

HALAMAN PENGESAHAN

**APLIKASI TERJEMAHAN BAHASA INDONESIA – BAHASA TOMBULU
MENGUNAKAN VOICE READ TEXT BERBASIS ANDROID**

Oleh

RIO CAHYACHRISTY SANGEROKI

NIM : 11 024 126

Tugas Akhir ini telah diterima dan disahkan sebagai persyaratan
untuk menyelesaikan pendidikan Program **Diploma IV** Jurusan Teknik Elektro
Program Studi Teknik **Informatika**
Politeknik Negeri Manado

Manado, Agustus 2015

Ketua Panitia Tugas Akhir

Dosen Pembimbing

Fanny Jouke Doringin, ST. MT
NIP.19670403 199203 1 003

Edwin Lumunon, ST, MIT
NIP. 19740121 199802 1001

Mengetahui,
Ketua Jurusan Teknik Elektro

Ir. Jusuf Luther Mappadang. MT
NIP. 19610601 199003 1 002

PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Rio Cahyacristy Sangeroki

NIM : 11 024 126

Program Study : Teknik Informatika

Menyatakan dengan sebenarnya bahwa Tugas Akhir yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan tulisan atau pemikiran orang lain. Apabila kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan Tugas Akhir ini hasil karya orang lain, saya bersedia menerima sanksi atas perbuatan saya.

Manado, Agustus 2015

Yang menyatakan

Rio Cahyacristy Sangeroki

ABSTRAK

Sangeroki,C.Rio.2015.*Aplikasi terjemahan bahasa Indonesia ke bahasa Tombulu menggunakan voice read text berbasis Android*. Tugas Akhir. Jurusan Teknik Elektro. Program Study Teknik Informatika.

Penelitian ini dilatarbelakangi oleh banyaknya masyarakat yang ada di kota Tomohon yang sudah tidak tahu berbicara bahasa khas Tomohon yaitu bahasa Tombulu. Oleh karena itu dibutuhkan fasilitas yang bisa digunakan oleh masyarakat kota Tomohon untuk dapat belajar bahasa Tombulu terlebih juga dapat memberikan pengetahuan bagi wisatawan yang datang ke kota Tomohon.

Tujuan penelitian ini untuk membuat suatu aplikasi berbasis android yang dapat digunakan oleh pengguna android untuk belajar bahasa Tombulu dengan mudah dari Handphone dan dapat digunakan oleh banyak kalangan masyarakat yang ingin belajar bahasa Tombulu.

Metode yang digunakan dalam penelitian ini adalah metode Waterfall dimana terdapat beberapa tahap yang dapat digunakan untuk membangun aplikasi yang akan dibuat yaitu pengumpulan dan pengolahan data bahasa Tombulu, bahasa Indonesia, bahasa Inggris yang berguna untuk mengolah data tersebut ke Aplikasi agar lebih efisien dan terstruktur. Pada pembuatan aplikasi Kamus bahasa ini menggunakan Eclipse IDE sebagai tools pembuatan aplikasi berbasis android.

Berdasarkan hasil penelitian dapat disimpulkan yaitu aplikasi dapat melakukan proses input data bahasa Inggris dan bahasa Indonesia, serta dapat melakukan proses pencarian data bahasa Inggris dan bahasa Indonesia menggunakan voice read text sesuai dengan kata atau bahasa yang ingin diterjemahkan dan dapat menerjemahkannya langsung ke bahasa Tombulu.

Kata kunci: Aplikasi terjemahan bahasa Tombulu, Android

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah Yang Maha Esa, karena atas rahmat dan karuni-Nya sehingga penulis dapat menyelesaikan penyusunan Tugas Akhir ini dengan Judul **“APLIKASI TERJEMAHAN BAHASA INDONESIA KE BAHASA TOMBULU DENGAN VOICE READ TEXT BERBASIS ANDROID”**

Adapun tujuan dari penulisan tugas akhir ini yaitu sebagai salah satu syarat untuk menyelesaikan pendidikan Diploma IV program studi Teknik Informatika Jurusan Teknik Elektro di Politeknik Negeri Manado.

Dalam penulisan tugas akhir ini penulis banyak menghadapi berbagai hambatan dan tantangan, namun berkat bantuan dari berbagai pihak sehingga penulis dapat menyelesaikan Tugas Akhir ini, untuk itu pada kesempatan ini penulis ingin menyampaikan banyak terima kasih yang sebesar – besarnya kepada :

1. Yesus Kristus karena dengan segala berkat, pertolongan dan kehendak-Nya, penulis dapat menyelesaikan penulisan tugas akhir ini.
2. Orang Tua dan Keluarga yang tak henti – hentinya memberikan segala kebutuhan untuk penulis, baik berupa doa, semangat maupun materi.
3. Bapak Ir. Jemmy J Rangan, MT selaku Direktur Politeknik Negeri Manado
4. Bapak Ir. Jusuf L Mapadang, MT selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Manado
5. Bapak Fanny J Doringin, ST.MT selaku Ketua Panitia Tugas Akhir.
6. Bapak Edwin Lumunon, ST.MIT selaku dosen pembimbing yang telah membimbing penulis dalam penyelesaian tugas akhir ini.
7. Seluruh dosen dan staff administrasi jurusan Teknik Elektro Politeknik Negeri Manado

8. Bapak Rein Heydemans yang selalu membantu penulis dalam menyelesaikan tugas akhir.
9. Ria Sangeroki selaku kakak yang selalu membantu dan memberikan semangat bagi penulis
10. Semua teman – teman se angkatan Teknik Elektro 2011
11. Rukun keluarga GBTC yang selalu mendoakan dan mengingatkan untuk segera menyelesaikan tugas akhir.
12. Andre Rondonuwu selaku sahabat yang selalu membantu disaat penulis berada dalam kesulitan saat menyelesaikan tugas akhir
13. Greggy Rarung dan Ripka Legi yang sudah menyediakan fasilitas berupa tempat tinggal dalam menyelesaikan tugas akhir
14. Maya Lioew dan Indiani Atika Laode yang telah membantu dalam penyelesaian pembuatan tugas akhir
15. Torang – torang group yang selalu memberikan semangat, penghiburan dan motivasi bagi penulis
16. Dan untuk seluruh pihak yang telah memberikan bantuan dan dukungan yang tidak dapat penulis cantumkan satu persatu.

Penulis juga menyadari bahwa dalam penyusunan tugas akhir ini masih terdapat kekurangan dan kesalahan, maka dari itu penulis meminta maaf untuk kesalahan dalam penyusunan tugas akhir ini.

Harapan penulis agar sekiranya tugas akhir ini dapat bermanfaat bagi semua pembaca. Terima Kasih

Manado, Agustus 2015

Rio Cahyacristy Sangeroki

DAFTAR ISI

LEMBAR PENGESAHAN	
ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	vi
DAFTAR TABEL	vii

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan Penelitian.....	2
1.4. Manfaat Penelitian	2
1.5. Ruang Lingkup Dan Batasan Masalah.....	3
1.6. Metodologi Penulisan.....	3
1.7. Sistematika Penulisan.....	4

BAB II TINJAUAN PUSTAKA

2.1. Android	5
2.1.1 Android: Platform Masa Depan	7
2.1.2 The Dalvik Virtual Machine (DVM)	9
2.1.3 Versi Android.....	11
2.1.4 Libraries... ..	15
2.1.4 Tool dalam Pemrograman Android.....	16
2.1.5.1 Java SE (Standart Edition).....	17
2.1.5.2 Eclipse.....	17
2.1.5.3 Android SDK... ..	19
2.1.5.4 Android Virtual Device Manager (AVD).. ..	20
2.1.5.5 JDK/JRE... ..	20
2.1.5.6 FRAMEWORK-RES.APK... ..	21
2.1.5.7 Android Package... ..	21
2.1.5.8 Android Development Tools (ADT)... ..	21
2.1.5.9 Arsitektur Android... ..	23
2.1.5.10 Fundamental Aplikasi.. ..	25

2.2. Database	27
2.2.1 Bahasa Pada Database	27
2.2.2 Structured Query Language	28
2.2.3 SQLite	29
2.2.3.1 Fitur-fitur	29
2.2.3.2 Integrasi Dengan Bahasa Lain	30
2.2.3.3 SQLite Manager	31

BAB III METODOLOGI DAN PERANCANGAN

3.1 Waktu dan Tempat	32
3.2 Kerangka Konseptual	32
3.2.1 Metode Pengumpulan Data	33
3.2.1.1 Sumber Data Primer	33
3.2.1.2 Sumber Data Sekunder	34
3.2.2 Metode Pembuatan Aplikasi	34
3.2.3 Metode Perancang Interface	34
3.2.4 Implementasi & Testing	35
3.3 Diagram Alur (Flowchart)	35
3.4 Komponen User Interface (UI)	36
3.5 AndroidManifest.xml	39
3.6 Activity	43
3.6.1 Siklus Activity	43
3.7 Android Widget	48
3.8 DataKamus.Java	50
3.9 ShowKamus.Java	51

BAB IV PENGUJIAN DAN IMPLEMENTASI SISTEM

4.1	I
Implementasi Program	55
4.2	T
Ampilan Akhir Kamus	55

BAB V KESIMPULAN DAN SARAN

5.1. Kesimpulan	58
5.2. Saran	58
DAFTAR PUSTAKA	59

DAFTAR GAMBAR

Gambar 2.1 Tampilan <i>Interface</i> Eclipse.....	19
Gambar 2.2 Tampilan <i>Interface</i> AVD.....	20
Gambar 2.3 Arsitektur Android	25
Gambar 2.4 Tampilan <i>Interface</i> SQLite Manager.....	32
Gambar 3.1 Metodologi Waterfall	32
Gambar 3.2 Flowchart	35
Gambar 3.3 Menampilkan Interface	36
Gambar 3.4 Lifecycle Activity... ..	44
Gambar 4.1 Tampilan Interface Kamus	36
Gambar 4.2 Hasil Terjemahan.....	44
Gambar 4.3 Hasil terjemahan yang tidak tersedia.....	44

DAFTAR TABEL

Tabel 3.1 Lifecycle Activity	46
Tabel 3.2 Jeni-jenis (Classes) Android Widgets... ..	49
Tabel 4.1 Daftar Uji Coba... ..	49

PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini:

Nama : Deanty Mokodompit
Nim : 11 024 012
Program Study : Teknik Informatika

Menyatakan dengan sebenarnya bahwa Tugas Akhir yang saya tulis ini benar- benar merupakan hasil karya sayasendiri, bukan merupakan pengambilalihan tulisan atau pemikiran orang lain. Apabila kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan Tugas Akhir ini hasil karya orang lain, saya bersedia menerima sanksi atas perbuatan saya.

2015

Manado, Agustus

Yang menyatakan

Deanty Mokodompit

LEMBAR PENGESAHAN

**SISTEM INFORMASI ONLINE PERPUSTAKAAN
PROVINSI SULAWESI UTARA BERBASIS
SMS GATEWAY**

Oleh

Deanty Mokodompit

NIM : 11 024 012

*Tugas Akhir ini telah diterima dan disahkan sebagai persyaratan untuk
menyelesaikan Pendidikan **Diploma IV** Teknik Elektro*

*Program Studi Teknik **Informatika***

Politeknik Negeri Manado

Manado, Agustus 2015

Ketua Panitia Tugas Akhir,

Dosen Pembimbing,

Fanny Jouke Doringin, ST.,MT
NIP. 19670430 199203 1 003

I Gede Para Atmaja, ST.,MT
NIP. 196901301993031003

Mengetahui
Ketua Jurusan Teknik Elektro,

Ir Jusuf Luther Mappadang,MT
NIP.196106011990031002

KATA PENGANTAR

Segala puji syukur dipanjatkan kehadirat Allah Swt yang maha sumber segala ilmu dan pengetahuan, yang telah memberikan hikmat, tuntunan, penyertaan serta lindungan, selama penulis menyelesaikan studi di jurusan Teknik Elektro Politeknik Negeri Manado, sehingga penulis dapat menyelesaikan pembuatan tugas akhir ini. Meskipun dalam penyusunan tugas akhir ini, banyak menghadapi berbagai macam hambatan, rintangan dan tantangan yang harus dilalui, tetapi berkat rahmat dan tuntunan dari Allah Swt dan dukungan dari berbagai pihak sehingga tugas akhir ini dapat terselesaikan.

Penulis menyadari bahwa dengan keterbatasan kemampuan yang ada sehingga penulisan jauh dari kesempurnaan dan masih banyak kekurangan. Oleh karena itu dengan segala kerendahan hati penulis mengharapkan kritik dan saran yang bersifat membangun dari semua pihak yang bertujuan kearah penyempurnaan tugas akhir ini.

Selama proses penyelesaian tugas akhir ini banyak ditunjang dengan bantuan tenaga, pemikiran baik moral maupun material dari banyak pihak. Oleh karena itu, sepantasnyalah bila pada kesempatan ini dengan segala kerendahan hati penulis menyampaikan banyak-banyak terimakasih yang setulus-tulusnya kepada:

1. Ir. Jemmy J. Rangan, MT. SelakuDirekturPoliteknikNegeri Manado
2. Ir. JusufL.Mappadang, MT selakuKetuaJurusanTeknikElektro
3. Ir. Nikita A.E. Sajangbati, SelakuKetua Program StudiTeknikInformatika
4. I Gede Para Atmaja , ST,. MT, Selaku dosen pembimbing yang telah memberikan banyak masukan sehingga tugas akhir ini dapat terselesaikan.
5. Dosen pengajar di TeknikElektroterlebihkhususdosenTeknikInformatika yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada penulis.
6. KeduaOrang tua sertaadikyang telah memberikandukunganbaikdalamDoa maupunmaterisehingga proses penulisan tugasakhir ini dapat terselesaikan.

7. Kepada Muhammad Indra yang selalumembantudanmemberikanperhatiankepadapenulisselamapenyusunantugasakhirinidapatterselesaikan.
8. Kepada teman penulis, Stieven, Livi, Devrit, Jimmy dan temanlainnyayang selalumemberikan support danbanyakmembantudalampenyusunanTugasAkhirini.
9. Kepada rekan-rekan mahasiswa Teknik Informatika D-IV yang telah banyak membantu dalam proses penyelesaian tugas akhir ini.

Dengan segala kerendahan hati, penulis berharap semoga apa yang tertulis dalam tugas akhir ini dapat bermanfaat bagi semua pihak.

Manado, Agustus 2015

Deanty Mokodompit

ABSTRAK

Mokodompit, Deanty.2015. *Sistem Informasi Online Perpustakaan Berbasis SMS Gateway*. Tugas Akhir. Jurusan Teknik Elektro. Program Studi Teknik Informatika.

Saat ini pada kantor Badan Perpustakaan Arsip dan Dokumentasi Provinsi Sulawesi Utara dalam melakukan pengisian data sudah menggunakan suatu aplikasi Ms.Excel yang berguna untuk menyimpan data buku yang ada. Tetapi pada kenyataannya aplikasi tersebut belum sesuai dengan yang diharapkan dan pengisian data masih terlalu lama dan hanya membuang-buang waktu dalam pemakaiannya. Dan pencatatan peminjaman buku yang masih manual sehingga sering terjadi data ganda dan lama pengembalian buku oleh peminjam karena faktor lupa. Untuk itu dibutuhkan suatu sistem yang dapat melakukan pengolahan data secara cepat dan efektif untuk menunjang kinerja setiap pegawai dalam melakukan pekerjaannya.

Tujuan dari penelitian ini adalah untuk mengembangkan sistem pengolahan data buku dan peminjaman yang ada di kantor BPAD Provinsi Sulawesi Utara khususnya untuk data buku yang digunakan, serta memberikan informasi peringatan jadwal pengembalian buku dan informasi keterlambatan pengembalian kepada peminjam.

Metode yang digunakan dalam penelitian ini adalah metode siklus hidup pengembangan sistem atau System Development Life Cycle (SDLC) di mana terdapat beberapa tahap yang dapat digunakan untuk membangun aplikasi yang akan dibuat yaitu pengolahan data buku dan data peminjaman pada BPAD Provinsi Sulut yang berguna untuk mengolah data lebih efisien dan terstruktur.

Berdasarkan hasil penelitian, maka dapat disimpulkan yaitu aplikasi dapat melakukan proses penginputan data, menampilkan laporan peminjaman dan pengembalian serta memberikan informasi peringatan jadwal pengembalian buku kepada peminjam menggunakan SMS Gateway.

Kata kunci: Aplikasi, Perpustakaan, Pengolahan data..

DAFTAR ISI

LEMBAR PERNYATAAN	i
LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
ABSTRAK	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian Tugas Akhir.....	3
1.5. Metodologi Penulisan	4
1.6. Sistematika Penulisan Tugas Akhir	4
BAB II TINJAUAN PUSTAKA.....	6
2.1. Penelitian Sebelumnya Sebagai Gagasan	6
2.2. Pengertian Sistem	8
2.3. Pengertian Informasi dan Data	8

2.4. Pengolahan Data	8
2.5. Basis Data (Databases)	9
2.6. Sistem Basis Data	10
2.7. Diagram Konteks	11
2.8. Data Flow Diagram (DFD)	11
2.9. Hierarchy Input Proses Output (HIPO)	13
2.10. ERD	13
2.11. HTML	16
2.12. PHP	16
2.13. MYSQL	17
2.14. Pengenalan SMS Gateway	18
2.15. Aplikasi SlekiSMS	20
BAB III METODOLOGI PENELITIAN	21
3.1. Tempat dan Waktu Penelitian	21
3.2. Metode Penelitian	22
3.3. Analisa Masalah	29
3.4. Analisa Kebutuhan Sistem	29
3.5. Perancangan Sistem	30
3.5.1. Database Sistem Informasi Perpustakaan	31
3.5.2. Entity Relation Database	38
3.5.3. Database Model	39
3.5.4. Perancangan Flowchart	39

3.5.5. Diagram Konteks	45
3.5.6. Data Flow Diagram (DFD)	45
3.5.7. Use Case Diagram	48
3.6. Perancangan Antarmuka (Interface)	52
3.7. Perancangan Pengujian	72
BAB IV PEMBAHASAN DAN PENGUJIAN	59
4.1. Pembahasan Aplikasi	59
4.2. Pengujian Aplikasi	69
BAB V PENUTUP	81
5.1. Kesimpulan	84
5.2. Saran	84
DAFTAR PUSTAKA	85
LAMPIRAN	

DAFTAR GAMBAR

1. Gambar 2.1 Sistem Pengolahan Data	9
2. Gambar 2.2 Simbol Proses	14
3. Gambar 2.3 Simbol Harddisk	15
4. Gambar 2.4 Simbol Kondisi	15
5. Gambar 2.5 Simbol Terminator	15
6. Gambar 2.6 Simbol Garis Alir	15
7. Gambar 2.7 Simbol Data	16
8. Gambar 2.8. Ilustrasi Sms Gateway	18
9. Gambar 2.9. Diagram sms gateway	19
10. Gambar 2.10. Arsitektur sms gateway	19
11. Gambar 2.11. Aplikasi Sleki SMS untuk android	20
12. Gambar 3.1 Pengembangan Software Metode Waterfall	23
13. Gambar 3.2. Blok diagram metodologi penelitian	26
14. Gambar 3.3 Entity Relation Database	38
15. Gambar 3.4 Model database perpustakaan	39
16. Gambar 3.5 Flowchart Sistem Informasi Perpustakaan	40
17. Gambar 3.6 Flowchart Sistem Informasi Perpustakaan	41
18. Gambar 3.7 Flowchart Sistem Informasi Perpustakaan	42
19. Gambar 3.8. Flowchart program auto send sms peringatan	43

20. Gambar 3.9. Flowchart program auto send sms pemberitahuandenda	44
21. Gambar 3.10. Gambar Diagram Konteks Sistem Informasi Perpustakaan	45
22. Gambar 3.11. Gambar DFD Level 0 Sistem Informasi Perpustakaan .	46
23. Gambar 3.12. Gambar DFD Level 1 Sistem Informasi Perpustakaan .	46
24. Gambar 3.13. Gambar DFD Level 2 Sistem Informasi Perpustakaan .	47
25. Gambar 3.14. Gambar DFD Level 3 Sistem Informasi Perpustakaan ..	48
26. Gambar 3.15. Gambar Use case Sistem Informasi Perpustakaan	49
27. Gambar 3.16. Form Menu Pengunjung	50
28. Gambar.3.17 Form Login Admin	51
29. Gambar.3.18 Form Menu Utama Admin	51
30. Gambar.3.19 Form Input Data Klasifikasi	52
31. Gambar.3.20 Form Input Data Rak	52
32. Gambar.3.21 Form Input Data Buku	53
33. Gambar.3.22 Form Input Data Peminjaman Buku	54
34. Gambar.3.23 Form Input Data Pengembalian Buku	55
35. Gambar.3.24 Form Input Data Anggota	56
36. Gambar.3.25 Form Data Laporan	57
37. Gambar 4.1. Tampilan Form Login	59
38. Gambar 4.2. Tampilan Menu Admin	60
39. Gambar 4.3. Tampilan Data Buku	60

40. Gambar 4.4. Tampilan Data Klasifikasi.....	61
41. Gambar 4.5. Tampilan Data Rak	61
42. Gambar 4.6. Tampilan Data Peminjaman	62
43. Gambar 4.7. TampilanPengisian Data Peminjaman	62
44. Gambar 4.8. TampilanPengisian Data Pengembalian.....	63
45. Gambar 4.9. Tampilan Data Anggota	63
46. Gambar 4.10. Tampilan Data LaporanPeminjaman.....	64
47. Gambar 4.11. Tampilan Data LaporanPengembalian	64
48. Gambar 4.12. TampilanCetakLaporanPengembalian	65
49. Gambar 4.13. Hasil tampilan laporan pengembalian berdasarkan tanggal.....	65
50. Gambar 4.14. Tampilan cetak pertanggal Laporan Denda	66
51. Gambar 4.15. Hasil tampilan laporan denda berdasarkan tanggal.....	66
52. Gambar 4.16. Hasil tampilan eksekusi SMS Peringatan otomatis.....	67
53. Gambar 4.17. Eksekusi SMS Pemberitahuan Denda ke-2.....	67
54. Gambar 4.18. Eksekusi SMS Pemberitahuan Denda ke-3.....	68
55. Gambar 4.19. Eksekusi SMS Pemberitahuan Denda ke-4.....	68
56. Gambar 4.20. Eksekusi SMS Pemberitahuan Denda ke-5.....	69
57. Gambar 4.21. Hasil tampilan SMS peringatan kepada anggota.....	69
58. Gambar 4.22. Hasil tampilan SMS Pemberitahuan denda ke-2.....	70
59. Gambar 4.23. Hasil tampilan SMS Pemberitahuan denda ke-3.....	70
60. Gambar 4.24. Hasil tampilan SMS Pemberitahuan denda ke-4.....	71

61. Gambar 4.25. Hasil tampilan SMS Pemberitahu denda ke-5.....	71
---	----

DAFTAR TABEL

1. Tabel 2.1	7
2. Tabel 3.1. TabelBuku	31
3. Tabel 3.2. TabelKlasifikasi.....	32
4. Tabel 3.3. TabelRak.....	32
5. Tabel 3.4. TabelAnggota	33
6. Tabel 3.5. TabelDenda.....	33
7. Tabel 3.6. TabelPeminjaman	34
8. Tabel 3.7. Tabelpeminjaman_item	34
9. Tabel 3.8. TabelPengembalian	35
10. Tabel 3.9. Tabel Admin	35
11. Tabel 3.10. Tabelz_sms_inbox	36
12. Tabel 3.11. Tabelz_sms_outbox	37
13. Tabel 3.12 Rencanapengujian Program	58
14. Tabel 4.1. Pengujian Login Admin.....	72
15. Tabel 4.2 Pengujian Data Klasifikasi.....	73
16. Tabel 4.3. Pengujian Data Rak.....	74
17. Tabel 4.4. Pengujian Data Buku	75
18. Tabel 4.5. Pengujian Data Anggota	76
19. Tabel 4.6. PengujianPengisian Data Peminjaman	77

20. Tabel 4.7. PengujianPengisianPengembalian	78
21. Tabel 4.8. PengujianLaporanDenda.....	79
22. Tabel 4.9.PengujianLaporanPengembalian.....	80
23. Tabel 4.10. PengujianLaporanPengiriman SMS Peringatan.....	81
24. Tabel 4.11. PengujianLaporan SMS Pemberitahuandenda.....	81
25. Tabel 4.12. Pengujian Database	82

DAFTAR LAMPIRAN

1. Kode program halaman Login	86
2. Kodeprogram form Input, update buku.....	88
3. Kode program insert, edit anggota	92
4. Kode program data peminjaman	94
5. Kode program Pengembalianbuku	99
6. Kode Program SMS Peringatan auto send (kirimotomatis).....	105
7. Kode program SMS pemberitahuan auto send (kirimotomatis)	106

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bahasa sangat penting dan menjadi dasar bagi kita manusia agar dapat berinteraksi dengan sesama. Di setiap Negara memiliki bahasa mereka masing-masing. Bahkan di Indonesia setiap daerah memilikih bahasa khas mereka sendiri. Tomohon adalah salah satu kota yang memiliki bahasa khas yaitu bahasa tombulu. Namun seiring berkembangnya teknologi dan gaya hidup modern saat ini sangat mempengaruhi pengetahuan kita terhadap budaya kita terutama bahasa khas yang terdapat pada daerah kita. Banyak sekali masyarakat terutama anak-anak yang ada di Kota Tomohon yang sudah tidak tahu berbicara bahasa khas Tomohon yaitu bahasa Tombulu.

Hal ini dikarenakan masyarakat Kota Tomohon yang sudah mengikuti perkembangan zaman, menganggap itu sudah tidak terlalu penting. Media pembelajaran juga sangat berpengaruh karna fasilitas dan tempat yang disediakan kurang untuk menarik perhatian bagi masyarakat untuk belajar bahasa tombulu.

Maka dari itu dibutuhkan sebuah fasilitas yang bisa digunakan oleh masyarakat khususnya bagi anak-anak untuk dapat belajar bahasa tombulu terlebih juga dapat memberikan pengetahuan bagi wisatawan yang datang ke Kota Tomohon agar dapat belajar bahasa khas Tomohon yaitu bahasa tombulu.

Dengan adanya fasilitas yang mudah dan dapat digunakan dimanapun dapat membuat masyarakat dengan sendirinya belajar bahasa tombulu sehingga bahasa khas Kota Tomohon bisa terus di realisasikan.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas dapat kita rumuskan masalah sebagai berikut :

1. Bagaimana membuat masyarakat Kota Tomohon dapat belajar bahasa tombulu dengan mudah?
2. Dengan padatnya kegiatan masyarakat saat ini, Bagaimana membuat fasilitas yang dapat digunakan setiap saat untuk belajar?
3. Bagaimana memperkenalkan bahasa khas Kota Tomohon kepada wisatawan yang datang?
4. Bagaimana membuat anak-anak di kota Tomohon tertarik untuk belajar bahasa tombulu dengan fasilitas yang ada?

1.3 Tujuan Penelitian

Tujuan penelitian ini untuk membuat suatu aplikasi berbasis android yang dapat digunakan oleh pengguna android untuk belajar bahasa tombulu

1.4 Manfaat Penelitian

Penelitian ini bermanfaat bagi Masyarakat Kota Tomohon, wisatawan dan penulis.

1. Bagi masyarakat Kota Tomohon yaitu mempermudah untuk belajar bahasa tombulu dan dapat melestarikan bahasa tersebut
2. Bagi wisatawan dapat belajar bahasa tombulu dengan mudah
3. Bagi penulis sebagai mahasiswa yaitu mempermudah dalam membuat study kasus dan memperbanyak pengetahuan dalam pembuatan

aplikasi ini serta sebagai salah satu persyaratan untuk menyelesaikan study D4 di Politeknik Negeri Manado Jurusan Teknik Elektro.

1.5 Ruang Lingkup dan Batasan Masalah

1. Sebuah aplikasi yang dapat berjalan di gadget yang memiliki sistem operasi android
2. Membuat aplikasi terjemahan yang dapat bersuara menggunakan fasilitas dari android
3. Membuat terjemahan yaitu bahasa Inggris ke bahasa Indonesia dan bahasa Tombulu.

1.6 Metodologi Penulisan

Dalam penulisan tugas akhir ini, penulis menggunakan beberapa metode penelitian sebagai berikut :

1. Studi Lapangan
Metodi ini dilakukan penulis untuk mendesain, menguji dan mengimplementasikan aplikasi yang penulis buat dengan menggunakan gadget android
2. Studi Pustaka
Studi pustaka dilakukan untuk mencari refrensi yang dapat dijadikan dasar kajian dan landasan teori yang mendukung data – data informasi sebagai acuan dalam melakukan perencanaan, pembuatan dan percobaan pada aplikasi maupun laporan tugas akhir ini.

1.7 Sistematika Penulisan

Sistematika penulisan terdiri dari 5 bab yang terdiri dari :

1. BAB 1 : bagian pertama adalah pendahuluan, pada bagian ini diuraikan tentang latar belakang penelitian, permasalahan – permasalahan yang pada penelitian, tujuan penelitian, serta batasan dan ruang lingkup dari penelitian. Dan dibagian akhir bab ini diuraikan sistematika penyajian laporan penelitian.
2. BAB 2 : bagian kedua mengenai tentang landasan teori dalam pembuatan tugas akhir ini yang membahas tentang device android, dasar pemrogramannya, mikrokontroler arduino uno, Ethernet shield, media komunikasi wifi, serta penjelasan singkat mengenai komponen – komponen elektronik yang digunakan dalam rangkaian relay dan catu daya.
3. BAB 3 : bagian ini memaparkan tentang perancangan system.
4. BAB 4 : bagian ini berisi mengenai informasi dari hasil uji coba dan implementasi terhadap alat dibuat.
5. BAB 5 : menyajikan kesimpulan serta saran yang bisa digunakan sebagai dasar dalam pengembangan alat ini untuk kedepannya.

BAB II

TINJAUAN PUSTAKA

2.1 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android, Inc. didirikan di Palo Alto, California, pada bulan Oktober 2003 oleh Andy Rubin (pendiri Danger), Rich Miner (pendiri Wildfire Communications, Inc.), Nick Sears (mantan VP T-Mobile), dan Chris White (kepala desain dan pengembangan antarmuka WebTV) untuk mengembangkan "perangkat seluler pintar yang lebih sadar akan lokasi dan preferensi penggunanya". Tujuan awal pengembangan Android adalah untuk mengembangkan sebuah sistem operasi canggih yang diperuntukkan bagi kamera digital, namun kemudian disadari bahwa pasar untuk perangkat tersebut tidak cukup besar, dan pengembangan Android lalu dialihkan bagi pasar telepon pintar untuk menyaingi Symbian dan Windows Mobile (iPhone Apple belum dirilis pada saat itu). Meskipun para pengembang Android adalah pakar-pakar teknologi yang berpengalaman, Android Inc. dioperasikan secara diam-diam, hanya diungkapkan bahwa para pengembang sedang menciptakan sebuah perangkat lunak yang diperuntukkan bagi telepon seluler.

Android adalah 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.

Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang bagi yang membuat peranti 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.

Didunia 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 jenis smartphone yang menggunakan Android sebagai sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia dipasaran 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, Toshiba 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 pembaruan berupa perbaikan bug dan penambahan fitur baru.

Pada masa saat ini kebanyakan vendor – vendor smartphone sudah memproduksi smartphone berbasis android, vendor – vendor itu antara lain HTC, Motorola, Samsung, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus SciPhone, WayteQ, Sony Ericsson, Acer, Philips, T-mobile, Nexian, IMO, Asus dan masih banyak lagi vendor smartphone didunia yang memproduksi android. Hal ini karena android itu adalah sistem operasi yang open source sehingga bebas didistribusikan dan dipakai oleh vendor manapun.

Tidak hanya menjadi sistem operasi dismartphone, saat ini android menjadi pesaing utama dari apple pada sistem operasi Table PC. Pesatnya pertumbuhan android selain factor yang disebutkan diatas 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 didunia, sehingga android terus berkembang pesat baik dari segi teknologi maupun dari segi jumlah device yang ada di dunia.

2.1.1. Android: Platform Masa Depan

Android dipuji sebagai “platform mobile pertama yang lengkap, terbuka dan bebas”.

- 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.
- Terbuka (Open Source Platform) : Platform android disediakan melalui lisensi open source. Pengembangan dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux Kernel 2.6.
- Free (Free Platform) : Android adalah platform/aplikasi yang bebas untuk develop. Tidak ada lisensi atau biaya royalty 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 memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari android dilisensikan di bawah GNU, General Public Lisensi Versi 2 (GPLv2), yang sering dikenal dengan istilah “copyleft” lisensi dimana setiap perbaikan pihak ketiga harus terus jatuh di bawah terms. Android didistribusikan dibawah 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. Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka dibawah skema lisensi apapun yang mereka inginkan.

Pengembangan memiliki beberapa pilihan ketika membuat aplikasi yang berbasis android. Kebanyakan pengembang menggunakan eclipse yang tersedia secara bebas untuk merancang dan mengembangkan aplikasi android. Eclipse adalah Ide yang paling populer untuk pengembangan android, karena memiliki android plug-in yang tersedia untuk memfasilitasi pengembangan android. Selain itu eclipse juga mendapat dukungan langsung dari google untuk menjadi 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. Tetapi hal diatas tidak menutup kemungkinan untuk menggunakan IDE yang lain seperti NetBeans untuk melakukan pengembangan android.

2.1.2 The Dalvik Virtual Machine (DVM)

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 (JVM) 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 oleh Dan Botnsten dan beberapa engineers google lainnya. Jadi bisa kita katakan “Dalvik equals(Java) == False”. Dalvik Virtual Machine menggunakan kernel Linux untuk menangani fungsionalitas tingkat rendah termasuk keamanan, threading, dan proses serta manajemen memori. Ini memungkinkan kita untuk menulis aplikasi C / C + sama halnya seperti pada OS Linux kebanyakan. Meskipun dalam kenyataanya kita harus banyak memahami arsitektur dan proses istem dari kernel 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 sangat kecil. The executable file diciptakan dengan mengubah kelas bahasa java dan dikompilasi menggunakan tools yang disediakan dalam SDK Android.

2.1.3 Versi Android

1. Android 1.0 (API level 1)

Android 1.0, Versi komersil dirilis pada 23 september 2008, dengan menggunakan device HTC DREAM.

2. Android 1.1

Pada maret 2009 google merilis Android versi 1.1 pada versi ini Android sudah di lengkapi dengan pembaharuan estis pada aplikasi,jam,alarm,voice search,pengirim pesan dan gmail,serta pembaharuan email.

3. Android 1.5 (cupcake)

Pada 27 April 2009, Android 1.5 dirilis, menggunakan kernel Linux 2.6.27. Versi ini adalah rilis pertama yang secara resmi menggunakan nama kode berdasarkan nama-nama makanan pencuci mulut ("Cupcake"), nama yang kemudian digunakan untuk semua versi rilis selanjutnya. Pembaruan pada versi ini termasuk beberapa fitur baru dan perubahan UI.

4. Android 1.6 (donut)

Menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang diintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine; kemampuan dial kontak; teknologi

text to change speech (tidak tersedia pada semua ponsel pengadaaan resolusi VWGA.

5. Android 2.0/2.1 (Éclair)

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.

6. Android 2.2 (Froyo)

Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, intergrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi Android Market.

7. Android 2.3 (gingerbread)

Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User Interface) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

8. Android 3.0/3.1 (honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Perangkat tablet dengan platform Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama Eee Pad Transformer produksi dari Asus.

9. Android 4.0 (ice cream sandwich)

Membawa fitur Honeycomb untuk smartphone dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan menggunakan NFC.

10. Android 4.1 (Jelly Bean)

dirilis tahun 2012. Android Jelly Bean ini diperuntukkan untuk komputer tablet dan memungkinkan untuk digunakan pada sistem operasi PC atau Komputer. Sehingga rumornya kemunculan Android Jelly Bean ini untuk menyaingi rilis terbaru Windows 8 yang juga akan segera dirilis. Karena kita ketahui bersama

perbincangan versi Android sebelumnya yaitu Android Ice Cream Sandwhich pun masih hangat di telinga.

11. Android 4.2 (Jelly Bean API level 17)

Dirilis pada 13 november 2012, versi ini melengkapi kekurangan maupun bugs yang sering terjadi pada JB 4.1, seperti perbaikan bug pada aplikasi ‘people’, penambahan tampilan nirkabel (*miracast*), perbaikan aksesibilitas, VPN yang selalut terhubung dan lain – lain.

12. Android 4.4 (KitKat)

Dirlis pada 31 Oktober 2013, versi ini memiliki antarmuka terbaru dengan status bar dan navigasi transparan pada layar depan, webviews berbasis *Chromium*, mendukung media komunikasi Infra merah yang memungkinkan devices bisa menjadi remote untuk smart tv.

13. Android 5.0 (Lollipop)

Pada versi terbaru ini google selaku pengembang membuat sebuah desain antarmuka terbaru yang dinamakan “Material Design”, serta mendukung 64bit ART compiler, dan menambahkan system keamanan yang bernama ‘*factory reset protection*’ yang berfungsi ketika smartphone hilang, ia tidak bisa direset ulang tanpa memasukkan id dan password akun google.

2.1.4 Libraries

Bertempat di level yang sama dengan Android Runtime adalah Libraries. Android menyertakan satu set library-library dalam bahasa C/C++ yang digunakan oleh berbagai komponen yang ada pada sistem Android. Kemampuan ini dapat diakses oleh programmer melewati Android application framework. Sebagai contoh Android mendukung pemutaran format audio, video, dan gambar. Berikut ini beberapa core library tersebut :

System C library,

diturunkan dari implementasi standard C system library (libc) milik BSD, dioptimasi untuk piranti embedded berbasis Linux.

Media Libraries,

berdasarkan PacketVideo's OpenCORE; library-library ini mendukung playback dan recording dari berbagai format audio and video populer, meliputi MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG.

Surface Manager,

mengatur akses pada display dan lapisan composites 2D and 3D graphic dari berbagai aplikasi.

LibWebCore,

web browser engine modern yang mensupport Android browser maupun embeddable web view.

SGL,

the underlying 2D graphics engine.

3D libraries,

implementasi berdasarkan OpenGL ES 1.0 APIs; library ini menggunakan hardware 3D acceleration dan highly optimized 3D software rasterizer

FreeType,

bitmap dan vector font rendering

SQLite,

relational database engine yang powerful dan ringan tersedia untuk semua aplikasi.

2.1.5 Tool dalam Pemograman android

Pada setiap pembuatan aplikasi atau program, seorang programmer tentu saja membutuhkan tools – tools untuk mendukung proses pembuatan aplikasi yang ia buat. Tools yang dimaksud disini bisa berupa aplikasi berbasis bahasa pemograman yang memudahkan seorang programmer dalam mengetikkan coding ataupun mendesain tampilan dari aplikasi yang akan ia buat. Dalam pemograman aplikasi untuk dekstop komputer kita biasa mendengar yang namanya NetBeans, Delphi, Visual Basic, dll. Netbeans dan Delphi inilah yang disebut tool untuk pemograman.

Untuk memulai pembuatan sebuah aplikasi Android pun, kita membutuhkan tools – tools yang nantinya akan mempermudah proses pembuatan aplikasi tersebut. Adapun tools – tools yang dibutuhkan untuk membuat aplikasi android yaitu :

1. Java
2. Eclipse IDE
3. Android SDK

2.1.5.1 Java SE (Standart Edition)

Java platform Standart Edition merupakan sebuah paket dalam bahasa pemrograman java, dimana berarti Java SE bisa digunakan untuk membuat aplikasi berbasis Java. Dalam lingkup pemrograman Android, Java SE merupakan sebuah dasar dari pemrograman Android. Mengapa ? Karena bahasa pemrograman asli untuk Android adalah Java. Jadi untuk memulai pembuatan aplikasi Android kita harus terlebih dahulu memiliki Java SE yang terinstal di komputer/laptop yang akan kita gunakan untuk membuat aplikasi Android.

2.1.5.2 Eclipse

Eclipse adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan disemua platform. 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. Multi 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

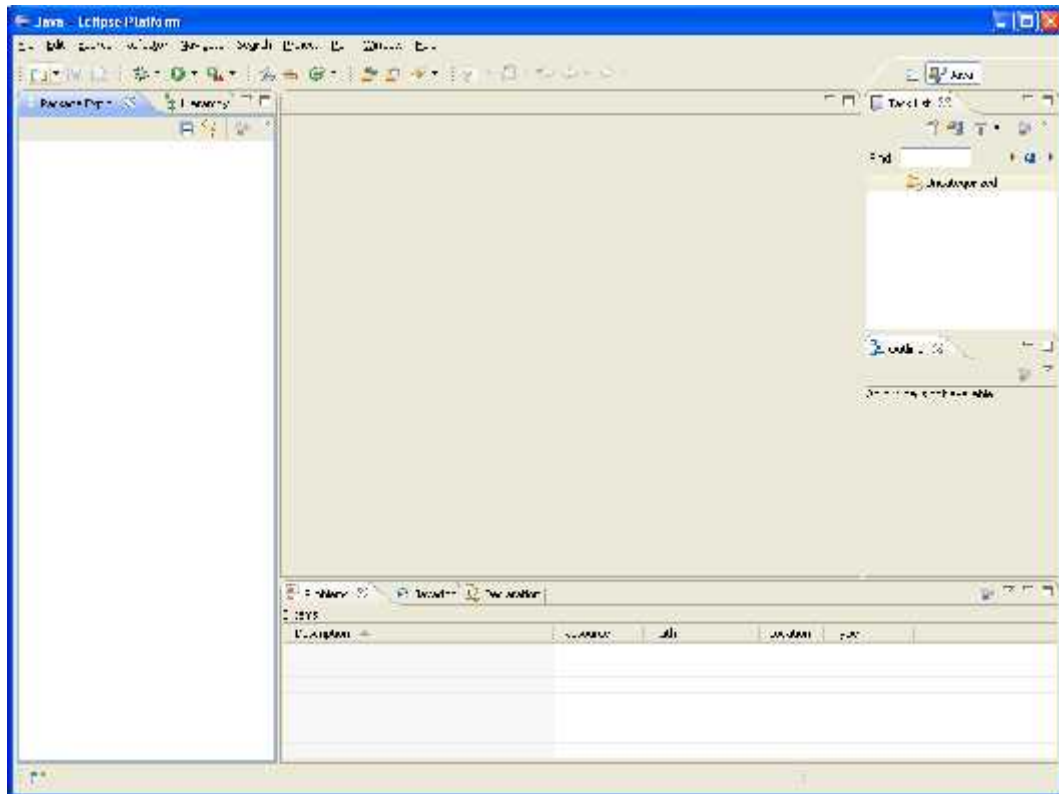
Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak seperti dokumentasi, test perangkat lunak, pengembangan Web dan lain – lain.

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.

Seperti yang dijelaskan tadi untuk membuat sebuah aplikasi programmer membutuhkan tools – tools diantaranya tools yang digunakan untuk menuliskan kode – kode program. Untuk pemrograman Android, tools untuk menuliskan program yang dibutuhkan adalah Eclipse.

Eclipse digunakan untuk menuliskan kode dan menguji serta menjalankan kode program di satu tempat saja tanpa harus menggunakan aplikasi tambahan. Oleh

karena itu Eclipse sering disebut dengan istilah Integrated Development Environment(IDE), versi Eclipse yang akan digunakan untuk pemrograman Android ini bisa didapatkan dengan mendownload Android SDK.



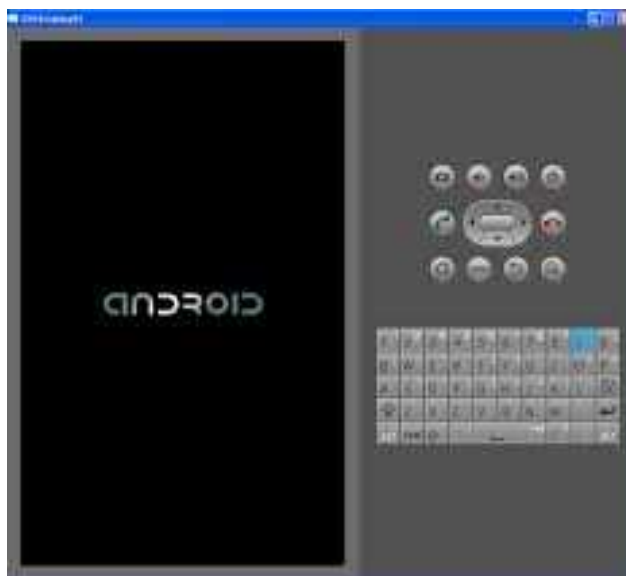
Gambar 2.1 Tampilan *Interface* Eclipse

2.1.5.3 Android SDK

Tools terakhir yang digunakan adalah Android SDK (Software Development Kit), didalam tools ini terdapat file – file dan utilities (alat bantu) lainnya yang berfungsi untuk mempermudah pembuatan aplikasi Android. Android SDK bekerja sama dengan Eclipse IDE untuk membantu programmer menciptakan aplikasi secara cepat.

2.1.5.4 Android Virtual Device Manager (AVD)

Android virtual device manager merupakan sebuah tool yang memungkinkan programmer untuk menguji aplikasi android yang akan dibuat memakai alat menyerupai ponsel atau gadget. AVD ini akan berperan sebagai emulator, yaitu software yang memiliki tampilan dan cara kerja menyerupai device android yang asli.



Gambar 2.2 Tampilan *Interface* AVD

2.1.5.5 JDK/JRE

JDK merupakan singkatan dari Java Development Kit, yaitu software yang digunakan untuk membangun aplikasi java. Tanpa JDK kita tidak akan bisa membuat berbagai macam aplikasi berbasis java. Sedangkan JRE adalah

singkatan dari Java Runtime Environment, yaitu software yang diciptakan oleh Sun Microsystem yang berguna hanya untuk menjalankan program atau aplikasi java yang sudah melewati proses compile atau sudah jadi.

2.1.5.6 FRAMEWORK-RES.APK

Framework-Res adalah sebuah inti dari sistem operasi Android, didalam file ini terselipkan beberapa coding penting dan beberapa settingan icon agar device dapat dijalankan dan biasanya untuk memodifikasi framework ini dibutuhkan pengetahuan tentang pemograman Android yang lebih lanjut

2.1.5.7 Android Package

Application Package File (APK), adalah format (extension) yang digunakan untuk mendistribusikan dan memasang software ke ponsel dengan sistem operasi Android, mirip dengan paket MSI pada Windows atau Deb pada OS Debian.

2.1.5.8 Android Development Tools (ADT)

Android Development Tools (ADT) adalah plugin yang didesain untuk IDE Eclipse yang memberikan kita kemudahan dalam mengembangkan aplikasi android dengan menggunakan IDE Eclipse. Dengan menggunakan ADT untuk eclipse akan memudahkan kita dalam membuat aplikasi project android, memuat GUI aplikasi, dan menambahkan komponen – komponen yang lainnya, begitu

juga kita dapat melakukan running aplikasi menggunakan 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 menggunakan ADT di eclipse sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi android. Berikut adalah versi ADT untuk Eclipse yang sudah dirilis :

- ADT 12.0.0 (July 2011)
- ADT 11.0.0 (June 2011)
- ADT 10.0.1 (March 2011)
- ADT 10.0.0 (February 2011)
- ADT 9.0.0 (January 2011)
- ADT 8.0.1 (December 2010)
- ADT 8.0.0 (December 2010)
- ADT 0.9.9 (September 2010)
- ADT 0.9.8 (September 2010)
- ADT 0.9.7 (May 2010)
- ADT 0.9.6 (March 2010)
- ADT 0.9.5 (December 2009)
- ADT 0.9.4 (October 2009)

Semakin tinggi platform android yang kita gunakan, dianjurkan menggunakan ADT yang lebih terbaru, karena biasanya munculnya platform baru diikuti oleh munculnya versi ADT yang terbaru. Untuk melakukan instalasi ADT di eclipse dapat dilakukan secara online maupun offline.

2.1.5.9 Arsitektur Android

Secara garis besar arsitektur android dapat dijelaskan dan digambarkan sebagai berikut:

- Applications dan Widgets

Applications dan Widgets ini adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita download aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Dilayer terdapat aplikasi inti termasuk klien email, program sms, kalender, peta, browser, kontak, dan lain – lain. Semua aplikasi ditulis menggunakan bahasa pemrograman java.

- Applications Frameworks

Android adalah “Open Development Platform” yaitu android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi resources, menjalankan service background, mengatur alarm, dan memiliki akses penuh menuju API framework seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan 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 disistem operasi android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti content – providers yang berupa sms dan panggilan telepon.

Komponen – komponen yang termasuk didalam applications frameworks adalah sebagai berikut :

- a. Views
- b. Content Provider

- c. Resource Manager
 - d. Notification Manager
 - e. Activity Manager
- Libraries

Libraries ini adalah layer dimana fitur – fitur android berada, biasanya para pembuat aplikasi mengakses libraries untuk menjalankan aplikasinya. Berjalan diatas kernel, layer ini meliputi berbagai library C/C+ inti seperti Libc dan SSL, serta :

 - a. Libraries media untuk pemutaran media audio dan video
 - b. Libraries untuk manajemen tampilan
 - c. Libraries Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D
 - d. Libraries SQLite untuk dukungan database
 - e. Libraries SSL dan WebKit terintegrasi dengan web browser dan security
 - f. Libraries LiveWebcore mencakup modern web browser dengan engine embedded web view
 - g. Libraries 3D yang mencakup implementasi OpenGL ES 1.0 API's
- Android Run Time

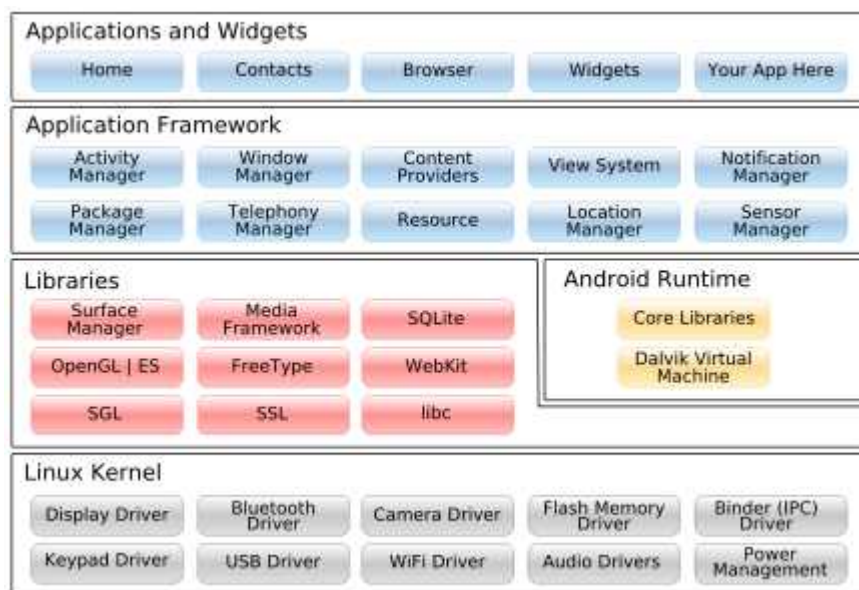
Layer yang membuat aplikasi android dapat dijalankan dimana dalam prosesnya menggunakan implementasi linux. Dalvik Virtual Machine (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi android. Didalam android run time dibagi menjadi dua bagian yaitu :

 - a. Core Libraries : aplikasi android dibangun dalam bahasa java, sementara Dalvik sebagai virtual mesinnya bukan Virtual Machine Java, sehingga diperlukan sebuah libraries yang berfungsi untuk menterjemahkan bahasa java/c yang ditangani oleh Core Libraries.
 - b. Dalvik Virtual Machine : virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi – fungsi secara efisien,

dimana merupakan pengembangan yang mampu membuat linux kernel untuk melakukan threading dan manajemen tingkat rendah.

- **Linux Kernel**

Linux kernel adalah layer dimana inti dari operating sistem dari android itu berada. Berisi file – file sistem yang mengatur sistem processing, memory, resource, drivers, dan sistem – sistem operasi android lainnya. Linux kernel yang digunakan android adalah linux kernel release 2.6.



Gambar 2.3 Arsitektur Android

2.1.5.10 Fundamental Aplikasi

Aplikasi android ditulis dalam bahasa pemrograman java. Kode java dikompilasi bersama dengan data file resource yang dibutuhkan oleh aplikasi, dimana prosesnya dipackage oleh tools yang dinamakan “apt tools” ke dalam paket android sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut denan aplikasi, dan nantinya dapat di install diperangkat mobile.

Ada enam jenis komponen pada aplikasi android yaitu :

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 aplikasi tersebut. Satu activity biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai user interface (UI) saat aplikasi diperlihatkan kepada user. Untuk pindah dari satu even, misalnya click tombol, memilih opsi atau menggunakan triggers tertentu. Secara hirarki sebuah windows activity dinyatakan dengan method `Activity setContentView()`. `Content View` adalah objek yang berada pada root hirarki.

b. Service

Service tidak memiliki Graphic User Interface (GUI), tetapi service berjalan secara background, sebagai contoh dalam memainkan music, service mungkin memainkan music atau mengambil data dari jaringan, tetapi setiap service harus berada dalam kelas 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 jalan. Untuk menjaga music tetap dijalankan, activity player dapat menjalankan service. Service 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 perubahan referensi bahasa yang digunakan. Aplikasi juga dapat menginisiasi broadcast misalnya memberikan informasi pada aplikasi lain bahwa ada data yang telah diunduh 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 Notification Manager untuk memberitahu 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 dalam file sistem seperti 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.

2.2 Database

Database (basis data) adalah kumpulan informasi yang disimpan didalam computer secara sistematis sehingga dapat diperiksa menggunakan suatu program computer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kuery (query) basis data disebut system manajemen basis data (database management system, DBMS). Sistem basis data dipelajari dalam ilmu informasi.

2.2.1 Bahasa Pada Database

Terdapat dua jenis bahasa computer yang digunakan saat kita ingin membangun dan memanipulasi sebuah basis data, yaitu :

1. Data Definition Language (DDL)

DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek – objek yang diperlukan dalam basis data, misalnya tabel, view, user, dan sebagainya. Secara umum DDL yang digunakan adalah

create untuk membuat objek baru, *use* untuk menggunakan objek, *alter* untuk mengubah objek yang sudah ada, dan *drop* untuk menghapus objek. DDL biasanya digunakan oleh administrator basis data dalam pembuatan sebuah aplikasi basis data.

2. Data Manipulation Language (DML)

DML digunakan untuk memanipulasi data yang ada dalam suatu tabel.

Perintah yang umum dilakukan adalah :

- SELECT untuk menampilkan data
- INSERT untuk menambahkan data baru
- UPDATE untuk mengubah data yang sudah ada
- DELETE untuk menghapus data

2.2.2 Structured Query Language

SQL (Structured Query Language) adalah sebuah bahasa yang digunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara de facto merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen basis datanya.

Sejarah SQL dimulai dari artikel seorang peneliti dari IBM bernama Jhonny Oracle yang membahas tentang ide pembuatan basis data relasional pada bulan juni 1970. Artikel ini juga membahas kemungkinan pembuatan bahasa standar untuk mengakses data dalam basis data tersebut. Bahasa tersebut kemudian diberi nama SEQUEL (Structured English Query Language).

Standarisasi SQL dimulai pada tahun 1986, ditandai dengan dikeluarkannya standar SQL oleh ANSI. Standar ini sering disebut dengan SQL86. Standar tersebut kemudian diperbaiki pada tahun 1989 kemudian diperbaiki lagi pada tahun 1992. Versi terakhir dikenal dengan SQL92. Pada

tahun 1999 dikeluarkan standar baru yaitu SQL99, akan tetapi kebanyakan implementasi mereferensi pada SQL92.

2.2.3 SQLite

SQLite merupakan sebuah system manajemen basis data relasional yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp.

Tidak seperti paradigma client-server pada umumnya, inti SQLite bukanlah sebuah sistem yang mandiri yang berkomunikasi dengan sebuah program, melainkan sebagai integral dari sebuah program secara keseluruhan. Sehingga protocol komunikasi utama yang digunakan adalah melalui pemanggilan API secara langsung melalui bahasa pemrograman. Mekanisme seperti ini tentunya membawa keuntungan karena dapat mereduksi overhead, latency times, dan secara keseluruhan lebih sederhana. Seluruh elemen basis data (definisi data, tabel, indeks, dan data) disimpan sebagai sebuah file. Kesederhanaan dari sisi disain tersebut bisa diraih dengan cara mengunci keseluruhan file basis data pada saat sebuah transaksi dimulai.

2.2.3.1 Fitur-fitur

Pustaka SQLite mengimplementasikan hampir seluruh elemen-elemen yang berlaku pada SQL-92, termasuk transaksi yang bersifat atomic, konsistensi basisdata, isolasi, dan durabilitasi (ACID), trigger, dan kueri-kueri yang kompleks. Tidak ada pengecekan tipe sehingga data bisa di entrikan dalam bentuk string untuk sebuah kolom bertipe integer.

Beberapa proses ataupun thread dapat berjalan secara bersamaan dan mengakses basisdata yang sama tanpa mengalami masalah. Hal ini disebabkan karena akses baca data dilakukan secara parallel. Sementara itu akses tulis data hanya bisa dilakukan jika tidak ada proses tulis lain yang sedang dilakukan. Jika tidak, proses tulis tersebut akan gagal dan mengembalikan kode kesalahan. Hanya saja ketika sebuah tabel temporer dibuat, mekanisme penguncian pada proses multithread akan menyebabkan masalah. Update terkini (versi 3.3.4) dikatakan telah memperbaiki masalah ini. Sebuah program yang mandiri dinamakan *sqlite* disediakan dan bisa digunakan untuk mengeksekusi kueri dan manajemen file – file basis data SQLite. Program tersebut juga merupakan contoh implementasi penulisan aplikasi yang menggunakan pustaka SQLite.

2.2.3.2 Integrasi Dengan Bahasa Lain

- SQLite termasuk dalam framework REALbasic. Yang memungkinkan aplikasi yang dikembangkan dengan menggunakan REALbasic dapat memanfaatkan basis data SQLite.
- Pustaka SQLite bisa digunakan secara langsung pada bahasa C/C++, namun untuk Tcl dan beberapa bahasa pemrograman berbasis script juga tersedia.
- Modul DBI/DBD untuk Perl juga tersedia pada CPAN, DBD:SQLite, namun modul ini bukanlah antarmuka dengan SQLite melainkan memasukan SQLite secara keseluruhan dalam modul tersebut.
- Modul Python juga tersedia (PySQLite) yang diimplementasikan pada DB API Python versi 2.0 (PEP 249).
- PHP dimulai dengan PHP5 telah memasukan SQLite, versi PHP4 sebelumnya bisa digunakan untuk mengakses SQLite, namun modul SQLite tidak dimasukan secara standar bawaan.

- Dimulai Lazarus versi 0.9.8 dan Free Pascal 2.0.0, SQLite didukung untuk digunakan oleh para programmer.
- Meskipun Borland tidak memaketkan SQLite secara standar bawaan, Delphi sudah mendukung SQLite juga menggunakan pustaka yang dibuat oleh pihak ketiga (Aducom dan Zeos). Banyak program administrasi basis data SQLite baik komersil ataupun bebas yang menawarkan kemudahan-kemudahan dalam manajemen basis data SQLite terdapat di internet dibuat dengan menggunakan bahasa ini.
- SQLite juga termasuk dalam paket yang dibundle secara standar bawaan pada Mac OS X, dan digunakan sebagai salah satu pilihan mekanisme penyimpanan data pada API Apple.

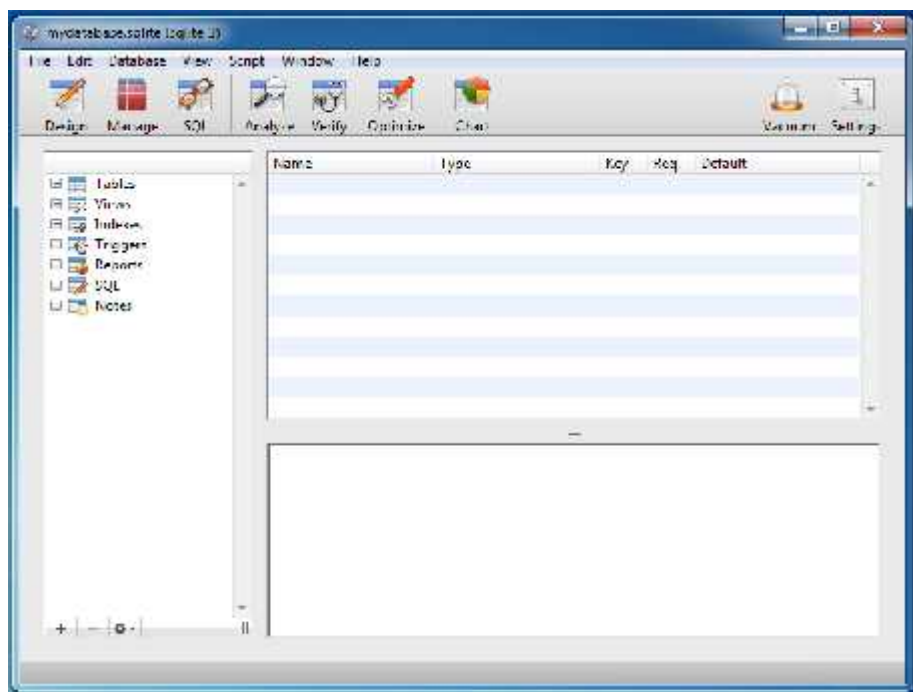
2.2.3.3 SQLite Manager

SQLite manager adalah multibahasa/tools berbasis web untuk mengelolah database SQLite. Bahasa pemrograman yang digunakan adalah : PHP4 dan PHP5. Bekerja dengan baik pada platform Linux, Windows dan Mac OS X. Tools ini dalam masa development, tetapi sangat fungsional. Berbagai fungsinya adalah :

- Manajemen beberapa database
- Creation 'koneksi' ke beberapa database
- Properti dan pengelolah pilihan dari database yaitu insert, update, delete tabel.
- Import data dari file – eksekusi manual permintaan dari SQLite format atau dari MYSQL format
- Membuat, update dan hapus dari view
- Membuat, update dan hapus dari trigger
- Manajemen dari fungsi yang ditetapkan pengguna.
- Ekspor dan struktur data ke format SQL

SQLite merupakan salah satu embedded database yang wajib dipertimbangkan untuk digunakan pada aplikasi bukan enterprise karena memiliki beberapa kelebihan yaitu

- Secara umum SQLite sangat stabil
- Performa SQLite lebih efisien dan cepat dibandingkan database yang lain
- Sedikit menggunakan memori, hanya membutuhkan single library untuk mengakses database
- Bejalan dibanyak platform yang berbeda, dan dapat dipindahkan dengan mudah tanpa setting administrasi yang rumit.
- Memenuhi standart ACID (Atomicity, Consistency, Isolation, and Durability)
- Mendukung ANSI 92 SQL standar
- Mempunyai API untuk bahasa C++, PHP, Perl, Python, Tcl.
- Lisensi public domain, free untuk redistributed.



Gambar 2.4 Tampilan *Interface* SQLite Manager

2.3 Tombulu

Tombulu adalah salah satu sub etnis Suku Minahasa yang mendiami wilayah tengah Minahasa, yaitu di wilayah Kota Tomohon dan Kecamatan Tombulu Kabupaten Minahasa. Orang-orang Tombulu menggunakan Bahasa Tombulu atau Bahasa Minahasa sebagai bahasa ibu. Saat ini terdapat organisasi adat yang terus melestarikan dan mengembangkan budaya sub etnis Tombulu, yaitu Pakasaan Tombulu yang dipimpin oleh mantan Wakil Walikota Tomohon Syennie-Smits Watoelangkow. Organisasi masyarakat ini telah membangun suatu amfiteater dengan pemandangan indah ke gunung api Lokon yang masih aktif dan gunung Empung dimana suku Tombulu dulu menganggap terdapat tempat dewa – dewa mereka. Di sekitar amfiteater terdapat banyak mata air yang konon kabarnya bisa menyembuhkan atau membuat orang yang mandi disitu mejadi lebih pintar. Juga di kompleks organisasi pakasaan Tombulu yang bernama Rano Walanda, terdapat banyak Waruga. Waruga adalah batu yang berlubang dimana masyarakat Minahasa dulu meletakkan mayat orang di dalam dan menutupnya dengan batu berukir besar. Lokasi amfiteater Rano Walanda terdapat di desa Woloan I di Kota Tomohon.

2.4 Voice Read Text

Voice read text adalah suatu sistem yang dapat melakukan konversi dari teks menjadi ucapan. Saat ini terdapat berbagai sistem yang dapat melakukan fungsi seperti itu, misalnya sistem IVR (interface voice response) yang banyak digunakan untuk layanan informasi otomatis melalui telpon (call center). Sistem IVR biasanya menggunakan rekaman kata atau kalimat yang direkam secara utuh. Voice Read Text sama halnya dengan Text to speech.

BAB III

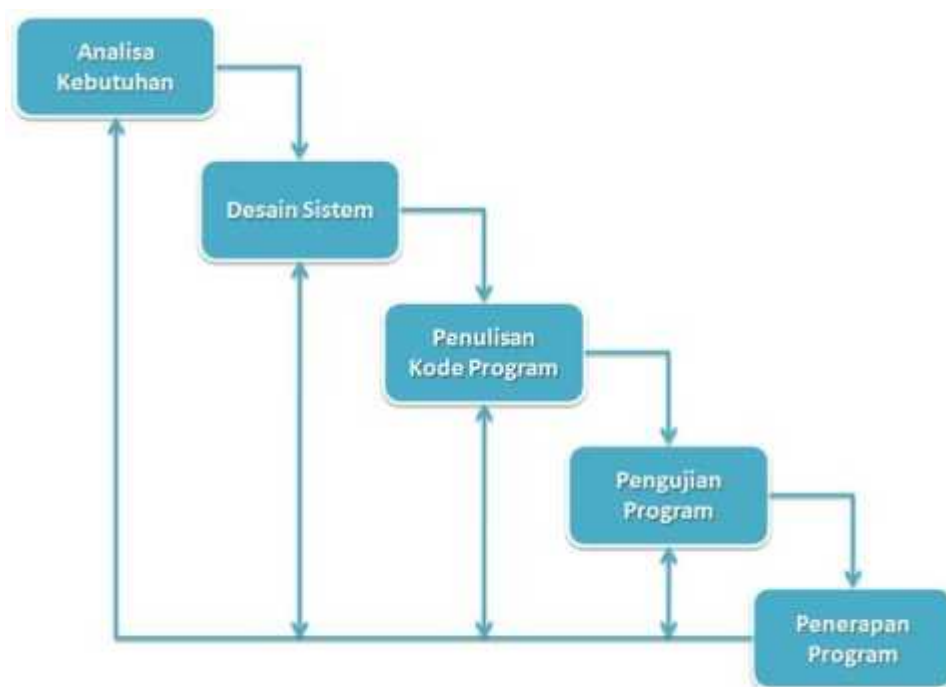
METODOLOGI DAN PERANCANGAN

3.1 Waktu dan Tempat

- Lokasi : Sekretariat Grup Torang-Torang
- Waktu : Mei – Juli 2015

3.2 Metode Penelitian Yang Digunakan

Waterfall adalah suatu metodologi pengembangan perangkat lunak yang mengusulkan pendekatan kepada perangkat lunak sistematis dan sekuensial yang mulai pada tingkat kemajuan sistem pada seluruh analisis, design, kode, pengujian dan pemeliharaan.



Gambar 3.1 Metodologi Waterfall

Langkah – langkah yang harus dilakukan pada metodologi waterfall adalah sebagai berikut :

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak. Untuk memahami sifat program yang dibangun, rekayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk kerja dan antar muka (interface) yang diperlukan. Kebutuhan baik untuk sistem maupun perangkat lunak di dokumentasikan dan dilihat dengan pelanggan.

b. Desain

Desain perangkat lunak sebenarnya adalah proses multi langka yang berfokus pada empat atribut sebuah program yang berbeda. Struktur data, arsitektur perangkat lunak, representasi interface dan detail (algoritma) procedural. Proses desain menerjemahkan syarat/kebutuhan kedalam sebuah representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.

c. Penulisan Kode Program

Desain harus diterjemahkan dalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

d. Pengujian Program

Proses pengujian dilakukan pada logika internal untuk memastikan semua pernyataan sudah diuji. Pengujian eksternal fungsional untuk menemukan kesalahan – kesalahan dan memastikan bahwa input akan memberikan hasil yang actual sesuai yang dibutuhkan.

e. Penerapan Program

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja.

3.2.1 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data adalah mencari dan menggunakan sumber data primer dan sumber data sekunder untuk itu akan dijelaskan secara singkat mengenai pemahaman metode tersebut sebagai berikut :

3.2.1.1 Sumber Data Primer

Data primer adalah data yang diperoleh langsung dari instansi baik melalui pengamatan maupun pencatatan terhadap objek penelitian, dalam memperoleh data primer penulis menggunakan metode sebagai berikut :

a. Metode Observasi

Metode ini dipakai penulis untuk mendapatkan data dan informasi dengan melakukan peninjauan atau pengamatan secara langsung ke kota Tomohon guna untuk pembuatan kamus yang membahasakan bahasa tombulu

b. Metode Wawancara

Penulis mendapatkan data serta informasi melalui dialog langsung dengan masyarakat kota Tomohon guna untuk membuat aplikasi. Penulis juga melakukan wawancara dengan beberapa pendatang di kota Tomohon untuk mengetahui apa yang dibutuhkan untuk

perancangan aplikasi kamus bahasa tomohon menggunakan voice read text ini.

3.2.1.2 Sumber Data Sekunder

Selain sumber data primer, penyusunan skripsi ini juga mencari dan mengumpulkan data dari sekunder, berikut adalah metode yang digunakan dalam mencari dan mengumpulkan data dari sumber sekunder.

a. Metode Studi Pustaka

Penulis melakukan kajian pustaka secara literatur yaitu melalui buku – buku, karangan – karangan, kumpulan buku kuliah, informasi melalui internet yang relevan dengan variabel yang diteliti dan dari para narasumber yang berhubungan langsung dengan masalah pada objek penelitian.

b. Dokumentasi

Teknik ini dilakukan dengan mempelajari material yang didapat dari proses voice read text. Metode yang diamati yaitu penulisan dan cara penyebutan.

3.2.2 Metode Pembuatan Aplikasi

Aplikasi yang akan dibuat menggunakan Eclipse IDE sebagai tools pembuatan aplikasi berbasis android. Pembuatan aplikasi ini juga melibatkan gadget yang memiliki sistem operasi android.

3.2.3 Metode Perancangan Interface

Perancangan antarmuka (interface) merupakan rancangan dasar aplikasi, untuk menampilkan tampilan dari aplikasi tersebut.



Gambar 3.2 Rancangan User Interface

Ada beberapa bagian pada interface aplikasi ini, mempunyai fungsi masing – masing yaitu :

- Input text bahasa Inggris

Pada edit text tersebut kita dapat memasukan kosakata bahasa Inggris yang tersedia

- Terjemahan

Button Terjemahan ini adalah buton yang jika di klik akan menerjemahkan kosakata bahasa Inggris yang telah di input

- Output Bahasa Indonesia dan Bahasa Tombulu

Pada Edit Text ini menampilkan hasil dari terjemahan, setelah kita mengklik button terjemahan. Maka pada setiap Edit text akan menghasilkan output kosakata.

- Speak

Button speak berfungsi untuk menampilkan voice read text dari kosakata bahasa Tombulu.

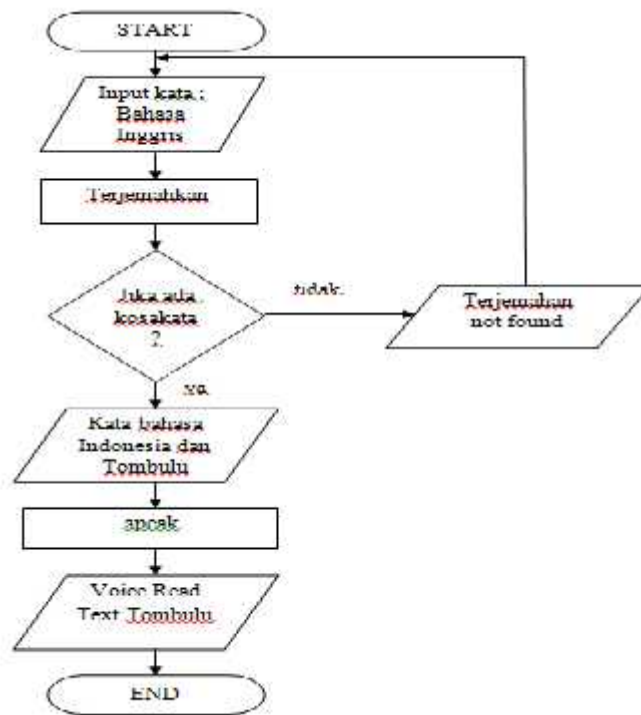
3.2.4 Implementasi & Testing

Aplikasi yang telah dibuat kemudian akan diuji dan diimplementasikan pada sebuah gadget android dengan tujuan untuk melihat tampilan dan fungsi dari aplikasi tersebut.

3.3 Diagram Alur (Flowchart)

Pada pembuatan aplikasi ini dibutuhkan suatu teknik perancangan yang mempunyai struktur yang baik, biasanya diawali dengan pembuatan diagram alur (flowchart) Diagram alur digunakan untuk menggambarkan terlebih dahulu apa yang harus dikerjakan sebelum mulai merancang atau membuat suatu system seperti yang akan dijelaskan dibawah ini.

Berikut adalah diagram alur (flowchart) dari aplikasi android.



Gambar 3.3 Flowchart

3.4 AndroidManifest.xml

File AndroidManifest.xml diperlukan oleh setiap aplikasi android, file ini berada pada folder root aplikasi, File ini mendeskripsikan variable global dari paket aplikasi yang kita gunakan, file ini juga berfungsi untuk mendeskripsikan resource apa saja yang akan digunakan oleh project seperti koneksi internet, sms, mengakses gps, dll. Berikut adalah contoh file AndroidManifest.xml yang akan kita gunakan dalam program UIcoding.

Didalam AndroidManifest.xml ini lah berisi semua pendefinisian hal – hal yang dibuthkan oleh aplikasi android kita, beberapa elemen yang terdapat didalam file ini adalah sebagai berikut :

<manifest>

Titik root utama dari AndroidManifest.xml, berisi atribut package aplikasi serta paket activity dalam program android.

Disini terlihat bahwa package kita adalah com.Booting beserta version code yang digunakan.

<uses-permission>

Menjelaskan tentang user permission/security permission yang harus diberikan agar aplikasi dapat berjalan sebagaimana mestinya misalnya, apabila kita menggunakan resources yang tersedia dari sistem, seperti ketika mengirim sms, dan sebagainya.

<permission>

Menjelaskan batasan tentang user permission/security permission.

<instrumentation>

Mendeklarasikan komponen instrument yang tersedia untuk menguji fungsionalitas dari paket aplikasi yang kita gunakan dalam aplikasi android kita.

<application>

Elemen root yang berisi deklarasi aplikasi android kita.

<intent filter>

Mendeklarasikan intent yang dibutuhkan oleh aplikasi android yang kita gunakan, atribut – atribut bisa berikan disini untuk mensupply label, icon, data dan informasi yang kita gunakan dalam aplikasi android.

<action>

Berisi tentang action type yang didukung oleh komponen – komponen yang berada dalam aplikasi android.

<category>

Mendeklarasikan category – category yang didukung oleh aplikasi android

<data>

Mendeklarasikan tipe MIME, URL, authority penggunaan URL serta penentuan path yang digunakan dalam URL.

<meta – data>

Mendeklarasikan meta data yang kita butuhkan sebagai tambahan data yang ada kita gunakan dalam aplikasi android.

<receiver>

Mendeklarasikan dimana aplikasi kita diberikan informasi mengenai sesuatu perubahan atau aksi yang terjadi, seperti menerima sms.

<service>

Mendeklarasikan komponen yang dapat berjalan sebagai service (berjalan di background).

<provider>

Mendeklarasikan komponen – komponen yang mengelola data dan mempublikasikannya untuk dikelola/dipakai oleh aplikasi lain.

<uses-sdk- uses-sdk android:minSdkVersion=?>

Mendeklarasikan sdk android yang digunakan, bisa juga menentukan sdk minimum yang digunakan. Sampai buku ini ditulis, tersedia kemungkinan nilai minSdkVersion sebagai berikut:

- 1 mewakili Android 1.0 SDK
- 2 mewakili Android 1.1 SDK
- 3 mewakili Android 1.5 SDK
- 4 mewakili Android 1.6 SDK

- 5 mewakili Android 2.0 SDK
- 6 mewakili Android 2.0 SDK
- 7 mewakili Android 2.0 Update (2.0.1) SDK
- 8 mewakili Android 2.2 SDK
- 9 mewakili Android 4.2.2 SDK
- 10 mewakili Android 4.2.2.3 SDK
- 11 mewakili Android 3.0
- 12 mewakili Android 3.1
- 13 mewakili Android 3.2
- 14 mewakili Android 4.0
- 15 mewakili Android 4.0.1
- 16 mewakili Android 4.1
- 17 mewakili Android 4.2.1
- 18 mewakili Android 4.2.2
- 19 mewakili Android 4.3

3.6 Activity

Activity merupakan public class dalam aplikasi Android. Setiap activity merupakan sesuatu yang unik atau single, yang ditujukan untuk menghandle macam – macam hal yang bisa dilakukan oleh user. Umumnya, activity berhubungan dengan user dimana activity menciptakan windows atau UI yang mana ditampilkan dengan concepts `setContentView(View)`. Ada 2 method yang pasti dimiliki oleh satu activity, yaitu :

- a. `onCreate` untuk menginisiasi suatu activity, biasanya dipanggil dengan perintah `setContentView(int)` untuk resource yang kita defenisikan di layout UI, dan perintah `findViewById(int)` untuk memanggil widget yang dibutuhkan UI untuk berinteraksi dengan aplikasi.
- b. `onPause` untuk menyatakan ketika user meninggalkan suatu activity

Untuk penggunaan dengan `Context.startActivity()`, semua kelas activity harus sesuai dengan `<activity>` yang dideklarasikan dalam suatu paket di `AndroidManifest.xml`. Activity adalah bagian penting dari keseluruhan siklus aplikasi, bagaimana activity dijalankan merupakan bagian penting dari model aplikasi.

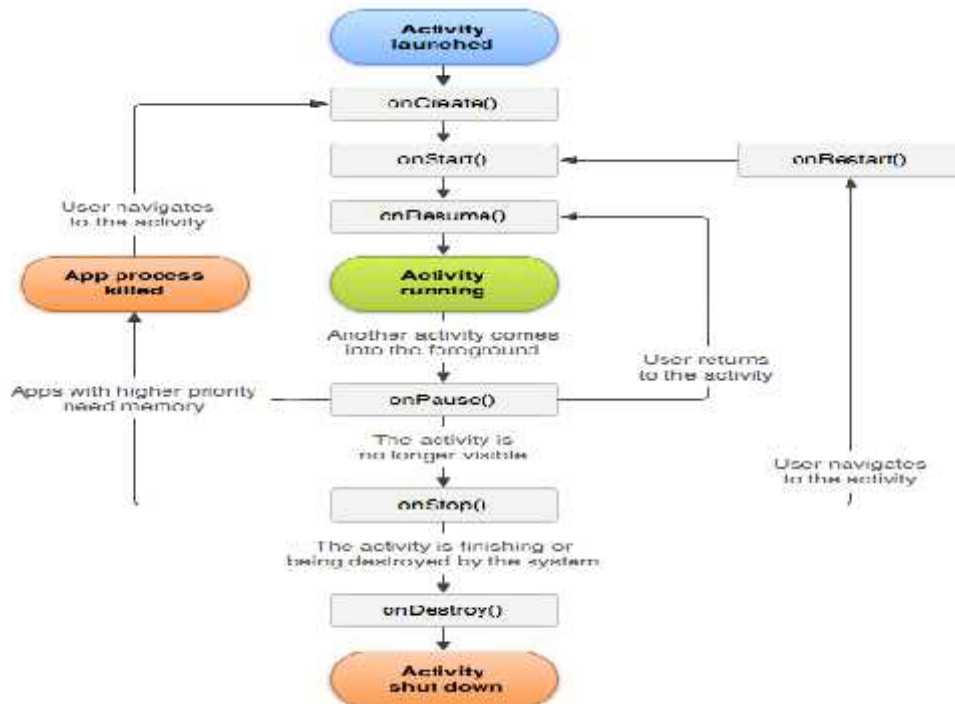
3.6.1 Siklus Activity

Activity aplikasi android dikelola dengan sistem yang dikenal dengan activity stack. Ketika suatu activity start, activity diletakkan pada stack yang paling atas dan activity – activity yang sudah jalan berada dibawahnya dan akan terus berada pada posisi atas stack sampai muncul activity yang baru.

Pada dasarnya activity memiliki empat keadaan, yaitu :

1. active/running, jika activity berada pada posisi atas stack.
2. Pause, jika activity tidak dipakai atau dibutuhkan pada suatu saat tertentu, tetapi activity itu masih ada atau visible, ketika activity baru yang ditangani oleh sistem activity yang lama disebut pause dan masih berada di memory, bisa jadi suatu activity yang sudah keadaan pause tidak ada di memori yang kemungkinan disebabkan oleh keterbatasan memory.
3. Stopped, jika activity sudah tidak dipakai dan digantikan oleh activity lain, activity yang sudah stopped tidak akan pernah dipanggil lagi, dan secara permanen memory pun tidak menyimpan info mengenai activity ini.
4. Restart, jika activity pause atau stopped, sistem dapat mendrop activity ini dari sistem memory, dan ketika user membutuhkan activity tersebut, activity akan kembali ke keadaan awal, artinya activity tersebut mengalami proses restart.

Berikut ini adalah gambaran mengenai activity lifecycle atau siklus dari activity didalam android sistem.



Gambar 3.4 Lifecycle Activity

Berdasarkan gambar lifecycle activity diatas, secara umum dapat kita jelaskan bahwa looping (perulangan) yang mungkin dialami oleh activity ada tiga perulangan, yaitu :

1. Entire lifetime yaitu activity yang terjadi mulai dari onCreate() sampai dengan onDestroy(). Biasanya activity ini akan dibuat setup global ketika mendefinisikannya.
2. Visible lifetime yaitu activity yang terjadi mulai dari onCreate() sampai dengan onStop().
3. Foreground lifetime yaitu activity yang terjadi diantara onResume() dan onPause()

Method	Description	Killable	Next
onCreate()	Ketika sebuah activity dibuat, pada method inilah kita melakukan inisialisasi seperti create view, list data, dll. Method onCreate() selalu diikuti oleh onStart()	N	onStart()
onRestart()	Ketika sebuah activity dihentikan, dan merupakan prioritas untuk memanggil activity itu kembali. Method ini selalu diikuti oleh onStart()	N	onStart()
OnStar()	Ketika sebuah activity dipanggil sebelum diperlihatkan ke user, method ini diikuti oleh onResume() atau onStop()	N	onResume() or onStop()
onResume	Ketika sebuah activity start/ mulai melakukan interaksi dengan user, pada saat ini activity berada pada posisi teratas dari activity stack yang mana user akan melakukan input. Selalu diikuti oleh onPause()	N	onPause()
onPause()	Ketika sebuah activity lainnya dipanggil atau dimulai, method ini digunakan ketika data tidak harus disimpan kedalam sistem secara permanen, method ini diikuti oleh onResume() atau onStop()	Y	onResume() or onStop()
onStop()	Ketika sebuah activity tidak lagi dibutuhkan	Y	onDestroy()

	atau tidak terlihat lagi oleh user, method ini diikuti oleh onRestart() atau onDestroy()		
onDestroy()	Ketika sebuah activity secara permanent tidak lagi dibutuhkan (activity dihancurkan), dapat juga dilakukan dengan fungsi finish() atau isFinishing()	Y	-

Tabel 3.1 Lifecycle Activity

Killable disini maksudnya adalah apakah memungkinkan bagi sistem untuk mematikan proses host activity atau tidak. Sehingga method yang ditandai dengan N itu berarti melindungi host activity (tidak dimatikan).

Sebagai catatan, activity yang secara teknis killable adalah N bisa saja dimatikan secara otomatis oleh sistem apabila terjadi low memory atau tidak cukupnya resource untuk menjaga activity itu tetap hidup. Kejadian tidak cukupnya resource bisa jadi terjadi karena banyak proses yang sedang dieksekusi dan memerlukan kapasitas memory yang besar, sehingga terjadi low memory yang menyebabkan beberapa activity otomatis hilang dari memory.

3.7 Android Widget

Android widget sudah kita implementasikan sedikit pada bagian activity, pada bagian android widget ini akan penulis jelaskan lebih detail mengenai komponen – komponen yang terdapat dalam android widget. Paket widget pada dasarnya merupakan visualisasi dari elemen user interface (UI) yang digunakan pada layar aplikasi android dimana kita dapat merancang sendiri sesuai kebutuhan kita.

Widget didalam android ditampilkan dengan konsep View. Dimana aplikasi android pada umumnya menggunakan widget sebagai layout XML. Untuk megimplementasikan widget, selain file java kita juga membutuhkan tambahan

dua file. Berikut ini adalah file – file yang umumnya kita butuhkan apabila kita membuat widget :

a. File Java

Berupa file java dimana merupakan file yang mengimplementasikan aksi dari widget. Jika anda mendefinisikan suatu widget beserta posisinya dilayar yang didefiisikan dari file XML, kita harus melakukan coding di file java dimana kita dapat mengambil semua nilai atribut dari tata letak file XML yang didefinisikan.

b. File XML

Sebuah file XML yang berada di res/values/ yang mendefinisikan komponen elemen – elemen XML yang digunakan untuk inisialisasi widget serta atribut yang mendukungnya (file yang kita kenal dengan main.xml pada bab bagian – bagian sebelumnya).

c. Layout XML [optional]

File ini tidak harus ada, file XML berada didalam res/layout yang menggambarkan atau penambahan keterangan pada layout widget anda. Sebenarnya ini kita dapat dilakukan di file java kita.

AbsListView	CursorAdapter
AbsListView.LayoutParams	CursorTreeAdapter
AbsoluteLayout	DatePicker
AbsSeekBar	DialerFilter
AbsSpinner	DigitalClock
AdapterView<T extends Adapter>	EditText

AdapterView.AdapterContextMenuInfo	ExpandablesListView
AlphabetIndexer	ExpandableListView.ExpandableListContextMenuInfo
AnalogClock	Filter
ArrayAdapter<T>	Filter.FilterResults
AutoCompleteTextView	FrameLayout
BaseAdapter	FrameLayout.LayoutParams
BaseExpandableListAdapter	Gallery
Button	Gallery.LayoutParams
CheckBox	GridView
CheckedTextView	HeaderViewListAdapter
Chronometer	HorizontalScrollView
ImageSwitcher	SeekBar
ImageView	SimpleAdapter
LinearLayout	SimpleCursorAdapter
LinearLayout.LayoutParams	SimpleCursorTreeAdapter
ListView	SimpleExpandableListAdapter
ListView.FixedViewInfo	SlidingDrawer
MediaController	Spinner
MultiAutoCompleteTextView	TabHost
MultiAutoCompleteTextView.Co	TabHost.TabSpec

mmaTokenizer	
OverScroller	TableLayout
PopupWindow	TableLayout.LayoutParams
ProgressBar	TableRow
QuickContactBadge	TableRow.LayoutParams
RadioButton	TabWidget
RadioGroup.LayoutParams	TimePicker
RemoteViews	Toast
ResourceCursorAdapter	ToggleButton
ResourceCursorTreeAdapter	TwoLineListItem
Scroller	VideoView
ScrollView	ViewAnimator

Tabel 3.2 Jenis – jenis (classes) android widgets

BAB IV

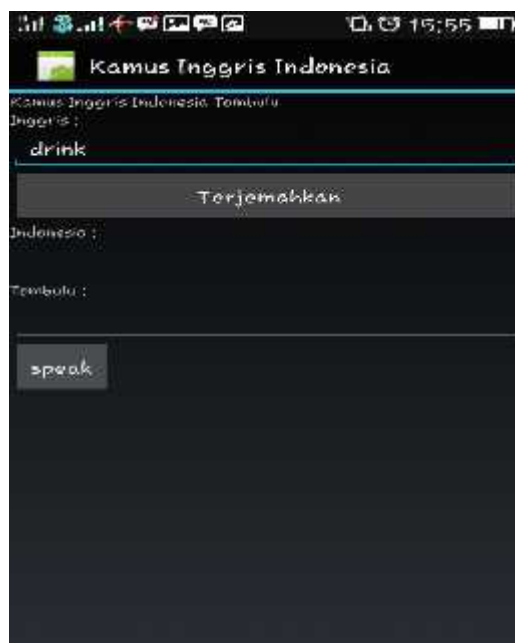
PENGUJIAN DAN IMPLEMENTASI SISTEM

Desain sistem ini bertujuan untuk membuat sistem yang dapat memenuhi sasaran atau tujuan yang diinginkan.

4.1 Implementasi Program

Tampilan hasil akhir dari Aplikasi Kamus terjemahan Bahasa Inggris ke Bahasa Indonesia dan Tomohon menggunakan voice read text.

4.2 Tampilan Akhir Kamus



Gambar 4.1 Tampilan Inteface Kamus

Pada gambar diatas menampilkan tampilan aplikasi yang akan menerjemahkan kata “drink” ke bahasa Indonesia dan Tombulu.



Gambar 4.2 Hasil Terjemahan

Pada gambar diatas menampilkan hasil terjemahan dari kata “drink”. Untuk melakukan fungsi voice read text, tekan button “speak” untuk mendengar suara dari bahasa Tomohon (tombulu).



Gambar 4.3 Tampilan untuk terjemahan yang tidak tersedia

Pada gambar diatas menampilkan hasil terjemahan yang muncul ketika user menginput kata yang tidak tersedia oleh aplikasi.

4.3 Data Uji Coba

Data uji coba ini diambil dari hasil penilaian dari para pengguna aplikasi terjemahan bahasa Indonesia ke bahasa Tombulu menggunakan voice read text berbasis android.

No	Nama Pengguna	Tingkat Penilaian	
		Puas	Tidak Puas
1	Ripka Legi	×	
2	Ivan Gilang		×
3	Gregory Guit		×
4	Patrick Soleman	×	
5	Dimas Saputra		×
6	Maya Kotulus	×	
7	Indiani Atika		×
8	Marko Lumi	×	
9	Trizy Paat	×	
10	Amanda Wuisan		×

Tabel 4.1 Daftar Uji Coba

BAB V

KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dan saran secara menyeluruh berdasarkan tujuan dan hasil dari “Aplikasi Terjemahan Bahasa Inggris – Bahasa Indonesia dan Tomohon dengan Voice Read Text”.

5.1 Kesimpulan

Kesimpulan diperoleh setelah melakukan penelitian dan pengujian aplikasi adalah sebagai berikut :

1. Dengan adanya aplikasi terjemahan dengan voice read text maka tujuan untuk membuat kamus yang dapat mengeluarkan suara terpenuhi
2. Dengan adanya aplikasi terjemahan dengan voice read text dapat memudahkan pengguna gadget android untuk belajar bahasa khas Tomohon yaitu bahasa Tombulu
3. Dengan adanya aplikasi terjemahan dengan voice read text yang berbasis android, maka pengguna gadget android dapat dengan mudah belajar bahasa Tombulu dimanapun dan kapanpun.

5.2 Saran

Berdasarkan hasil dan perancangan pengimplementasian aplikasi. Aplikasi terjemahan dengan voice read text ini dapat diberikan saran – saran sebagai bahan penyempurnaan dan pengembangan lebih lanjut. Berikut ini adalah saran – saran yang mungkin dapat digunakan untuk pengembang aplikasi “Aplikasi Terjemahan Bahasa Inggris – Bahasa Indonesia dan Tomohon dengan Voice Read Text” ini :

1. Aplikasi ini dapat dikembangkan dengan menambahkan lebih banyak kosakata yang belum ada pada aplikasi ini
2. Aplikasi ini dapat dikembangkan dalam skala yang lebih besar dengan menambahkan bahasa yang baru.
3. Aplikasi ini dapat dikembangkan sistem interface menjadi tampilan yang lebih baik.

DAFTAR PUSTAKA

Mufti Yusus (2012). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung, INFORMATIKA.

Nazruddin Safaat H (2014). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android Revisi Kedua*. Bandung, INFORMATIKA.

Yudistira Yuan (2013). *Membuat Aplikasi iPhone Android & BlackBerry Itu Gampang*. Jakarta Selatan, Media Kita.

<http://www.belajarandroid.com/kamus-bersuara>, diakses pada tanggal 15 Juli 2015

<http://ilmusay.blogspot.co.id/2012/11/metodelogi-waterfall.html>, diakses pada tanggal 15 Juli 2015

<https://id.wikipedia.org/wiki/Tombulu>, diakses pada tanggal 15 Juli 2015

LAMPIRAN

1. Coding Class showKamus.java

```
package com.Bootling;

import android.app.Activity;
import android.os.Bundle;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.Button;
import android.widget.Toast;

//import tambahan
import java.util.Locale;
import android.content.Intent;
import android.speech.tts.TextToSpeech;
import android.view.View.OnClickListener;
import android.speech.tts.TextToSpeech.OnInitListener;

public class showKamus extends Activity implements OnInitListener{
    //percobaan
    private TextToSpeech membaca;
    private int KODE_UNTUK_PRIKSA_TTS = 0;
    private Button speak;
    //batas percobaan
    private SQLiteDatabase db = null;
    private Cursor kamusCursor = null;
    private EditText txtInggris;
    private EditText txtIndonesia;
    private EditText txtTombulu;
    private DataKamus datakamus = null;
    public static final String INDONESIA = "indonesia";
    public static final String INGGRIS = "inggris";
    public static final String TOMBULU = "tombulu";
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        datakamus = new DataKamus(this);
        db = datakamus.getWritableDatabase();
        datakamus.createTable(db);
        datakamus.generateData(db);

        setContentView(R.layout.main);
        membaca = new TextToSpeech(this, this);
        txtInggris = (EditText) findViewById(R.id.txtInggris);
        txtIndonesia = (EditText) findViewById(R.id.txtIndonesia);
        txtTombulu = (EditText) findViewById(R.id.txtTombulu);
        //percobaan
        speak = (Button) findViewById(R.id.speak);
        speak.setOnClickListener(new View.OnClickListener() {

            public void onClick(View arg0) {
                speakOut();
            }

        });
    }
}
```

```

//batas percobaan
public void getTerjemahan(View view) {
    String result= "";
    String result2= "";

    String englishword = txtInggris.getText().toString();
    kamusCursor = db.rawQuery("SELECT ID, INGGRIS, INDONESIA, TOMBULU " + "FROM
kamus where INGGRIS = " + englishword + "ORDER BY INGGRIS", null);

    //Membaca data dari kamusCursor dan menampungnya di result
    //Membaca string di index 2 yaitu bahasa inggris
    if (kamusCursor.moveToFirst()) {
        result = kamusCursor.getString(2);
        for (;!kamusCursor.isAfterLast(); kamusCursor.moveToNext()) {
            result = kamusCursor.getString(2);
        }
    }

    //Membaca data dari kamusCursor dan menampungnya di result
    //Membaca string di index 3 yaitu bahasa tombulu
    if (kamusCursor.moveToFirst()) {
        result2 = kamusCursor.getString(3);
        for (;!kamusCursor.isAfterLast(); kamusCursor.moveToNext()) {
            result2 = kamusCursor.getString(3);
        }
    }

    //Menampilkan not found jika data tidak sesuai atau tidak ada di database
    if (result.equals("")) {
        result = "Terjemahan Not Found";
    }

    if (result2.equals("")) {
        result2 = "Terjemahan Not Found";
    }

    //Menampilkan datanya setelah ditampung
    txtIndonesia.setText(result);
    txtTombulu.setText(result2);
}

@Override
public void onDestroy() {
    super.onDestroy();
    kamusCursor.close();
    db.close();
}

//percobaan lagi
public void onInit(int status) {
    // TODO Auto-generated method stub

    if (status == TextToSpeech.SUCCESS) {

        int result = membaca.setLanguage(Locale.US);

        if (result == TextToSpeech.LANG_MISSING_DATA

```

```

        || result ==
TextToSpeech.LANG_NOT_SUPPORTED) {
    Toast.makeText(this, "Language not supported",
Toast.LENGTH_LONG).show();
    Log.e("TTS", "Language is not supported");
    } else {
        speak.setEnabled(true);
    }
    } else {
        Log.e("TTS", "Initialization Failed");
    }
}

private void speakOut() {
    String text = txtTombulu.getText().toString();
    if (text.length() == 0) {
        membaca.speak("You haven't typed text",
TextToSpeech.QUEUE_FLUSH, null);
    } else {
        membaca.speak(text, TextToSpeech.QUEUE_FLUSH, null);
    }
}
//batas percobaan
}

```

2. Coding Class DataKamus.java

```

package com.Bootng;

import android.content.ContentValues;

import android.content.Context;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

public class DataKamus extends SQLiteOpenHelper{

    private static final String DATABASE_NAME = "dbkamus";

    public static final String INGGRIS = "inggris";

    public static final String INDONESIA = "indonesia";

    public static final String TOMBULU = "tombulu";

    //Constructor DataKamus untuk initiate database

    public DataKamus(Context context) {

```

```

        super(context, DATABASE_NAME, null, 1);
    }

    //Method createTable untuk membuat table kamus
    public void createTable(SQLiteDatabase db) {
        db.execSQL("DROP TABLE IF EXISTS kamus");
        db.execSQL("CREATE TABLE if not exists kamus (id INTEGER PRIMARY KEY
        AUTOINCREMENT, inggris TEXT, indonesia TEXT, tombulu TEXT);");
    }

    //Method generateData untuk mengisi data ke kamus.
    public void generateData(SQLiteDatabase db) {
        ContentValues cv = new ContentValues();
        cv.put(INGGRIS,"and");
        cv.put(INDONESIA, "dan");
        cv.put(TOMBULU, "wo");

        db.insert("kamus", INGGRIS, cv);

        cv.put(INGGRIS, "one");
        cv.put(INDONESIA, "satu");
        cv.put(TOMBULU, "esa");

        db.insert("kamus", "INGGRIS", cv);

        cv.put(INGGRIS, "drink");
        cv.put(INDONESIA, "minum");
        cv.put(TOMBULU, "melep");

        db.insert("kamus", INGGRIS, cv);

        cv.put(INGGRIS, "five");
        cv.put(INDONESIA, "lima");
    }

```

```
cv.put(TOMBULU, "lima");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "far");
```

```
cv.put(INDONESIA, "jauh");
```

```
cv.put(TOMBULU, "rou");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "star");
```

```
cv.put(INDONESIA, "bintang");
```

```
cv.put(TOMBULU, "tetamun");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "dust");
```

```
cv.put(INDONESIA, "debu");
```

```
cv.put(TOMBULU, "awu");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "two");
```

```
cv.put(INDONESIA, "dua");
```

```
cv.put(TOMBULU, "rua");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "three");
```

```
cv.put(INDONESIA, "tiga");
```

```
cv.put(TOMBULU, "telu");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "back");
cv.put(INDONESIA, "belakang");
cv.put(TOMBULU, "wuri");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "bad");
cv.put(INDONESIA, "buruk");
cv.put(TOMBULU, "lewoan");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "all");
cv.put(INDONESIA, "semua");
cv.put(TOMBULU, "pelen");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "big");
cv.put(INDONESIA, "besar");
cv.put(TOMBULU, "sela");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "bird");
cv.put(INDONESIA, "burung");
cv.put(TOMBULU, "burun");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "blood");
cv.put(INDONESIA, "darah");
cv.put(TOMBULU, "ra");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "black");
cv.put(INDONESIA, "hitam");
cv.put(TOMBULU, "remdem");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "bone");
cv.put(INDONESIA, "tulang");
cv.put(TOMBULU, "rui");
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "breath");
cv.put(INDONESIA, "bernafas");
cv.put(TOMBULU, "maasen");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "burn");
cv.put(INDONESIA, "bakar");
cv.put(TOMBULU, "riket");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "child");
cv.put(INDONESIA, "anak");
cv.put(TOMBULU, "oki");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "cloud");
cv.put(INDONESIA, "awan");
cv.put(TOMBULU, "ninaa");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "wind");
cv.put(INDONESIA, "angin");
cv.put(TOMBULU, "rate");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "come");
cv.put(INDONESIA, "datang");
cv.put(TOMBULU, "meye");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "count");
cv.put(INDONESIA, "hitung");
cv.put(TOMBULU, "reken");
db.insert("kamus", "INGGRIS", cv);
cv.put(INGGRIS, "cut");
cv.put(INDONESIA, "potong");
cv.put(TOMBULU, "kator");

db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "dirty");
cv.put(INDONESIA, "kotor");
cv.put(TOMBULU, "keremus");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "dog");
cv.put(INDONESIA, "anjing");
cv.put(TOMBULU, "asu");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "stupid");
cv.put(INDONESIA, "bodoh");
cv.put(TOMBULU, "wenel");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "ear");
cv.put(INDONESIA, "telinga");
cv.put(TOMBULU, "luntan");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "earth");
cv.put(INDONESIA, "tanah");
cv.put(TOMBULU, "tana");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "egg");
cv.put(INDONESIA, "telur");
cv.put(TOMBULU, "atelu");
```



```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "eye");
```

```
cv.put(INDONESIA, "mata");
```

```
cv.put(TOMBULU, "weren");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "fall");
```

```
cv.put(INDONESIA, "jatuh");
```

```
cv.put(TOMBULU, "ragos");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "far");
```

```
cv.put(INDONESIA, "jauh");
```

```
cv.put(TOMBULU, "rou");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "weak");
```

```
cv.put(INDONESIA, "lemah");
```

```
cv.put(TOMBULU, "weles");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "father");
```

```
cv.put(INDONESIA, "ayah");
```

```
cv.put(TOMBULU, "ama");
```

```
db.insert("kamus", "INGGRIS", cv);
```

```
cv.put(INGGRIS, "fear");
```

```
cv.put(INDONESIA, "takut");
cv.put(TOMBULU, "inde");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "fish");
cv.put(INDONESIA, "ikan");
cv.put(TOMBULU, "sera");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "flow");
cv.put(INDONESIA, "mengalir");
cv.put(TOMBULU, "masoso");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "flower");
cv.put(INDONESIA, "bunga");
cv.put(TOMBULU, "wuna");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "fly");
cv.put(INDONESIA, "terbang");
cv.put(TOMBULU, "tewel");

db.insert("kamus", "INGGRIS", cv);

cv.put(INGGRIS, "clouded");
cv.put(INDONESIA, "kabut");
cv.put(TOMBULU, "semu");
```

```

        db.insert("kamus", "INGGRIS", cv);

        cv.put(INGGRIS, "jump");
        cv.put(INDONESIA, "lompat");
        cv.put(TOMBULU, "konta");

        db.insert("kamus", "INGGRIS", cv);

        cv.put(INGGRIS, "fruit");
        cv.put(INDONESIA, "buah");
        cv.put(TOMBULU, "wua");

        db.insert("kamus", "INGGRIS", cv);

        cv.put(INGGRIS, "give");
        .put(INDONESIA, "memberi")
        cv.put(TOMBULU, "we");

        db.insert("kamus", "INGGRIS", cv);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        //TODO Auto-generated method stub
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        //TODO Auto-generated method sub
    }
}

```

}

3. Coding Class Main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Kamus Inggris Indonesia Tomblu"
    />
<TextView
    android:text="Inggris : "
    android:id="@+id/TextView01"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
</TextView>
<EditText
    android:text=""
    android:id="@+id/txtInggris"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    />
<Button
    android:text="Terjemahkan"
    android:id="@+id/btnTerjemah"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:onClick="getTerjemahan"
    />
<TextView
    android:text="Indonesia : "
    android:id="@+id/TextView03"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    />
<EditText
    android:text=""
    android:id="@+id/txtIndonesia"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:editable="false"
    />
<TextView
    android:text="Tomblu : "
    android:id="@+id/TextView04"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    />
<EditText
    android:text=""
    android:id="@+id/txtTomblu"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:editable="false"
    />
<Button
    android:id="@+id/speak"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="speak"
    />
</LinearLayout>
```

4. Coding Class manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.Booting"
    android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="17" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".showKamus"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```