

**APLIKASI KAMUS NAMA ILMIAH TUMBUHAN
DAN HEWAN BERBASIS ANDROID DENGAN
METODE SEQUENTIAL SEARCHING**



SKRIPSI

Diajukan untuk memenuhi salah satu syarat kelulusan Strata Satu (S1)

AHMAD RIKI SETIAWAN

12135076

Program Studi Teknik Informatika
Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri
Jakarta
2016

PERSEMBAHAN

Pada lembar persembahan ini, penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang sangat mendukung penulis dalam pembuatan dan penyusunan skripsi ini, adapun yang dimaksud adalah sebagai berikut:

1. Kepada orang tua penulis dan keluarga penulis yang selalu memberikan dukungan berupa materi dan do'a kepada penulis sehingga penulis bisa menyelesaikan karya tulis ini.
2. Kepada Ahmad Mustaqim yang selalu memberikan motivasi dan semangat.
3. Kepada Yudha Finandhi yang memberikan semangat dan tak pernah berhenti untuk mendukung penulis.
4. Kepada Fahri Marasabessy yang telah mengajari penulis cara membuat aplikasi android dan juga yang telah membimbing dalam pembuatan aplikasi.
5. Kepada Muhammad Arifin yang selalu setia dan memberikan dukungan yang luar biasa kepada penulis.
6. Kepada sahabat-sahabat yang selalu ada disaat penulis membutuhkan.

Semoga Tuhan Yang Maha Kuasa memberikan balasan yang setimpal dengan kebaikan yang penulis sampaikan diatas. Amin Ya Rabbal'Alamin.

SURAT PERNYATAAN KEASILAN SKRIPSI

Yang bertanda tangan dibawah ini:

Nama : Ahmad Riki Setiawan

NIM : 12135076

Perguruan Tinggi : STMIK Nusa Mandiri Jakarta

Dengan ini menyatakan bahwa skripsi yang telah saya buat dengan judul: **“Aplikasi Kamus Nama Ilmiah Tumbuhan dan Hewan Berbasis Android Dengan Metode Sequential Searching”**, adalah asli (orisinil) atau tidak plagiat (menjiplak) dan belum pernah diterbitkan/dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya tanpa ada paksaan dari pihak manapun juga. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa skripsi yang telah saya buat adalah karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari **Sekolah Tinggi Manajemen Informatika & Komputer Nusa Mandiri** dicabut/dibatalkan.

Dibuat di : Jakarta

Pada tanggal : 06 Januari 2017

Yang menyatakan

Ahmad Riki Setiawan

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Yang bertanda tangan dibawah ini, saya:

Nama : Ahmad Riki Setiawan
NIM : 12135076
Program Studi : Teknik Informatika
Perguruan Tinggi : STMIK Nusa Mandiri Jakarta

Dengan ini menyetujui untuk memberikan ijin kepada pihak **Sekolah Tinggi Manajemen Informatika & Komputer Nusa Mandiri**, Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti-Free Right*) atas karya ilmiah kami berjudul: **“Aplikasi Kamus Nama Ilmiah Tumbuhan dan Hewan Berbasis Android Dengan Metode Sequential Searching”**, beserta perangkat yang diperlukan(apabila).

Dengan **Hak Bebas Royalti Non-Eksklusif** ini pihak **Sekolah Tinggi Manajemen Informatika & Komputer Nusa Mandiri** berhak menyimpan mengalih-media atau *format*-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya dan menampilkan atau mempublikasikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari kami selama tetap mencantumkan nama kami sebagai penulis/pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak **Sekolah Tinggi Manajemen Informatika & Komputer Nusa Mandiri**, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jakarta
Pada tanggal : 06 Januari 2017
Yang menyatakan

Ahmad Riki Setiawan

PERSETUJUAN DAN PENGESAHAN SKRIPSI

Skripsi ini diajukan oleh:

Nama : Ahmad Riki Setiawan
NIM : 12135076
Program Studi : Teknik Informatika
Jenjang : Strata Satu (S1)
Judul Skripsi : Aplikasi Kamus Nama Ilmiah Tumbuhan dan Hewan
Berbasis Android Dengan Metode Sequential Searching

Telah dipertahankan pada periode II-2016 dihadapan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh Sarjana Ilmu Komputer (S.Kom) pada Program Strata Satu (S1) Program Studi Teknik Informatika di Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri.

Jakarta, 16 Februari 2017

PEMBIMBING SKRIPSI

Dosen Pembimbing I : Pas Mahyu Akhirianto S.Pd, M.Kom

D E W A N P E N G U J I

Penguji I : Desmulyati, ST, M.Kom

Penguji II : Giatika Chrisnawati, ST, M.Kom

PANDUAN PENGGUNAAN HAK CIPTA

Skripsi sarjana yang berjudul “**Aplikasi Kamus Nama Ilmiah Tumbuhan dan Hewan Berbasis Android Dengan Metode Sequential Searching**” adalah hasil karya tulis asli AHMAD RIKI SETIAWAN dan bukan hasil terbitan sehingga peredaran karya tulis hanya berlaku dilingkungan akademik saja, serta memiliki hak cipta. Oleh karena itu, dilarang keras untuk menggandakan baik sebagian maupun seluruhnya karya tulis ini tanpa seizin penulis.

Referensi kepustakaan diperkenankan untuk dicatat tetapi pengutipan atau peringkasan isi tulisan hanya dapat dilakukan dengan seizin penulis dan disertai ketentuan pengutipan secara ilmiah dengan menyebutkan sumbernya.

Untuk keperluan perizinan pada pemilik dapat menghubungi informasi yang tertera di bawah ini:

Nama	: Ahmad Riki Setiawan
Alamat	: Jl. Kayu Besar RT.003/012 Jakarta Barat
No. Telp	: 081218469935
Email	: ahmadc242@gmail.com

KATA PENGANTAR

Puji syukur alhamdulillah, penulis kehadiran Allah SWT, yang telah melimpahkan rahmat dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan Skripsi ini dengan baik. Dimana skripsi ini penulis sajikan dalam bentuk buku yang sederhana. Adapaun judul skripsi yang penulis ambil sebagai berikut, “APLIKASI KAMUS NAMA ILMIAH TUMBUHAN DAN HEWAN DENGAN METODE SEQUENTIAL SEARCHING”.

Tujuan penulisan skripsi ini dibuat salah satu syarat kelulusan program Strata Satu (S1) STMIK Nusa Mandiri. Sebagai bahan penulisan diambil berdasarkan hasil penelitian (eksperimen), observasi dan beberapa sumber literatur yang mendukung penulisan ini. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan skripsi ini tidak akan lancar. Oleh karena itu pada kesempatan ini, izinkanlah penulis menyampaikan ucapan terima kasih kepada:

1. Ketua STMIK Nusa Mandiri.
2. Pembantu Ketua I STMIK Nusa Mandiri.
3. Ketua Program Studi Teknik Informatika STMIK Nusa Mandiri.
4. Bapak Pas Mahyu Akhirianto SPd, M.Kom selaku Dosen Pembimbing I Skripsi
5. Bapak/Ibu Dosen Teknik Informatika STMIK Nusa Mandiri yang telah memberikan penulis dengan semua bahan yang diperlukan.
6. Staff / Karyawan / Dosen di lingkungan STMIK Nusa Mandiri.
7. Orang tua tercinta yang telah memberikan dukungan moral maupun spiritual.

8. Rekan-rekan mahasiswa kelas 12.7AA.05

Serta semua pihak yang terlalu banyak untuk disebut satu persatu sehingga terwujudnya penulisan ini. Penulis menyadari penulisan skripsi ini masih jauh sekali dari sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan dimasa yang akan datang.

Akhir kata semoga skripsi ini dapat berguna bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Jakarta, 22 Januari 2017

Penulis

Ahmad Riki Setiawan

ABSTRAKSI

Ahmad Riki Setiawan (12135076), Aplikasi Kamus Nama Ilmiah Tumbuhan dan Hewan Berbasis Android Dengan Metode Sequential Searching.

Kamus Nama Ilmiah atau bahasa latin hewan dan tumbuhan berbasis android adalah suatu pengembangan aplikasi *mobile* yang memungkinkan penggunanya dapat mencari kata dengan mudah dan cepat. Aplikasi ini memungkinkan pengguna dengan cepat mencari arti dari bahasa latin hewan maupun tumbuhan. Proses yang terjadi pada aplikasi ini meliputi pencarian kesamaan kata dengan menggunakan metode pencarian *Sequential Searching*. *Input*-an berupa nama hewan atau tumbuhan merupakan nilai *string* yang akan dijadikan parameter pencarian didalam *database*. Hasil pengujian Kamus pencarian menggunakan metode *Sequential Searching* berhasil dengan kecepatan kurang dari satu detik waktu pencarian. Dalam pengujian ini, digunakan kata terdepan dalam *database* sehingga pencarian kata lebih cepat. Kecepatan proses pencarian menggunakan metode *Sequential Searching* bergantung kata yang dicari dalam indeks pencarian.

Kata Kunci: Android, Kamus, Bahasa Latin, Sequential Searching

ABSTRACTION

Ahmad Riki Setiawan (12135076), Application Dictionary of Scientific Name Plant and Animal-Based Android With Sequential Searching Method.

Dictionary of Scientific or Latin name of plants and animals is an android based mobile application development that allows users to search for words easily and quickly. This application allows users to quickly search for the meaning of the Latin animals and plants. The process that occurs in these applications include word similarity search using the search method Sequential Searching. Input form animal or plant form of the name is a string value that will be used as search parameters in the database. The test results using methods Sequential Searching Dictionary search successfully at speeds of less than one second of time searching. In this test, used the word leader in the database so that the search word more quickly. Speed search process using the method relies Sequential Searching search word in the search index.

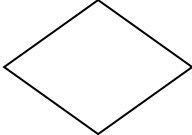
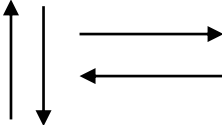
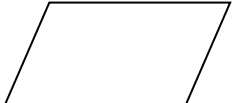
Keywords: Android, Dictionaries, Latin, Sequential Searching

DAFTAR ISI

LEMBAR JUDUL SKRIPSI.....	i
LEMBAR PERSEMBAHAN	ii
LEMBAR PERNYATAAN KEASLIAN SKRIPSI	iii
LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	iv
LEMBAR PERSETUJUAN DAN PENGESAHAN SKRIPSI.....	v
LEMBAR PANDUAN PENGGUNAAN HAK CIPTA	vi
 Kata Pengantar	vii
Abstraksi.....	ix
Daftar Isi	xi
Daftar Simbol	xiii
Daftar Gambar	xvii
Daftar Tabel.....	xviii
Daftar Lampiran	xix
 BAB I PENDAHULUAN.....	1
1.1.Latar Belakang	1
1.2.Perumusan Masalah	4
1.3.Maksud dan Tujuan.....	4
1.4.Metode Penelitian.....	5
1.4.1. Teknik Pengumpulan Data	5
1.4.2. Metode Pengembangan Sistem	5
1.5.Ruang Lingkup.....	6
 BAB II LANDASAN TEORI	7
2.1. Tinjauan Jurnal.....	7
2.2. Konsep Dasar Pemograman	7
2.2.1. Android	8
2.2.2. Kamus	9
2.2.3. Sejarah Android	9
2.2.4. Arsitektur Android	11
2.2.5. Java.....	15
2.2.6. Android Studio	16
2.2.7. Android Virtual Device.....	16
2.2.8. Android SDK	17
2.2.9. Object Oriented Program(OOP).....	17
2.2.10 SQLite Database	20
2.3. Metode Algoritma	20
2.4. Pengujian Software	22
2.4.1. Pengujian White Box	22
2.4.2. Pengujian Black Box.....	22
2.5. UML (Unified Modeling Language).....	23
2.5.1 Jenis-Jenis Diagram	24

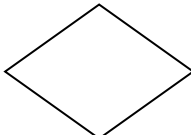
BAB III	ANALISA DAN RANCANGAN SOFTWARE	29
	3.1. Analisa Kebutuhan Software	29
	3.1.1. Identifikasi Masalah	29
	3.1.2. Analisa Kebutuhan	30
	3.2. Desain.....	32
	3.2.1. Rancangan Algoritma	33
	3.2.2. Software Architecture	33
	3.2.3. User Interface	43
	3.3. Implementasi	50
	3.4. Testing.....	56
	1. White Box	56
	2. Black Box.....	59
	3.5. Support	60
BAB IV	PENUTUP	61
	4.1. Kesimpulan	61
	4.2. Saran.....	61
	DAFTAR PUSTAKA	63
	DAFTAR RIWAYAT HIDUP	64
	LEMBAR KONSULTASI BIMBINGAN	65
	LAMPIRAN.....	66

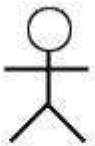
DAFTAR SIMBOL

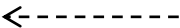
a.Simbol Flow Chart	
	TERMINAL Digunakan untuk menggambarkan awal dan akhir dari suatu kegiatan.
	DECISION Digunakan untuk menggambarkan proses pengujian suatu kondisi yang ada.
	FLOW LINE Digunakan untuk menggambarkan hubungan proses dari satu proses ke proses lainnya.
	PROSES Digunakan untuk menggambarkan pengolahan yang dilakukan oleh siste,
	INPUT / OUTPUT Digunakan untuk menggambarkan proses memasukan data yang berupa pembacaan data dan sekaligus proses keluaran yang berupa pencetakan data.

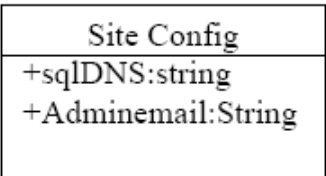
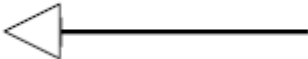
b. Simbol Use Case	
	ACTOR Menspesifikasikan peran yang pengguna gunakan ketika berinteraksi dengan use case
	SYSTEM BOUNDARY Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	GENERALIZATION Hubungan dimana anak objek berbagi perilaku dan struktur data dari objek yang ada diatasnya atau objek induk.
	ASSOCIATION Menghubungkan antara objek satu dengan objek lainnya.

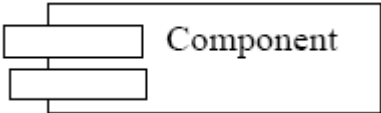
	USE CASE Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
	EXTEND Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.

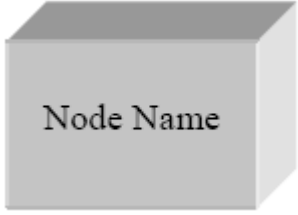
c. Simbol Activity Diagram	
	ACTIVITY Menunjukkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
	ACTION State dari sistem yang mencerminkan eksekusi dari suatu aksi
	INITIAL NODE Bagaimana objek dibentuk atau diawali atau bisa disebut titik awal.
	ACTIVITY FINAL NODE Bagaimana objek diakhiri atau bisa disebut titik akhir.
	DECISION Pilihan untuk mengambil keputusan.

d. Simbol Sequence Diagram	
	OBJECT LIFELINE <i>Object Lifeline</i> : menggambarkan <i>object</i> apa saja yang terlibat.
	ACTOR <i>Actor</i> : menggambarkan hubungan <i>actor</i> yang terlibat.

	ACTIVATION <i>Activation</i> : menggambarkan hubungan antara <i>object</i> dengan <i>message</i> .
	MESSAGE (CALL) <i>Message(call)</i> : menggambarkan alur <i>message</i> yang merupakan kejadian objek pengirim <i>lifeline</i> ke objek penerima <i>lifeline</i> .
	MESSAGE (RETURN) <i>Message(return)</i> : menggambarkan alur pengambilan <i>message</i> ke objek pemanggil dan tanda bahwa objek penerima telah menyelesaikan prosesnya.

e. Simbol Class Diagram	
	CLASS <i>Class</i> adalah blok - blok pembangun pada pemrograman berorientasi obyek. Sebuah <i>class</i> digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari <i>class</i> . Bagian tengah mendefinisikan <i>property/atribut class</i> . Bagian akhir mendefinisikan <i>method-method</i> dari sebuah <i>class</i> .
	DEPENDENCY Kadangkala sebuah <i>class</i> menggunakan <i>class</i> yang lain. Hal ini disebut <i>dependency</i> . Umumnya penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain.
	GENERALISATION Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

f. Simbol Deployment Diagram	
	COMPONENT Pada deployment diagram, komponen-komponen yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka

	NODE Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.
	ASSOCIATION Sebuah association digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara element-elemen hardware
	DEPENDENCY Kebergantungan antara node, arah panah mengarah anak panah yang dipakai

DAFTAR GAMBAR

	Halaman
Gambar II.1 Arsitektur Android.....	15
Gambar II.2 Contoh Use Diagram	24
Gambar II.3 Contoh Activity Diagram	25
Gambar II.4 Contoh Sequence Diagram	26
Gambar II.5 Contoh Class Diagram.....	27
Gambar II.6 Contoh Deploymnet Diagram.....	28
Gambar III.1 Diagram Use Case.....	34
Gambar III.2 Activity Diagram Menu Utama.....	35
Gambar III.3 Activity Diagram Menu Hewan	36
Gambar III.4 Activity Diagram Menu Tumbuhan	36
Gambar III.5 Activity Diagram Menu Daftar Kamus.....	37
Gambar III.6 Activity Diagram Menu Info.....	37
Gambar III.7 Activity Diagram Menu Keluar.....	38
Gambar III.8 Sequence Diagram Menu Hewan.....	39
Gambar III.9 Sequence Diagram Menu Tumbuhan.....	39
Gambar III.10 Sequence Diagram Menu Daftar Kamus.....	40
Gambar III.11 Sequence Diagram Menu Info.....	40
Gambar III.12 Sequence Diagram Menu Keluar	41
Gambar III.13 Class Diagram	42
Gambar III.14 Deployment Diagram	43
Gambar III.15 Rancangan Interface Menu Utama	44
Gambar III.16 Rancangan Interface Menu Hewan	45
Gambar III.17 Rancangan Interface Menu Tumbuhan	46
Gambar III.18 Rancangan Interface Menu Daftar Kamus	47
Gambar III.19 Rancangan Interface Menu Daftar Kamus Hewan.....	48
Gambar III.20 Rancangan Interface Menu Daftar Kamus Tumbuhan.....	49
Gambar III.21 Rancangan Interface Menu Info	49
Gambar III.22 Splash Screen	51
Gambar III.23 Menu Utama.....	51
Gambar III.24 Menu Terjemah Hewan.....	52
Gambar III.25 Menu Terjemah Tumbuhan.....	53
Gambar III.26 Menu Daftar Kamus	53
Gambar III.27 Daftar Kamus Hewan	54
Gambar III.28 Daftar Kamus Tumbuhan	55
Gambar III.29 Menu Info.....	55
Gambar III.30 Tampilan Keluar.....	56
Gambar III.31 Flowchart Diagram.....	57
Gambar III.32 Diagram Alir	58

DAFTAR TABEL

	Halaman
Tabel II.1 Versi Android.....	11
Tabel III.1 Pengujian Black Box.....	59
Tabel III.2 Kebutuhan Perangkat Aplikasi.....	60

DAFTAR LAMPIRAN

	Halaman
Lampiran A. Emulator Alternatif.....	66
Lampiran B. Spesifikasi Minimal Komputer untuk Android Studio	67
Lampiran C. Listing Source Code Program.....	67

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi sangat pesat penggunaan ponsel lebih didominasi oleh ponsel pintar yang dikenal dengan nama *smartphone*. Seiring berjalan waktu, fungsi dasar dari ponsel biasa telah tergantikan dengan hadirnya *smartphone*. Ponsel pada saat ini tidak hanya berfungsi sebagai alat komunikasi telepon dan sms, namun pada saat ini ponsel lebih dikembangkan dengan berbagai aplikasi yang menarik. *Smartphone* dengan sistem operasi *android* menawarkan berbagai fitur layanan aplikasi yang sangat banyak di *Google Play Store*. Aplikasi yang terdapat pada *playstore* memungkinkan pengguna untuk menginstall berbagai aplikasi yang dibutuhkan pengguna. Fitur-fitur yang disediakan menampilkan beberapa aplikasi untuk memenuhi kebutuhan dan daya tarik tersendiri bagi para penggunanya.

Kamus merupakan sebuah media yang dapat diartikan sebagai buku yang berisikan tentang arti suatu kata dari bahasa atau istilah asing. Misalkan untuk istilah nama ilmiah makhluk hidup, hal ini sangat jelas bahwa kamus tersebut berisikan tentang arti kata dari istilah nama ilmiah makhluk hidup. Dan keberadaan kamus sekarang sangatlah diperlukan sebagai media yang dapat menambah pengetahuan dan wawasan. Pada saat ini masih mendominasi penggunaan kamus yang berbentuk buku untuk memperoleh informasi khususnya tentang istilah nama ilmiah. Hal ini menjadikan masyarakat kurang tertarik ketika mencari istilah tentang nama ilmiah makhluk hidup, Sehingga memberikan efek

malas untuk mencari pengetahuan dan wawasan tentang nama ilmiah. Sekarang ini banyak sekali kamus yang beredar, mulai dari bentuk buku, kamus elektronik, aplikasi kamus berbasis *dekstop* dan *web*.

Mahkluk hidup yang ada di bumi terdiri dari manusia, hewan dan tumbuhan. Manusia memiliki karakteristik berbeda manusia sendiri bisa bernafas, bergerak, peka terhadap rangsangan dan memerlukan makanan untuk bertahan hidup, begitu juga dengan hewan dan tumbuhan memiliki karakteristik yang sama karena mereka termasuk kedalam mahkluk hidup. Hewan dan tumbuhan memiliki nama ilmiah, nama ilmiah ini menggunakan bahasa latin, mengapa harus bahasa latin karena bahasa tersebut sekarang sudah tidak digunakan lagi. Bahasa yang sudah tidak digunakan maka bahasa tersebut tidak mengalami perubahan kata, sehingga para ilmuwan yang ingin meneliti hewan dan tumbuhan menggunakan bahasa latin ini untuk mengidentifikasi atau mengenalinya. Maka dari itu untuk mempermudah pembelajaran dalam mengenal nama ilmiah hewan dan tumbuhan kamus ini dibuat.

Menurut Nugraha dkk (2015:2) “Aplikasi adalah penggunaan dalam suatu komputer, instruksi atau pernyataan yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*. Media pembelajaran adalah segala sesuatu yang dapat menyampaikan dan menyalurkan pesan dari sumber secara terencana sehingga tercipta lingkungan belajar yang kondusif dimana penerimanya dapat melakukan proses belajar secara efektif dan efisien.”

Dari kutipan jurnal diatas penulis akan menggunakan metode sekuensial karena metode ini cocok untuk pengembangan perangkat lunak. Metode sekuensial termasuk ke dalam metode yang paling simpel dan sederhana. Metode sekuensial ini banyak dipakai oleh peneliti karena memang sangat mudah

digunakan, sehingga dapat menghemat waktu dan biaya untuk melakukan penelitian karya ilmiah. Maka dari itu penulis menggunakan metode sekuensial ini untuk bisa menghemat waktu dan biaya, itu salah satu dari kelebihan dari metode ini. Selain metode penelitian penulis juga menggunakan algoritma *sequential searching* algoritma ini juga sangat simpel dan sederhana sama seperti metode penelitian sekuensial. *Sequential searching* adalah algoritma pencarian data dari sekumpulan data dimana data dicari dari awal sampai akhir atau atas sampai bawah tanpa harus mengurutkan data terlebih dahulu. Kekurangan dari *sequential searching* jika data yang dicari paling akhir maka akan membutuhkan waktu yang relatif lama begitu juga sebaliknya.

Oleh karena itu penulis mencoba untuk membuat sebuah aplikasi sebagai sarana untuk menambah pengetahuan dan wawasan terhadap masyarakat tentang kamus nama ilmiah makhluk hidup *android mobile*. Aplikasi ini berisi tentang istilah – istilah nama ilmiah makhluk hidup yang disajikan dengan desain yang bagus untuk menarik masyarakat. Masyarakat tidak kesulitan apabila ingin mencari pengetahuan dan wawasan tentang nama ilmiah makhluk hidup karena dengan aplikasi ini masyarakat lebih mudah dan lebih efisien untuk mencarinya. Dikarenakan satu aplikasi bisa digunakan banyak orang. Di dalam aplikasi ini terdapat dua pilihan untuk mencari nama-nama ilmiah makhluk hidup yaitu hewan dan tumbuhan. Salah satunya adalah nama ilmiah hewan, dimana pada kategori hewan ini adalah untuk menngartikan nama ilmiah hewan saja, demikian juga dengan tumbuhan. Dengan begitu penulis ingin menyampaikan karya ilmiah ini dengan judul “**Aplikasi Kamus Nama Ilmiah Tumbuhan Dan Hewan Berbasis Android Dengan Metode Sequential Searching**”.

1.2 Perumusan Masalah

Masalah akan dirumuskan sebagai berikut:

- 1) Mengidentifikasi kebutuhan apa saja untuk membuat aplikasi kamus berbasis *android*.
- 2) Studi literasi diperlukan untuk melakukan kategori yang diteliti dalam membuat kasus nama ilmiah hewan dan tumbuhan.
- 3) Pengumpulan data atas objek penelitian yaitu nama ilmiah hewan dan tumbuhan.
- 4) Simulasi dan Hasil Simulasi dibutuhkan untuk membuat aplikasi kamus nama ilmiah hewan dan tumbuhan.
- 5) Realisasi dan evaluasi dibutuhkan untuk menuangkan metode *sequential searching*.

1.3 Maksud dan Tujuan

Maksud dari penulisan skripsi ini adalah:

- a. Bagaimana mengidentifikasi kebutuhan untuk membuat aplikasi ini.
- b. Studi literasi dibutuhkan untuk membuat membuat aplikasi sehingga dapat membuat aplikasi sesuai keinginan.
- c. Pengumpulan data ini dilakukan melalui buku, jurnal, dan internet.
- d. Simulasi meliputi pembuatan *design*, *database*, model data, dan pengujian aplikasi ke dalam *emulator*. Hasil simulasi berupa *output* dari suatu kata melalui proses *input*.
- e. Evaluasi dilakukan untuk apakah aplikasi tersebut berjalan dengan lancar.

Tujuan penulisan skripsi ini adalah untuk melengkapi satu syarat kelulusan Program Strata 1 (S1) Program Studi Teknik Informatika pada Sekolah Tinggi

Manajemen Informatika dan Komputer Nusa Mandiri(STIMIK Nusa Mandiri)
Jakarta Periode II tahun 2016.

1.4 Metode Penelitian

Dalam penulisan skripsi ini penulis menggunakan metode sebagai berikut:

1.4.1 Teknik Pengumpulan Data

Untuk mendapatkan data dan informasi yang diperlukan, penulisan menggunakan metode:

1. Pengamatan Langsung

Dalam membuat aplikasi ini, metode penelitian yang dilakukan penulis berupa pengamatan interaksi langsung dengan pengguna baik anak-anak, orang tua dan guru.

2. Studi Pustaka

Penulis melakukan pengumpulan data dan informasi dengan cara membaca buku-buku sebagai referensi yang dapat dijadikan acuan dalam penulisan ini.

1.4.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan penulis adalah metode pengembangan sistem *waterfall*. Berikut tahapan:

a) Mengidentifikasi Masalah

Identifikasi Masalah merupakan awal dari perancangan aplikasi ini dan untuk mengetahui masalah tersebut.

b) Analisa Kebutuhan

Pada tahap ini pengamatan tentang jurnal, dan hasil pengamatan dijadikan bahan pertimbangan untuk membuat perancangan.

c) Desain

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan.

d) Implementasi

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*. Setiap *unit* dikembangkan dan diuji untuk fungsionalitas.

e) Pengujian

Seluruh *unit* yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing *unit*.

1.5 Ruang Lingkup

Dalam penulisan skripsi ini, penulis membatasi ruang lingkup permasalahan aplikasi nama ilmiah hewan dan tumbuhan yaitu nama ilmiah hewan dan nama ilmiah tumbuhan. Dalam aplikasi terdapat 50 nama ilmiah baik hewan maupun tumbuhan. Periode II 2016 STMIK Nusa Mandiri Jakarta.

BAB II

LANDASAN TEORI

2.1. Tinjauan Jurnal

Menurut Parno,dkk(2011:121), “Kebutuhan masyarakat terhadap layanan teknologi berbasis IT sangat bervariasi, salah satu kebutuhan adalah akan ketersediaan kamus dengan berbagai kepentingan mulai dari kamus yang bersifat umum hingga kamus istilah-istilah khusus seperti kamus politik, kamus ekonomi, kamus psikologi dan sebagainya.”

Menurut Damastuti Natalia dan Aulia Siti Aisjah (2015:36) “Prinsip kerja dari algoritma ini adalah dengan terlebih dahulu melakukan pencarian lokasi kosong pada dermaga, yang kemudian dipisahkan antara lokasi kosong dengan yang sudah terisi. Kemudian dilakukan looping dari data awal hingga akhir yang diinginkan dan disimpan dalam suatu database.”

2.2. Konsep dasar Pemrograman

Menurut Yuswanto (2008:7), “Konsep pemrograman terstruktur merupakan peranan terpenting dalam merancang, menyusun, memelihara dan mengembangkan suatu program, khususnya program aplikasi yang besar dan kompleks.” Program merupakan kata, ekspresi, pernyataan atau kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur, berupa urutan langkah untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan

bahasa pemrograman sehingga dapat dieksekusi oleh komputer. Pemrograman merupakan proses mengimplementasikan urutan langkah untuk menyelesaikan suatu bahasa pemrograman. Bahasa pemrograman merupakan prosedur atau tata cara penulisan program. Pada bahasa pemrograman terdapat dua faktor penting, yaitu sintak dan semantik. Sintak adalah aturan-aturan gramatikal yang mengatur tata cara penulisan kata, ekspresi dan pernyataan.

Sistem adalah sekumpulan komponen yang mengimplementasikan model dan fungsionalitas yang dibutuhkan. Komponen-komponen tersebut saling berinteraksi di dalam sistem guna mentransformasi *input* yang diberikan kepada sistem tersebut menjadi *output* yang berguna dan bernilai bagi pengguna. Hal lain yang penting dari sebuah sistem adalah batasan dari sistem itu sendiri (*system boundary*). Batasan sistem itulah yang akhirnya menentukan kompleksitas sebuah sistem dan pemodelannya. Sebuah sistem harus memiliki batasan yang jelas dan apabila sistem tidak memiliki batasan yang jelas maka itu bukan sistem.

2.2.1. Android

Menurut Sadeli (2014:2) mengemukakan bahwa “*Android* merupakan sistem operasi berbasis *Linux* yang digunakan untuk perangkat *mobile* (*smartphone*) atau pun perangkat tablet (PDA).” Sifat *platform android* yang terbuka bagi para pengembang untuk mengembangkan aplikasi buatan sendiri membuat *android* menjadi sistem operasi *mobile* yang populer hingga saat ini. Selain itu masyarakat sekarang sudah terbiasa dengan *android* karena penggunaan yang mudah dan *user interface friendly* sehingga anak-anak dan orang dewasa mudah mengoperasikan.

2.2.2. Kamus

Menurut Kamus Besar Bahasa Indonesia “Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata. Ia berfungsi untuk membantu seseorang mengenal perkataan baru.” Selain menerangkan maksud kata, kamus juga mungkin mempunyai pedoman sebutan, asal-usul (etimologi) sesuatu perkataan dan juga contoh penggunaan bagi sesuatu perkataan. Kamus juga dapat berupa kamus bahasa, kamus istilah, kamus budaya, kamus nama ilmiah salah satu yang akan dibahas oleh penulis, kamus ini bukan buku melainkan kamus elektronik.

2.2.3. Sejarah *Android*

Menurut Hermawan (2011:2), “Pada awalnya sistem aplikasi *android* dikembangkan oleh *Android Inc.* Yang kemudian dibeli *Google* pada tahun 2015. Dalam usaha mengembangkan *android*, pada tahun 2007 dibentuklah *Open Handset Alliance* (OHA), sebuah kongsortium dari beberapa perusahaan, yaitu *Texas Instruments*, *Broadcom Corporation*, *Google*, *HTC*, *Intel*, *LG*, *Marvell Technology Group*, *Motorola*, *Nvidia*, *Qualcomm*, *Samsung Electronics*, *Sprint Nextel*, dan *T-Mobile* dengan tujuan untuk mengembangkan standar terbuka untuk perangkat *mobile*. Pada tanggal 9 Desember 2008, diumumkan bahwa 14 anggota akan bergabung proyek *android*, termasuk *Packet Video*, *ARM Holdings*, *Atheros Communication*, *Asustek Computer Inc*, *Garmin Ltd*, *Softbank*, *Sony Ericsson*, *Toshiba Corp*, dan *Vodafone Group Plc.*” Di bawah ini merupakan paparan dari beberapa keunggulan *android* yang mungkin tidak kita ketahui sebelumnya adalah:

a. *Open Source*

Android merupakan sistem operasi berbasis *Linux* yang berarti menyediakan *platform* terbuka jadi siapa saja bisa menjadi pengembang.

b. *Magnet Google*

Android merupakan salah satu aset yang dimiliki *Google*. Jadi semua yang berhubungan dengan *Google* akan otomatis tersinkronisasi dengan ponsel *android* seperti: *Youtube*, *Gmail*, *Google Maps*, *Google Talk* dan masih banyak lagi.

c. *Multitasking*

Ponsel *android* mampu mengerjakan tugas secara bersamaan tanpa harus menunggu aplikasi yang lain dihentikan, itulah yang dinamakan multitasking. Misalnya dapat melakukan *browsing* serta mendengarkan lagu dalam waktu yang bersamaan.

d. Dukungan dari *Vendor* Papan Atas

Kecanggihan *android* mampu mengambil hati para *vendor* papan atas sehingga mampu melonjakan popularitas serta kemampuan *android*. Hadirnya *android* membawa berkah yang besar bagi para *vendor* karena dengan *android* penjualan produk mereka menjadi bertambah.

e. *Easy Notification*

Android memberikan kemudahan untuk mengatur pemberitahuan, mulai dari terima sms, terima *email*, terima pesan *facebook* dan masih banyak lagi dengan nada yang berbeda-beda.

f. *Beautiful widget*

Dalam *homescreen* ponsel *android* dapat ditambahkan *widget* yang bermacam-macam seperti halnya dalam sebuah komputer. Dalam ponsel *android* dapat diberikan menu *shortcut* sebuah aplikasi. Dan *shortcut* bisa dimodifikasi sesuai keinginan pengguna.

g. Modding

Sistem operasi terbuka benar-benar dimanfaatkan untuk orang-orang yang gemar mengutak-atik atau modifikasi ponsel. Hal itu terbukti dengan adanya modifikasi dari versi ke versi.

Tabel II.1 Versi *Android*

Versi <i>Android</i>	<i>API Level</i>	Diluncurkan	Nama <i>Android</i>
1	1	23 September 2008	<i>Astro</i>
1.1	2	9 Februari 2009	<i>Bender</i>
1.5	3	30 April 2009	<i>Cupcake</i>
1.6	4	15 September 2009	<i>Donut</i>
2.0 - 2.1	5 – 7	26 Oktober 2009	<i>Eclair</i>
2.2	8	20 Mei 2010	<i>Froyo</i>
2.3	9 – 10	6 Desember 2010	<i>Gingerbread</i>
3.0 - 3.2	11 – 13	22 Februari 2011	<i>Honeycomb</i>
4.0	14 – 15	19 Oktober 2011	<i>Ice Cream Sandwich</i>
4.1 - 4.3	16 – 18	27 Juni 2012	<i>Jelly Bean</i>
4.4 - 4.4W	19 – 20	14 Oktober 2014	<i>Kitkat</i>
5.0 - 5.1	21 – 22	17 Oktober 2014	<i>Lollipop</i>
6.0	23	5 Oktober 2015	<i>Marshmallow</i>
7.0 - 7.1	24 – 25	23 Agustus 2016	<i>Nougat</i>

Sumber : <http://techijau.com/urutan-versi-nama-android/>

2.2.4. *Arsitektur Android*

Menurut Perbanga (2015), secara garis besar *Arsitektur Android* dapat dijelaskan dan digambarkan sebagai berikut:

1. *Application dan Widgets*

Application dan *Widgets* ini adalah *layer* yang berhubungan dengan aplikasi saja, biasanya dilakukan *download* kemudian di-*install* dan aplikasi tersebut dijalankan. Di-*layer* terdapat aplikasi inti termasuk klien *email*, program SMS, kalender, peta, *browser*, kontak, dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman *Java*.

2. *Applications dan 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 menambahkan *status notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju *API framework* seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur dirancang agar dapat dengan mudah menggunakan kembali komponen yang sudah digunakan (*reuse*). Jadi, *Applications Frameworks* ini adalah *layer* dimana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi *Android*, karena pada layer ini lah aplikasi dapat dirancang dan dibuat, seperti *content-providers* yang berupa sms dan panggilan telpon.

Komponen-komponen yang termasuk didalam *Applications Frameworks* adalah sebagai berikut:

- A. *Views* Digunakan untuk mengambil sekumpulan *button*, *list*, *grid*, dan *text box* yang digunakan didalam antarmuka pengguna.

- B. *Content Provider* Digunakan untuk memungkinkan aplikasi mengakses data dari aplikasi lain (seperti *contacts*) atau untuk membagikan data mereka sendiri.
- C. *Resource Manager* Digunakan untuk mengakses sumber daya yang bersifat bukan kode seperti *string lokal*, *bitmap*, deskripsi dari *layout file* dan bagian eksternal lain dari aplikasi.
- D. *Notification Manager* Digunakan untuk mengatur tampilan peringatan dan fungsi-fungsi lain.
- E. *Activity Manager* Mengatur siklus dari aplikasi dan menyediakan navigasi *backstack* untuk aplikasi yang berjalan pada proses yang berbeda.
- F. *Package Manager* Untuk melacak aplikasi yang di-*instal* pada perangkat.
- G. *Telephony Manager* Berisi sekumpulan *API* yang diperlukan untuk memanggil aplikasi.

3. *Libraries*

Android memiliki sekumpulan *library C/C++* yang digunakan oleh berbagai komponen dalam sistem *Android*. Kemampuan-kemampuan ini dilihat oleh para pengembang melalui kerangka kerja aplikasi. Beberapa dari *library* utama dijelaskan sebagai berikut :

- a. *System C Library*
- b. *Media Libraries*
- c. *Surface Manager*
- d. *LibWebCore*
- e. *Scalable Graphics Library*
- f. *3D Libraries*

g. *Free Type Library*

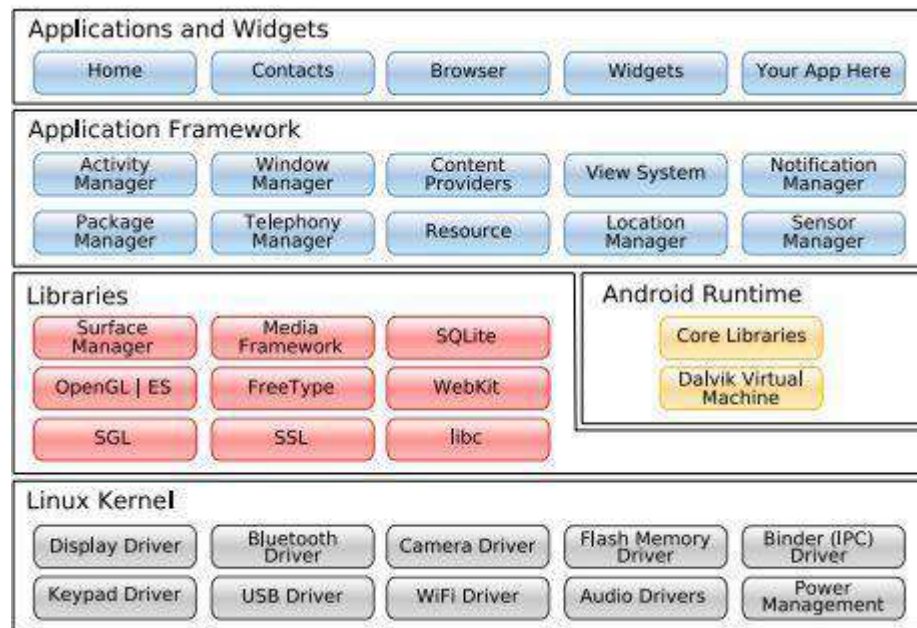
h. *SQLite*

4. *Android Runtime*

Merupakan lokasi dimana komponen utama dari *DVM* ditempatkan. *DVM* dirancang secara khusus untuk *Android* pada saat dijalankan pada lingkungan yang terbatas, dimana baterai yang terbatas, *CPU*, memori dan penyimpanan data menjadi fokus utama. *Android* memiliki sebuah *tool* yang terintegrasi yaitu “*dx*” yang mengkonversi *generated byte code* dari (*JAR*) ke dalam file (*DEX*) sehingga *byte code* menjadi lebih efisien untuk dijalankan pada prosesor yang kecil. Hal ini memungkinkan untuk memiliki beberapa jenis dari *DVM* berjalan pada suatu peralatan tunggal pada waktu yang sama. *Core libraries* ditulis dalam bahasa *Java* dan berisi kumpulan *class*, *I/O* dan peralatan lain.

5. *Linux Kernel*

Arsitektur *Android* berdasarkan pada *Linux 2.6 kernel* yang dapat digunakan untuk mengatur keamanan, manajemen memori, manajemen proses, *network stack*, dan *driver model*. Kernel juga bertindak sebagai lapisan abstrak antara perangkat keras dan seluruh *software stack*.



(Sumber <http://www.perbangga.com/2015/11/arsitektur-andorid.html>)

Gambar II.1 Arsitektur Android

2.2.5. Java

Menurut Hermawan (2011:20), “Java merupakan bahasa pemrograman untuk membangun aplikasi pada sistem aplikasi *android*. Oleh karena itu, untuk membangun aplikasi pada sistem operasi ini diperlukan dasar tentang pemrograman *java*.” Java merupakan pemrograman berorientasi objek. Oleh karena itu, setiap konsep yang diimplementasikan dalam *java* berbentuk dalam kelas. Kelas ini mendefinisikan objek-objek yang memiliki kesamaan perilaku dan keadaan. Pada *java* terdapat kumpulan kelas standar yang dikenal dengan *Application Programing Interface (API) Java*, selain itu dapat juga dideskripsikan kelas sendiri sesuai kebutuhan. Pemograman *java* tidak hanya untuk membuat aplikasi *mobile* saja tapi juga bisa *dekstop*, *web* dan lainnya. Maka dari itu banyak orang yang mempelajari bahasa pemograman *java* karena memang pemograman *java* ini cocok untuk membuat suatu aplikasi.

2.2.6. *Android Studio*

Android Studio adalah sebuah *IDE* untuk *Android Development* yang diperkenalkan *google* pada acara *Google I/O 2013*. *Android Studio* merupakan pengembangan dari *Eclipse IDE*, dan dibuat berdasarkan *IDE Java* populer, yaitu *IntelliJ IDEA*. *Android Studio* merupakan *IDE* resmi untuk pengembangan aplikasi *Android*. Sebagai pengembangan dari *Eclipse*, *Android Studio* mempunyai banyak fitur-fitur baru dibandingkan dengan *Eclipse IDE*. Berbeda dengan *Eclipse* yang menggunakan *Ant*, *Android Studio* menggunakan *Gradle* sebagai *build environment*. Fitur-fitur lainnya adalah sebagai berikut :

1. Menggunakan *Gradle-based build system* yang fleksibel.
2. Bisa mem-build *multiple APK*.
3. *Template support* untuk *Google Services* dan berbagai macam tipe perangkat.
4. *Layout editor* yang lebih bagus.
5. *Built-in support* untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*.
6. *Import library* langsung dari *Maven repository*.

2.2.7. *AVD(Android Virtual Device)*

AVD merupakan *emulator* untuk menjalankan program aplikasi *android*. *AVD* ini nantinya yang akan kita jadikan sebagai tempat untuk melakukan tes atau *debugging* dan menjalankan aplikasi *android* yang kita buat. *Android Studio* sendiri sudah memiliki *AVD* sendiri dan *AVD* di *android studio* bukan hanya

menyediakan *AVD* untuk *smartphone* ada juga untuk perangkat TV dan Jam Tangan berbasis *android*.

2.2.8. *Android Software Development Kit(SDK)*

Android Software Development Kit(SDK) adalah *tools Application Programming Interface(API)* yang diperlukan untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman *java*. *SDK android* berisi *debugger, library, emulator*, dokumentasi contoh kode program dan tutorial. *Android* memberi kesempatan untuk membuat aplikasi yang dibutuhkan, namun bukan merupakan aplikasi bawaan *smartphone*.

2.2.9. *OOP (Object Oriented Program)*

Menurut Shalahudin (2013:99) menyebutkan bahwa “Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya”.

Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek dan pengujian berorientasi objek, hasil dari satu tahapan pengembangan ketahap berikutnya, keuntungan metode berorientasi objek sebagai berikut:

1. Meningkatkan produktifitas

Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut. Sehingga objek yang satu dengan yang lainnya bisa saling berhubungan.

2. Kecepatan pengembangan

Karena sistem yang dibangun dengan baik dan benar pada saat analisa dan perancangan akan menyebabkan kekurangan kesalahan pada saat pengkodean.

3. Kemudahan pemeliharaan

Karena dengan model-model, pola-pola yang cenderung tetap dan stabil dapat dipisahkan pola-pola yang mungkin sering berubah-ubah.

4. Adanya konsistensi kualitas perangkat lunak

Karena pendekatan pengembangan lebih dekat dengan dunia nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan. Setiap komponen dalam sistem tersebut dapat mewarisi atribut dan sifat dan komponen lainnya dapat berinteraksi satu sama lain. Berikut ini adalah beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek:

1. Kelas (*class*)

Kelas adalah kumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi statis dan himpunan objek yang sama yang mungkin lahir atau diciptakannya dan kelas tersebut. Sebuah kelas akan mempunyai sifat (*attribute*), kelakuan (*method*), hubungan (*relationship*) dan arti. Suatu kelas dapat diturunkan pada kelas yang lainnya, dimana atribut dari kelas semula dapat diwariskan kekelas yang baru.

2. Objek (*object*)

Objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, atau hal lain-lainnya yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan

mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya.

3. Metode (*method*)

Operasi atau metode pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metode struktural. Sebuah kelas boleh memiliki lebih dari satu metode operasi, metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri.

4. Atribut (*attribute*)

Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis, nama.

5. Absrtaksi (*abstraction*)

Prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek yang tidak sesuai permasalahan.

6. Enkapsulasi (*encapsulation*)

Pembungkusan atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya.

7. Pewarisan (*inheritance*)

Mekanisme yang memungkinkan suatu objek mewarisi sebagian atau seluruh definisi dan objek lainnya sebagian dari dirinya.

8. Polimorfism

Kemampuan suatu objek untuk digunakan sebanyak tujuan yang berbeda dengan nama yang sama sehingga menghemat baris program.

9. Antarmuka (*interface*)

Antarmuka atau *interface* sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode sebuah *interface* dapat diimplementasikan oleh kelas lain. Sebuah kelas mengimplementasikan lebih dari satu antarmuka dimana kelas ini akan mendeklarasikan metode pada antarmuka yang dibutuhkan oleh kelas itu sekaligus mendefinisikan isinya pada kode program kelas itu.

2.2.10 SQLite Database

Menurut Andi (2013:195) “*SQLiteDatabase* adalah *class* dasar untuk *database SQLite* di android. Disini pengguna bisa membuka *query,update*,dan menutup *database*. *SQLiteDatabase* punya banyak *method*,seperti *method insert()*,*update()*,dan *delete()*.Ada juga *execSQL()* yang bisa mengeksekusi *SQL* secara langsung.” Nilai *object ContentValues* memungkinkan mendefinisikan *key/value*.*Key* merepresentasikan identitas kolom tabel,seangkan *value* merepresentasikan konten dari *record* tabel di kolom.*ContentValues* bisa digunakan untuk *insert* dan *update entri database*.

2.3. Metode Algoritma

Dalam skripsi ini, penulis menggunakan metode algoritma *Sequential Searching* teknik ini adalah untuk mencari kata dalam array dari awal sampai

akhir, dimana data tidak perlu diurutkan terlebih dahulu. Algoritma pencarian dapat dituliskan sebagai berikut:

1. $i \leftarrow 0$
2. ditemukan $\leftarrow$ false
3. Selama (tidak ditemukan) dan $(i \leq N)$ kerjakan baris 4
4. Jika $(Data[i] = x)$ maka ditemukan $\leftarrow$ true, jika tidak $i \leftarrow i + 1$
5. Jika (ditemukan maka i adalah indeks dari data yang dicari, jika tidak data tidak ditemukan.

```
int SequentialSearch(int x)
```

```
{ int i = 0; bool
```

```
    ditemukan = false;
```

```
    while ((!ditemukan) && (i < Max))
```

```
    { if(Data[i] == x)
```

```
        Ditemukan = true;
```

```
    else
```

```
        i++; }
}
```

```
if(ditemukan)
```

```
    return i;
```

```
else
```

```
    return -1; }
```


2.4. Pengujian Software

Pengujian ini digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada dua metode untuk melakukan *unit testing* yaitu:

2.4.1. Pengujian White Box

Shalahuddin(2013:276) “Pengujian *White Box* yaitu menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan dan keluaran yang sesuai dengan spesifikasi kebutuhan. Pengujian kotak putih dilakukan dengan memeriksa logik dari kode program. Pembuatan kasis uji bisa mengikuti standar pengujian dari standar pemograman yang seharusnya. Contoh dari pengujian kotak putih misalkan menguji alur(dengan menelusuri) pengulangan(*looping*) pada logika pemrograman.”

2.4.2. Pengujian Black Box

Shalahuddin(2013:275) “Pengujian *Black Box* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan.” Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji yang dibuat adalah:

- 1) Jika user memasukan nama pemakai (*username*) dan kata sandi (*password*) yang benar
- 2) Jika user memasukan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau sebaliknya atau keduanya salah”.

2.5. *UML*

Menurut Nikko (2015) “*UML (Unified Modelling Language)* suatu metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek. *UML* merupakan suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem *software*. Saat ini *UML* sudah menjadi bahasa standar dalam penulisan *blue print software*.” Fungsi dari penggunaan *UML* diantaranya:

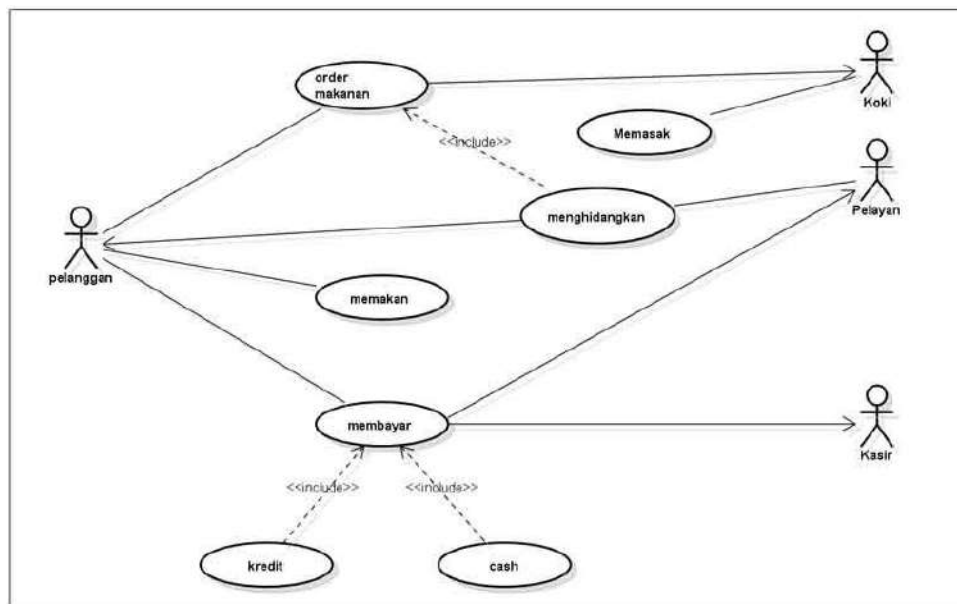
1. Dapat memberikan bahasa permodelan *visual* kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa. Dapat menyatukan praktek-praktek terbaik yang ada dalam permodelan.
2. Dapat memberikan model yang siap untuk digunakan, merupakan bahasa permodelan *visual* yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah.
3. Dapat berguna sebagai *blue print*, sebab sangat lengkap dan detail dalam perancangannya yang nantinya akan diketahui informasi yang *detail* mengenai koding suatu program.
4. Dapat memodelkan sistem yang berkonsep berorientasi objek, jadi tidak hanya digunakan untuk memodelkan perangkat lunak (*software*) saja.

5. Dapat menciptakan suatu bahasa permodelan yang nantinya dapat dipergunakan oleh manusia maupun oleh mesin.

2.5.1. Jenis-jenis diagram UML

a. Use case diagram

Use case diagram yaitu salah satu jenis diagram pada UML yang menggambarkan interaksi antara sistem dan aktor, *use case diagram* juga dapat mendeskripsikan tipe interaksi antara pemakai sistem dengan sistemnya.

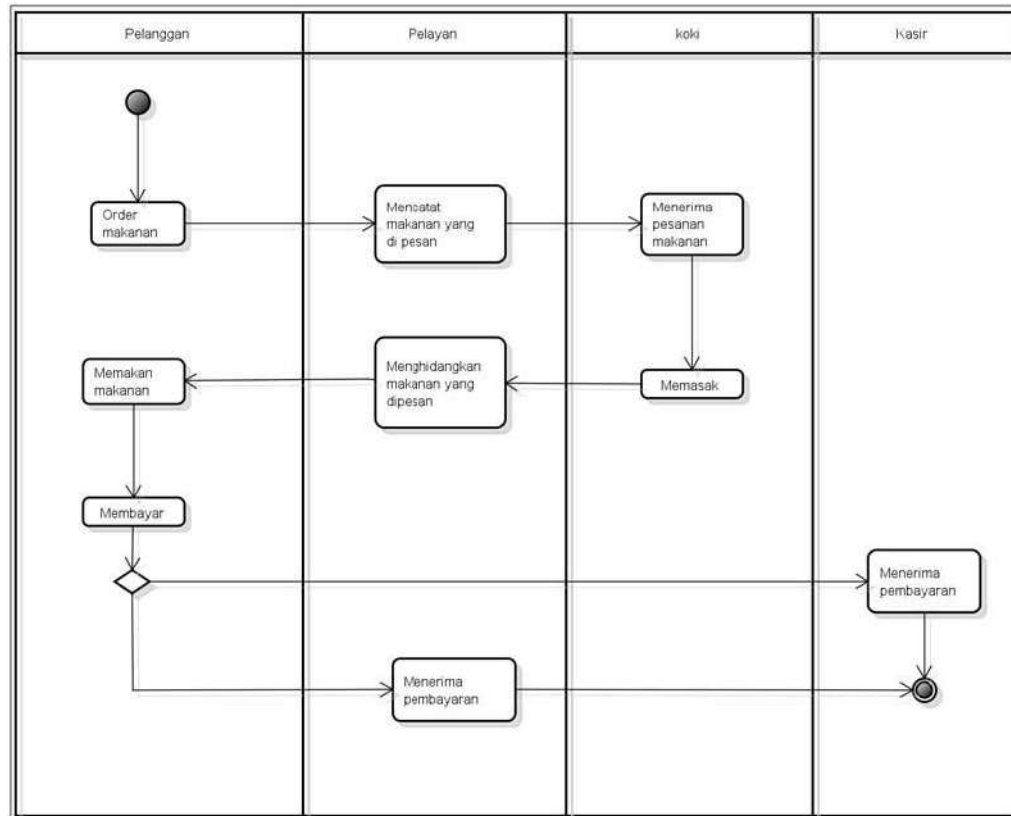


Sumber : Nikko (2015)

Gambar II.2 Contoh Use Case Diagram

b. Activity Diagram

Activity diagram yaitu salah satu jenis diagram pada UML yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.

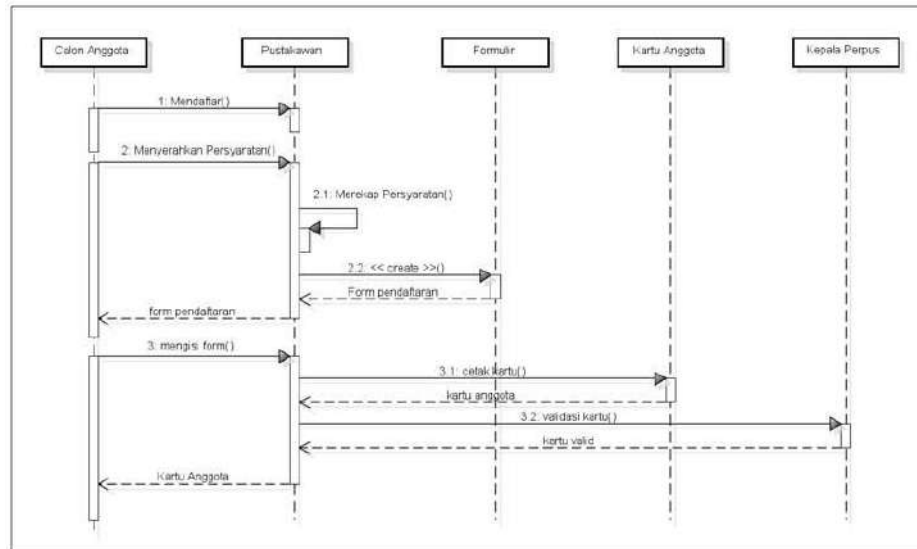


Sumber : Nikko (2015)

Gambar II.3 Contoh Activity Diagram

c. *Sequence diagram*

Sequence diagram yaitu salah satu jenis diagram pada *UML* yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence diagram* juga dapat menggambarkan urutan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada *use case diagram*.

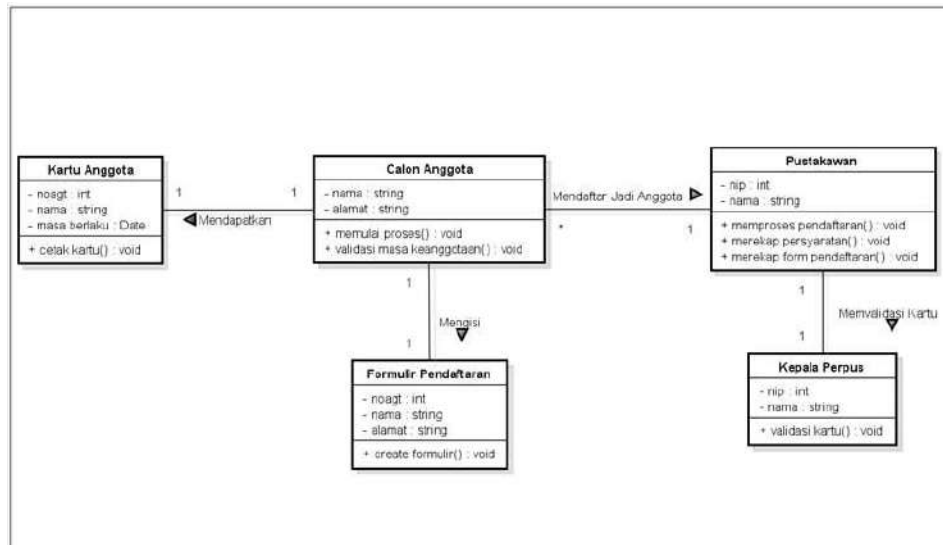


Sumber : Nikko (2015)

Gambar II.4 Contoh Sequence Diagram

d. *Class diagram*

Class diagram yaitu salah satu jenis diagram pada *UML* yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut.

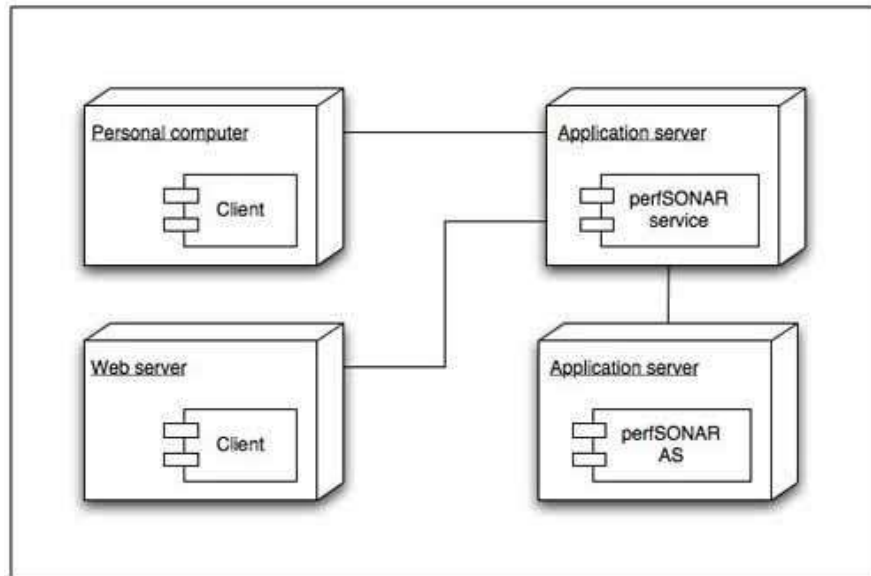


Sumber : Nikko (2015)

Gambar II.5 Contoh Class Diagram

e. *Deployment diagram*

Deployment diagram yaitu salah satu diagram pada *UML* yang menunjukkan tata letak suatu sistem secara fisik, dapat juga dikatakan untuk menampilkan bagian-bagian *software* yang terdapat pada *hardware* dan digunakan untuk menerapkan suatu sistem dan hubungan antara komponen *hardware*. Jadi *Deployment diagram* intinya untuk menunjukkan letak *software* pada *hardware* yang digunakan sistem.



Sumber : Nikko (2015)

Gambar II.6 Contoh *Deployment Diagram*

BAB III

ANALISA DAN RANCANGAN SOFTWARE

3.1. Analisa Kebutuhan *Software*

Dalam pembuatan aplikasi kamus nama ilmiah hewan dan tumbuhan berbasis *android*, untuk mempermudah masyarakat umum mengenal nama ilmiah hewan dan tumbuhan dibutuhkan analisis kebutuhan sistem yang meliputi:

a. Input

User membuka aplikasi selanjutnya aplikasi tersebut menampilkan layar atau halaman awal aplikasi.

b. Proses

User memilih menu didalam aplikasi seperti: nama ilmiah hewan dan tumbuhan.

c. Output

Menampilkan isi menu yang telah dipilih user.

3.1.1 Identifikasi Masalah

Pada penulisan Skripsi ini permasalahan yang akan diteliti dalam pembuatan aplikasi kamus nama ilmiah hewan dan tumbuhan adalah:

1. Mempermudah pengguna untuk mencari nama ilmiah.
2. Minimnya pengetahuan tentang nama ilmiah hewan dan tumbuhan.
3. Kurangnya efisien waktu untuk mencari buku tentang nama ilmiah.

3.1.2. Analisa Kebutuhan

Tahap analisa kebutuhan mencakup *hardware*, *software*, aplikasi dan *output* yang digunakan adalah sebagai berikut:

A. Komponen *Hardware*

Komputer yang digunakan penulis mempunyai klasifikasi sebagai berikut:

- 1) Vendor : *Sony VAIO*
- 2) Tipe : *VGN-TT26GG*
- 3) *Processor* : *Intel Core2Duo U9400 (Dual Core)*
- 4) *Display Adapter* : *Intel Graphics Media Accelerator(GMA)*
- 5) *RAM* : 3 GB
- 6) *Hardisk* : 128 GB

Dalam pembuatan aplikasi *android* minimal harus menggunakan *processor Dual Core* karena akan mempengaruhi terhadap pembuatan *virtual android* yang akan digunakan.

Perangkat *Gadget* atau *Smartphone* yang digunakan penulis mempunyai klasifikasi sebagai berikut:

- 1) Vendor : *Evercoss*
- 2) Tipe : *A74A*
- 3) *Processor* : *Quad Core 1.2 GHz*
- 4) *RAM* : 1 GB
- 5) *Memory Internal* : 8 GB
- 6) *Memory Eksternal*: 2 GB
- 7) *Resolution* : 4 Inch, 480 x 800 px

B. Komponen *Software*

Komponen perangkat lunak yang digunakan untuk membuat sistem tersebut sebagai berikut:

1) *Microsoft Windows 7*

Penulis menggunakan sistem operasi *Windows 7* sebagai sistem operasi dalam perancangan aplikasi kamus nama ilmiah hewan dan tumbuhan ini.

2) *Android Studio*

Android Studio adalah sebuah *IDE (Integrate Developmnet Environment)* yang digunakan untuk mengembangkan aplikasi mulai dari *android* versi pertama hingga versi terbaru. Jika pengembang ingin mengembangkan aplikasi *android* versi terbaru, maka *android studio* menjadi pilihan utama dikarenakan *android studio* mendapatkan dukungan langsung dari *google* selaku pemilik *android*.

3) *JDK (Java Developmnet Kit)*

JDK(Java Development Kit) adalah program *developmnet environment* untuk menulis *Java Applets* dan aplikasi. *JDK* terdiri dari *runtime environment* yang ada diatas *layer* sistem operasi serta *tool* dan program yang memerlukan *compile*, *debug*, dan *run applets* dan aplikasi yang ditulis menggunakan bahasa pemrograman *Java*. *Java Developmnet Kit* digunakan untuk *plugins* bahasa pemrograman *java*. *JDK* ini dibutuhkan karena *android* menggunakan bahasa pemograman *java*.

4) *Emulator*

Emulator adalah *software* yang dapat menjalankan aplikasi berbasis *android* di laptop yang menggunakan sistem operasi *windows*. *Software* ini

digunakan penulis untuk menjalankan aplikasi yang dibuat atau melakukan *testing aplikasi*.

5) *Adobe Photoshop*

Adobe Photoshop adalah *software* aplikasi desain yang digunakan untuk mendesain gambar, mengedit *image* grafis dan mengolah foto digital. *Software* ini berguna untuk mendesain tampilan gambar-gambar yang akan digunakan dalam program *android*.

6) *Enterprise Architect*

Enterprise Architect merupakan sebuah *platform* visualisasi untuk merancang dan membangun sebuah sistem perangkat lunak dan untuk permodelan proses bisnis. *Enterprise Architect* adalah sebuah alat yang sangat maju yang mencakup seluruh aspek mulai dari awal tahap mendesain sampai mengontrol perkembangan, pemeliharaan, pengujian dan perubahan kontrol. *Tools Enterprise Architect* dikembangkan oleh *Sparx Systems*, sebuah perusahaan Australia dengan inovasi dan pengembangan dalam pemodelan *UML*. *Sparx System* adalah anggota dari *Object Management Group(OMG)*.

3.2. Desain

Sebelum aplikasi *Android* ini di implementasikan dalam bentuk *package.apk*, maka perlu dirancang terlebih dahulu. Tahap perancangan bertujuan untuk memenuhi kebutuhan pengguna dan memberikan gambaran yang jelas mengenai aplikasi yang kita buat. Keseluruhan dari perancangan ini akan di implementasikan dalam *Gadget* berbasis *android* dengan menggunakan perangkat lunak *Android Studio*.

3.2.1 Rancangan Algoritma

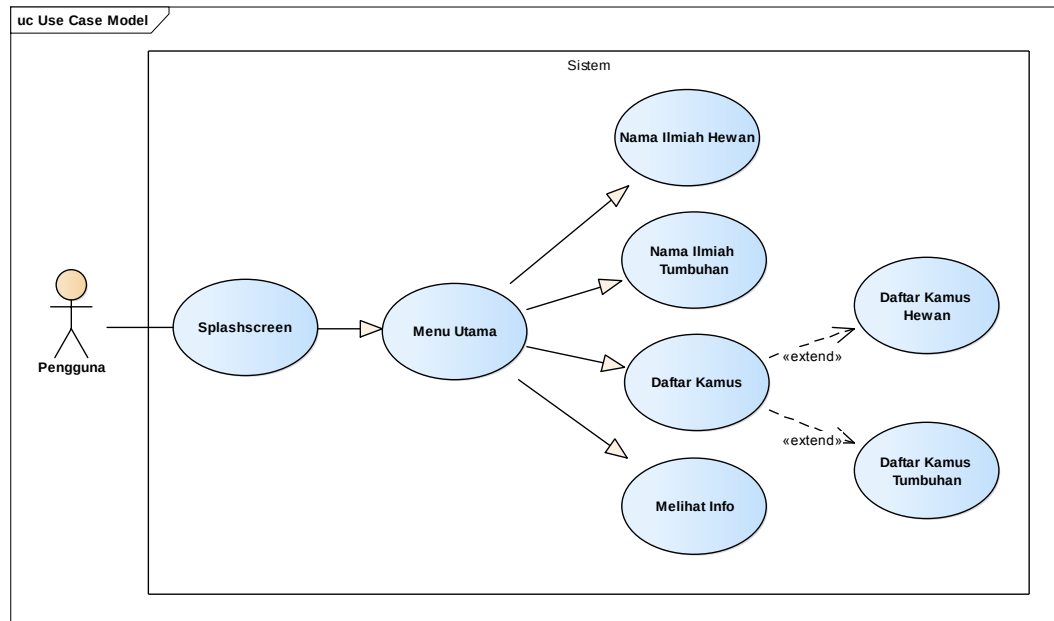
Untuk mengimplementasikan ke dalam program aplikasi dibutuhkan algoritma, yaitu langkah-langkah instruksi sehingga dicapai hasil yang diinginkan. Algoritma *sequential* merupakan sederetan pernyataan-pernyataan urutan dan pelaksanaan eksekusinya runtut, yang lebih dahulu ditemukan (dibaca) akan dikerjakan (dieksekusi)

3.2.2. Software Architecture

Rekayasa perangkat lunak merupakan pembangunan yang menggunakan prinsip dan konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin. Perangkat lunak banyak dibuat dan pada akhirnya sering tidak digunakan karena masalah-masalah non teknis.

1. Diagram Use Case

Use Case merupakan model diagram *UML* yang digunakan untuk menggambarkan *requirement* fungsional yang diharapkan dari sebuah sistem diagram *Use Case* menggunakan khususnya pada “siapa” melakukan “apa” dalam lingkungan perangkat lunak yang akan dibangun. Gambar menerangkan dimana pada menu awal aplikasi pertama kali pengguna akan menemui menu utama yang terdiri dari empat *button*: hewan, tumbuhan, daftar kamus dan info. Dan didalam menu daftar kamus terdapat 2 *button* hewan dan tumbuhan Adapun *use case diagram* dalam aplikasi ini sebagai berikut:



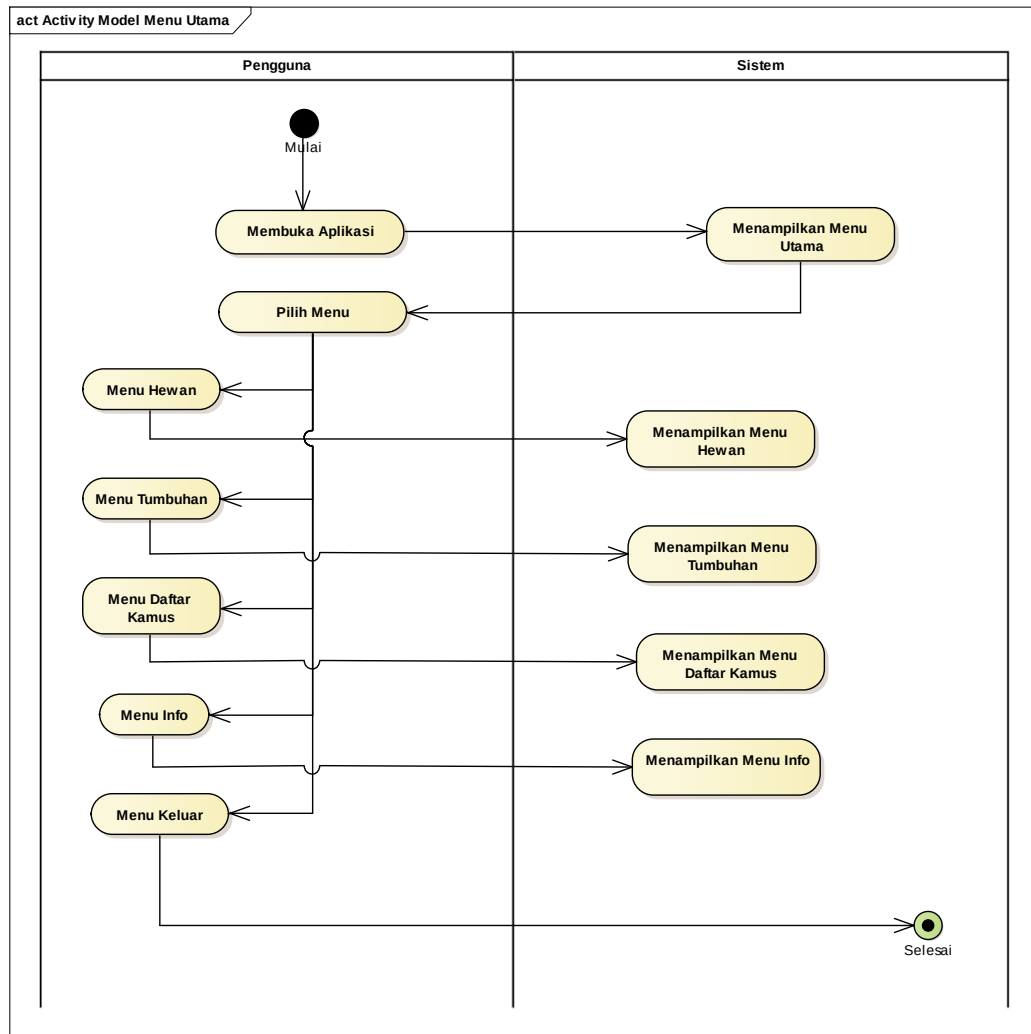
Sumber : Hasil Penelitian

Gambar III. 1 Diagram Use Case

2. Activity Diagram

Diagram *activity* menggambarkan berbagai aliran aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan akhir dari aktivitas. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Berikut merupakan beberapa diagram *activity* dari aplikasi yang dibuat:

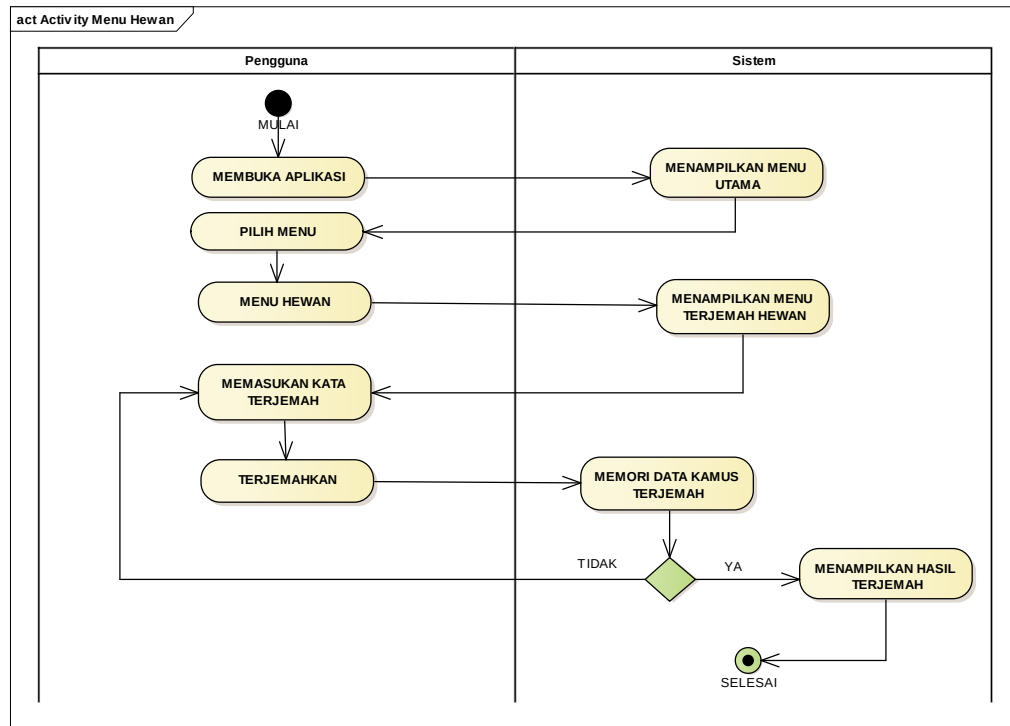
A) Activity Diagram Menu Utama



Sumber : Hasil Penelitian

Gambar III. 2 Activity Diagram Menu Utama

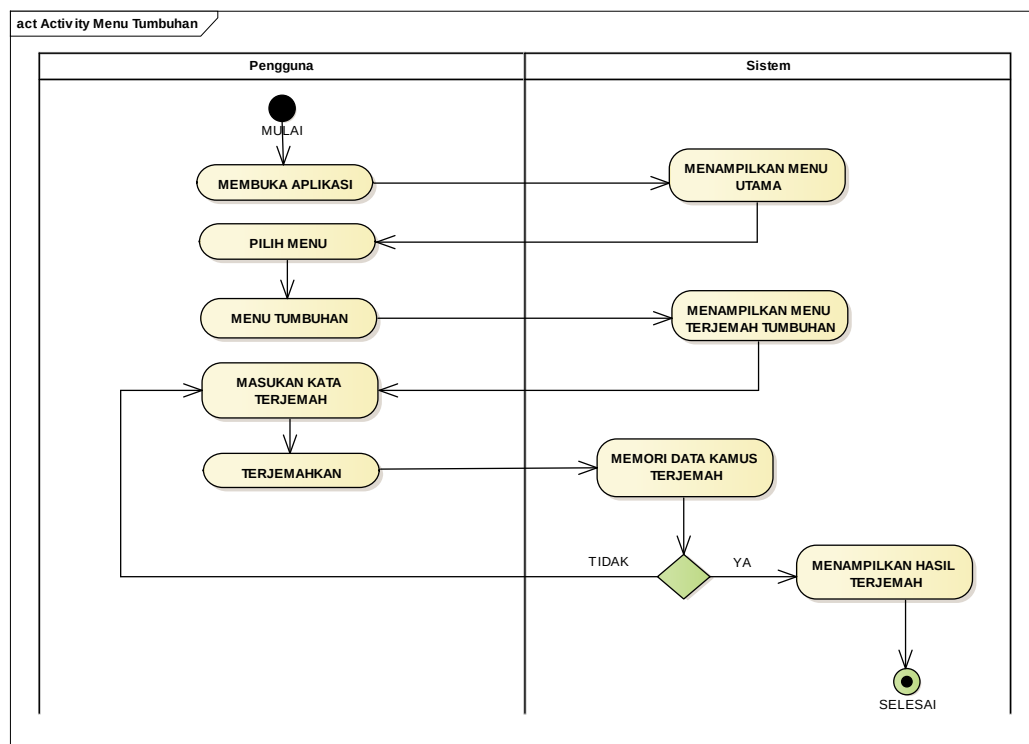
B) Activity diagram menu hewan



Sumber : Hasil Penelitian

Gambar III. 3 Activity Diagram Menu Hewan

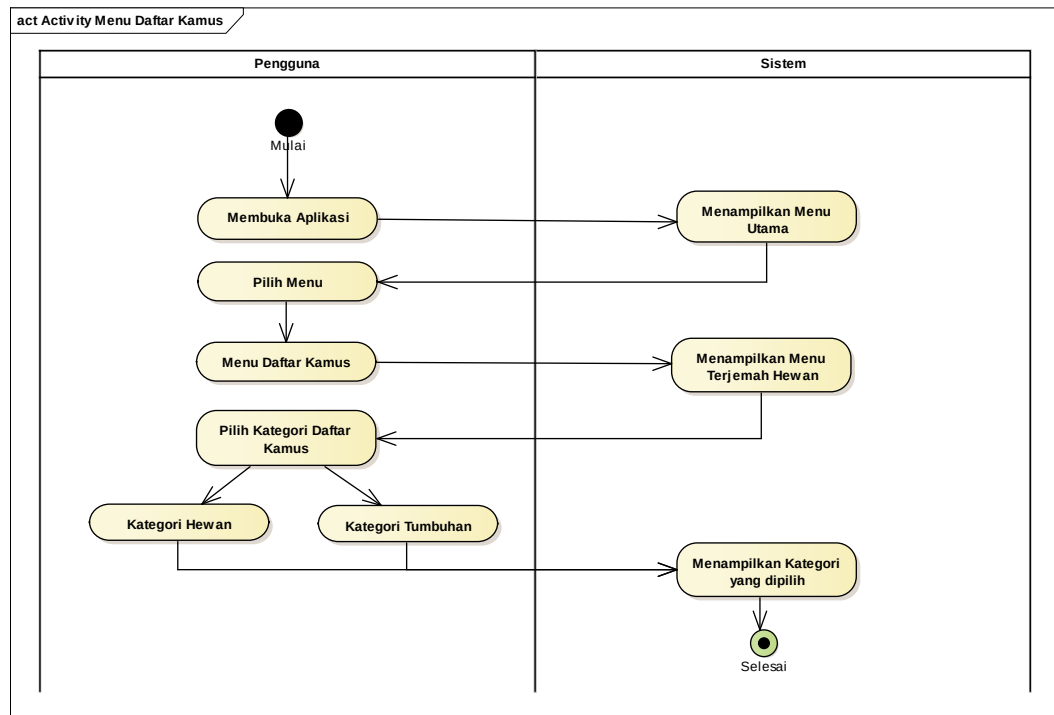
C) Activity Diagram Menu Tumbuhan



Sumber : Hasil Penelitian

Gambar III. 4 Activity Diagram Menu Tumbuhan

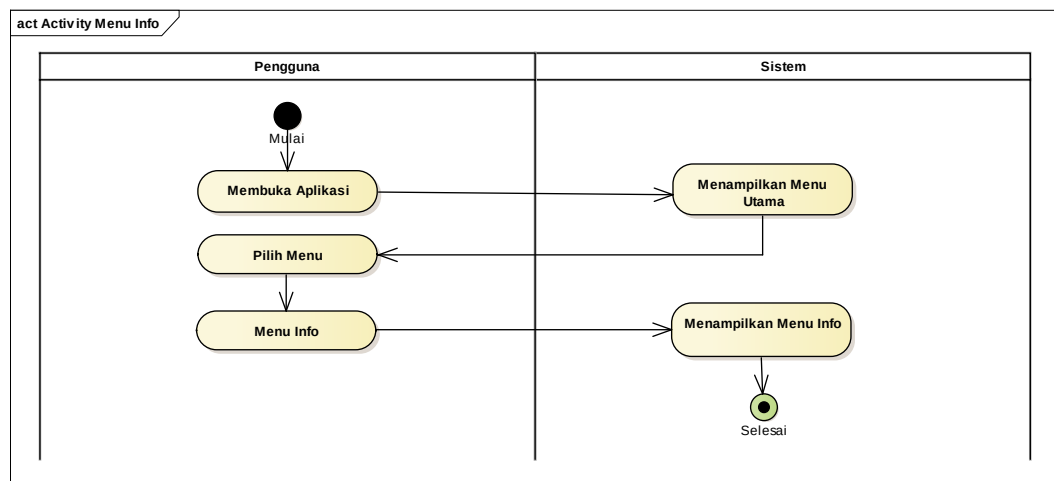
D) Activity Diagram Menu Daftar Kamus



Sumber : Hasil Penelitian

Gambar III. 5 Activity Diagram Menu Daftar Kamus

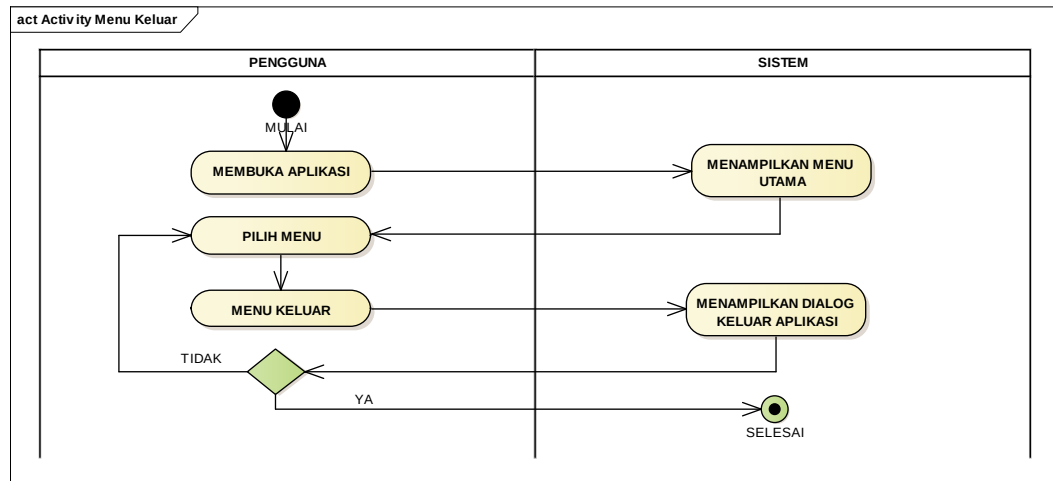
E) Activity Diagram Menu Info



Sumber : Hasil Penelitian

Gambar III. 6 Activity Diagram Menu Info

F) Activity Diagram Menu keluar



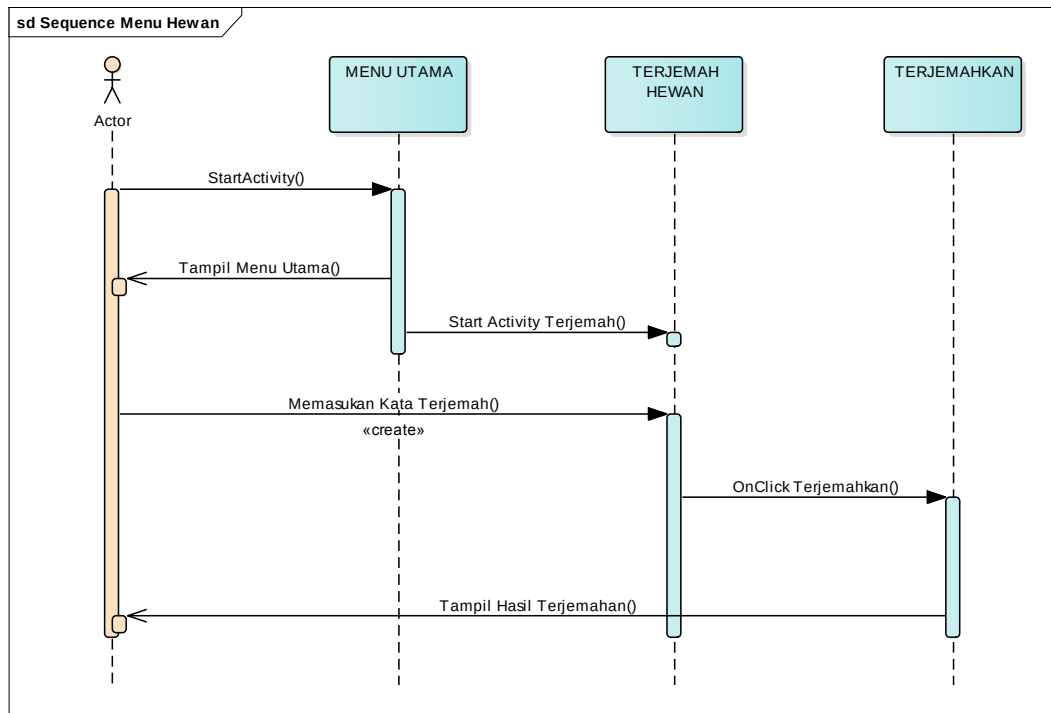
Sumber : Hasil Penelitian

Gambar III. 7 Activity Diagram Menu Keluar

3. Sequence Diagram

Sequence diagram dibutuhkan untuk melihat skenario yang ada pada *use case*. *Sequence* diagram juga menjelaskan diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek tersebut. Adapun *sequence diagram* pada aplikasi ini sebagai berikut:

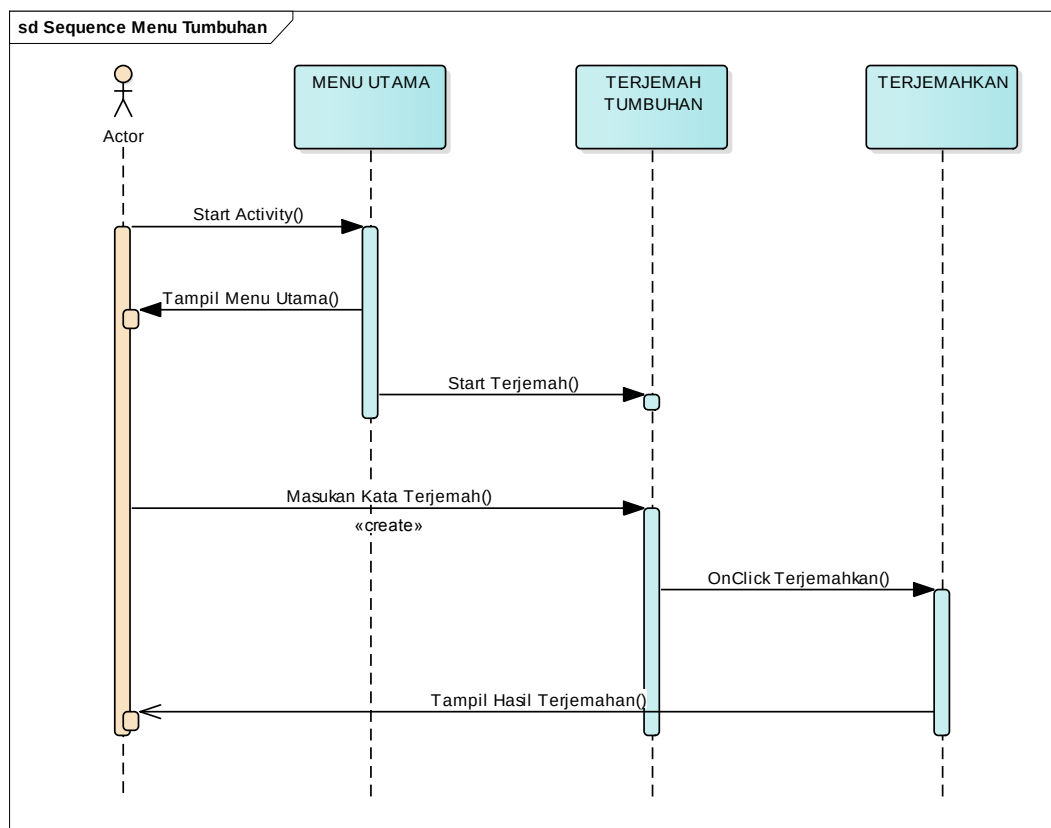
a) *Sequence Diagram Menu Hewan*



Sumber : Hasil Penelitian

Gambar III. 8 Sequence Diagram Menu Hewan

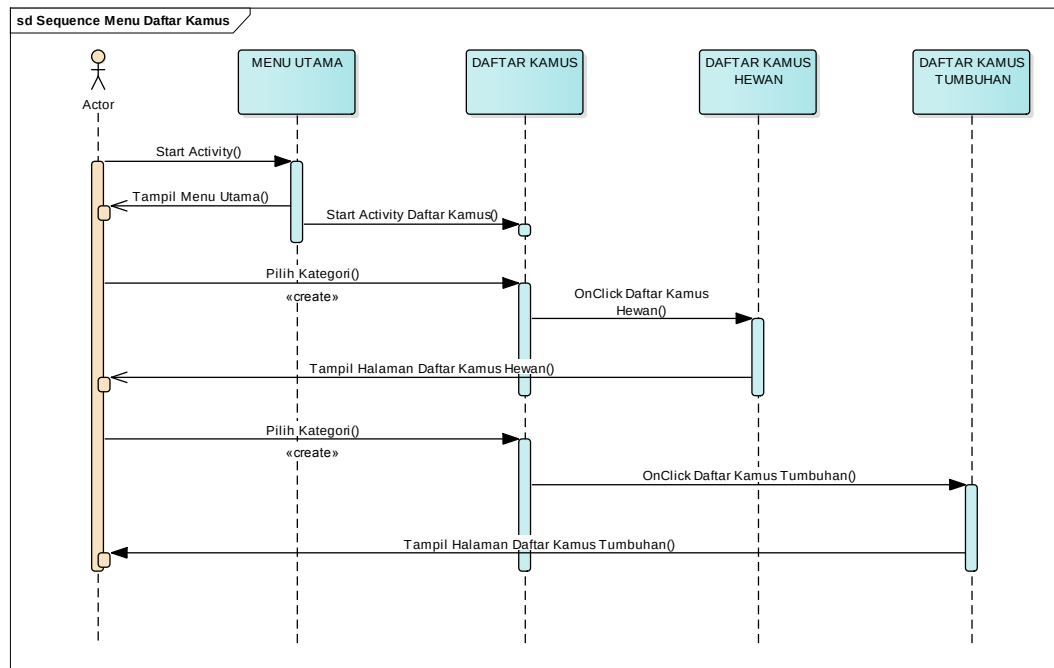
b) *Sequence Diagram Menu Tumbuhan*



Sumber : Hasil Penelitian

Gambar III.9 Sequence Diagram Menu Tumbuhan

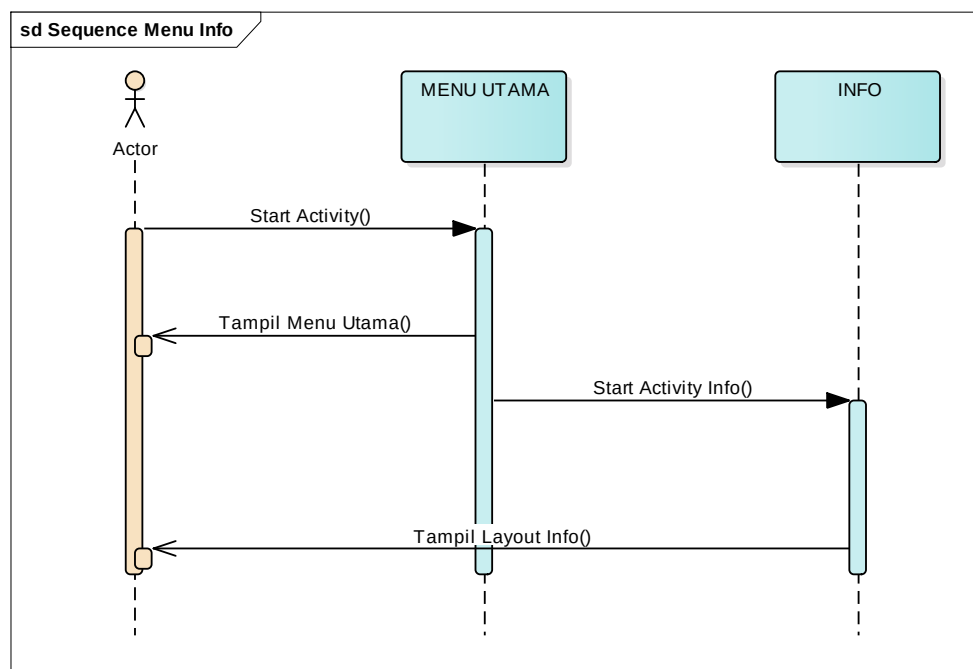
c) *Sequence Diagram Menu Daftar Kamus*



Sumber : Hasil Penelitian

Gambar III.10 Sequence Diagram Menu Daftar Kamus

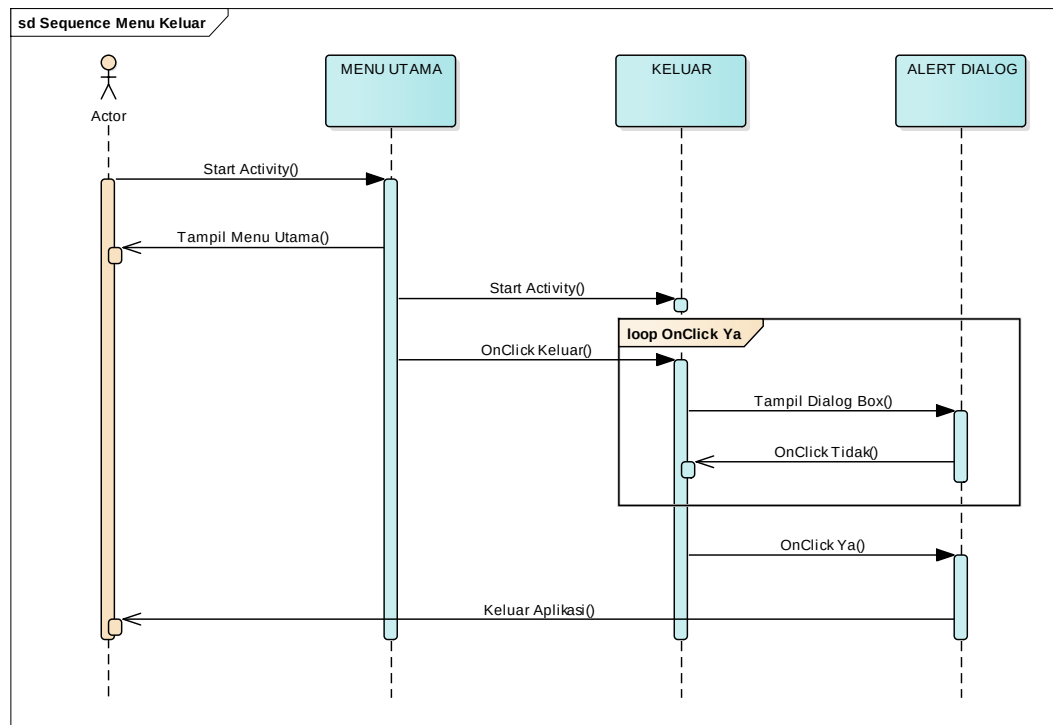
d) *Sequence Diagram Menu Info*



Sumber : Hasil Penelitian

Gambar III.11 Sequence Diagram Menu Info

e) *Sequence Diagram Menu Keluar*

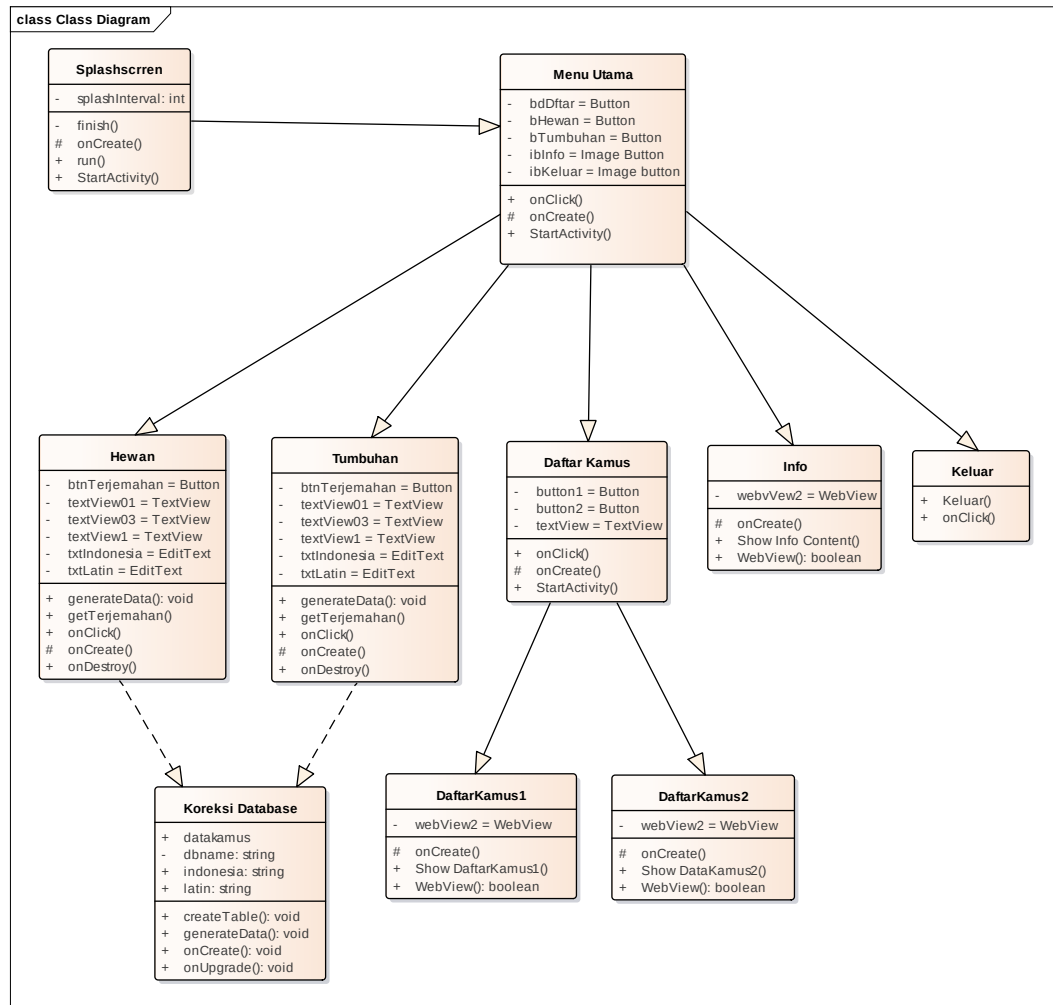


Sumber : Hasil Penelitian

Gambar III. 12 Sequence Diagram Menu Keluar

4. Class Diagram

Class Diagram merupakan diagram yang memperlihatkan atau menampilkan struktur dari sebuah sistem. Sistem tersebut akan menampilkan kelas, atribut, dan hubungan antar kelas ketika suatu sistem telah selesai membuat diagram. Adapaun *class diagram* pada aplikasi ini sebagai berikut:

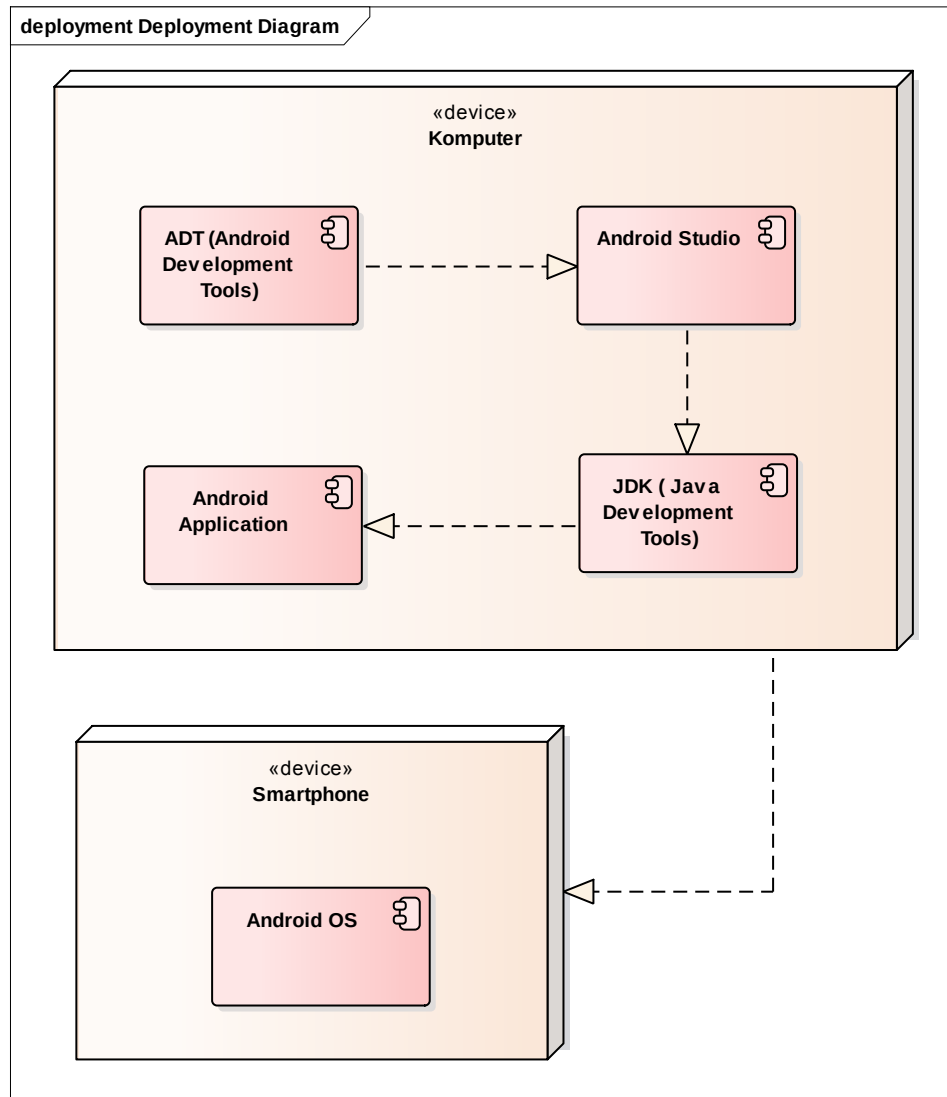


Sumber : Hasil Penelitian

Gambar III.13 Class Diagram

5. Deployment Diagram

Deployment Diagram merupakan gambaran proses-proses yang berbeda dari suatu sistem berjalan. Diagram ini menjelaskan tentang bagaimana relasi yang terdapat didalamnya. Adapun *Deployment* diagram dari aplikasi ini sebagai berikut:



Sumber : Hasil Penelitian

Gambar III.14 Deployment Diagram

3.2.3. User Interface

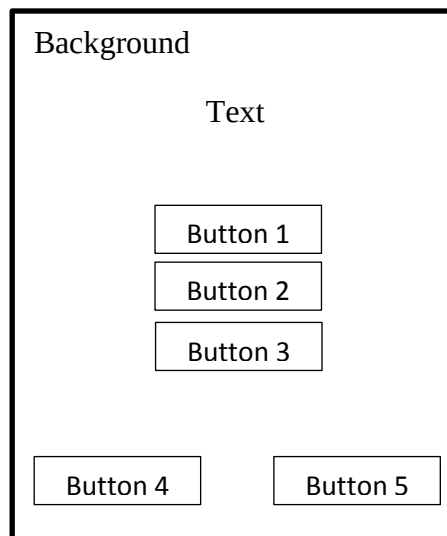
Antarmuka pengguna (*user interface*) merupakan mekanisme komunikasi antara pengguna (*user*). Tujuan utamanya adalah mengkomunikasikan fitur-fitur yang terdapat pada sistem agar *user* mengerti dan dapat menggunakan aplikasi dengan baik.

User Interface pada aplikasi diantaranya *Activity* dan *User Interface* yang terdiri dari komponen. Semuanya yang saling berhubungan dengan *user interface* pada aplikasi *android* yang terdapat pada lokasi *res layout menu.xml*. adapun *interface* yang digunakan dalam aplikasi ini adalah sebagai berikut:

1) Tampilan Menu Utama

Pada tampilan menu utama di desain sederhana yang didalamnya terdapat *background* dengan *format file .jpg* dan *.png* teks judul dengan tulisan “Kamus Nama Ilmiah Mahkluk Hidup” dan terdapat empat *button* yang terdiri dari Hewan, Tumbuhan, Info, dan Keluar.

i. Rancangan tampilan Menu Utama



Sumber : Hasil Penelitian

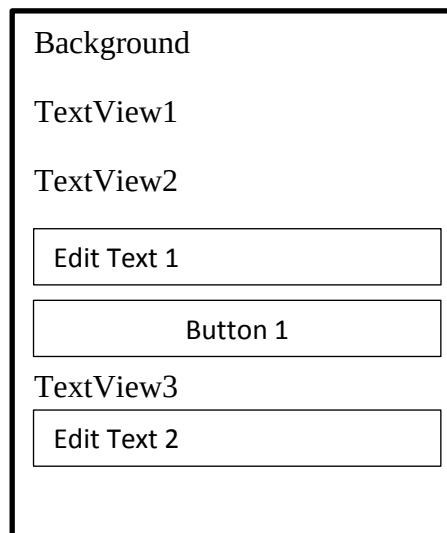
Gambar III.15 Rancangan *Interface* Menu Utama

Keterangan dari tampilan *interface* menu utama sebagai berikut:

- a. *Background* menggunakan gambar dengan nama *bg1.jpg* yang terletak di *folder res/drawable/bg1.jpg*.
- b. *Text* ini menggunakan *background* yang berbentuk seperti gambar yang berformat *.png*.

- c. *Button* 1 menggunakan gambar dengan nama *hewan1.jpg* yang terletak di *folder res/drawable/hewan1.jpg*.
- d. *Button* 2 menggunakan gambar dengan nama *tumbuhan1.jpg* yang terletak di *folder res/drawable/tumbuhan1.jpg*.
- e. *Button* 3 menggunakan gambar dengan nama *daftar_kamus.jpg* yang terletak di *folder res/drawable/daftar_kamus.jpg*.
- f. *Button* 4 menggunakan gambar dengan nama *info.png* yang terletak di *folder res/drawable/info.png*.
- g. *Button* 5 menggunakan gambar dengan nama *keluar.png* yang terletak di *folder res/drawable/keluar.png*.

2) Rancangan tampilan *interface* menu hewan



Sumber : Hasil Penelitian

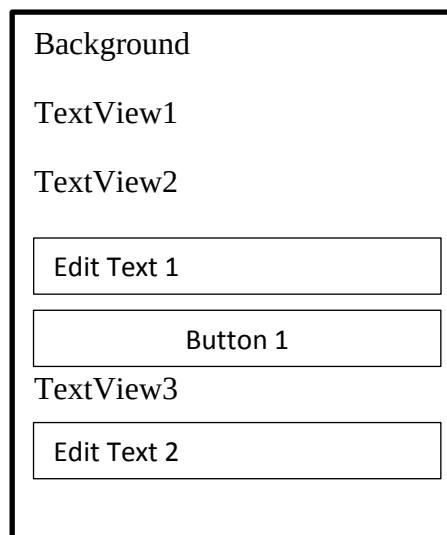
Gambar III.16 Rancangan *interface* menu hewan

Keterangan dari tampilan *interface* menu hewan sebagai berikut:

- a. *Background* menggunakan gambar dengan nama *hewan.jpg* yang terletak pada *folder res/drawable/hewan.jpg*.

- b. *TextView1* menggunakan gambar dengan nama *nhewan.png* yang terletak pada *folder res/drawable/nhewan.png*.
- c. *TextView2* menggunakan gambar dengan nama *bahasa.png* yang terletak pada *folder res/drawable/bahasa.png*.
- d. *EditText1* digunakan untuk memasukan kata untuk diterjemahkan ke dalam bahasa latin atau nama ilmiah.
- e. *Button1* menggunakan gambar dengan nama *terjemahkan.jpg* yang terletak pada *folder res/drawable/terjemahkan.jpg*.
- f. *TextView3* menggunakan gambar dengan nama *nil.png* yang terletak pada *folder res/drawable/nil.png*.
- g. *EditText2* ini digunakan untuk menampilkan hasil terjemahan dari *field EditText1*.

3) Rancangan tampilan *interface* menu tumbuhan



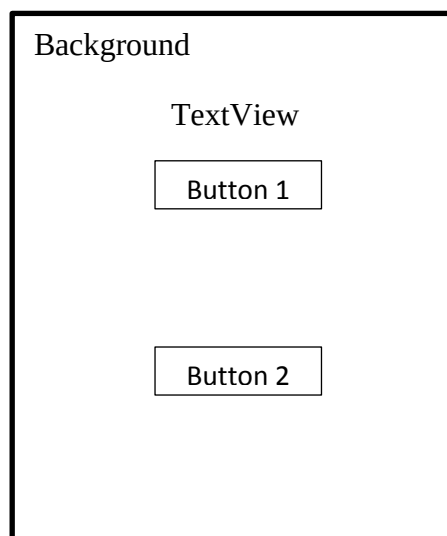
Sumber : Hasil Penelitian

Gambar III. 17 Rancangan *interface* menu tumbuhan

Keterangan dari tampilan *interface* menu tumbuhan sebagai berikut:

- a. *Background* menggunakan gambar dengan nama *tumbuhan.png* yang terletak pada *folder res/drawable/tumbuhan.png*.
- b. *TextView1* menggunakan gambar dengan nama *ntumbuhan.png* yang terletak pada *folder res/drawable/ntumbuhan.png*.
- c. *TextView2* menggunakan gambar dengan nama *bahasa.png* yang terletak pada *folder res/drawable/bahasa.png*.
- d. *EditText1* digunakan untuk memasukan kata untuk diterjemahkan ke dalam bahasa latin atau nama ilmiah.
- e. *Button1* menggunakan gambar dengan nama *terjemahkan.jpg* yang terletak pada *folder res/drawable/terjemahkan.jpg*.
- f. *TextView4* menggunakan gambar dengan nama *nil.png* yang terletak pada *folder res/drawable/nil.png*.
- g. *EditText* ini digunakan untuk menampilkan hasil terjemahan dari *field EditText1*.

4) Rancangan tampilan *interface* menu daftar kamus



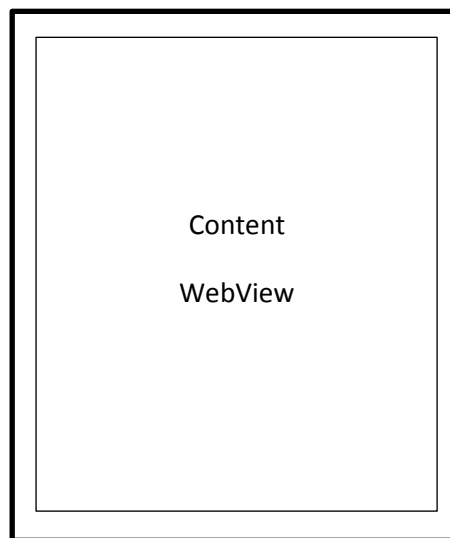
Sumber : Hasil Penelitian

Gambar III.18 Rncangan *interface* menu daftar kamus

Keterangan dari tampilan *interface* menu datar kamus sebagai berikut:

- a. *Background* menggunakan gambar dengan nama pilihan.jpg yang terletak pada folder *res/drawable/pilihan.jpg*.
- b. *TextView1* menggunakan gambar dengan nama pilih.png yang terletak pada folder *res/drawable/pilih.png*.
- c. *Button1* menggunakan gambar dengan nama hewan1.jpg yang terletak pada folder *res/drawable/hewan1.jpg*.
- d. *Button2* menggunakan gambar dengan nama tumbuhan1.jpg yang terletak pada folder *res/drawable/tumbuhan1.jpg*.

5) Rancangan tampilan *interface* menu daftar kamus hewan



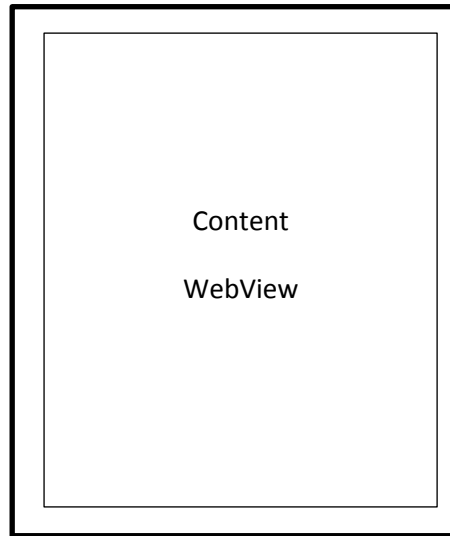
Sumber : Hasil Penelitian

Gambar III.19 Rancangan *interface* menu daftar kamus hewan

Keterangan dari tampilan *interface* menu daftar kamus hewan sebagai berikut:

- a. *WebView* digunakan untuk menampilkan tampilan yang berupa *.html* dan *WebView* ini bisa di *scroll down or up* (geser horizontal atau vertikal), *WebView* sendiri mengambil *source* dari *file* bernama Daftar Nama Ilmiah Hewan.html di folder *assets*.

6) Rancangan tampilan *interface* menu daftar kamus tumbuhan



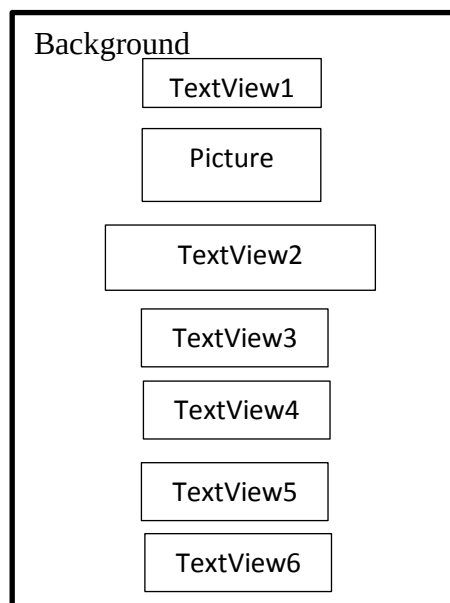
Sumber : Hasil Penelitian

Gambar III.20 Rancangan *interface* menu daftar kamus tumbuhan

Keterangan dari tampilan *interface* menu daftar kamus tumbuhan sebagai berikut:

- a. *WebView* digunakan untuk menampilkan tampilan berupa *.html* dan *WebView* ini bisa di *scroll down or up* (geser horizontal atau vertikal), *WebView* sendiri mengambil *source* dari *file* bernama Daftar Nama Ilmiah Tumbuhan.*html* yang terletak di *folder assets*.

7) Rancangan tampilan *iinterface* menu info



Sumber : Hasil Penelitian

Gambar III.21 Rancangan *interface* menu info

Keterangan dari tampilan *interface* menu info sebagai berikut:

- a. *Background* menggunakan gambar dengan nama *bg3.jpg* yang terletak pada folder *ress/drawable/bg3.jpg*.
- b. *TextView1* merupakan text yang bertuliskan “INFO”
- c. *Picture* menggunakan gambar dengan nama *capture.png* yang terletak pada folder *ress/drawable/capture.png*.
- d. *TextView2* merupakan text yang menerangkan tentang penjelasan aplikasi.
- e. *TextView3* merupakan text yang bertuliskan “Contact”
- f. *TextView4* merupakan text yang bertuliskan *email* pembuat aplikasi.
- g. *TextView5* merupakan text yang bertuliskan “STMIK NUSA MANDIRI JAKARTA”.
- h. *TextView6* merupakan text yang bertuliskan “2017”.

3.3. Implementasi

Aplikasi kamus nama ilmiah ini merupakan aplikasi berbasis *mobile android*. Jadi aplikasi ini dapat dijalankan pada *smatphone* yang memiliki sistem operasi *android*. Adapun tampilan yang merancang menggunakan *software android studio* sebagai berikut:

1. Splash Screen

Sebelum menampilkan menu utama terlebih dahulu tampil *splashscreen* yang berguna sebagai *feedback* bahwa aplikasi tersebut sedang melakukan proses *loading*.



Sumber : Hasil Penelitian

Gambar III. 22 Splash Screen

2. Menu utama

Menu utama berisi *buttun-button* yang akan mengarah ke halaman selanjutnya. *Button-button* tersebut adalah hewan, tumbuhan, daftar kamus, info dan keluar.

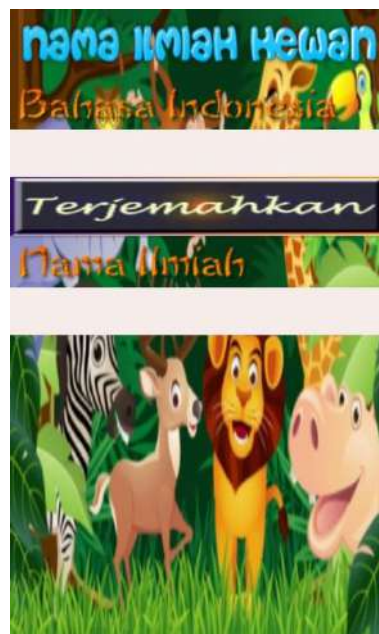


Sumber : Hasil Penelitian

Gambar III.23 Menu Utama

3. Menu hewan

Menu hewan ini berisi *button*, *edit text*, dan *text view* dalam menu ini nantinya pengguna akan bisa memasukan kata yang ingin diterjemahkan dalam konteks nama ilmiah. Apabila kata yang dicari tidak terdapat di dalam *database* maka kolom bahasa latin akan menampilkan “KATA TIDAK DITEMUKAN”.



Sumber : Hasil Penelitian

Gambar III.24 Menu Terjemah Hewan

4. Menu Tumbuhan

Menu tumbuhan ini berisi *button*, *edit text*, dan *text view* dalam menu ini nantinya pengguna akan bisa memasukan kata yang ingin diterjemahkan dalam konteks nama ilmiah. Apabila kata yang dicari tidak terdapat di dalam *database* maka kolom bahasa latin akan menampilkan “KATA TIDAK DITEMUKAN”.



Sumber : Hasil Penelitian

Gambar III.25 Menu Terjemah Tumbuhan

5. Menu daftar kamus

Menu daftar kamus ini berisi *button* pilihan yang akan mengarahkan ke halaman selanjutnya, yaitu halaman daftar kamus hewan dan daftar kamus tumbuhan.



Sumber : Hasil Penelitian

Gambar III.26 Menu Daftar Kamus**6. Menu daftar kamus hewan**

Menu ini berisi daftar kamus hewan yang diterjemahkan kedalam nama ilmiah atau bahasa latin. Dalam daftar kamus ini disusun sesuai urutan abjad yaitu dari A-Z, jadi pengguna hanya dengan melakukan geser keatas atau kebawah.

Daftar Nama Ilmiah Hewan	
A	
Anjing	= Canis Lupus
Anoa	= Bubalus Quarlesi
Ayam	= Gallus Gallus
B	
Banteng	= Bos Sundaicus
Beo	= Gracula Religiosa
Buaya	= Crocodylus Porosus
Bunglon	= Gonyoschepalus
Novaeguinae	
C	
Camar	= Larus sp
Cendrawasih	= Paradisaea Apoda
Cicak	= Hemidactylus Frenatus
D	
Domba Argali	= Ovis Ammon

Sumber : Hasil Penelitian

Gambar III.27 Daftar Kamus Hewan**7. Menu daftar kamus tumbuhan**

Menu ini berisi daftar kamus tumbuhan yang diterjemahkan kedalam nama ilmiah atau bahasa latin. Dalam daftar kamus ini disusun sesuai urutan abjad yaitu dari A-Z, jadi pengguna hanya dengan melakukan geser keatas atau kebawah.

Daftar Nama Ilmiah Tumbuhan	
A	
Alang Alang	= Imperata Cylindrical
Apel	= Pyrus Malus
Aren	= Arenga Pinnata
B	
Bayam	= Amaranthus Hibridus
Bengkuang	= Pachirrhysus Erosus
Bawang Merah	= Allium Cepa
C	
Cemara	= Cupressus Jusitanica
Ceremai	= Phillanthus Acidus
D	
Damar	= Podocarpus Damara
Delima	= Punica Granatum
Durian	= Durio Zibenthinus
E	

Sumber : Hasil Penelitian

Gambar III.28 Daftar Kamus Tumbuhan

8. Menu info

Menu info ini menampilkan sekilas penjelasan tentang aplikasi dan pembuat aplikasi serta kontak yang bisa dihubungi. Serta terdapat nama sekolah tinggi yang menunjukkan bahwa aplikasi ini dibuat untuk penyusunan skripsi.

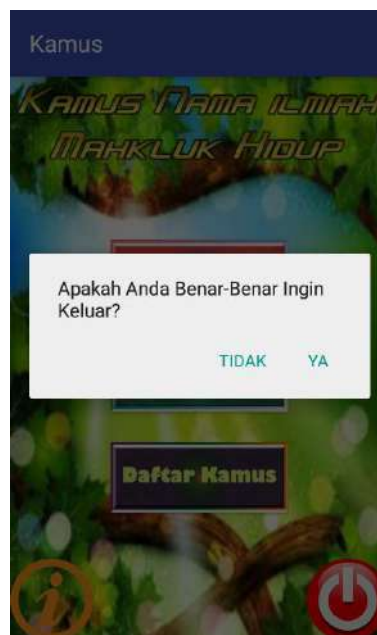


Sumber : Hasil Penelitian

Gambar III.29 Menu Info

9. Button keluar

Saat ditekan tombol keluar, sistem akan menampilkan dialog keluar aplikasi. Apabila pengguna ingin keluar tekan “YA” dan apabila masih ingin tetap berada dalam aplikasi tekan “TIDAK”.



Sumber : Hasil Penelitian

Gambar III.30 Tampilan Keluar

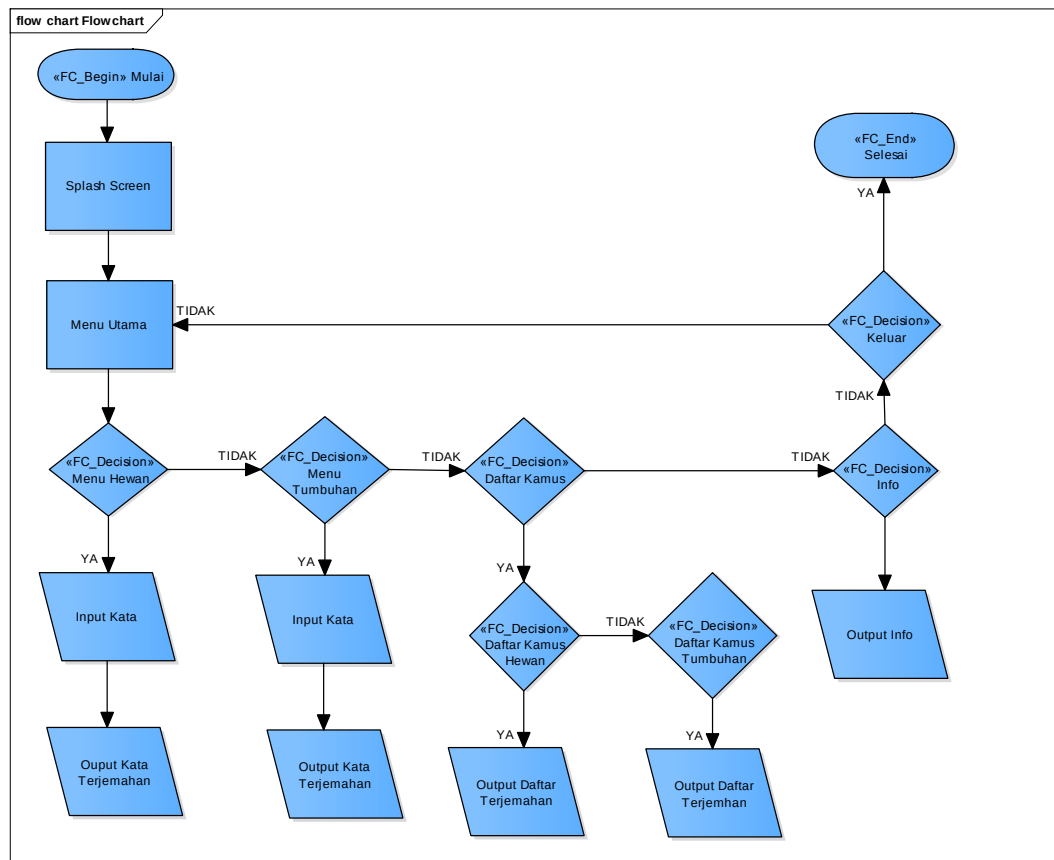
3.4. Testing

Testing dengan menggunakan *white box* dan *black box* untuk pengujian *white box*. Dengan menggunakan skema diagram alir, berikut merupakan diagram alir dalam aplikasi kamus nama ilmiah tumbuhan dan hewan.

1. White Box

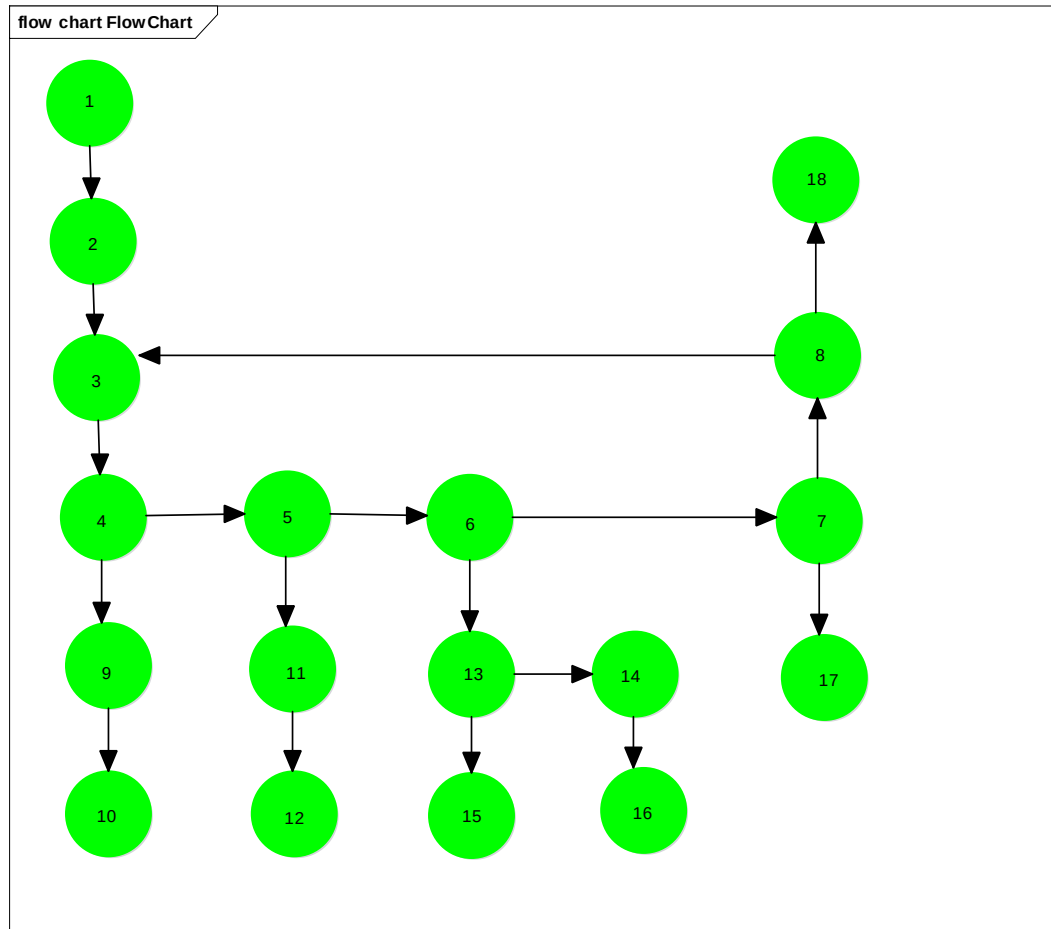
Dalam tahap pengujian *white box* ini penulis menggunakan skema diagram alir untuk mendapatkan hasil pengujian yang akurat. Berikut

merupakan *flowchart* dan skema diagram alir aplikasi kamus nama ilmiah tumbuhan dan hewan berbasis android:



Sumber : Hasil Penelitian

Gambar III.31 Flowchart Diagram



Sumber : Hasil Penelitian

Gambar III.32 Diagram Alir

Dari skema diagram alir diatas dapat diperoleh kompleksitas siklomatis dengan rumus sebagai berikut:

$$V(G) = E - N + 2$$

E = Jumlah *Edge* yang ditentukan dengan gambar anak panah

N = Jumlah simpul grafik alir yang ditentukan dengan gambar lingkaran

$$V(G) = 22 - 18 + 2 = 6$$

- a) 1-2-3-4-9-10
- b) 1-2-3-4-5-11-12
- c) 1-2-3-4-5-6-13-15

d) 1-2-3-4-5-6-13-14-16

e) 1-2-3-4-5-6-7-17

f) 1-2-3-4-5-6-7-8-18

Ketika aplikasi dijalankan ,maka akan terlihat satu set baris yang dihasilkan adalah 1-2-3-4-9-10-3-4-5-11-12-3-4-5-6-13-15-3-4-5-6-13-14-3-4-5-6-13-14-16-3-4-5-6-7-17-3-4-5-6-7-8-3-4-5-6-7-8-18 dan terlihat bahwa simpul telah dieksekusi satu kali.

2. Black Box

Pengujian selanjutnya dilakukan untuk memastikan suatu masukan menjelaskan proses yang tepat dan menghasilkan *output* yang sesuai dengan rancangan.

Tabel III.1 Pengujian *Black Box*

No.	Menu	Unjuk Kerja	Keterangan
1.	<i>Splash Screen</i>	Menampilkan tampilan <i>splash screen</i>	Berhasil
		Menampilkan <i>loading</i> saat <i>splash screen</i> sedang berjalan	Berhasil
2.	Menu Utama	Menampilkan tampilan <i>home</i>	Berhasil
		Menampilkan menu hewan, tumbuhan, daftar kamus, info dan keluar	Berhasil
		Menampilkan menu hewan setelah <i>user menklik</i> menu hewan dan menampilkan <i>layout</i> berikutnya	Berhasil
		Menampilkan menu tumbuhan setelah <i>user menklik</i> menu tumbuhan	Berhasil
		Menampilkan menu daftar kamus setelah <i>user menklik</i> dan menampilkan	Berhasil
		Menampilkan menu info	Berhasil
		Menampilkan kotak dialog keluar aplikasi	Berhasil
3.	Menu hewan	Memasukan kata terjemah dalam kolom <i>EditText</i>	Berhasil
		Menampilkan <i>button</i> terjemahkan	Berhasil

		Menampilkan kata sesudah <i>user</i> <i>menklik button</i> terjemahkan	Berhasil
		Menampilkan hasil terjemahan	Berhasil
4	Menu Tumbuhan	Memasukan kata terjemah dalam kolom <i>EditText</i>	Berhasil
		Menampilkan <i>button</i> terjemahkan	Berhasil
		Menampilkan kata sesudah <i>user</i> <i>menklik button</i> terjemahkan	Berhasil
		Menampilkan hasil terjemahan	Berhasil
5.	Daftar Kamus	Menampilkan menu pilih kategori hewan dan tumbuhan	Berhasil
		Menampilkan <i>button</i> hewan	Berhasil
		Menampilkan <i>button</i> tumbuhan	Berhasil
6.	Daftar Kamus Hewan	Menampilkan daftar tentang terjemahan nama ilmiah hewan	Berhasil
7.	Daftar Kamus Tumbuhan	Menampilkan daftar tentang terjemahan nama ilmiah tumbuhan	Berhasil
8.	Menu Info	Menampilkan keterangan pembuat aplikasi, foto pembuat aplikasi dan <i>user</i> dapat melihat kontak pembuat aplikasi serta terdapat informasi tentang aplikasi yang dibuat.	Berhasil
9.	Keluar	Menampilkan kotak dialog keluar aplikasi dan juga tombol “ya” dan “tidak” jika <i>user</i> menekan ya maka akan keluar dari aplikasi jika tidak kembali ke menu utama.	Berhasil

Sumber : Hasil Penelitian

3.5. Support

Dalam bahasan di *point* ini berisikan spesifikasi *hardware* dan *software* yang akan digunakan untuk pemasangan aplikasi yang telah diujikan.

Table III.2 Kebutuhan Perangkat Aplikasi

Kebutuhan	Keterangan
Sistem Operasi	Android v4.0 Ice Cream Sandwich
Prosesor	Dual Core 1Ghz
RAM	512MB
Memori Penyimpanan	50MB

Sumber : Hasil Penelitian

BAB IV

PENUTUP

4.1. Kesimpulan

Kesimpulan dalam penulisan skripsi ini adalah sebagai berikut:

1. Aplikasi *android* memang membantu dalam proses pembelajaran apalagi sekarang jaman digital yang semuanya tidak terlepas dari *gadget*.
2. Aplikasi kamus memberikan kemudahan dalam mencari suatu kata maupun arti sehingga media pembelajaran melalui aplikasi kamus ini sangat efektif.
3. Selain kamus ada juga buku sebagai media pembelajaran, mengingat kebutuhan masyarakat akan mobilitas tinggi, sehingga aplikasi kamus berbasis *android* sangat membantu karena tidak perlu repot lagi membawa buku yang mungkin tebalnya melebihi dari kamus bahasa.
4. Kamus ini bertujuan untuk memberikan pembelajaran kepada masyarakat umum yang membutuhkan atau hanya sekedar ingin tahu tentang nama ilmiah dari hewan dan tumbuhan.

4.2. Saran

Berdasarkan hasil penelitina, maka penulis merekomendasikan atau menyarankan beberapa hal mengenai karya ilmiah ini:

1. Untuk meningkatkan akan baca maka kamus elektronik adalah hal yang tepat untuk melakukan pembelajaran secara mandiri, selain itu mudah digunakan dan efisien.
2. Pengaruh zaman digital yang tidak dapat dihindarkan maka dari itu pengetahuan tentang hal-hal yang kecil itu penting, apalagi kalau hal kecil itu dapat bermanfaat untuk kehidupan masyarakat.
3. Secara keseluruhan aplikasi ini kurang dari kata sempurna karena dengan tampilan yang sederhana. Dengan tampilan sederhana maka memori yang dibutuhkan akan kecil, dan dapat disimpan untuk *smartphone* dengan memori kecil.
4. Saran untuk pembaca dari penulis semoga pembaca mengerti dan paham apa yang penulis jabarkan ke dalam karya ilmiah dan penulis juga tidak berharap juga pembaca akan mengerti semuanya.

Penulis berharap pembaca dapat mengambil hikmah dari karya tulis ini, penulis ucapkan terima kasih.

DAFTAR PUSTAKA

- A.S, Rosa, & M.Shalahudin. (2013). *Rekayasa Perangkat Lunak*. Bandung: Bandung Informatika.
- Andi. (2013). *Android Programing With Eclipse*. Yogyakarta: CV Andi Offset.
- Dryad, Pandu. (20 September 2015). *Urutan Tingkatan Versi Nama Android*. Dipetik 28 Oktober, 2016, dari Techijau: <http://techijau.com/urutan-versi-nama-android/>
- Hermawan, Stephanus S. (2011). *Mudah Membuat Aplikasi Android*. Yogyakarta: CV Andi Offset.
- JadiBaru. (9 November 2015). *Pengenalan Android Studio*. Dipetik 30 Oktober 2016, dari Jadibaru: <http://www.jadibaru.com/android/pengenalan-android-studio-2/>
- Kurniawan, Dwi Ely, Simon Reinhard, & Irwansyah. (2013). Aplikasi Kamus Aneka Bahasa Daerah Berbasis Smartphome Android. *Teknologi Informasi, Komunikasi dan Industri*, 5.
- Nasional, Pusat Bahasa. (2002). *Kamus Besar Bahasa Indonesia*. Jakarta: Balai Pustaka.
- Nikko, Sora. (24 September 2015). *Pengertian UML dan Jenis-Jenisnya Serta Contoh Diagramnya*. Dipetik 2 November 2016, dari Pengertian Apapun: <http://www.pengertianku.net/2015/09/pengertian-uml-dan-jenis-jenisnya-serta-contoh-diagramnya.html>
- Nugraha, Taufik Sapta, & Tresnawati Dewi. (2015). Pengembangan Aplikasi Pengenalan Kesenian Daerah. *Jurnal Algoritma*, 1.
- Parno, Darmayanti, & Rahmansyah Nandang. (2011). Aplikasi Mobile Kamus Istilah Psikologi Berbasis Android 2.2. *Psikologi, Ekonomi, Sastra, Arsitektur & Sipil*, 121.
- Perbanga, Abrizeni Duha. (November 2015). *Arsitektur Android*. Dipetik 29 Oktober 2016, dari Perbanga: <http://www.perbanga.com/2015/11/arsitektur-andorid.html>
- Sadeli, Muhammad. (2014). *Toko Buku Online Dengan Android*. Palembang: Maxikom.
- Yuswanto. (2008). *Algoritma dan Pemograman Dengan Visual Basic.net 2005*. Jakarta: Cerdas Pustaka Publisher.

DAFTAR RIWAYAT HIDUP

I. Biodata Mahasiswa

NIM : 12135076
Nama Lengkap : Ahmad Riki Setiawan
Tempat dan Tanggal Lahir : Pati, 24 Februari 1995
Alamat lengkap : Jl. Kayubesar Dalam Rt 003/012,
Kel.Tegal Alur, Kec.Kalideres,
Jakarta Barat.11820.

II. Pendidikan formal

1. SDN Gesengan 01 Pati, Lulus tahun 2006.
2. SMPN 1 Cluwak Pati, Lulus tahun 2009
3. SMK Gajah Mada 01 Margoyoso, Lulus tahun 2012.
4. AMIK Bina Sarana Informatika Jakarta, Lulus tahun 2016.

III. Riwayat Pengalaman berorganisasi / pekerjaan

1. PT. Slamet Sempurna Tbk. Kapuk Kamal No.88 Jakarta Barat Tahun 2012-2014.



Jakarta, 6 Januari 2017

Ahmad Riki Setiawan

	LEMBAR KONSULTASI BIMBINGAN SKRIPSI
	SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER NUSA MANDIRI

NIM : 12135076
 Nama Lengkap : Ahmad Riki Setiawan
 Dosen Pembimbing : Pas Mahyu Akhirianto SPd, M.Kom
 Judul Skripsi : Aplikasi Kamus Nama Ilmiah Tumbuhan dan
 Hewan Berbasis Android dengan Metode
 Sequential Searching

No.	Tanggal Bimbingan	Pokok Bahasan	Paraf Dosen Pembimbing
1	25 Okt 2016	Bimbingan Perdana	
2	11 Nop 2016	Pengajuan Judul, dan Bab 1	
3	25 Nop 2016	Acc Judul dan Bab 1	
4	22 Des 2016	Pengajuan Bab 2 & Aplikasi	
5	13 Jan 2017	Acc Bab 2 dan Pengajuan Bab 3	
6	24 Jan 2017	Acc Bab 3 dan Pengajuan Bab 4	
7	06 Feb 2017	Acc Bab 4 dan Aplikasi	
8	21 Feb 2017	Revisi dan Bimbingan Akhir	

Bimbingan Skripsi

- Dimulai pada tanggal : 25 Oktober 2016
- Diakhiri pada tanggal : 21 Februari 2017
- Jumlah pertemuan bimbingan : 8 Kali Pertemuan

Disetujui oleh,
Dosen Pembimbing

(Pas Mahyu A. SPd, M.Kom)

LAMPIRAN

A. Emulator Alternatif

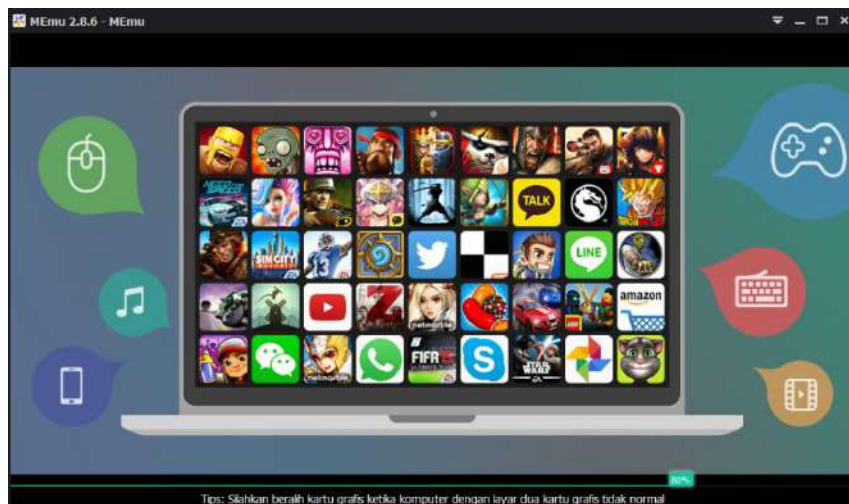
Memu Emulator adalah emulator android keluaran China, memu mengkalim sebagai salah satu Emulator Android dengan performa terbaik dan juga paling *smooth*. MEMu juga bisa digunakan di beberapa perangkat seperti PC, Notebook hingga tablet. MEMu masih berbasis Android 4.2.2 Jelly Bean, cukup *jadul* memang, namun setidaknya cukup stabil digunakan.

Spesifikasi Minimum untuk menjalankan Memu Emulator:

Sistem Operasi	: Windows XP SP3 / 7 / 8.1 / 10.
Processor	: Intel / AMD
Ram	: 2 GB (Minimum)
Directx	: 9.0c
Hardisk	: 5 GB

Beberapa prosessor AMD yang tidak support sebagai berikut:

- AMD Athlon (tm) 64 X2 DUAL CORE Processor 4000+
- AMD Athlom(tm) X4 905e processor 2.3
- X4 610e
- X2 Dual 5600+
- X2 DUAL 4800+
- X4 965
- X2 250
- X2 4600+
- X2 4200+



Gambar Memu Emulator

B. Spesifikasi Minimal Komputer untuk Android Studio

Dibawah ini adalah spesifikasi minimal komputer untuk menjalankan aplikasi Android Studio:

- OS Windows 7/8 32 - 64 bit
- RAM 2 GB , Disarankan memakai RAM 4 GB
- 2 GB Minimum Ruang di Hardisk 4 GB Disarankan
- 1280 x 800 Minimum Resolusi Layar
- Java Development Kit (JDK) 7 or Java Development Kit (JDK) 8 (Support API 24 ,Android Nougat
- Untuk bisa mengakses emulator terbaru pastikan intel processor kalian mendukung Intel® VT-x, Intel® EM64T (Intel® 64), dan Execute Disable (XD) Bit functionality

C. Listing Source Code Program

1. Listing Program Layout (.xml)

a. Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1"
    android:background="@drawable/bg1">

    <Button
        android:layout_width="150dp"
        android:layout_height="60dp"
        android:id="@+id/bHewan"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/hewan1"
        android:layout_marginTop="60dp"
        android:layout_below="@+id/imageButton2"
        android:layout_centerHorizontal="true" />

    <Button
        android:layout_width="150dp"
        android:layout_height="60dp"
        android:id="@+id/bTumbuhan"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/tumbuhan1"
        android:layout_below="@+id/bHewan"
        android:layout_alignLeft="@+id/bHewan"
        android:layout_alignStart="@+id/bHewan"
        android:layout_marginTop="25dp" />

    <ImageButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```

        android:id="@+id/imageButton2"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:background="@drawable/judul" />

<Button
    android:layout_width="150dp"
    android:layout_height="60dp"
    android:id="@+id/bDaftar"
    android:layout_below="@+id/bTumbuhan"
    android:layout_centerHorizontal="true"
    android:background="@drawable/daftar_kamus"
    android:layout_marginTop="25dp" />

<Button
    android:layout_width="70dp"
    android:layout_height="70dp"
    android:id="@+id/ibInfo"
    android:background="@drawable/info"
    android:layout_alignParentBottom="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />

<Button
    android:layout_width="70dp"
    android:layout_height="70dp"
    android:id="@+id/ibKeluar"
    android:layout_alignBottom="@+id/ibInfo"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:background="@drawable/keluar" />
</RelativeLayout>

```

b. Splashscreen.xml

```

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

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:src="@drawable/kamus"
        android:scaleType="fitCenter"
        android:background="#73b9f2" />

    <ProgressBar
        android:id="@+id/progressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="58dp" />
</RelativeLayout>

```

c. Activity_hewan.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    tools:context="com.ars.sony.kamus2.MainActivity"
    android:background="@drawable/hewan">

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"

        android:textAppearance="?android:attr/textAppearanceLarge"
        android:background="@drawable/nhewan" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView01"
        android:background="@drawable/bahasa" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:id="@+id/txtIndonesia"
        android:text=""
        android:background="#f4ede9">

    </EditText>
    <Button
        android:id="@+id/btnTerjemahan"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:onClick="getTerjemahan"
        android:background="@drawable/terjemahkan" />
    <TextView
        android:id="@+id/TextView03"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@drawable/nil" />

    <EditText
        android:id="@+id/txtLatin"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:background="#f4ede9">
    </EditText>

</LinearLayout>

```

d. Activity_tumbuhan.xml

```

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

```



```

xmlns:tools="http://schemas.android.com/tools"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:orientation="vertical"
tools:context="com.ars.sony.kamus2.MainActivity"
android:background="@drawable/tumbuhan">

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:gravity="center"

    android:textAppearance="?android:attr/textAppearanceLarge"
    android:background="@drawable/ntumbuhan" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textView01"
    android:background="@drawable/bahasa" />

<EditText
    android:layout_width="fill_parent"
    android:layout_height="40dp"
    android:id="@+id/txtIndonesia"
    android:text=""
    android:background="#f4ede9">

</EditText>
<Button
    android:id="@+id/btnTerjemah"
    android:layout_width="fill_parent"
    android:layout_height="50dp"
    android:onClick="getTerjemahan"
    android:background="@drawable/terjemahkan" />
<TextView
    android:id="@+id/TextView03"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@drawable/nil" />

<EditText
    android:id="@+id/txtLatin"
    android:layout_width="fill_parent"
    android:layout_height="40dp"
    android:background="#f4ede9">
</EditText>

</LinearLayout>

```

e. Activity_daftar_kamus.xml

```

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

```

```

        android:background="@drawable/pilihan">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="New Text"
            android:id="@+id/textView"
            android:layout_alignParentTop="true"
            android:layout_centerHorizontal="true"
            android:background="@drawable/pilih" />

        <Button
            android:layout_width="150dp"
            android:layout_height="60dp"
            android:id="@+id/button1"
            android:layout_below="@+id/textView"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="100dp"
            android:background="@drawable/hewan1" />

        <Button
            android:layout_width="150dp"
            android:layout_height="60dp"
            android:id="@+id/button2"
            android:layout_below="@+id/button1"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="96dp"
            android:background="@drawable/tumbuhan1" />
    </RelativeLayout>

```

f. Activity_daftar_kamus_hewan.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1">

    <WebView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/webView"
        android:layout_gravity="center_horizontal" />
</LinearLayout>

```

g. Activity_daftar_kamus_tumbuhan.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1">

    <WebView
        android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:id="@+id/webView2"
        android:layout_gravity="center_horizontal" />
</LinearLayout>

```

h. Activity_info.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg3">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="INFO"
        android:id="@+id/textView2"
        android:layout_gravity="center_horizontal"
        android:textSize="30dp" />

    <ImageView
        android:layout_width="150dp"
        android:layout_height="174dp"
        android:id="@+id/imageView"
        android:layout_gravity="center_horizontal"
        android:background="@drawable/capture"
        android:layout_marginTop="20dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Aplikasi Kamus Nama Ilmiah Tumbuhan dan
Hewan ini dibuat untuk mencari nama ilmiah makhluk hidup yang
dalam kasus ini adalah hewan dan tumbuhan. Aplikasi ini mudah,
praktis dan juga efisien."
        android:id="@+id/textView3"
        android:layout_gravity="center_horizontal"
        android:textAlignment="center"
        android:textStyle="bold"
        android:layout_marginTop="30dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Contact"
        android:id="@+id/textView4"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="20dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="ahmadc242@gmail.com"
        android:id="@+id/textView5"
        android:layout_gravity="center_horizontal"
        android:textColor="#033dfc"

```

```

        android:layout_marginTop="10dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="STMIK NUSA MANDIRI JAKARTA"
    android:id="@+id/textView6"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="30dp"
    android:textStyle="bold"
    android:textSize="20dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="2017"
    android:id="@+id/textView7"
    android:layout_gravity="center_horizontal"
    android:textStyle="bold"
    android:textSize="20dp" />

</LinearLayout>

```

2. Listing Kode Program (.java)

a. MainActivity.java

```

package com.ars.sony.kamus2;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.KeyEvent;
import android.view.View;
import android.view.WindowManager;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);

        Button bHewan = (Button) findViewById(R.id.bHewan);

        bHewan.setOnClickListener(new View.OnClickListener() {

            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent i = new Intent(MainActivity.this,
                    ActivityHewan.class);
                startActivity(i);
            }
        });
    }
}

```

```

        Button bTumbuhan = (Button) findViewById(R.id.bTumbuhan);

        bTumbuhan.setOnClickListener(new View.OnClickListener() {
            public void onClick(View arg0) {
                Intent j = new Intent(MainActivity.this,
ActivityTumbuhan.class);
                startActivity(j);
            }
        });

        Button bDaftar = (Button) findViewById(R.id.bDaftar);

        bDaftar.setOnClickListener(new View.OnClickListener() {

            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent k = new Intent(MainActivity.this,
daftarkamus.class);
                startActivity(k);
            }
        });

        Button bInfo = (Button) findViewById(R.id.ibInfo);

        bInfo.setOnClickListener(new View.OnClickListener() {

            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent l = new Intent(MainActivity.this,
info.class);
                startActivity(l);
            }
        });

        Button bKeluar = (Button) findViewById(R.id.ibKeluar);

        bKeluar.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                // TODO Auto-generated method stub
                close();
            }
        });
    }

    public void close() {
        AlertDialog.Builder builder = new
AlertDialog.Builder(this);
        builder.setMessage("Apakah Anda Benar-Benar Ingin
Keluar?")
            .setCancelable(false).setPositiveButton("Ya", new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int id) {
                    finish();
                }
            })
            .setNegativeButton("Tidak", new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,

```

```

int id) {
    dialog.cancel();
    }
    }).show();
}

public boolean onKeyDown(int keyCode, KeyEvent event) {
    if (keyCode == KeyEvent.KEYCODE_BACK) {
        close();
    }
    return super.onKeyDown(keyCode, event);
}
}

```

b. Splashscreen.java

```

package com.ars.sony.kamus2;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.Window;
import android.view.WindowManager;

public class splashscreen extends Activity {
    //Set waktu lama splashscreen
    private static int splashInterval = 2000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);
        setContentView(R.layout.splashscreen);
        new Handler().postDelayed(new Runnable() {

            @Override
            public void run() {
                // TODO Auto-generated method stub
                Intent i = new Intent(splashscreen.this,
MainActivity.class);
                startActivity(i);

                //jeda selesai Splashscreen
                this.finish();
            }

            private void finish() {
                // TODO Auto-generated method stub

            }
        }, splashInterval);
    };
}

```