementara versi 4.0 sudah merupakan *platform* yang bisa dipakai di *smartphone* dan tablet pc. Demikian beberapa versi Android yang sudah dirilis sampai dengan buku ini ditulis, kemungkinan besar versi tersebut akan terus berkembang seiring dengan kebutuhan yang sangat kompleks di bidang penggunaan *smartphone.* Untuk informasi detail versi *platform* atau untuk

mendapatkan informasi versi terbaru Android Anda dapat melihat di www.developer.android.com (Nazruddin, 2012).

2.5 *Eclipse*

Menurut Arif Akbarul (2011) *Eclipse* adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua Platform (platform-independent).

Berikut ini adalah sifat dari Eclipse:

1. *Multi-platform*: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
2. *Mulit-language*: *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
3. *Multi-role*: Selain sebagai IDE untuk pengembangan aplikasi, *Eclipse* pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, *test* perangkat lunak, pengembangan *web*, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in.*

Secara *default editor* ini belum bisa dipakai untuk mendevelop *android.* Agar bisa digunakan untuk membuat aplikasi *android* maka harus diinstall *plugin* dulu namanya ADT (*Android Development Tools*). Setelah terinstal, maka *eclipse* sudah siap digunakan, hanya saja belum dilengkapi library dan emulator. Untuk memenuhinya, maka perlu diinstal SDK *Android* (*StKamurt Development Kit*).

BAB III
ANALISA DAN PERANCANGAN SISTEM

3.1 Analisa Kebutuhan

Dalam pembuatan aplikasi pengenalan rambu lalu lintas berbasis *Android* ini dilakukan analisa sistem mulai dari analisa kebutuhan fungsional dan analisa kebutuhan non fungsional yang meliputi analisa kebutuhan perangkat lunak (*software*), analisa kebutuhan perangkat keras (*hardware*), dan analisa kebutuhan sumber daya manusia (*brainware*).

3.1.1 Analisa Kebutuhan Fungsional

Analisa kebutuhan fungsional berisi tentang proses-proses apa saja yang nantinya dilakukan oleh aplikasi dan berisi informasi-informasi apa saja yang harus ada dalam aplikasi. Berikut ini adalah kebutuhan fungsional dari aplikasi pengenalan rambu lalu lintas berbasis android, yaitu :

1. Aplikasi dapat menampilkan informasi yang berhubungan dengan rambu lalu lintas, seperti pengertian rambu jenis-jenis rambu lalu lintas, aturan umum rambu lalu lintas, persyaratan bentuk dan warna rambu, dan kekuatan hukum rambu.
2. Aplikasi dapat menampilkan gambar dari rambu larangan, rambu peringatan, rambu perintah, rambu petunjuk, dan rambu tambahan yang disertai dengan penjelasan teks dari setiap gambar, dan disertai juga dengan suara penjelasan dari gambar tersebut.
3. Aplikasi dapat menampilkan undang-undang lalu lintas no 22 tahun 2009, tampilan dari undang-undang lalu llintas pada aplikasi ini berupa pasal-pasal dan isinya.
4. Aplikasi dapat menampilkan soal kuis sebanyak sepuluh soal dengan jawaban pilihan ganda, untuk satu halaman akan ditampilkan satu soal kuis disertai dengan jawaban pilihan ganda.
5. Aplikasi dapat menampilkan nilai atau skor yang diperoleh dari hasil jawaban soal yang benar.

3.1.2 Analisa Kebutuhan Non Fungsional

Analisa kebutuhan non fungsional dilakukan untuk menghasilkan spesifikasi kebutuhan kebutuhan *non-fungsional*. Spesifikasi kebutuhan *non-fungsional* adalah spesifikasi yang rinci tentang hal-hal yang akan dilakukan sistem ketika diimplementasikaan serta komponen-komponen yang akan dilibatkan pada sistem yang akan dibagun meliputi analisa perangkat lunak (*software*), analisa perangkat keras (*hardware*), dan analisa kebutuhan sumber daya manusia (*brainware*).

a) Perangkat Lunak (*software*)

Adapun perangkat lunak (*software*) yang digunakan dalam pembuatan aplikasi pengenalan rambu lalulintas ini adalah sebagai berikut :

1. **Eclipse**

   Eclipse digunakan untuk membuat aplikasi pengenalan rambu lalu lintas berbasis *Android* yang berfungsi sebagai

2. **Adobe Photoshop CS3**

   Adobe Photoshop CS4 digunakan untuk editing gambar yang akan dipakai dalam aplikasi. Editing gambar yang dilakukan adalah pengolahan agar ukuran file gambar tidak terlalu besar pada saat dipasang pada aplikasi. Karena jika file gambar yang terlalu besar dipasang pada aplikasi akan menyebabkan besarnya ukuran file aplikasi sehingga memungkinkan aplikasi tidak dapat dijalankan pada *device* target.

   Fungsi dari Adobe Photoshop CS3 antara lain :

   a) Membuat *Image Button*

   b) Membuat *background*

   c) Editing gambar

b) Perangkat Keras (*hardware*)

Dalam pembuatan aplikasi pengenalan rambu lalu lintas berbasis *Android*, perangkat keras yang digunakan adalah komputer dengan spesifikasi sebagai berikut :

1. Hardisk 500 GB
2. 14''WXGA LED, Max. *Resolution* 1366 x 768, *Clear SuperView* LED.
3. Prosesor *Intel* ® *Inside* ™ Core ™ i3.

c) Analisa Kebutuhan Sumber Daya Manusia (*brainware*)

Sumber daya manusia adalah orang yang terlibat dalam perancangan, pembuatan, dan implementasi aplikasi pengenalan rambu lalu lintas berbasis *Android* ini. Sumber daya manusia yang dibutuhkan dalam pembuatan aplikasi ini adalah sebagai berikut :

1. *Disainer*

   Disainer yaitu pihak yang mengimplementasikan dan merancang aplikasi secara keseluruhan.

2. *User*

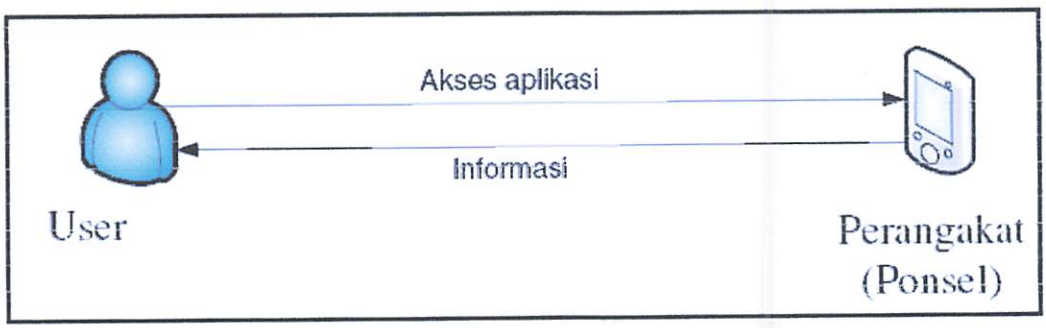
   User yaitu pihak yang mengunakan aplikasi pengenalan rambu lalu lintas berbasis *Android* ini.

3.2 Perancangan Aplikasi

Dalam pembuatan aplikasi pengenalan rambu lalu lintas berbasis *Android* ini setelah melakukan analisa kebutuhan adalah membuat *desain* aplikasinya. *Desain* aplikasi dibuat untuk memudahkan dalam implementasi aplikasi yang akan dibuat nantinya. D*esain* aplikasi yang dilakukan adalah sebagai berikut :

3.2.1 Perancangan Sistem

Perancangan sistem menunjukkan interaksi antara *user* atau pengguna dengan aplikasi ditunjukkan dengan gambar 3.1 dibawah ini.



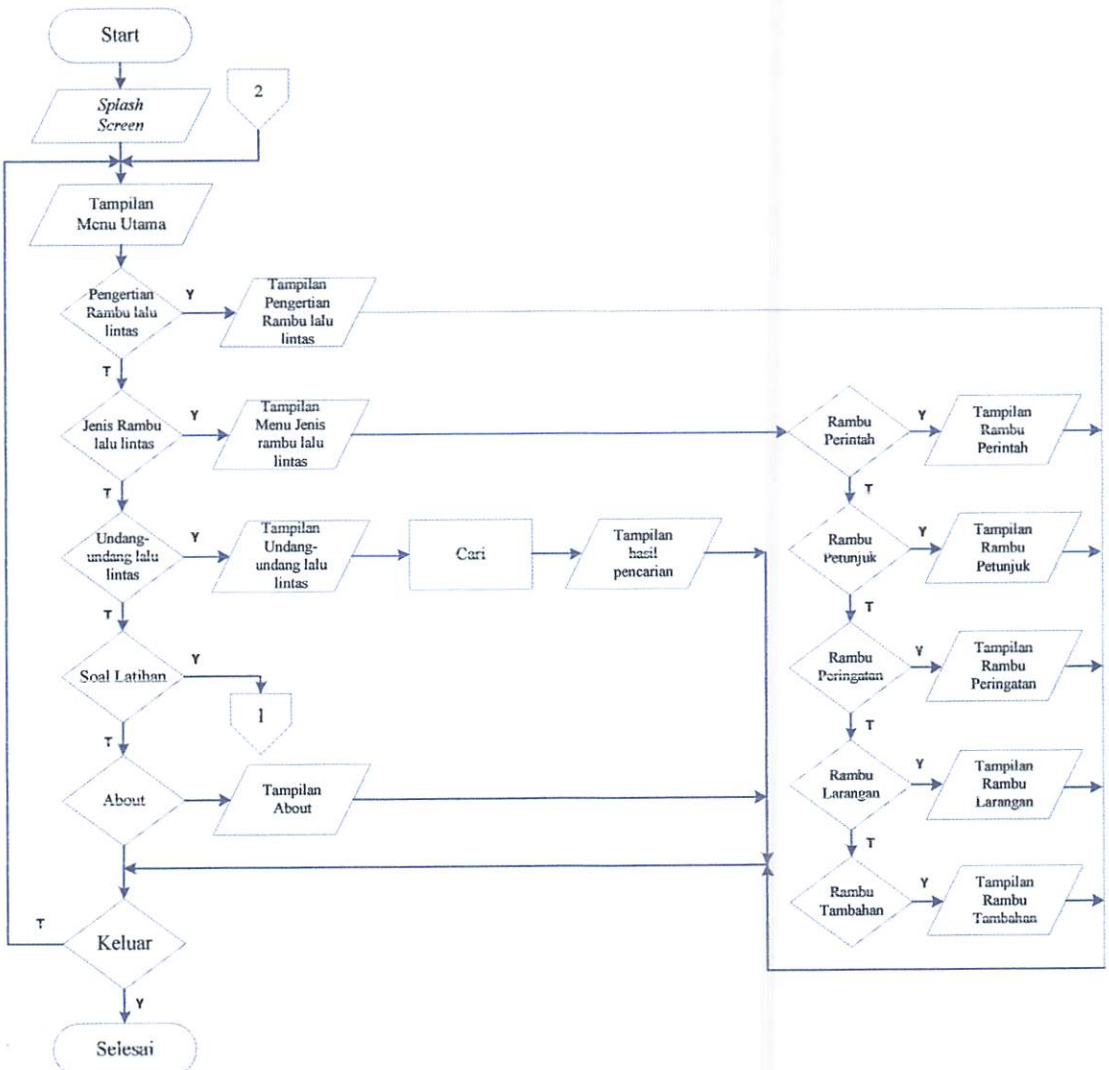
Gambar 3.1 Blok diagram

Blok diagram tersebut menunjukkan interaksi antara *user* atau pengguna dengan aplikasi yang terpasang pada perangkat ponsel. Dimana dalam blok diagram tersebut dapat dijelaskan *user* mengakses atau membuka aplikasi melalui

perangkat ponsel dan memberi perintah dengan memilih menu atau fitur-fitur yang terdapat pada aplikasi. Setelah memberikan perintah (memilih menu), aplikasi akan menampilkan informasi atau data kepada *user* sesuai dengan perintah yang diberikan oleh *user*.

3.2.2 Perancangan *Flowchart* Aplikasi

Pembuatan *flowchart* aplikasi bertujuan memberikan gambaran konsep mengenai apa saja yang ditampilkan dalam aplikasi yang dibuat. Supaya alur kerja dalam aplikasi dapat digambarkan dengan jelas maka perlu dibuat suatu *flowchart* untuk menggambarkan hubungan dalam aplikasi tersebut. Secara garis besar alur kerja dari aplikasi pengenalan rambu berbasis *Android* ini ditunjukkan dengan *flowchart* pada gambar 3.2.

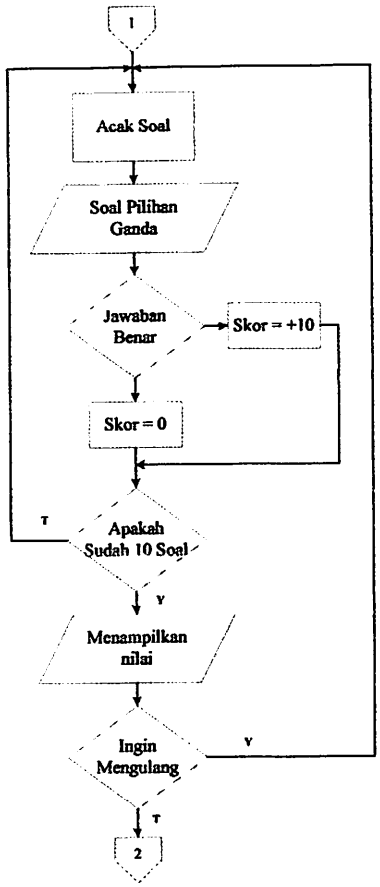


Gambar 3.2 *Flowchart* Aplikasi

Flowchart pada gambar 3.2 menggambarkan pada saat aplikasi dibuka akan dimulai dari halaman pembuka kemudian masuk ke halaman menu utama. Pada halaman menu utama ditampilkan lima pilihan menu yaitu, menu tentang rambu, jenis rambu, undang-undang lalu lintas, dan soal quis. Jika *user* memilih menu tentang rambu maka akan tampil penjelasan tentang rambu lalu lintas, aturan umum, pengertian jenis-jenis rambu lalu lintas, persyaratan bentuk dan warna rambu, dan ke kuatan hukum rambu. Jika pada menu utama *user* memilih menu jenis rambu maka akan tampil menu jenis rambu yaitu, rambu larangan, rambu peringatan, rambu perintah,rambu petunjuk, dan rambu tambahan, dari jenis rambu tersebut jika *user* memilih salah satu maka akan tampil gambar dari jenis rambu yang di pilih beserta dengan penjelasn tentang gambar rambu tersebut dan juga akan disertai dengan suara yang menjelaskan arti dari gambar rambu tersebut. Jika pada menu utama *user* memilih menu undang-undang lalu lintas maka akan tampil *list* pasal undang-undang lalulintas, karena pasal dan isinya banyak *user* bisa mengetikkan pada kotak yang tersedia pasal berapa yang ingin dilihat. Jika *user* memilih menu soal maka akan tampil soal-soal berurutan mulai dari soal pertama sampai soal ke sepuluh, setelah *user* selesai menjawab semua soal-soal maka akan tampil hasil nilai dari jawaban yang benar.

3.2.3 Perancangan *Flowchart* Soal Kuis

Flowchart soal kuis pada gambar 3.3 menjelaskan apabila *user* memilih menu soal maka aplikasi akan menampilkan soal-soal kuis, kemudian *user* harus menjawab soal yang telah tersedia, soal yang tersedia sebanyak 10 soal, untuk satu halama di tampilkan satu soal disertai jawaban pilihan ganda, yang apabila soal yang dijawab benar nilai ditambah 10, jika jawaban salah nilai 0. Jika semua soal sudah selesai dijawab maka aplikasi akan menampilkan hasil nilai yang diperoleh *user* dari jawaban yang benar, jika ingin mengulang nanti akan ada pilihan ya untuk mengulang soal, dan jika tidak ingin mengulang soal maka pilih tidak.

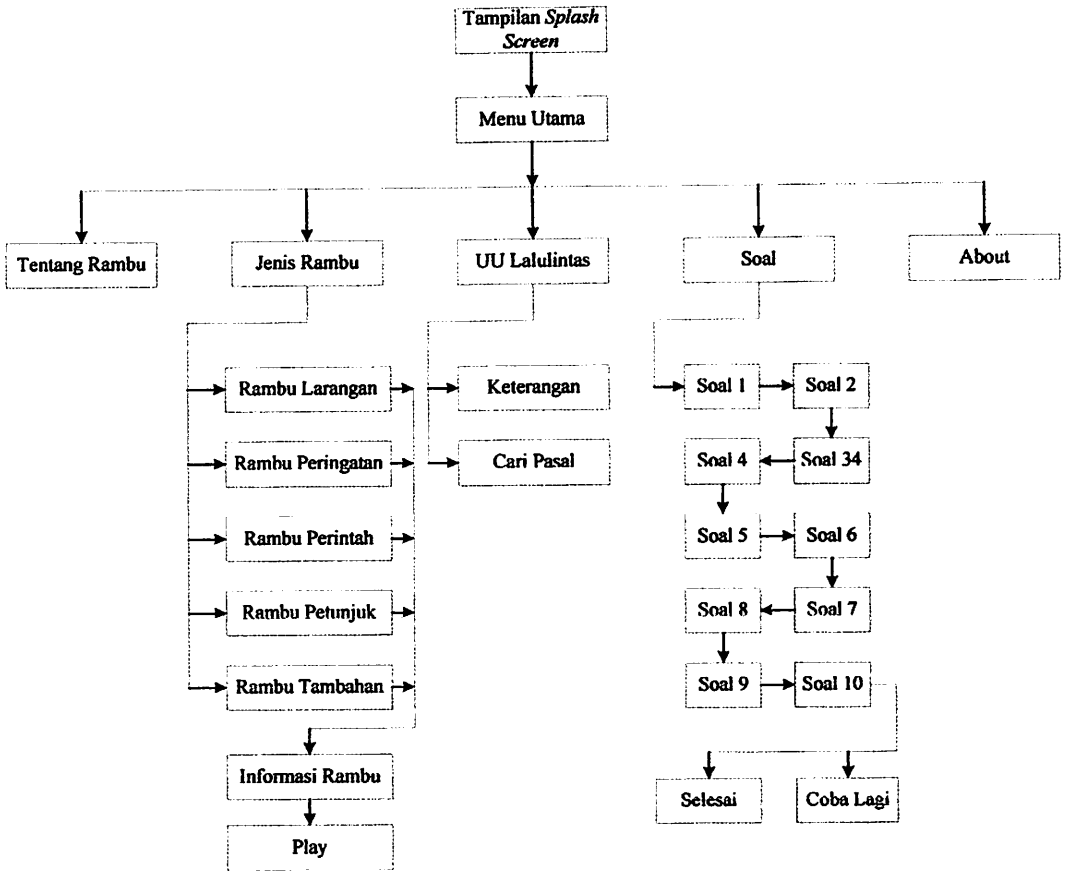


Gambar 3.3 *Flowchart* Kuis

3.2.4 Perancangan Struktur Navigasi Aplikasi

Struktur navigasi merupakan struktur atau alur dari suatu program yang merupakan rancangan hubungan dari beberapa area yang berbeda yang dapat membantu dalam mengorganisasikan seluruh elemen dalam pembuatan program. Navigasi berfungsi untuk membuka halaman-halaman dalam aplikasi. Dengan adanya navigasi memudahkan *user* untuk melakukan interaksi denga tombol-tombol pada aplikasi untuk membuka halaman-halaman pada aplikasi.

Menentukan struktur navigasi bertujuan untuk mempermudah pembuatan aplikasi dalam membuat interaksi pada aplikasi. Karena dengan struktur navigasi dapat digambarkan dengan jelas rencana interaksi dari menu-menu dan sub-sub menu pada aplikasi yang dibuat. Struktur navigasi dalam aplikasi pengenalan rambu lalu lintas berbasis *Android* ini dapat ditunjukkan pada gambar 3.4.



Gambar 3.4 Struktur Navigasi *User*

a. Tampilan Awal

Tampilan awal merupakan halaman awal pada saat aplikasi pertama kali dibuka. Tamapilan awal terhubung dengan halaman menu utama.

b. Menu Utama

Menu utama merupakan halaman utama pada aplikasi ini. Pada halaman menu utama terdapat pilihan-pilihan menu yang merupakan isi dari aplikasi. Halaman menu utama terhubung dengan halaman tentang rambu, jenis rambu, undang-undang lalu lintas, soal dan *about*.

c. Tentang Rambu

Halaman tentang rambu merupakan halaman sub menu dari menu utama. Di dalam halaman tentang rambu ini berisi penjelasan tentang rambu lalu lintas, aturan umum, pengertian jenis-jenis rambu lalu lintas, persyaratan bentuk dan warna rambu, dan ke kuatan hukum rambu.

d. Jenis Rambu

Halaman jenis rambu merupakan halaman sub menu dari menu utama. Dalam halaman jenis rambu terdapat seb menu yaitu menu rambu larangan yang menampilkan gambar-gambar dari rambu larangan beserta penjelasan tulisan dan suara, menu rambu peringatan yang menampilkan gambar-gambar dari rambu peringatan beserta penjelasan tulisan dan suara, menu rambu perintah yang menampilkan gambar-gambar dari rambu perintah beserta penjelasan tulisan dan suara, menu rambu petunjuk yang menampilkan gambar-gambar dari rambu petunjuk beserta penjelasan tulisan dan suara, dan menu rambu tambahan yang menampilkan gambar-gambar dari rambu tambahan beserta penjelasan tulisan dan suara.

e. Undang-undang lalu lintas

Halaman undang-undang lalu lintas merupakan halaman sub menu dari menu utama. Di dalam halaman undang-undang lalu lintas ini berisi undang-undang lalu lintas berupa pasal-pasal dan isinya.

f. Soal

Halaman soal merupakan halaman sub menu dari menu utama. Di dalam halaman soal berisi soal-soal berurutan mulai dari soal pertama sampai soal ke sepuluh secara bergantian.

g. *About*

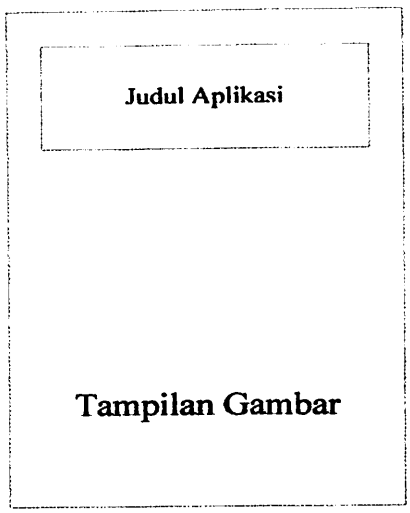
Halaman *About* merupakan sub menu dari menu utama. *About* berisi penjelasan tentang judul dari aplikasi, nama pembuat dan nim.

3.3 Perancangan Halaman Aplikasi

Desain aplikasi merupakan tampilan berupa gambar bagian-bagian halaman dari aplikasi yang akan dibuat yang nantinya akan ditampilkan pada aplikasi tersebut. Berikut ini adalah desain dari halaman-halaman yang akan ditampilkan pada aplikasi :

3.3.1. Halaman Pembuka (*Splashscreen*)

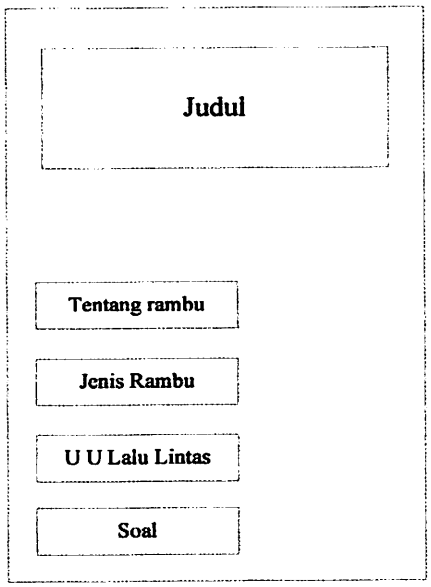
Pada halaman pembuka (*splashscreen*) menampilkan judul dari aplikasi dan gambar sebelum masuk ke halaman berikutnya yaitu halaman menu utama. Gambar 3.5 merupakan gambar desain untuk halaman pembuka (*splashscreen*).



Gambar 3.5 Desain Halaman Pembuka (*splashscreen*)

3.3.2. Halaman Menu Utama

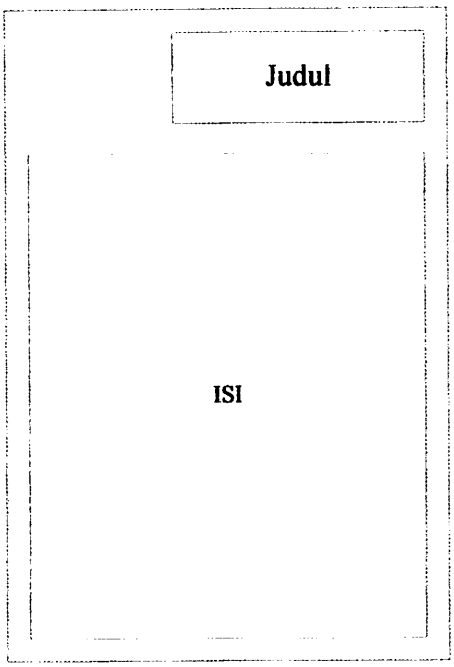
Halaman menu utama merupakan halaman utama pada aplikasi. Pada halaman menu utama menampilkan menu-menu yang terdapat pada aplikasi. Menu-menu tersebut yang nantinya berfungsi langsung untuk masuk ke halaman-halaman isi dari aplikasi. Pada gambar 3.6. menunjukkan bahwa pada halaman menu utama hanya terdapat judul halaman, yaitu menu utama dan ada empat pilihan menu yaitu tentang rambu, jenis rambu, undang-undang rambu dan soal. Keempat tulisan pilhan menu tersebut yang digunakan sebagai tombol untuk masuk ke halaman dari pilihan-pilihan menu tersebut.



Gambar 3.6 Desain Halaman Menu Utama

3.3.3. Halaman Menu Tentang Rambu

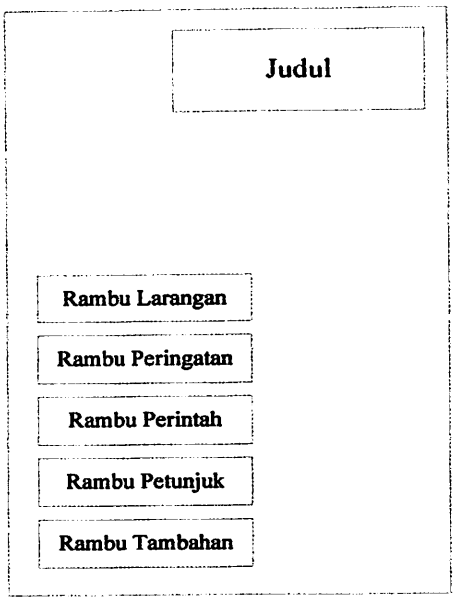
Dalam halaman menu tentang rambu ini berisi penjelasan tentang rambu lalu lintas, aturan umum, pengertian jenis-jenis rambu lalu lintas, persyaratan bentuk dan warna rambu, dan ke kuatan hukum rambu. Untuk gambar menu tentang rambu ditunjukkan pada gambar 3.7.



Gambar 3.7 Desain Halaman Menu Tentang Rambu

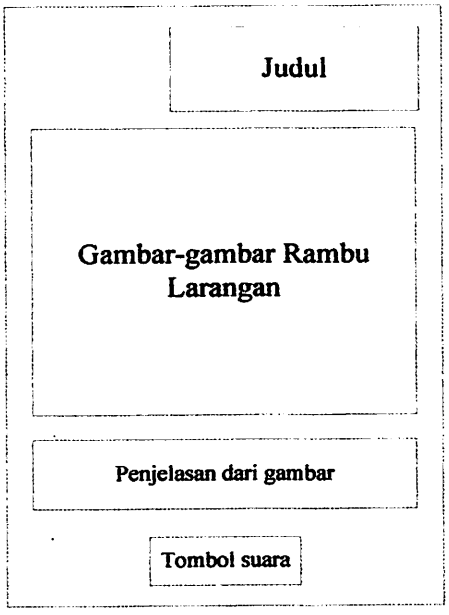
3.3.4. Halaman Menu Jenis Rambu

Dalam menu jenis rambu terdapat lima halaman yang merupakan isi atau konten dari menu jenis rambu tersebut. Pada gambar 3.8 dapat dilihat bahwa pada desain halaman jenis rambu terdapat lima buah sub menu, yaitu menu rambu larangan, rambu peringatan, rambu perintah, rambu petunjuk dan rambu tambahan, dimana dalam setiap sub menu tersebut merupakan tombol-tombol untuk menampilkan informasi dari setiap halaman.



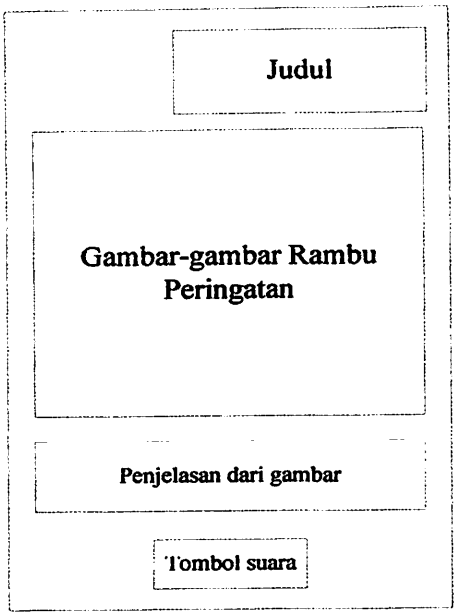
Gambar 3.8 Desain Halaman Menu Jenis Rambu

Sedangkan pada gambar 3.9 menunjukkan desain halaman isi pada menu rambu larangan yang menampilkan gambar-gambar dari rambu larangan beserta penjelasan tulisan dan suara.



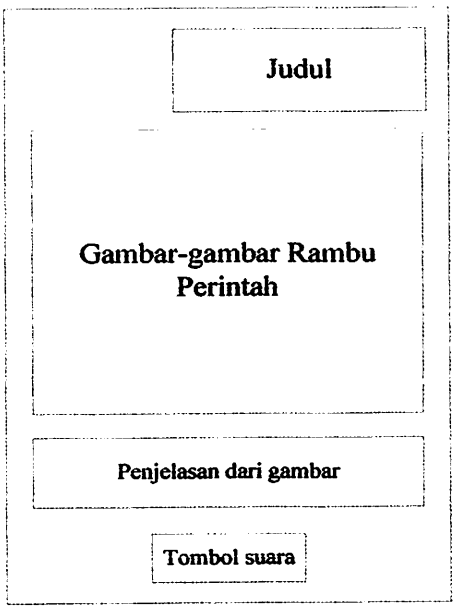
Gambar 3.9 Desain Halaman Menu Rambu Larangan

Sedangkan pada gambar 3.9 menunjukkan desain halaman isi pada menu rambu peringatan yang menampilkan gambar-gambar dari rambu peringatan beserta penjelasan tulisan dan suara.



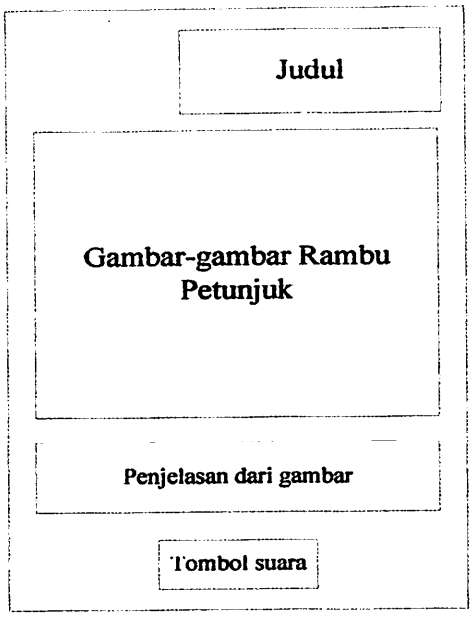
Gambar 3.10 Desain Halaman Menu Rambu Peringatan

Sedangkan pada gambar 3.11 menunjukkan desain halaman isi pada menu rambu perintah yang menampilkan gambar-gambar dari rambu perintah beserta penjelasan tulisan dan suara.



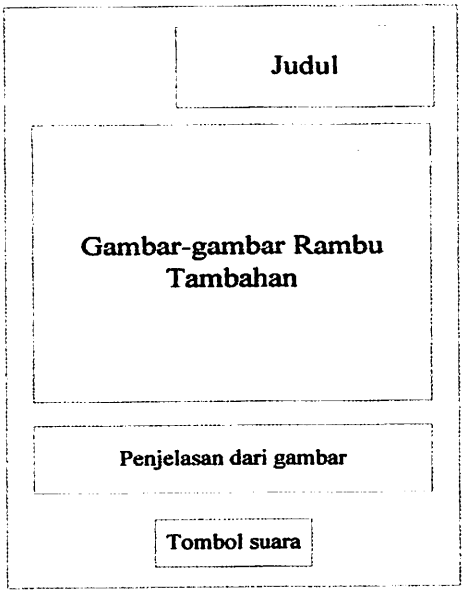
Gambar 3.11 Desain Halaman Menu Rambu Perintah

Sedangkan pada gambar 3.12 menunjukkan desain halaman isi pada menu rambu perintah yang menampilkan gambar-gambar dari rambu perintah beserta penjelasan tulisan dan suara.



Gambar 3.12 Desain Halaman Menu Rambu Petunjuk

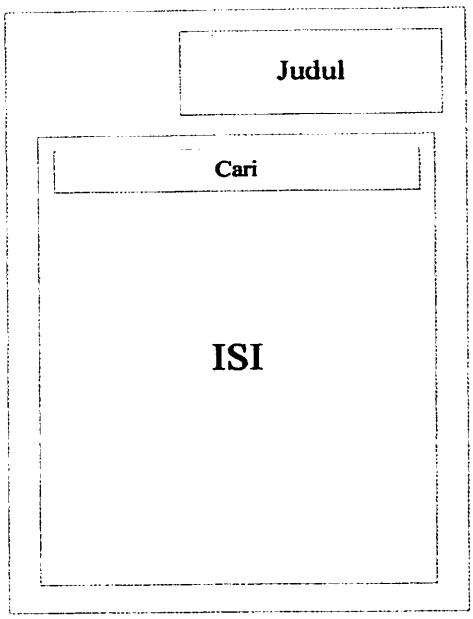
Sedangkan pada gambar 3.13 menunjukkan desain halaman isi pada menu rambu tambahan yang menampilkan gambar-gambar dari rambu tambahan beserta penjelasan tulisan dan suara.



Gambar 3.13 Desain Halaman Menu Rambu Tambahan

3.3.5. Halaman Menu Undang-undang Lalu Lintas

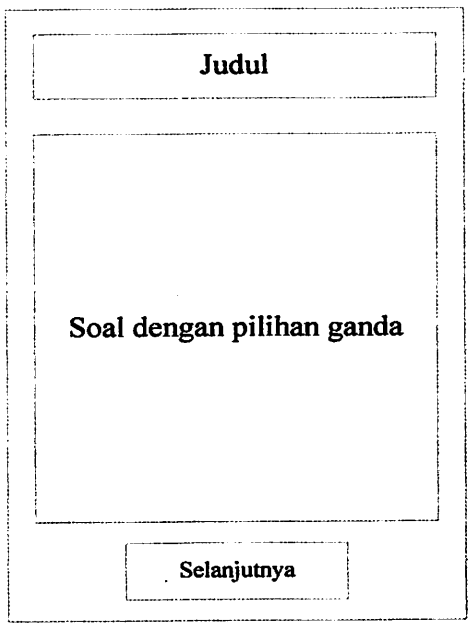
Dalam halaman menu undang-undang lintas ini berisi undang-undang lalu lintas berupa pasal-pasal dan isinya.



Gambar 3.14 Desain Halaman Menu Undang-undang Lalu Lintas

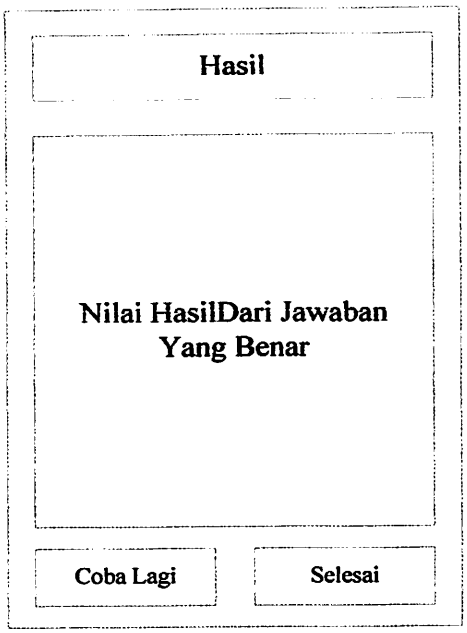
3.3.6. Halaman Menu Kuis

Di dalam halaman soal berisi soal-soal berurutan mulai dari soal pertama sampai soal ke sepuluh. Dimana pada halaman menu soal ini terdapat dua desain tampilan halaman, yaitu desain tampilan halaman soal dan desain tampilan halaman hasil atau nilai.



Gambar 3.15 Desain Halaman Kuis

Pada gambar 3.16 menunjukkan desain halaman hasil atau skor pada menu kuis yang menampilkan nilai dari jawaban yang benar.



Gambar 3.16 Desain Halaman Nilai Hasil

Pada gambar 3.16 dapat dilihat bahwa pada desain halaman nilai hasil terdapat satu tombol coba lagi yang berfungsi untuk mengulang kembali soal, selain itu terdapat tombol selesai yang berfungsi untuk kembali ke menu utama.

BAB IV
IMPLEMENTASI DAN PENGUJIAN APLIKASI

4.1. Implementasi Sistem

Implementasi sistem adalah proses penerapan rancangan sistem yang telah dibuat menjadi suatu aplikasi yang bisa dijalankan pada kenyataannya. *Implementasi* sistem berfungsi untuk menerapkan sistem sesuai dengan tujuan sistem. Disamping *implementasi* berfungsi untuk menerapkan sistem, fungsi lainnya adalah untuk mengetahui tingkat keberhasilan dari rancangan yang telah dibuat.

Aplikasi pengenalan rambu lalu lintas berbasis android ini menggunakan *eclipse* sebagai software utamanya. Didalam aplikasi ini mempunyai empat menu utama yang masing-masing menu terdapat informasi mengenai rambu lalu lintas dan terdapat soal kuis untuk menguji pengetahuan *user*.

4.2. Implementasi *User Interface*

Implementasi *user interface* pada sistem aplikasi *mobile Android* tentang pengenalan rambu lalulintas ini mempunyai 20 (duapuluh) tampilan *interface* yaitu:

1. Tampilan *Splash Screen*
2. Tampilan Menu Utama
3. Tampilan Menu About
4. Tampilan Menu Tentang Rambu
5. Tampilan Menu Tampilan Jenis Rambu
6. Tampilan *List* Rambu Larangan
7. Tampilan Detail Rambu Larangan
8. Tampilan *List* Rambu Peringatan
9. Tampilan Detail Rambu Peringatan
10. Tampilan *List* Rambu Perintah
11. Tampilan Detail Rambu Perintah
12. Tampilan *List* Rambu Petunjuk

13. Tampilan Detail Rambu Petunjuk
14. Tampilan *List* Rambu Tambahan
15. Tampilan Detail Rambu Tambahan
16. Tampilan Keterangan pada Undang-undang Lalulintas
17. Tampilan *List* Pasal Undang-undang lalulintas
18. Tampilan Detail Pasal Undang-undang lalulintas
19. Tampilan Menu Tampilan Soal
20. Tampilan Hasil Nilai Kuis

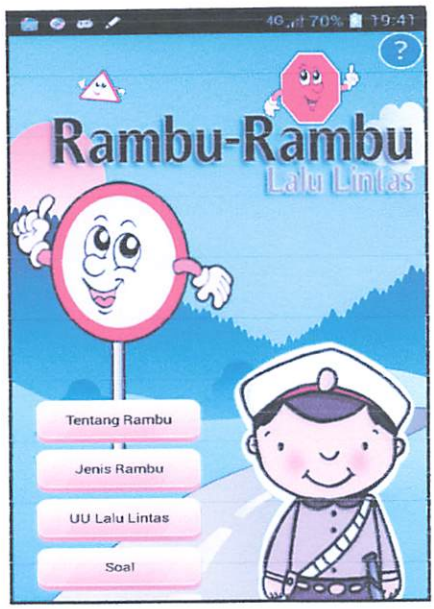
4.2.1. Tampilan Munu *Splash Screen*

Tampilan *splash screen* akan muncul pada saat aplikasi pertama kali dijalankan, setelah lima detik kemudian akan muncul tampilan menu utama. Tampilan *splash screen* ditunjukkan pada gambar 4.1.

Gambar 4.1 Tampilan *Splash Screen*

4.2.2. Tampilan Menu Utama

Tampilan menu utama merupakan tampilan yang keluar setelah *splash screen* selesai dijalankan. Didalam tampilan menu utama terdapat beberapa tombol menu diantaranya menu tentang rambu, menu jenis-jenis rambu, menu undang-undang lalu lintas, dan menu soal kuis dan menu *about* seperti ditunjukkan pada gambar 4.2.



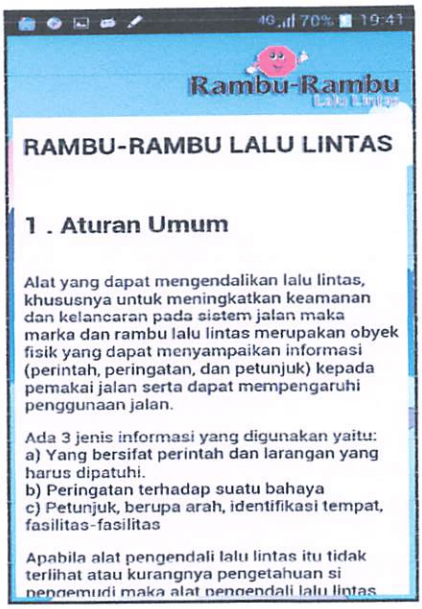
Gambar 4.2 Tampilan Menu Utama

Pada menu utama terdapat sebuah menu yang berisi info tentang judul aplikasi, nama dan nim. Untuk gambar dari tampilan dari menu *about* ditunjukkan pada gambar 4.3.

Gambar 4.3 Tampilan *About*

4.2.3. Tampilan Menu Tentang Rambu

Pada tampilan menu tentang rambu terdapat informasi tentang rambu lalu lintas, seperti penjelasan tentang rambu lalu lintas, aturan umum, pengertian jenis-jenis rambu lalu lintas, persyaratan bentuk dan warna rambu lalu lintas, dan kekuatan hokum rambu. Untuk tampilan menu tentang rambu ditunjukkan pada gambar 4.4.



Gambar 4.4 Tampilan Menu Tentang Rambu

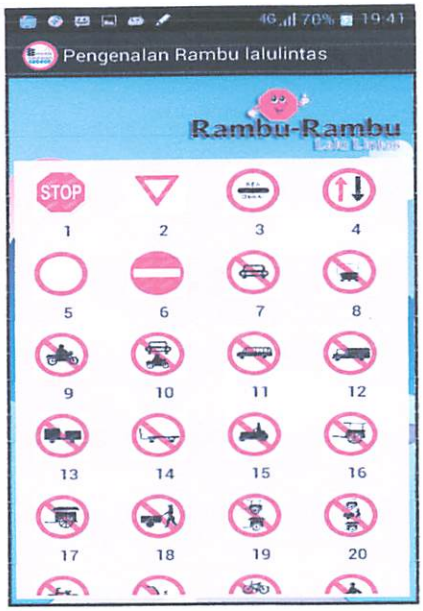
4.2.4. Tampilan Menu Jenis Rambu

Pada tampilan menu jenis rambu terdapat lima sub menu yaitu menu rambu larangan, menu rambu peringatan, menu rambu perintah, menu rambu petunjuk, dan menu rambu tambahan, dimana dalam setiap sub menu tersebut menampilkan informasi berupa gambar dari rambu tersebut beserta penjelasan *teks* maupun penjelasan suara. Untuk tampilan menu jenis rambu ditunjukkan pada gambar 4.5.

Gambar 4.5 Tampilan Menu Jenis Rambu

4.2.5. Tampilan Menu Rambu Larangan

Pada tampilan rambu larangan berisi tampilan galeri dari gambar-gambar rambu larangan sebanyak 50 gambar. Untuk gambar dari tampilan rambu larangan ditunjukkan pada gambar 4.6.



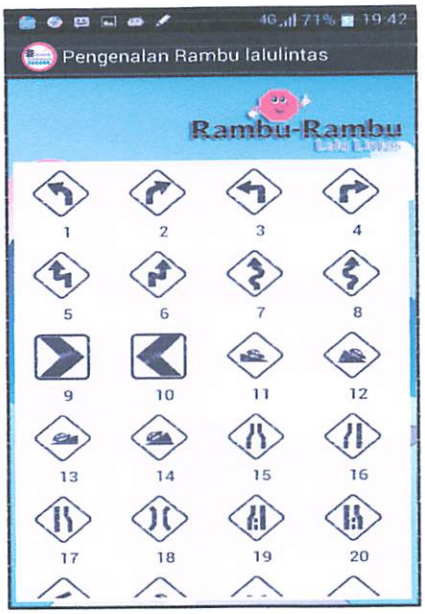
Gambar 4.6 Tampilan Rambu Larangan

Dimana pada setiap gambar dari rambu larangan mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu larangan ditunjukkan pada gambar 4.7.

Gambar 4.7 Keterangan Rambu Larangan

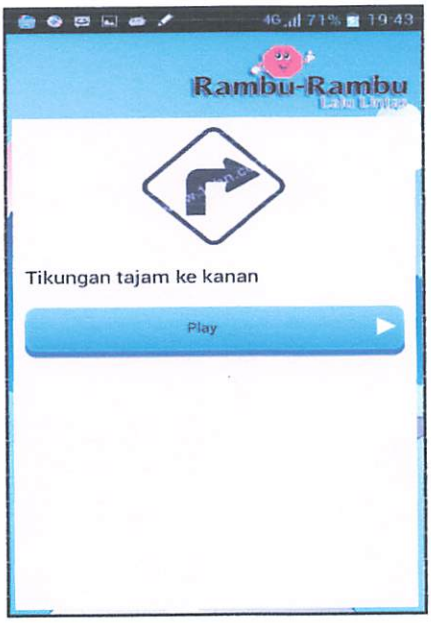
4.2.6. Tampilan Menu Rambu Peringatan

Pada tampilan rambu peringatan berisi tentang tampilan galeri dari gambar-gambar rambu peringatan. Untuk gambar dari tampilan rambu peringatan ditunjukkan pada gambar 4.8.



Gambar 4.8 Tampilan Rambu Peringatan

Dimana pada setiap gambar dari rambu peringatan mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu peringatan ditunjukkan pada gambar 4.9.



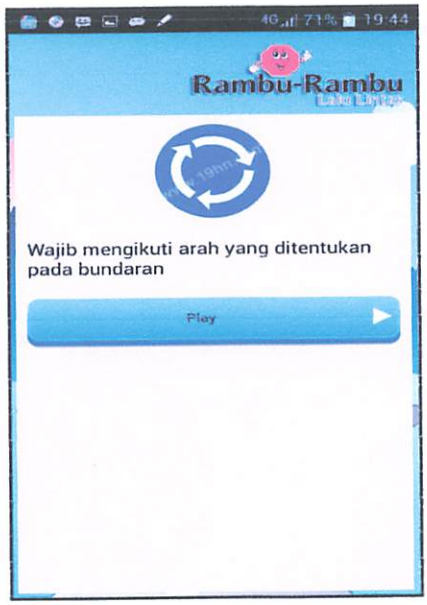
Gambar 4.9 Tampilan Rambu Peringatan

4.2.7. Tampilan Menu Rambu Perintah

Pada tampilan rambu perintah berisi tentang tampilan galeri dari gambar-gambar rambu perintah. Untuk gambar dari tampilan rambu perintah ditunjukkan pada gambar 4.10.

Gambar 4.10 Tampilan Rambu Perintah

Dimana pada setiap gambar dari rambu perintah mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu perintah ditunjukkan pada gambar 4.11.



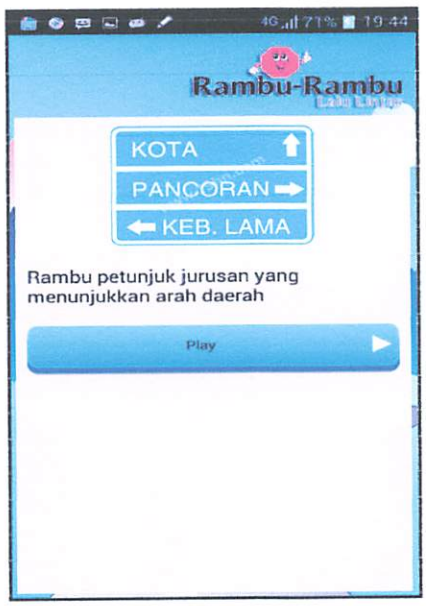
Gambar 4.11 Keterangan Rambu Perintah

4.2.8. Tampilan Menu Rambu Petunjuk

Pada tampilan rambu petunjuk berisi tentang tampilan galeri dari gambar-gambar rambu petunjuk. Untuk gambar dari tampilan rambu petunjuk ditunjukkan pada gambar 4.12.

Gambar 4.12 Tapilan Rambu Petunjuk

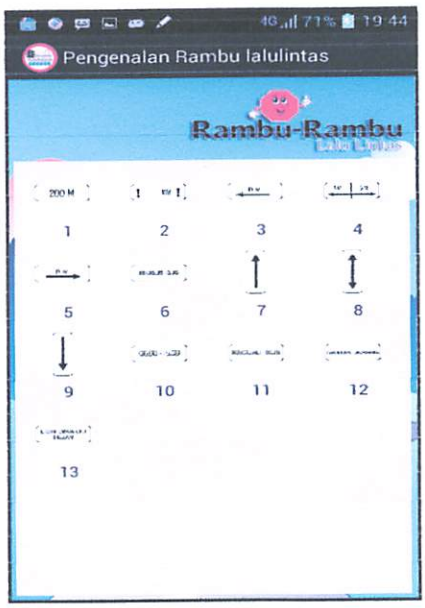
Dimana pada setiap gambar dari rambu petunjuk mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu petunjuk ditunjukkan pada gambar 4.13.



Gambar 4.13 Keterangan Rambu Petunjuk

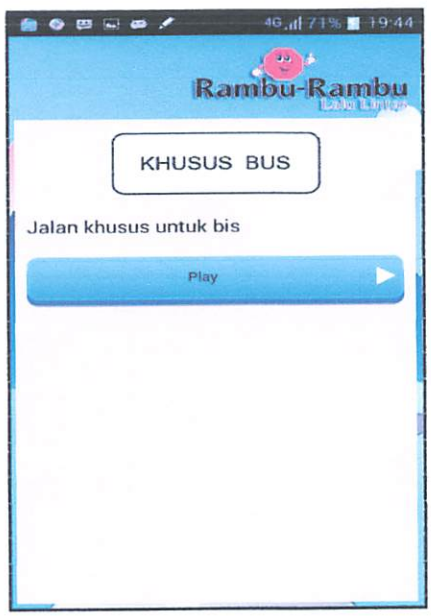
4.2.9. Tampilan Menu Rambu Tambahan

Pada tampilan rambu petunjuk berisi tentang tampilan galeri dari gambar-gambar rambu tambahan. Untuk gambar dari tampilan rambu petunjuk ditunjukkan pada gambar 4.14.



Gambar 4.14 Tampilan Rambu Tambahan

Dimana pada setiap gambar dari rambu tambahan mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu tambahan ditunjukkan pada gambar 4.15.



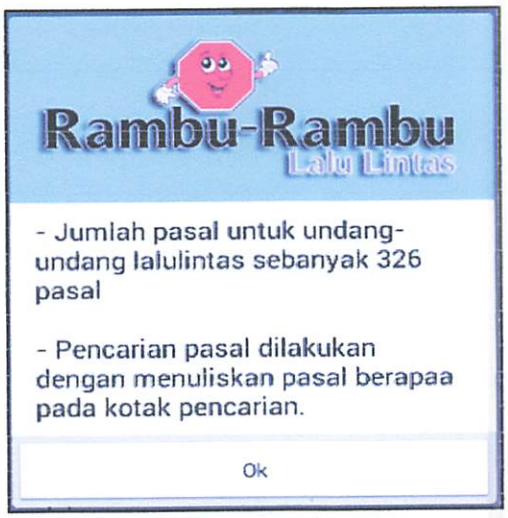
Gambar 4.15 Keterangan Rambu Tambahan

4.2.10. Tampilan Menu Undang-undang Lalulintas

Pada tampilan menu undang-undang lalulintas terdapat informasi tentang undang-undang lalu lintas nomor 22 tahun 2009 berupa pasal-pasal dan isinya. Untuk tampilan menu undang-undang lalulintas ditunjukkan pada gambar 4.16.

Gambar 4.16 Keterangan Undang-undang Lalulintas

Pada menu Undang-undang lalulintas terdapat sebuah menu keterangan yang berisi info tentang jumlah pasal-pasal undang-undang lalulintas, dan cara melakukan pencarian pasal. Untuk gambar dari tampilan dari menu keterangan ditunjukkan pada gambar 4.17.



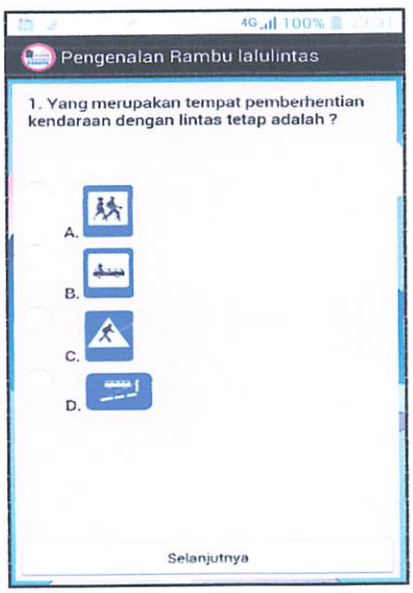
Gambar 4.17 Gambar Keterangan Undang-undang

Dimana pada setiap gambar dari rambu tambahan mempunyai keterangan atau pejelasan berupa teks atau kalimat dan penjelasan berupa suara. Untuk gambar dari penjelasan rambu tambahan ditunjukkan pada gambar 4.18.

Gambar 4.18 Keterangan Hasil Pencarian Pasal

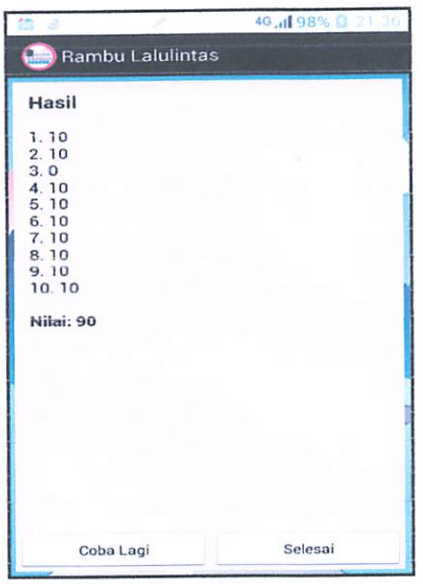
4.2.11. Tampilan Menu Kuis

Pada tampilan menu soal berisi soal-soal pilihan ganda yang akan ditampilkan secara satu persatu mulai dari soal nomer satu sampai soal nomor sepuluh secara berurutan. Untuk gambar dari tampilan soal ditunjukkan pada gambar 4.19.



Gambar 4.19 Gambar Tampilan Soal

Setelah selesai menjawab sepuluh soal yang sudah ditampilkan dengan pilihan ganda, maka nilai akan ditampilkan semua secara berurutan berdasarkan urutan soal yang sudah di jawab. Untuk gambar dari nilai hasil dari soal ditunjukkan pada gambar 4.20.



Gambar 4.20 Nilai Hasil dari Soal

4.3. Pengujian

4.3.1. Pengujian Fungsional Sistem

Pengujian fungsional sistem merupakan proses untuk menemukan adanya kesalahan atau tidak pada aplikasi, sebelum aplikasi tersebut diberikan kepada *user*. Selain itu pengujian ini sangatlah diperlukan untuk mengetahui tingkat keakuratan aplikasi media pembelajaran yang dirancang. Pengujian dikatakan baik dan berhasil jika memiliki peluang untuk memunculkan dan mendapatkan kesalahan yang belum diketahui. Bukan untuk memastikan tidak ada kesalahan tetapi untuk mencari sebanyak mungkin kesalahan yang ada pada sistem.

Pengujian harus dimulai dari lingkup yang kecil ke lingkup yang lebih besar. Pengujian tersebut harus sudah direncanakan jauh sebelum pengujian itu akan dilakukan. Pengujian ini juga harus dilakukan oleh beberapa pihak, jadi tidak hanya pada satu pihak saja, Penguji juga harus dapat menyimpulkan bahwa *system* yang dibangun telah sesuai dengan kasus yang diminta dan telah berjalan dengan baik dan dapat dipahami dengan mudah serta dimengerti oleh tingkat pemahaman *user*.

Tabel 4.1 Tabel Pengujian Fungsional Sistem Aplikasi Pengenalan Rambu Lalu Lintas Dalam Sistem Android.

Pengujian Kesesuain Fungsi dalam Sistem *Android*				***Spesifikasi Hardware***		***Operating System***				
No	**Menu**	**Proses**	**Fungsi button**	**L**	**P**	**A**	**B**	**C**	**D**	**E**
1	Splash Screen	Fadein, Fadeout	✓	X	✓	✓	✓	✓	✓	✓
2	Tombol About	*Link* ke About	✓	X	✓	✓	✓	✓	✓	✓
3	Tombol Menu Tentang Rambu	*Link* ke Tentang Rambu	✓	X	✓	✓	✓	✓	✓	✓
4	Tombol Jenis Rambu	*Link* ke Jenis Rambu	✓	X	✓	✓	✓	✓	✓	✓
5	Tombol Rambu Larangan	*Link* ke Rambu Larangan	✓	X	✓	✓	✓	✓	✓	✓
6	Tombol Rambu Peringatan	*Link* ke Rambu Peringatan	✓	X	✓	✓	✓	✓	✓	✓
7	Tombol Rambu Perintah	*Link* ke Rambu Perintah	✓	X	✓	✓	✓	✓	✓	✓
8	Tombol Rambu Petunjuk	*Link* ke Rambu Petunjuk	✓	X	✓	✓	✓	✓	✓	✓
9	Tombol Rambu Tambahan	*Link* ke Rambu Tambahan	✓	X	✓	✓	✓	✓	✓	✓
10	Tombol Play	*Link* ke Suara	✓	X	✓	✓	✓	✓	✓	✓
11	Tombol Menu UU Lalulintas	*Link* ke UU Lalulintas	✓	X	✓	✓	✓	✓	✓	✓
12	Tombol Keterangan	*Link* ke keterangan	✓	X	✓	✓	✓	✓	✓	✓
13	Tombol Pencarian	*Link* ke pasal	✓	X	✓	✓	✓	✓	✓	✓
14	Tombol Menu Soal	*Link* ke Soal	✓	X	✓	✓	✓	✓	✓	✓
15	Radio Botton	*Link* ke jawaban	✓	X	✓	✓	✓	✓	✓	✓
16	Tombol Selanjutnya	*Link* ke Soal Selanjutnya	✓	X	✓	✓	✓	✓	✓	✓
17	Tombol Coba Lagi	*Link* ke Soal	✓	X	✓	✓	✓	✓	✓	✓
18	Tombol Selesai	*Link* ke Mulai	✓	X	✓	✓	✓	✓	✓	✓

Keterangan simbol:

L = *Landscape Orientation*

P = *Potrait Orientation*

A = *Android Version Frozen Yoghurt (Froyo) Type.* 2.2.1 *Pixel* 320 x 480

B = *Android Version Gingerbread Type.* 2.3.3 *Pixel* 540 x 960

C = *Android Version Ice Cream Sandwich Type.* 4.0.0 *Pixel* 480 x 854

D = *Android Version Jelly Bean Type* 4.2.2 *pixel* 1024 x 600

E = *Android Version KitKat Type* (4.4.0)

✓ = Bisa

X = Tidak Bisa

Pengujian menunjukkan bahwa dari 18 pengujian pada aplikasi Android semuanya berfungsi dengan baik, sehingga memperoleh presentase 100 %.

4.3.2. Pengujian Spesifikasi *Hardware*

Pengujian aplikasi merupakan suatu pengujian dimana aplikasi media pembelajaran berbasis *Android* ini dapat dijalankan ke berbagai spesifikasi *smartphone*. Pengujian ini dilakukan untuk mengetahui *bug-bug* yang terjadi ketika menjalankan aplikasi *Android* ini.

Dari *Data* uji tabel 4.1. menjelaskan bahwa aplikasi ini dibuat dengan orientasi layar *portrait* yang dapat berjalan 100 % pada *smartphone* yang mempunyai *pixel* sebesar 240 x 320 sampai dengan *pixel* 1080 x 1920. Dan pada orientasi *landscape* gagal dikarenakan aplikasi ini hanya dibuat hanya untuk layout *portrait.*

4.3.3. Pengujian *Operating System Android*

Pengujian *Operating System Android* aplikasi Pengenalan Rambu Lalulintas ini. Pada pengujian menggunakan 5 macam *Operating System Android* dengan *type* yang berbeda. Seperti pada tabel 4.1 menjelaskan bahwa aplikasi Pengenalan Rambu Lalulintas ini dari 5 *Operating System* pada *Android* yang diujikan dapat berjalan di hampir semua *Operating System* antara versi *Froyo* (2.2) hingga ke versi *KitKat* (4.4) dengan rata – rata dari tingkat keakuratan hingga mencapai 100 %.

4.3.4. Pengujian User

Pengujian *user* pada aplikasi Pengenalan Rambu Lalulintas ini dilakukan dengan memberikan beberapa pertanyaan kepada *user* yang didasarkan atas

pengujian sistem aplikasi. Pengujian *user* ini dilakukan kepada 20 orang *responden* untuk memberikan penilaian terhadap aplikasi Pengenalan Rambu Lalulintas ini. Adapun hasil dari pengujian *user* ini ditunjukkan pada tabel dibawah ini.

Tabel 4.2. Hasil Pengujian sistem kepada pengguna *(user)*.

No	Pertanyaan	Baik	Cukup	Kurang
1	Tampilan	17	3	0
2	Kinerja Aplikasi	16	4	0
3	Manfaat Aplikasi	17	3	0
4	Tingkat Kemudahan Aplikasi	18	2	0
5	Fungsi Aplikasi Sebagai Media Pembelajaran	18	2	0
Jumlah		86	14	0

Prosentase Pengujian Pengguna (*user*)

$$\frac{jumlah\ responden}{jumlah\ soal} \times 100\%$$

1. Baik $\frac{86}{100} \times 100\ \% = 86\ \%$
2. Cukup $\frac{14}{100} \times 100\ \% = 14\ \%$
3. Kurang $\frac{0}{100} \times 100\ \% = 0\ \%$

Keterangan :

Pengujian dari lima pertanyaan yang diberikan kepada 20 *user* mengenai aplikasi pengenalan rambu lalulintas ini menunjukkan bahwa dari 20 *user* yang memilih baik sebanyak 86 poin dari 100 poin sehingga memperoleh prosentase 86 %, dan yang memilih cukup sebanyak 14 poin dari 100 poin sehingga memperoleh prosentase 14 %, dan 0 poin atau 0 % untuk yang memiih kurang.

Dari pengujian aplikasi kepada pengguna *(user)* dapat di tarik kesimpulan aplikasi pengenalan rambu lalulintas berbasis *Android* ini dapat diterima oleh para orang tua yang menggunakan aplikasi pengenalan rambu lalulintas ini sebagai alternatif untuk mengenalkan rambu lalulintas kepada anak.

BAB V

PENUTUP

5.1. Kesimpulan

Dari penelitian yang dilakukan maka diperoleh kesimpulan sebagai berikut:

1. Aplikasi ini dibuat dengan orientasi layar *portrait* yang dapat berjalan 100 % pada *smartphone* yang mempunyai *pixel* sebesar 240 x 320 sampai dengan *pixel* 1080 x 1920.
2. Aplikasi pengenalan rambu lalulintas ini dapat berjalan pada 5 *Operating System Android* yang telah diujikan, antara versi *Froyo* (2.2) hingga ke versi *KitKat* (4.4) dengan rata – rata dari tingkat keakuratan hingga mencapai 100 %.
3. Pengujian aplikasi pengenalan rambu lalulintas ini menunjukkan bahwa dari 20 *user* yang memilih baik sebanyak 86 %, dan yang memilih cukup sebanyak 14 %, dan 0 % untuk yang memiih kurang.

5.2. Saran

Dalam pembuatan aplikasi pengenalan rambu lalulintas berbasis Android ini masih jauh dari sempurna. Karena masih banyaknya kekurangan - kekurangan yang perlu untuk ditambahkan antara lain :

1. Aplikasi ini dapat dikembangkan lagi supaya aplikasi dapat dioperasikan selain *portrait* dapat juga di operasikan dengan *landscape*.
2. Aplikasi ini dapat di kembangkan lagi dengan menambahkan fasilitas video sertas animasi yang dapat diterapkan.

DAFTAR PUSTAKA

Afandi, Faisal. Rambu Lalulintas Jalan di Indonesia.

Huda, Arif Akbarul. "*24 Jam Pintar Pemrograman Android*". Yogyakarta :PENERBIT ANDI, 2012.

Kadir, Abdul. 2005. *Dasar Pemrograman Java 2.* Yogyakarta : ANDI Offset..

Kevin, Levandi Temotius. 2011. *Penggunaan Ternary Search Tree dalam Melakukan Autocomplete.* Program Study Teknik Informatika. Bandung.

Siregar, Ivan Michel. 2011. *Membongkar Source Code Berbagai Aplikasi ANDROID.* Yogyakarta : Gava Media.

Suprianto, Dodit dan Rini Agustina. 2012. *Pemograman Aplikasi Android.* Yogyakarta: MediaKom.

Syafaat, Nazarudin. 2013. *Aplikasi berbasis Android.* IF Informatika.

Undang-undang Lalulintas No.22 tahun 2009

LAMPIRAN

LAMPIRAN

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK
PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG
Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. RAYA Karanglo, Km2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 21 April 2014

Nomor : ITN-253/INF/TA/2014
Lampiran : ---
Perihal : *Bimbingan Skripsi*

Kepada : Yth. Bpk/Ibu Ir. Yusuf Ismail Nakhoda, MT
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama	:	DANAN EKATAMA KHUSNA
Nim	:	1018157
Prodi	:	Teknik Informatika S-1
Fakultas	:	Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 April 2014 S/D 21 September 2014

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
K e t u a,

Joseph Dedy Irawan,ST., MT.
NIP : 197404162005021002

Form S-4a

Nama : [illegible]
Kelompok : --
Perihal : [illegible]

Kepada .

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK
Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. RAYA Karanglo, Km2 Telp. (0341) 417636 Fax. (0341) 417634 Malang
T. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Malang, 21 April 2014

Nomor : ITN-253/INF/TA/2014
Lampiran : ---
Perihal : *Bimbingan Skripsi*

Kepada : Yth. Bpk/Ibu Febriana S.W.S.Kom.M.Kom
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : DANAN EKATAMA KHUSNA
Nim : 1018157
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 April 2014 S/D 21 September 2014

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan,ST., MT.
NIP : 197404162005021002

Form S-4a

INSTITUTE TEKNOLOGI NASIONAL MALANG
Fakultas Teknologi Nasional Malang
Program Studi Teknik Informatika S1

FORMULIR BIMBINGAN SKRIPSI

Nama : Danan Ekatama Khusna
Nim : 1018157
Masa Bimbingan : 04 Mei 2014 s/d 20 Agustus 2014
Judul Skripsi : Aplikasi Pengenalan Rambu Lalulintas Berbasis Android

No	Tanggal	Uraian	Paraf
1	13-05-2014	Bimbingan Bab 1 dan bab 2	
2	04-06-2014	Bimbingan Desain Tampilan Aplikasi	
3	26-06-2014	Bimbingan bab 3	
4	16-07-2014	Bimbingan bab 3 dan Aplikasi	
5	24-07-2014	Bimbingan Bab 4 dan bab 5	
6	11-08-2014	Bimbingan Aplikasi	
7	14-08-2014	Bimbingan Makalah + Aplikasi	
8	20-08-2014	Bimbingan Bab 1-5 + Aplikasi	

Malang, 20 Agustus 2014
Dosen Pembimbing 1

Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 1018800189

INSTITUTE TEKNOLOGI NASIONAL MALANG

Fakultas Teknologi Nasional Malang

Program Studi Teknik Informatika S1

FORMULIR BIMBINGAN SKRIPSI

Nama : Danan Ekatama Khusna

Nim : 1018157

Masa Bimbingan : 04 Mei 2014 s/d 20 Agustus 2014

Judul Skripsi : Aplikasi Pengenalan Rambu Lalulintas Berbasis Android

No	Tanggal	Uraian	Paraf
1	13-05-2014	Bimbingan Bab 1 dan bab 2	
2	04-06-2014	Bimbingan Desain Tampilan Aplikasi	
3	25-06-2014	Bimbingan Bab 3 dan Aplikasi	
4	11-07-2014	Bimbingan Bab 3	
5	12-07-2014	Bimbingan Bab 3 dan 4	
6	23-07-2014	Bimbingan Bab 4, 5 dan Aplikasi	
7	11-08-2014	Bimbingan Makalah Skripsi	
8	19-08-2014	Bimbingan Laporan Bab 1-5	

Malang, 20 Agustus 2014
Dosen Pembimbing 2

Febriana Santi W, SKom, MKom

NIP. P. 10310000425

INSTITUTE TEKNOLOGI NASIONAL MALANG

Fakultas Teknologi Nasional Malang

Program Studi Teknik Informatika S1

FORMULIR BIMBINGAN SKRIPSI

Nama : Danan Ekatama Khusna
Nim : 1018157
Masa Bimbingan : 04 Mei 2014 s/d 20 Agustus 2014
Judul Skripsi : Aplikasi Pengenalan Rambu Lalulintas Berbasis Android

Tanggal	Penguji	Uraian	Paraf
21-08-2014	I	- Tabel 2.3, 2.5, 2.6 disesuainkan dengan gambar - Perbaiki kesimpulan no 2 - Lampirkan beberapa quisioner terisi	
21-08-2014	II	- Abstrak - Tabel pengujian diganti menjadi user, bukan presentase - Pengujian ditambah menjadi 20 user - Kesimpuan no 3 diperbaiki	

Anggota Penguji :

Penguji I

Ali Mahmudi, BEng,Phd
NIP.P. 1031000429

Penguji II

Sonny Prasetio, ST, MT
NIP.P. 1031000433

Mengetahui

Dosen Pembimbing I

Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 1018800189

Dosen Pembimbing II

Febriana Santi W, SKom, MKom
NIP. P. 10310000425

ANGKET SKRIPSI

JUDUL : APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID

NAMA : Sutomo
ALAMAT : Ds Slorok RT 01 RW 02
PEKERJAAN : Swasta

TANDA TANGAN

1) Menurut anda bagaimana user interface atau tampilan dari aplikasi ini ?
   a) Baik
   b) Cukup
   c) Kurang
2) Bagaimana tanggapan anda tentang kinerja dari aplikasi pengenalan rambu lalulintas ini ketika aplikasi dijalankan ?
   a) Baik
   b) Cukup
   c) Kurang
3) Bagaimana tanggapan anda tentang tingkat kemudahan anak-anak dalam pemahaman aplikasi ini dari segi tampilan maupun hasil yang diberikan ?
   a) Baik
   b) Cukup
   c) Kurang
4) Bagaimana tanggapan anda tentang fungsi aplikasi pengenalan rambu lalulintas ini sebagai sarana pembelajaran atau pengenalan kepada anak-anak ?
   a) Baik
   b) Cukup
   c) Kurang
5) Bagaimana tanggapan anda tentang materi dari aplikasi pengenalan rambu lalulintas ini ?
   a) Baik
   b) Cukup
   c) Kurang

ANGKET SKRIPSI

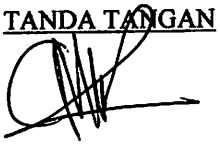
JUDUL : APLIKASI PENGENALAN RAMBU LALULINTAS BERBASIS ANDROID

NAMA : MOHAMAD AKBAR KURNIAWAN

ALAMAT : Jl. Kalimantan Dsn. Pc. Sari Ds. Slorok Garum

PEKERJAAN : WIRASWASTA

TANDA TANGAN



1) Menurut anda bagaimana user interface atau tampilan dari aplikasi ini ?
   a) Baik
   b) Cukup
   c) Kurang
2) Bagaimana tanggapan anda tentang kinerja dari aplikasi pengenalan rambu lalulintas ini ketika aplikasi dijalankan ?
   a) Baik
   b) Cukup
   c) Kurang
3) Bagaimana tanggapan anda tentang tingkat kemudahan anak-anak dalam pemahaman aplikasi ini dari segi tampilan maupun hasil yang diberikan ?
   a) Baik
   b) Cukup
   c) Kurang
4) Bagaimana tanggapan anda tentang fungsi aplikasi pengenalan rambu lalulintas ini sebagai sarana pembelajaran atau pengenalan kepada anak-anak ?
   a) Baik
   b) Cukup
   c) Kurang
5) Bagaimana tanggapan anda tentang materi dari aplikasi pengenalan rambu lalulintas ini ?
   a) Baik
   b) Cukup
   c) Kurang

Lampiran 1. Source Code untuk menampilkan pasal undang undang

```
package com.example.skripsi;
import java.io.File;
import java.io.IOException;
import com.example.skripsi_danan.R;
import com.example.skripsi_danan.R.id;
import android.app.Activity;
import android.content.res.AssetManager;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebView;
import android.widget.Toast;

public class DetailActivityUU extends Activity{
	@Override
	protected void onCreate(Bundle savedInstanceState) {
		// TODO Auto-generated method stub
		super.onCreate(savedInstanceState);
		setContentView(R.layout.list_data_detail);
		String lowerCase = getIntent().getStringExtra("D");
		if (lowerCase!=null) {
			WebView webView = (WebView)
findViewById(id.webView1);
			webView.setBackgroundColor(0xFFFFEFD5);
			if (!assetExists(getAssets(),
"pasal/"+lowerCase.replace(" ", "_")+".htm" )) {
				webView.setVisibility(View.GONE);
				Toast.makeText(getApplicationContext(),
"Pasal tidak ditemukan.", Toast.LENGTH_LONG).show();
				return;
			}
			try {
				webView.setVisibility(View.VISIBLE);

	webView.loadUrl("file:///android_asset/pasal/"+lowerCase.replace(
" ", "_")+".htm" );
				webView.setBackgroundColor(0xFFFFEFD5);

			}catch (Exception ex) {
				webView.setVisibility(View.GONE);
				Toast.makeText(getApplicationContext(),
"Pasal tidak ditemukan.", Toast.LENGTH_LONG).show();
			}
		}
	}
	private static boolean assetExists(AssetManager assets, String
name) {
	  try {
		File f = new File(name);
		String parent = f.getParent();
		if (parent == null) parent = "";
		String fileName = f.getName();
		// now use path to list all files
		String[] assetList = assets.list(parent);
```

```
            if (assetList != null && assetList.length > 0) {
                for (String item : assetList) {
                    if (fileName.equals(item))
                        return true;
                }
            }
        } catch (IOException e) {
            // Log.w(TAG, e); // enable to log errors
        }
        return false;
    }
}
```

Lampiran 2. Source Code untuk melakukan pencarian (*Autocomplete*)

```
public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.list_data_list);

        Intent I = getIntent();
        MODE = I.getStringExtra("MODE");

               autoCompleteTextView = (AutoCompleteTextView)
findViewById(id.autoCompleteTextView1);
        String[] dataComplete = new String[326];
        for (int j = 0; j < dataComplete.length; j++) {
              dataComplete[j] = "Pasal" + (j+1);
        }
        ArrayAdapter<String> adapters = new
ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,dataComple
te);
             autoCompleteTextView.setAdapter(adapters);

        bntCari = (ImageButton) findViewById(id.btnCari);
        bntCari.setOnClickListener(new View.OnClickListener() {

                    @Override
                    public void onClick(View v) {
                          // TODO Auto-generated method stub
      loadData(autoCompleteTextView.getText().toString().trim().toLower
Case());
                    }
             });

        Button btnKet = (Button) findViewById(id.button1);
        btnKet.setOnClickListener(new View.OnClickListener() {

                    @Override
                    public void onClick(View v) {
                          // TODO Auto-generated method stub
                          showKet();
                    }
             });

        ArrayList<HashMap<String, String>> dataList = new
ArrayList<HashMap<String, String>>();

        for (int j = 0; j < 326; j++) {

              HashMap<String, String> tmp = new HashMap<String,
String>();
              tmp.put("A", "Pasal " + (j+1));
              dataList.add(tmp);
             }
             ListView lvData = (ListView) findViewById(id.listView1);
          ListAdapter adapter = new SimpleAdapter(ListUUActivity.this,
dataList , R.layout.list_row_list,
                new String[] { "A"  },
```

```
                new int[] {  R.id.detail  });
            lvData.setAdapter(adapter);

            lvData.setOnItemClickListener(new OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View view,
int position, long id) {
                    loadData(("pasal
"+(position+1)).toString().trim().toLowerCase());
                    }
            });
```

Lampiran 3. Source Code Tombol Play

```
public class DetailActivity extends Activity {
    private String MODE;
      private String KET;
      private String ID;
      MediaPlayer m;

       AssetFileDescriptor descriptors ;
      /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        this.requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.detail);
        Intent I = getIntent();
        MODE = I.getStringExtra("MODE");
        KET = I.getStringExtra("KET");
        ID = I.getStringExtra("ID");

        KET = KET.split(":")[1];
        Button btn = (Button) findViewById(id.btnPlay);
        btn.setOnClickListener(new View.OnClickListener() {

                    @Override
                    public void onClick(View v) {
                          // TODO Auto-generated method stub
                          playBeep( );
                    }
             });
        ImageView img = (ImageView) findViewById(id.imgData);
        TextView txtKet = (TextView) findViewById(id.txtDetail);
        txtKet.setText(KET);
        m = new MediaPlayer();
        try {
             descriptors = getAssets().openFd(MODE+"/"+(ID)+".mp3");
                  m.setDataSource(descriptors.getFileDescriptor(),
descriptors.getStartOffset(), descriptors.getLength());
               descriptors.close();
               m.prepare();
               m.setVolume(1f, 1f);
               m.setLooping(false);
        } catch ( Exception e) {
                    // TODO Auto-generated catch block
              Toast.makeText(getApplicationContext(), e.toString(),
Toast.LENGTH_LONG).show();;
                    e.printStackTrace();
              }

       Bitmap homeIcon = getBitmapFromAsset(MODE+"/"+(ID)+".gif");
       img.setImageBitmap(homeIcon);
    }
    public void playBeep(  ) {
        try {
```

```
            if (m.isPlaying()) {
                m.stop();
            }
            m.start();
        } catch (Exception e) {
             Toast.makeText(getApplicationContext(), e.toString(),
Toast.LENGTH_LONG).show();;
            e.printStackTrace();
        }
    }
    @Override
    protected void onPause() {
       // TODO Auto-generated method stub
        if (m.isPlaying()) {
              m.stop();
        }
       super.onPause();
    }
    @Override
    protected void onDestroy() {
       // TODO Auto-generated method stub
       super.onDestroy();
    }
    public  Bitmap getBitmapFromAsset(String strName)
    {
        AssetManager assetManager = getAssets();
        InputStream istr = null;
        try {
            istr = assetManager.open(strName);
        } catch (IOException e) {
            e.printStackTrace();
        }
        Bitmap bitmap = BitmapFactory.decodeStream(istr);
        return bitmap;
    }
}
```

Lampiran 4. Source Code Soal

```
package com.example.skripsi;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Collections;
import com.example.skripsi_danan.R;
import com.example.skripsi_danan.R.id;
import android.app.Activity;
import android.content.Context;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.webkit.WebView;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.ScrollView;
import android.widget.Toast;

public class TestActivity extends Activity {

	WebView webViewSoal,webViewHasil ;
	ScrollView scrollView;

	WebView[] wVJawab;
	RadioButton[] RbJawab;
	Button btnNext,btnSelesai;
	ArrayList<Integer> idSoal;
	ArrayList<TSoal> listSoal;

	RadioGroup radioGroup;
	int CurrSoal;
	private int maxSoal=10;

	@Override
	protected void onCreate(Bundle savedInstanceState) {
		// TODO Auto-generated method stub
		super.onCreate(savedInstanceState);
		setContentView(R.layout.soal);

		wVJawab = new WebView[4];
		RbJawab = new RadioButton[4];

		scrollView = (ScrollView) findViewById(id.scrollView1);
		webViewHasil = (WebView) findViewById(id.webView4);

		webViewHasil.setVisibility(View.GONE);
		btnNext  = (Button) findViewById(id.btnNext);
		btnSelesai  = (Button) findViewById(id.btnSelesai);
```

```
            btnSelesai.setVisibility(View.GONE);
            RbJawab[0] = (RadioButton) findViewById(id.radio0);
            RbJawab[1] = (RadioButton) findViewById(id.radio1);
            RbJawab[2] = (RadioButton) findViewById(id.radio2);
            RbJawab[3] = (RadioButton) findViewById(id.radio3);

            webViewSoal = (WebView) findViewById(id.webViewSoal);
            wVJawab[0] = (WebView) findViewById(id.webView0);
            wVJawab[1] = (WebView) findViewById(id.webView1);
            wVJawab[2] = (WebView) findViewById(id.webView2);
            wVJawab[3] = (WebView) findViewById(id.webView3);

            ArrayList<String> tmplistSoal =
loadArrayFromAsset(getApplicationContext(),  "soal.txt");
            listSoal = new ArrayList<TestActivity.TSoal>();
            idSoal = new ArrayList<Integer>();
        for (int j = 0; j < tmplistSoal.size(); j++) {
            String strs = tmplistSoal.get(j);
             if (!strs.equals("")){
                    listSoal.add(new TSoal(strs));
                    idSoal.add(idSoal.size());
             }
            }
        Collections.shuffle(idSoal);
        CurrSoal = 0;
            ShowSoal();
            btnNext.setOnClickListener(new View.OnClickListener() {

                  @Override
                  public void onClick(View v) {
                        // TODO Auto-generated method stub
                        if (((Button)
v).getText().equals("Selanjutnya")) {
                              doNext();
                        }else{
                              reNew();
                        }
                  }
            });
            for (int i = 0; i < RbJawab.length; i++) {

                  RbJawab[i].setOnClickListener(new
View.OnClickListener() {
                        @Override
                        public void onClick(View v) {
                              // TODO Auto-generated method stub
                              unCheck();
                              RadioButton rb = (RadioButton) v;
                              rb.setChecked(true);
                        }
                  });
                   final int idx =i;
                   wVJawab[i].setOnTouchListener(new
View.OnTouchListener() {

                        @Override
                        public boolean onTouch(View v, MotionEvent
```

```
event) {
                                    // TODO Auto-generated method stub
                                    unCheck();
                                    RbJawab[idx].setChecked(true);
                                    return false;
                              }
                        });
                  }
            }
            public void doSelesai(View v) {
                  finish();
            }
            protected void reNew() {
                  // TODO Auto-generated method stub
                  CurrSoal=0;
                  unCheck();
                  Collections.shuffle(idSoal);
              CurrSoal = 0;
              ShowSoal();
              btnNext.setText("Selanjutnya");
              webViewHasil.setVisibility(View.GONE);
              btnSelesai.setVisibility(View.GONE);
              scrollView.setVisibility(View.VISIBLE);
            }
            public void unCheck() {
                  for (int i = 0; i < RbJawab.length; i++) {
                        RbJawab[i].setChecked(false);
                  }
            }
            protected void doNext() {
                  // TODO Auto-generated method stub
                  boolean isAns = false;
                  for (int i = 0; i < RbJawab.length; i++) {
                        if (RbJawab[i].isChecked()) {
                              listSoal.get(idSoal.get(CurrSoal)).Pilih =
String.valueOf( ((char) (i+65))).toLowerCase();
                              isAns =true;
                              break;
                        }

                  }
                  if (listSoal.get(CurrSoal).isBenar()) {
                  }
                  if (isAns) {
                        if (CurrSoal+1 < listSoal.size() && CurrSoal+1<
maxSoal ) {
                              CurrSoal++;
                              unCheck();
                              ShowSoal();
                        }else {
                              webViewHasil.setVisibility(View.VISIBLE);
                              scrollView.setVisibility(View.GONE);

                              final String mimeType = "text/html";
                          final String encoding = "utf-8";
                          String html = "<h3>Hasil</h3>" ;
                          int nilai =0;
```

```
                        for (int i = 0; i < maxSoal; i++) {
                           html += (1+i)+". ";
                           if (listSoal.get(idSoal.get(i)).isBenar()) {
                                  html +=  ""+10;
                                  nilai +=10;
                                  }else {
                                         html +=  ""+0;
                                         nilai +=0;
                                  }
                           html +=  "<br>"  ;
                           }
                        html +=  "<br><b>Nilai: "+nilai +"</b>";

webViewHasil.loadDataWithBaseURL("fake://not/needed", html, mimeType,
encoding, "");
                        webViewHasil.setBackgroundColor(0xFFFFEFD5);
                        btnNext.setText("Coba Lagi");
                        btnSelesai.setVisibility(View.VISIBLE);

                     }
              }else{
                     Toast.makeText(getApplicationContext(), "Silahkan
pilih salah satu jawaban.", Toast.LENGTH_LONG).show();
              }
       }
       public void ShowSoal() {

              final String mimeType = "text/html";
           final String encoding = "utf-8";
           String html = (CurrSoal+1)+".
"+listSoal.get(idSoal.get(CurrSoal)).Soal ;

           webViewSoal.loadDataWithBaseURL("fake://not/needed", html,
mimeType, encoding, "");
           webViewSoal.setBackgroundColor(0xFFFFEFD5);
              for (int i = 0; i < wVJawab.length; i++) {
                     html =     ((char) (i+65)) + ".
"+listSoal.get(idSoal.get(CurrSoal)).Jawab[i] ;
                     wVJawab[i].loadDataWithBaseURL("fake://not/needed",
html, mimeType, encoding, "");
                     wVJawab[i].setBackgroundColor(0xFFFFEFD5);
              }
       }
       class TSoal  {
              String Soal;
              String[] Jawab = new String[4];
              String Benar;
              String Pilih;
              public TSoal(String Datas) {

                     // TODO Auto-generated constructor stub

                     String[] tmpData = Datas.toString().split("#") ;
                     Soal = tmpData[1].trim();
                     Benar = tmpData[2].trim();

                     Jawab[0] = tmpData[3].trim();
```

```
                    Jawab[1] = tmpData[4].trim();
                    Jawab[2] = tmpData[5].trim();
                    Jawab[3] = tmpData[6].trim();

            }

            public boolean isBenar() {
                        return Benar.equals(Pilih);
            }
      }

       public ArrayList<String> loadArrayFromAsset(Context c, String
name) {
                    ArrayList<String> ret= new ArrayList<String>();

                    try {

                          InputStream inputStream =
c.getAssets().open(name);
                          BufferedReader r = new BufferedReader(new
InputStreamReader(inputStream));
                          String line;
                          while ((line = r.readLine()) != null) {
                                ret.add(line);
                          }
                          inputStream.close();
                          r.close();

                    } catch (IOException ex) {
                          ex.printStackTrace();
                          return null;
                    }
                    return ret;
             }

}
```

RIWAYAT HIDUP

Danan Ekatama Khusna dilahirkan di Blitar pada tanggal 26 Juni 1991, putra pertama dari pasangan bapak Sutomo dan ibu Sun Rofi'ah. Pendidikan dasar ditempuh di MI RN Pucung Sari Lor lulus pada tahun 2004, melanjutkan ke SMP Negeri 1 Garum lulus pada tahun 2007, dan pendidikan selanjutnya ditempuh di SMK Islam 1 Kota Blitar lulus pada tahun 2010.

Pada tahun 2010 menempuh pendidikan perguruan tinggi di Institute Teknologi Nasional Malang (ITN) dan mengambil program studi Teknik Informatika S-1 lulus pada tahun 2014 dengan gelar Sarjana Komputer (S,Kom).