

**IMPLEMENTASI ALGORITMA KNUTH-MORRIS-PRATT
PADA APLIKASI KAMUS BAHASA INDONESIA - BATAK
BERBASIS ANDROID**

SKRIPSI



**Disusun Oleh :
JEKSEN MANUEL SIMARMATA
NIM : 08.18.105**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

LEMBAR PERSETUJUAN

**IMPLEMENTASI ALGORITMA KNUTH-MORRIS-PRATT
PADA APLIKASI KAMUS BAHASA INDONESIA – BATAK
BERBASIS ANDROID**

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelur Sarjana Teknik Informatika Strata Satu (S-1)*

Disusun Oleh :

**JEKSEN MANUEL SIMARMATA
NIM : 08.18.105**

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II

**Joseph Dedy Irawan, ST, MT
NIP. 1974041620005021002**

**Yoseph Agus Pranoto, ST
NIP.P. 1031000432**

Mengetahui

Ketua Program Studi Teknik Informatika S-1

**Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002**

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

2013

Lampiran 1 : Surat Pernyataan



INTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

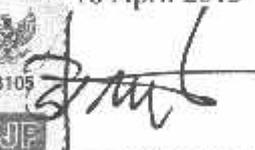
SURAT PERNYATAAN

Nama : Jeksen Manuel Simarmata
NIM : 0818105
Jurusan : Teknik Informatika S-1

Menyatakan bahwa karya skripsi saya yang berjudul :

**“Implementasi Algoritma Knut-Morris Prat pada kamus bahasa Indonesia –
Batak berbasis Android”**

Adalah bukan merupakan karya tulis orang lain, baik sebagian maupun keseluruhan, kecuali dalam bentuk kutipan yang kami sebutkan sumbernya. Demikian surat pernyataan ini kami buat dengan sebenar-benarnya dan apabila pernyataan ini tidak benar kami bersedia mendapatkan sanksi akademis.

18 April 2013
METERAI
TEMPEL
KEMENTERIAN KEHUTANAN
REPUBLIC OF INDONESIA
FE75CAAF135713105
ENKAM.SIDU.KUTZANI
6000 DJP

Jeksen Manuel Simarmata
08.18.105

KATA PENGANTAR

Puji dan syukur saya ucapkan kepada Tuhan Yesus Yang Maha baik dan Penyayang yang telah memberikan Rahmat yang berlimpah, sehingga penulis dapat menyelesaikan skripsi dengan judul **“IMPLEMENTASI ALGORITMA KNUTH-MORRIS-PRATT PADA APLIKASI KAMUS BAHASA INDONESIA – BATAK BERBASIS ANDROID”** dengan lancar. Skripsi merupakan persyaratan kelulusan Studi di Jurusan Teknik Informatika S-1 ITN Malang dan untuk mencapai gelar Sarjana Komputer.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Ir. Soeparno Djiwo, MT selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Anang Subardi, MT selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Joseph Dedy Irawan, ST, MT selaku Ketua Jurusan Teknik Informatika S-1 dan juga selaku Dosen Pembimbing I.
4. Bapak Yoseph Agus Pranoto, ST selaku Dosen Pembimbing II.
5. Mama tercinta Kastina Kristina br Damanik yang kasih dan sayangnya selalu menjadi inspirasi bagi penulis, damai di surga Ma
6. Papa tercinta J.Simarmata serta abang dan kakak – kakak dari penulis yang tiada pernah lelah memberikan doa dan dukungan, semoga Tuhan memberikan umur yang panjang bagi kalian semua.
7. Teman-teman serta saudara-saudara yang selalu memberikan doa restu, dorongan dan semangat dan semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penulis telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan ini. Untuk itu penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini. Harapan penulis semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Februari 2013

Penulis

IMPLEMENTASI ALGORITMA KNUTH-MORRIS-PRATT PADA APLIKASI KAMUS BAHASA INDONESIA – BATAK BERBASIS ANDROID

Jeksen Manuel Simarmata

Program Studi Teknik Informatika S-1,

Fakultas Teknologi Industri, Institut Teknologi Nasional Malang

Jln. Raya Karanglo Km 2 Malang

Email: jeksen.simarmata@gmail.com

Dosen Pembimbing : 1. Joseph Dedy Irawan ST, MT

2. Yoseph Agus Pranoto, ST

Abstrak

Perkembangan media elektronik pada saat ini berpengaruh sangat besar dalam kehidupan manusia, salah satu hal terbesar yang mempengaruhi tersebut adalah Mobile phone. Mobile phone saat ini tidak hanya sebagai alat untuk komunikasi saja tetapi juga sebagai alat penunjang kehidupan sehari-hari contohnya sebagai media pembelajaran untuk mempelajari suatu hal.

Kamus adalah salah satu alat bantu komunikasi antara satu orang dengan orang yang lain. Seiring pesatnya perkembangan teknologi mobile, kamus pada perangkat seluler pun sangat dibutuhkan. Ini dikarenakan kamus yang ada pada saat ini adalah kamus dalam bentuk buku, dimana penggunaannya sangat memakan tempat dan waktu. Kamus BAHASA INDONESIA - BATAK adalah kamus berbasis system operasi android yang dapat di jalankan pada android versi 2.3 sampai 4.0. Dibuat menggunakan perangkat lunak eclipse dan algoritma pencarian kata KNUTH-MORRIS-PRATT. kamus bahasa INDONESIA – BATAK berfungsi untuk mencari artian kata dari bahasa INDONESIA ke bahasa BATAK dan juga dari bahasa BATAK ke bahasa INDONESIA. Kamus ini dibuat sebagai salah satu cara untuk melestarikan budaya suku BATAK dan juga membantu para pengguna dari suku lain untuk belajar bahasa BATAK.

Aplikasi ini telah dilakukan pengujian terhadap beberapa user dan 50% dari user memberikan nilai baik atas keseluruhan dari aplikasi ini. Para user juga memberikan beberapa masukan yang membangun kepada penulis. Banyak dari mereka berharap agar aplikasi ini bisa dikembangkan lebih jauh lagi.

Kata Kunci : Kamus, batak, algoritma Knuth-Morris-Pratt, Indonesia, Android

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
Abstrak.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	2
1.5 Metode Penelitian.....	3
1.6 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	5
2.1 Android.....	5
2.1.1 Versi – versi Android.....	5
2.1.2 Karakteristik Android).....	16
2.1.3 Cara Kerja Sistem Android.....	17
2.2 Eclipse.....	22
2.3 Database.....	2
2.3.1 Database AQLite.....	16
2.4 Java.....	16
2.4.1 Java API.....	16
2.4.2 Java Virtual Machine.....	16
2.4.3 Sistem Operasi Java.....	16
2.4.4 Dasar Pemograman Java.....	16

2.5	Algoritma Pencarian Kata	
2.5.1	Algoritma Knuth Morris Pratt	
BAB III PERANCANGAN DAN ANALISA SISTEM		15
3.1	Spesifikasi Aplikasi	16
3.2	Perancangan Aplikasi	16
3.2.1	Flowchart	17
3.2.2	Perancangan Tabel Database	17
3.2.3	Perancangan Form Aplikasi	17
3.3	Analisa Sistem	17
3.3.1	Analisa Perangkat Keras (<i>Hardware</i>)	17
3.3.1	Analisa Perangkat Lunak (<i>Software</i>)	17
3.4	Memulai pembuatan Aplikasi dengan Eclipse	17
3.5	Database Sistem Pada Aplikasi Kamus	17
3.6	Algoritma Knuth-Morris-Pratt	17
BAB IV IMPLEMENTASI DAN PENGUJIAN		26
4.1	Blok Diagram	26
4.2	Pengujian Aplikasi pada perangkat android	26
4.2.1	Tampilan Aplikasi Setelah diinstal pada perangkat android	26
4.2.2	Tampilan awal aplikasi pertama kali	27
4.2.3	Tampilan aplikasi ketika dilakukan pencarian data	27
4.2.4	Pengujian bookmark	27
4.3	Hasil pengujian	28
4.3.1	Tabel Fungsionalitas Program	28
4.3.2	Tabel Pengujian User	30
BAB V PENUTUP		38
5.1	Kesimpulan	38

5.2	Saran.....	38
DAFTAR PUSTAKA		39

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Media elektronik pada saat ini berpengaruh sangat pesat dalam kehidupan manusia. Demikian juga dalam perkembangan dibidang teknologi mobile phone yang semakin pesat secara tidak langsung telah memberikan keinginan dalam pelestarian bahasa daerah khususnya dalam hal ini adalah bahasa batak berupa aplikasi kamus bahasa Indonesia Batak. Dengan adanya kamus ini, orang Indonesia dapat belajar bahasa Batak yang dalam aplikasi ini terdiri dari tiga bagian yaitu bahasa Batak Toha, bahasa Batak Karo dan bahasa Batak Simalungun. Dengan dilatar belakangi adanya permasalahan ini, maka muncul ide penulis untuk membuat aplikasi ini.

Dengan pemilihan Android sebagai salah satu sistem informasi yang berbasis mobile dikarenakan perkembangan mobile phone di Indonesia berkembang sangat pesat. Adalah system operasi mobile android yang sekarang paling depan perkembangannya. Android merupakan sistem operasi untuk perangkat *mobile* yang bersifat *open source* sehingga pengembangannya dapat dilakukan dengan luas. Selain harganya yang sangat ekonomis android juga ditopang dengan berbagai pengembang yang memadai. Banyak produsen perangkat mobile yang menggunakan system operasi android untuk produknya.

Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata. Ia berfungsi untuk membantu seseorang mengenal perkataan baru. Selain menerangkan maksud kata tentunya, kamus juga mempunyai pedoman sebutan asal-usul (etimologi) sesuatu perkataan dan juga contoh penggunaan bagi sesuatu perkataan. Untuk memperjelas kadang kala terdapat juga ilustrasi di dalam kamus atau cara pembacaan.

Aplikasi kamus bahas INONESIA-BATAK ini dibuat atas keprihatinan penulis akan mulai pudarnya pengetahuan masyarakat suku batak akan bahasanya sendiri. Dengan adanya kamus bahasa INDONESIA – BATAK ini kiranya dapat turut serta melestarikan budaya batak. Dan juga dapat membantu pengguna dari suku lain yang ingin belajar bahasa batak untuk memenuhi kosa kata dalam bahasa batak.

Aplikasi ini dibuat menggunakan algoritma pencarian string KNUTH-MORRIS-PRATT. Algoritma KNUTH-MORRIS-PRATT atau yang disingkat KMP ini merupakan salah stu algoritma yang digunakan untuk mencari apakah ada suatu kata terdapat dalam suatu string atau kumpulan kata. Cara kerja algoritma ini yaitu dengan cara mencocokkan kata yang akan dicari dalam string atau kumpulan kata sampai seluruh huruf dalam kata yang dicari tadi memenuhi padanannya dalam kata yang ada.

1.2 Rumusan Masalah

Adapun rumusan masalah yang akan dibahas adalah membuat dan merancang aplikasi kamus berbasis android untuk menerjemahkan kata dari bahasa yaitu Batak (Toba, Simalungun, Karo) ke bahasa Indonesia dan dari bahasa Indonesia ke bahasa Batak (Toba, Simalungun, Karo).

1.3 Tujuan

Membuat aplikasi kamus bahasa INDONESIA – BATAK dan BATAK – INDONESIA berbasis android untuk memudahkan pengguna dalam belajar bahasa batak lebih praktis dan lengkap karena aplikasi ini dilengkapi dengan 3 bahasa batak yaitu batak toba, batak simalungun dan batak karo.

1.4 Batasan Masalah

Batasan masalah pembahasan yang di ambil agar sesuai dengan tujuan dan tidak terjadi penyimpangan maksud dan tujuan utama, maka ditentukan ruang lingkup pembahasan sebagai berikut:

1. Menerjemahkan kata dari bahasa Indonesia ke bahasa batak dan sebaliknya.
2. Aplikasi ini hanya menterjemahkan kata saja.
3. Kata yang diterjemahkan dalam kamus ini adalah kata yang biasa digunakan oleh masyarakat batak dalam kehidupan sehari hari dan tidak menerjemahkan kata dalam bahasa tradisional batak.
4. Bahasa batak yang digunakan adalah bahasa batak toba, batak karo, batak simalungun.
5. Aplikasi ini hanya menggunakan database internal.
6. Menggunakan software eclipse helios sebagai sarana pembuatan aplikasi kamus Indonesia Batak.
7. Terjemahan database hanya 350 kata.
8. Aplikasi ini dapat berjalan pada sistem operasi android versi 2.3 sampai 4.0

1.5 Metode Penelitian

Metode yang digunakan dalam pembahasan ini adalah:

- 1 Studi Literatur
Pengumpulan data dilakukan dengan mencari bahan-bahan referensi dari berbagai sumber sebagai landasan teori yang berhubungan dengan permasalahan yang dijadikan objek penelitian.
 - 2 Analisa Kebutuhan Sistem
Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan suatu kerangka yang digunakan untuk acuan perancangan sistem Pembuatan program.
-

3 Eksperimen dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, berdasarkan fungsional program dan akan dilakukan koreksi.

1.6 Sistematika Penulisan

Dalam penyusunan skripsi ini agar lebih mudah dipahami maka dibuatlah suatu sistematika penulisan sebagai berikut:

BAB I : PENDAHULUAN

Berisi latar belakang, perumusan masalah, batasan masalah, tujuan dan metodologi penelitian.

BAB II : TINJAUAN PUSTAKA

Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan program yang akan dikerjakan

BAB III : PERANCANGAN SISTEM

Berisi tentang analisis dan perancangan system aplikasi.

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Berisi tentang implementasi dan ujicoba dari aplikasi.

BAB V : PENUTUP

Berisi tentang kesimpulan dan saran-saran yang digunakan untuk pengembangan program selanjutnya

BAB II

LANDASAN TEORI

2.1 ANDROID

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

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

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



Gambar 2.1 lambang android

2.1.1 Versi – versi android

Berikut ini adalah versi – versi sistem operasi android yang beredar hingga sekarang:

1. Android versi 1.0

Pertama kali dirilis pada 23 Spetember 2008. Sebenarnya Android versi pertama ini akan dinamai dengan nama "Astro" tapi karena alasan hak cipta dan trademark nama "Astro" tidak jadi disematkan pada versi pertama dari OS Android ini. HTC Dream adalah ponsel pertama yang menggunakan OS ini. Pada android versi pertama ini sudah mulai diperkenalkan Android Market, tempat untuk mendownload berbagai macam aplikasi dan update untuk Android. Kemudian terdapat fitur *synchronisasi* dengan berbagai layanan Google lainnya seperti *Google Contact, Calender, Maps, Talk*, dan lainnya.



Gambar 2.2 HTC dream menggunakan os android versi 1.0

2. Android 1.1 (Bender)

Pertama kali dirilis pada 9 Februari 2009. Versi Android kedua ini juga mengalami masalah penamaan yang sama dengan versi pertamanya. Pada awalnya Android ini akan diberi nama "Bender" akan tetapi karena alasan melanggar trademark, nama "Bender" tidak jadi disematkan pada versi Android ini. Awalnya versi OS Android ini dirilis untuk perangkat T-Mobile G1 saja. Versi ini merupakan update untuk memperbaiki beberapa bugs, mengganti API dan menambahkan beberapa fitur. Android versi ini dilengkapi dengan pembaharuan *Estetis* pada aplikasi *Jam alarm*, *Voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, pemberitahuan *email*. Berikut penampakan home screen OS Android v.1.1



Gambar 2.3 Home screen android versi 1.1 (Bender)

3 Android versi 1.5 (Cupcake)

Pertama kali dirilis pada 30 April 2009. Nah, mulai versi Android ini penamaan menggunakan nama makan pencuci mulut (dessert) mulai digunakan, karena ini merupakan versi yang ketiga maka penamaan diawali dengan huruf "C"

dan jadilah "Cupcake" menjadi nama resmi dari versi OS Android ketiga ini. OS ini berbasiskan pada kernel Linux 2.6.27 dan menambahkan beberapa update serta UI baru dari versi Android sebelumnya. Mulai terdapat "widget" yang dapat dibesar kecilkan. Kemudian ditambah kemampuan untuk meng-upload *video* dan gambar ke *Youtube* dan *Picasa*. Berikut tampilan Home screen android versi 1.5 dan logo dari android versi 1.5



Gambar 2.4 tampilan homescreen android versi 1.5



gambar 2.5 logo android versi 1.5

4 Android versi 1.6 (Donut)

Dirilis pertama kali pada 15 September 2009. Terdapat peningkatan pada fitur pencarian dan UI yang lebih user friendly. Pada versi ini juga sudah mendukung

teknologi *CDMA/EVDO*, *802.1x*, *VPNs*. Kemudian support layar dengan resolusi *WVGA*. Berikut penampakan Android v1.6 Donut. Tampilan yang sedikit lebih rapi. Berikut tampilan Home screen android versi 1.6 dan logo dari android versi 1.6.



Gambar 2.6 Tampilan homescreen android versi 1.6



gambar 2.7 logo android versi 1.6

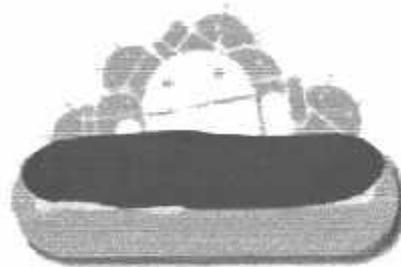
5 Android versi 2.0/2.1 (Éclair)

Dirilis pertama kali pada 9 Desember 2009. Terjadi penambahan fitur untuk pengoptimalan hardware, peningkatan *Google Maps 3.1.2*, perubahan *UI* dengan *browser* baru dan dukungan *HTML5*, daftar kontak yang baru, *dukungan flash* untuk kamera *3,2 MP*, *digital Zoom*, dan *Bluetooth 2.1*. Beberapa versi updatenya

antara *Android v.2.0* kemudian *v2.0.2* dan terakhir *v.2.1*. Berikut tampilan homescreen android versi 2.0/2.1 dan logonya.



Gambar 2.8 tampilan homescreen android versi 2.0/2.1



gambar 2.9 logo android versi 2.0/2.1

6 Android versi 2.2 (Frozen Yoghurt (Froyo))

Dirilis pertamakali pada 20 Mei 2010 pada smartphone Google Nexus One. Pada versi ini sudah support terhadap Adobe Flash Player 10.1. Peningkatan pada kecepatan membuka dan menutup aplikasi, serta penggunaan SD Card sebagai tempat penyimpanan aplikasi. Ketika Android Froyo hadir mulai muncul banyak diskusi

yang membahas mengenai persaingan antara Android dengan iOS yang akan semakin ketat di masa yang akan datang. Beberapa versi update yang dirilis antara lain Android v.2.2.1 hingga v.2.2.3. berikut tampilan homescreen dari android versi 2.2 dan gambar logonya.



Gambar 2.10 tampilan homescreen android versi 2.2



Gambar 2.11 logo dari android 2.2

7 Android versi 2.3 (Gingerbread)

Pertama kali diperkenalkan pada 6 Desember 2010. Terjadi banyak peningkatan pada versi Android yang satu ini dibandingkan dengan versi sebelumnya. Dirancang untuk memaksimalakan kemampuan aplikasi dan game. Serta mulai digunakannya *Near Field Communication (NFC)*. Perbaikan terhadap dukungan layar resolusi *WXGA* dan di atasnya. Beberapa versi update yang dirilis

antara lain v.2.3.3 hingga v.2.3.7. Sampai saat ini Android Gingerbread merupakan versi Android yang memiliki pengguna terbanyak dibandingkan dengan seri Android lainnya, yaitu mencapai 65% dari seluruh versi Android yang dirilis. Berikut tampilan homescreen android versi 2.3 dan logonya.



Gambar 2.12 tampilan utama android versi 2.3



Gambar 2.13 logo dari android versi 2.3 (Gingerbread)

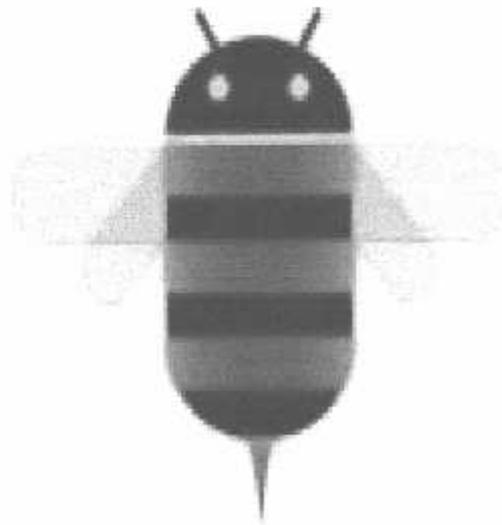
8 Android versi 3.0 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama

yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Berikut tampilan android versi 3.0 dan logonya.



Gambar 2.14 tampilan utama android versi 3.0



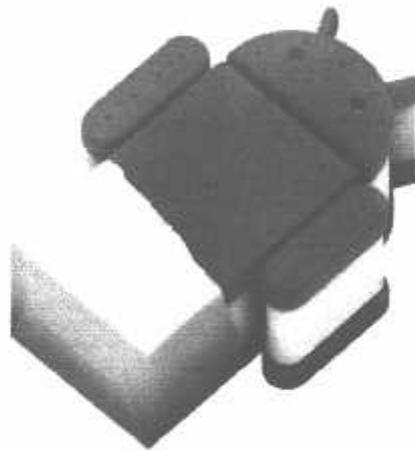
Gambar 2.15 logo android versi honeycomb

9 Android versi 4.0 (ICS: Ice Cream Sandwich)

Diumumkan pada tanggal 19 Oktober 2011, membawa fitur Honeycomb untuk smartphone dan menambahkan fitur baru termasuk *membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan menggunakan NFC*. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus. Berikut tampilan dari android versi 4.0 dan logonya.



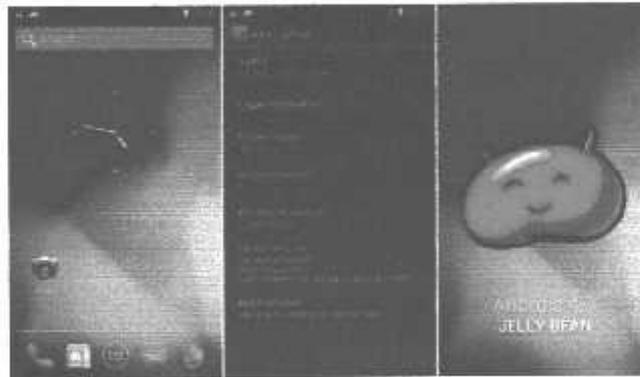
Gambar 2.16 tampilan utama dari android versi 4.0



Gambar 2.17 logo Ice Cream Sandwich

10 Android versi 4.1 (Jelly Bean)

Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input keyboard, desain baru fitur pencarian, *UI* yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Tak ketinggalan Google Now juga menjadi bagian yang diperbarui. *Google Now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7. Berikut tampilan android versi 4.1 dan logonya.



Gambar 2.18 tampilan utama android versi 4.1



Gambar 2.19 logo jelly bean

11 Android versi 4.2 (A New Flavor of Jelly Bean)

Ini merupakan versi android yang paling terbaru yang beredar. Fitur photo sphere untuk panorama, daydream sebagai screensaver, power control, lock screen widget, menjalankan banyak user (dalam tablet saja), widget terbaru. Android 4.2 Pertama kali dikenalkan melalui LG Google Nexus 4.. berikut tampilan LG Google Nexus 4



2.1.2 Karakteristik Android

Android memiliki karakteristik sebagai berikut:

1. Terbuka

Android dibangun untuk benar-benar terbuka sehingga sebuah aplikasi dapat memanggil salah satu fungsi inti perangkat seperti membuat panggilan, mengirim pesan teks, menggunakan kamera, dan lain-lain.

2. Semua aplikasi dibuat sama

Android tidak memberikan perbedaan terhadap aplikasi utama dari perangkat dan aplikasi yang dikembangkan oleh *user*. Semua aplikasi dibangun untuk memiliki akses yang sama terhadap kemampuan perangkat dalam menyediakan layanan dan aplikasi yang luas terhadap para *user*.

3. Pengembangan aplikasi yang cepat dan mudah

Lapisan selanjutnya adalah application framework, yang mencakup program untuk mengatur fungsi-fungsi dasar smartphone.

Application framework merupakan serangkaian tool dasar seperti alokasi resource smartphone, aplikasi telepon, pergantian antar-proses atau program, dan pelacakan lokasi fisik telepon. Para pengembang aplikasi memiliki akses penuh kepada tool-tool dasar tersebut, dan memanfaatkannya untuk menciptakan aplikasi yang lebih kompleks.

Dilapisan teratas bercokol aplikasi itu sendiri. Dilapisan inilah pengguna menemukan fungsi-fungsi dasar smartphone, seperti menelepon dan mengirim pesan singkat, menjalankan web browser, mengakses daftar kontak, dan lain-lain. Bagi rata-rata pengguna, lapisan inilah yang paling sering mereka akses. Mereka mengakses fungsi-fungsi dasar tersebut melalui user interface.

Ada 4 hal yang harus dipahami dalam membangun aplikasi berbasis Android :

Activity adalah tampilan grafis yang kita lihat ketika menjalankan sebuah aplikasi. Sebuah aplikasi dapat mempunyai lebih dari satu activity.

Intent adalah serangkaian value yang menunjukkan apa yang harus dilakukan ketika terjadi perpindahan layar. Service adalah layanan yang bekerja di Backend. Content provider memungkinkan sebuah aplikasi untuk dapat menyimpan dan menerima data dari database.



Gambar 2.21 arsitektur android

Berikut penjelasan dgambar stuktur arsitektur android:

Dalam paket sistem operasi android terdiri dari beberapa unsur seperti tampak pada gambar di bawah. Secara sederhana arsitektur android merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu framework yang menyediakan dan mengatur alur proses aplikasi.

1. Linux Kernel

Android dibangun di atas kernel Linux 2.6. Namun secara keseluruhan android bukanlah linux, karena dalam android tidak terdapat paket standar yang dimiliki oleh linux lainnya. Linux merupakan sistem operasi terbuka yang handal dalam manajemen memori dan proses. Oleh karenanya pada android hanya terdapat beberapa servis yang diperlukan seperti keamanan, manajemen memori, manajemen proses, jaringan dan driver. Kernel linux menyediakan driver layar, kamera, keypad, WiFi, Flash Memory, audio, dan IPC (Interprocess Communication) untuk mengatur aplikasi dan lubang keamanan.

2. Libraries

Android menggunakan beberapa paket pustaka yang terdapat pada C/C++ dengan standar Berkeley Software Distribution (BSD) hanya setengah dari yang aslinya untuk tertanam pada kernel Linux. Beberapa pustaka diantaranya:

1. Media Library untuk memutar dan merekam berbagai macam format audio dan video.
2. Surface Manager untuk mengatur hak akses layer dari berbagai aplikasi.
3. Graphic Library termasuk didalamnya SGL dan OpenGL, untuk tampilan 2D dan 3D.
4. SQLite untuk mengatur relasi database yang digunakan pada aplikasi.
5. SSI dan WebKit untuk browser dan keamanan internet.

Pustaka-pustaka tersebut bukanlah aplikasi yang berjalan sendiri, namun hanya dapat digunakan oleh program yang berada di level atasnya. Sejak versi

Android 1.5, pengembang dapat membuat dan menggunakan pustaka sendiri menggunakan Native Development Toolkit (NDK).

3. Android Runtime

Pada android tertanam paket pustaka inti yang menyediakan sebagian besar fungsi android. Inilah yang membedakan Android dibandingkan dengan sistem operasi lain yang juga mengimplementasikan Linux. Android Runtime merupakan mesin virtual yang membuat aplikasi android menjadi lebih tangguh dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat 2 bagian utama, diantaranya:

1. Pustaka Inti, android dikembangkan melalui bahasa pemrograman Java, tapi Android Runtime bukanlah mesin virtual Java. Pustaka inti android menyediakan hampir semua fungsi yang terdapat pada pustaka Java serta beberapa pustaka khusus android.
2. Mesin Virtual Dalvik, Dalvik merupakan sebuah mesin virtual yang dikembangkan oleh Dan Bornstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. Dalvik hanyalah interpreter mesin virtual yang mengeksekusi file dalam format Dalvik Executable (*.dex). Dengan format ini Dalvik akan mengoptimalkan efisiensi penyimpanan dan pengalamanan memori pada file yang dieksekusi. Dalvik berjalan di atas kernel Linux 2.6, dengan fungsi dasar seperti threading dan manajemen memori yang terbatas. [Nicolas Gramlich, *Andbook*, anddev.org]

4. Application Framework

Kerangka aplikasi menyediakan kelas-kelas yang dapat digunakan untuk mengembangkan aplikasi android. Selain itu, juga menyediakan abstraksi generik untuk mengakses perangkat, serta mengatur tampilan user interface dan sumber daya aplikasi. Bagian terpenting dalam kerangka aplikasi android adalah sebagai berikut [Hello Android 2nd Edition]:

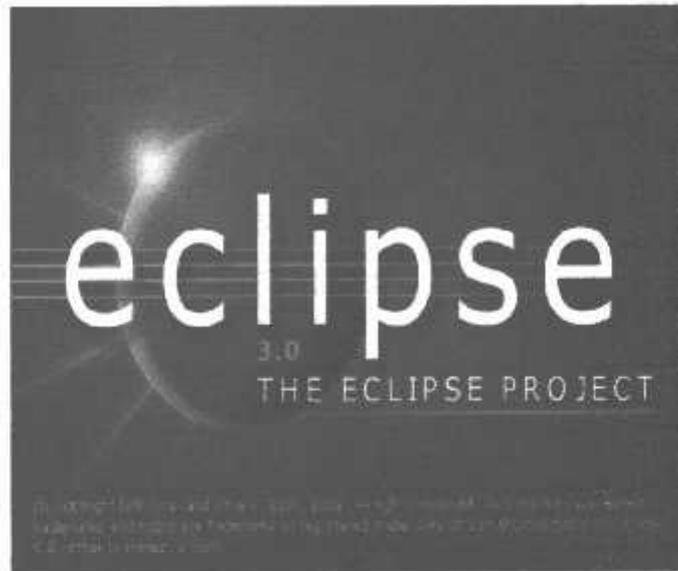
1. Activity Manager, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan "Backstack" untuk navigasi penggunaan.
2. Content Providers, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
3. Resource Manager, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan file layout.
4. Location Manager, berfungsi untuk memberikan informasi detail mengenai lokasi perangkat android berada.
5. Notification Manager, mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada status bar.

5. Application Layer

Puncak dari diagram arsitektur android adalah lapisan aplikasi dan widget. Lapisan aplikasi merupakan lapisan yang paling tampak pada pengguna ketika menjalankan program. Pengguna hanya akan melihat program ketika digunakan tanpa mengetahui proses yang terjadi dibalik lapisan aplikasi. Lapisan ini berjalan dalam Android runtime dengan menggunakan kelas dan service yang tersedia pada framework aplikasi.

Lapisan aplikasi android sangat berbeda dibandingkan dengan sistem operasi lainnya. Pada android semua aplikasi, baik aplikasi inti (native) maupun aplikasi pihak ketiga berjalan diatas lapisan aplikasi dengan menggunakan pustaka API (Application Programming Interface) yang sama.

2.2 Eclipse



Gambar 2.22 logo perangkat lunak eclipse

Eclipse adalah sebuah *Integrated Development Environment* atau IDE yang Bertujuan untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform (platform-independent)*.

Berikut ini adalah sifat dari Eclipse:

- 1 Multi-platform:** Eclipse dapat digunakan dalam berbagai sistem operasi antara lain Microsoft Windows, Linux, Solaris, AIX, HP-UX dan, Mac OS X.
- 2 Multi-role:** Digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

3 Multilanguage: eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, PHP, dan lain sebagainya.

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

Untuk keperluan programing eclipse menyediakan database secara default yang sudah ada di *library* yang disebut SQLite.

2.3 Database

Database atau basisdata adalah kumpulan informasi yang disimpan di dalam computer secara sistematis sehingga dapat diperiksa menggunakan suatu program computer untuk memperoleh informasi dari database tersebut.

2.3.1 Database SQLite

Database SQLite merupakan paket *software* yang bersifat *public domain* yang menyediakan sistem manajemen basis data relasional atau *Relational Database Management System* (RDBMS). Kata "Lite" pada SQLite tidak menunjuk pada kemampuannya, melainkan menunjuk pada sifat dari SQLite, yaitu ringan ketika dihubungkan dengan kompleksitas pengaturan, *administrative overhead*, dan pemakaian sumber. SQLite tidak memiliki *username* atau *login*. SQLite memiliki fitur-fitur sebagai berikut:

1. Tidak memerlukan *server*
 2. *Single File Database*
-

SQLite mengemas seluruh *database* ke dalam suatu *single file* yang berisi *layout database* dan data aktual yang berada pada tabel dan indeks untuk memudahkan *user* membuat, menyalin, ataupun mem-*backup database*.

3. *Zero Configuration*

SQLite tidak membutuhkan apapun untuk melakukan instalasi dan konfigurasi.

4. *Embedded Device Support*

Ukuran *code* dari SQLite bersifat kecil membuatnya cocok digunakan untuk *embedded system* yang berjalan terbatas pada sistem operasi.

5. Fitur-fitur yang unik

SQLite memungkinkan *user* untuk memasukkan nilai ke dalam kolom tanpa memperhatikan tipe data. Selain itu dapat memanipulasi lebih dari satu *database* pada satu waktu.

6. *Compatible License*

SQLite dan SQLite *code* tidak memiliki lisensi dan tidak dilindungi oleh *GNU's Not Unix (GNU) General Public License (GPL)* atau lisensi *open source* sejenisnya. Hal ini berarti *user* dapat memodifikasinya dengan berbagai cara dan mendistribusikan dengan berbagai cara

2.4 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai 24opular24 termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang

bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini java merupakan bahasa pemrograman yang paling 25opular digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web dan icon Java dapat dilihat pada gambar 2.2.



Gambar 2.2

2.4.1 Java API

Java API terdiri dari tiga bagian utama:

1. *Java Standart Edition (SE)*, sebuah standar API untuk merancang aplikasi dekstop dan *applets* dengan bahas dasar grafis.
2. *Java Eterprose Edition (EE)*, sebuah API yang dirancang untuk aplikasi server dengan mendukung basis data/database

Java Macro Edition (ME), sebuah API untuk merancang aplikasi yang berjalan pada alat kecil seperti telepon genggam, komputer, dan pager.

2.4.2 Java Virtual Machine

Java Virtual Machine (JVM) adalah spesifikasi untuk sebuah kompuer anstrak yang terdiri dari sebuah kelas pemanggil dan sebuah interpreter.

Java yang mengeksekusi kode arsitektur netral. Kelas pemanggil memanggil `file.class` dari kedua program Java dan API yang dieksekusi oleh interpreter Java. Interpreter Java mungkin sebuah perangkat lunak yang menerjemahkan satu kode byte pada satu waktu, atau mungkin sejumlah *just-intime* (JIT) kompiler yang menurunkan *bytecode* arsitektur netral ke dalam bahasa mesin untuk *host* komputer.

2.4.3 Sistem Operasi Java

Sistem operasi biasanya ditulis dalam sebuah kombinasi kode bahasa C dan assembly, terutama disebabkan oleh kelebihan performa dari bahasa tersebut dan memudahkan untuk berinteraksi dengan perangkat keras.

Satu kesulitan dalam merancang sistem basis bahasa adalah dalam hal proteksi memori, yaitu memproteksi sistem operasi dari pemakai program yang sengaja memproteksi pemakai program lainnya. Sistem operasi tradisional mengharapkan pada tampilan perangkat keras untuk menyediakan proteksi memori. Sistem basis bahasa mengandalkan pada tampilan keamanan dari bahasa, sebagai hasilnya sistem basis bahasa menginginkan pada saat perangkat keras kecil, yang memungkinkan kekurangan tampilan perangkat keras yang menyediakan proteksi memori.

2.4.4 Dasar Pemrograman Java

Java2 adalah generasi kedua dari Java *platform* (generasi awalnya adalah Java Development), dengan kata lain Java berdiri di atas sebuah mesin interpreter yang diberi nama JVM.

Platform Java terdiri *library*, kelas-kelas *loader* yang dipaket dalam sebuah lingkungan rutin Java, dan sebuah *compiler*, *debugger*, dan perangkat yang lain yang dipaket dalam Java Development Kit (JDK). Agar sebuah program Java dapat dijalankan maka ekstensi “java” harus dikompilasi menjadi file *bytecode* yang dibutuhkan JRE (*Java Runtime Environment*) yang memungkinkan program Java, hanya menjalankan dan tidak membuat kode baru lagi.

2.5 Algoritma pencarian kata

Algoritma pencarian string atau sering disebut juga pencocokan string adalah algoritma untuk melakukan semua kemunculan string pendek.

Pencocokan string merupakan permasalahan paling sederhana dari semua permasalahan string lainnya, dan dianggap sebagai bagian dari pemrosesan data, pengkompresian data, dan temu balik informasi. Teknik untuk menyelesaikan permasalahan pencocokan string biasanya akan menghasilkan implikasi langsung ke aplikasi string lainnya.

Algoritma pencocokan string dikategorikan menjadi tiga bagian menurut arah pencariannya, tiga kategori tersebut adalah :

1. Dari arah yang paling alami, dari kiri ke kanan, yang merupakan arah untuk membaca, algoritma yang termasuk kategori ini adalah:
 2. Algoritma Brute Force.
 1. Algoritma dari Morris dan Pratt, yang kemudian dikembangkan oleh Knuth, Morris, dan Pratt.
 2. Dari kanan ke kiri, arah yang biasanya menghasilkan hasil terbaik secara praktikal, contohnya adalah :

Algoritma Boyer dan Moore, yang memungkinkan banyak menjadi algoritma turbo Boyer-Moore, algoritma tuned Boyer-Moore, dan algoritma Zhu-takaoka.
 3. Dan kategori terakhir, dari arah yang ditentukan secara spesifik oleh algoritma tersebut, arah ini menghasilkan hasil terbaik secara teoritis, algoritma yang termasuk kategori ini adalah:
 - 1 Algoritma Colussi
 - 2 Algoritma Crochemore-Perrin.
-

2.5.1 Algoritma Knuth-Morris-Pratt

Secara sistematis, langkah-langkah yang dilakukan algoritma Knuth-Morris-Pratt pada saat mencocokkan string yang ditunjukkan pada gambar2.3:

1. Algoritma Knuth-Morris-Pratt mulai mencocokkan pola pada awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter pola dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi:
 - 1 Karakter di pola dan di teks yang dibandingkan tidak cocok (mismatch).
 - 2 Semua karakter di pola cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini
3. Algoritma kemudian menggeser pola berdasarkan tabel teks, lalu mengulangi langkah 2 sampai pola berada di ujung teks.

```

procedure KMPSearch(
  input m, n : integer,
  input P : array[0..n-1] of char,
  input T : array[0..m-1] of char,
  output ketemu : array[0..m-1] of boolean
)
Deklarasi:
i, j, next: integer
kmpNext : array[0..n] of integer
Algoritma:
preKMP(n, P, kmpNext)
i:=0
while (i<= m-n) do
  j:=0
  while (j < n and T[i+j] = P[j]) do
    j:=j+1
  endwhile
  if(j >= n) then
    ketemu[i]:-true;
  endif
  next:= j - kmpNext[j]
  i:= i+next
endwhile

```

gambar2.3

III PERANCANGAN SISTEM

Bab ini berisi tentang perancangan dari sistem yang dibuat. Perancangan ini meliputi : pokok-pokok bahasan dari perancangan, baik secara perangkat lunak atau perangkat keras yang dibuat.

3.1 Spesifikasi Aplikasi

Disesuaikan dengan tujuan awal aplikasi, spesifikasi aplikasi ini adalah sebagai berikut:

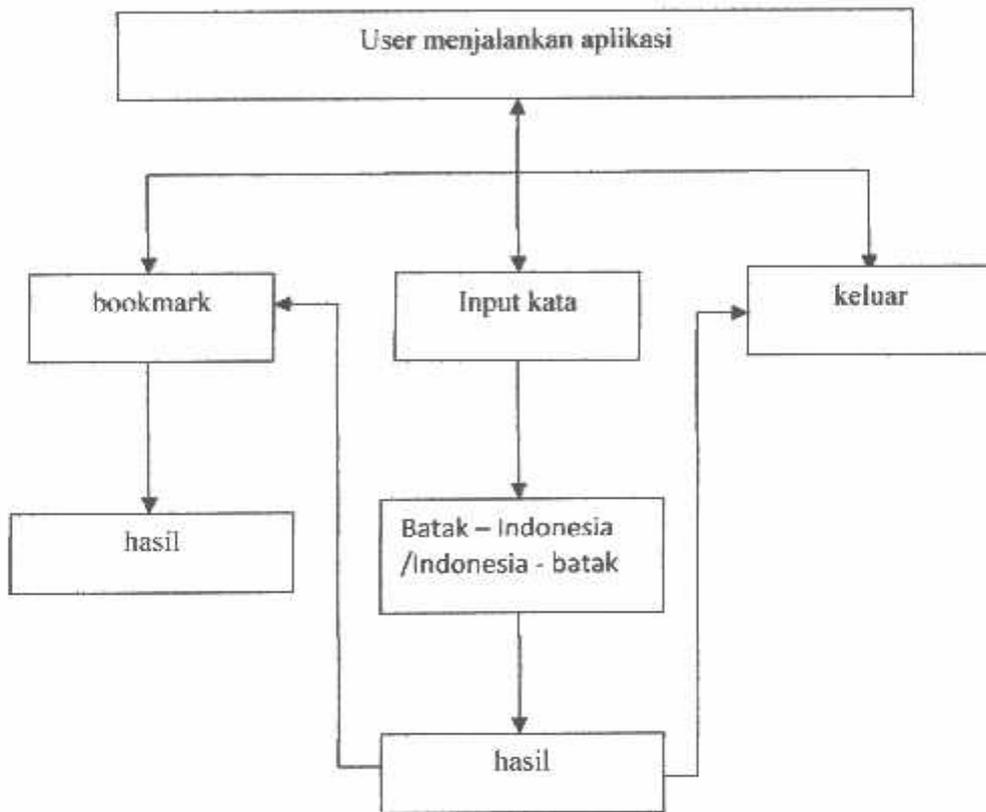
- 1 Pembuatan aplikasi dibuat dengan menggunakan bahasa Java. Bahasa pemrograman Java digunakan untuk mendukung pembuatan aplikasi kamus bahasa Indonesia-Batak berbasis android menggunakan algoritma Knuth-Morris-Pratt .
- 2 IDE yang digunakan adalah Eclipse Helios Perangkat Lunak digunakan dalam pembuatan aplikasi android
- 3 Aplikasi berjalan pada Android versi 2.3 (Gingerbread) sampai ke versi ke 4.0 (ICS)
- 4 Aplikasi kamus ini berjalan pada Android versi 2.3 sampai ke 4.0 karena para developer untuk mobile lebih banyak menggunakan versi – versi tersebut untuk handset yang mereka keluarkan.
- 5 Aplikasi tidak terhubung dengan internet.

Aplikasi kamus ini hanya menggunakan *database* dikarenakan data – data yang diinputkan hanya bersifat kata dan database internal saja cukup memungkinkan untuk melakukan penyimpanan data.

3.2 Perancangan Aplikasi

3.2.1 Block Diagram Sistem

Block diagram pada aplikasi ini terdapat pada gambar merupakan struktur yang berfungsi sebagai pencabangan untuk menampilkan setiap tampilan aplikasi kamus.

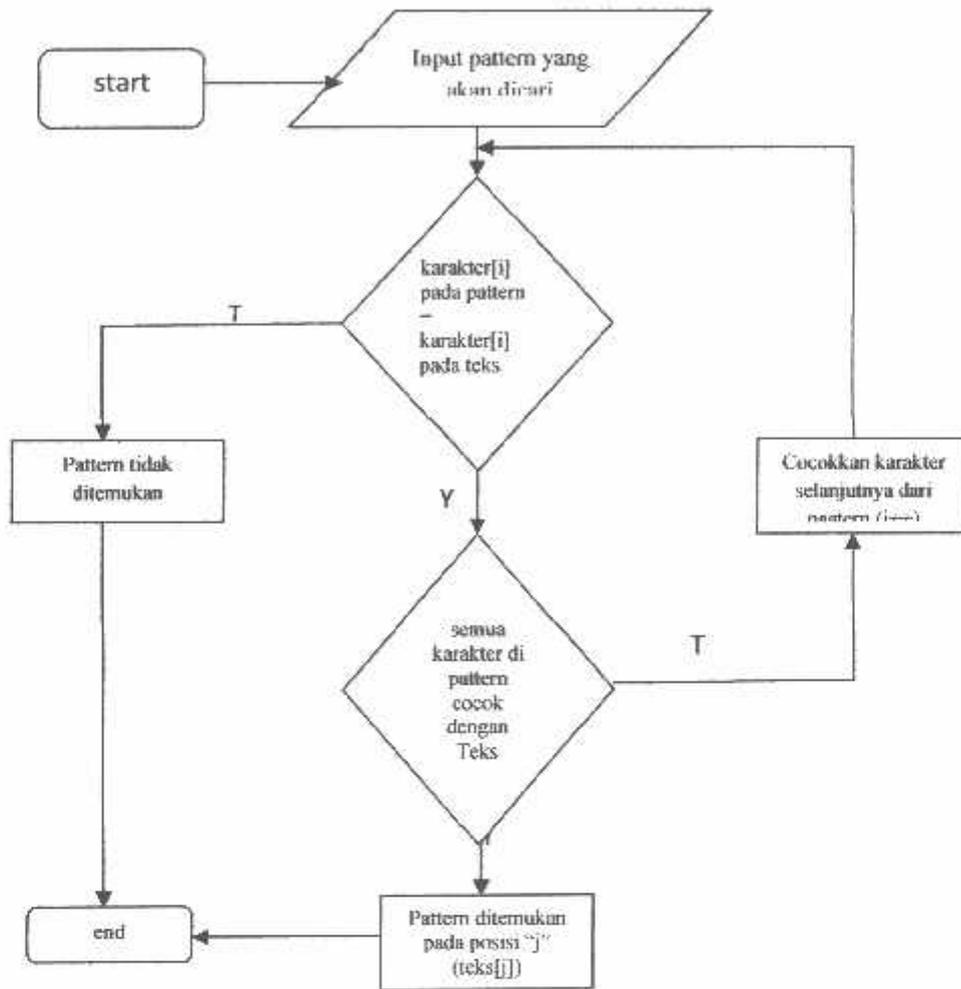


Gambar 2.4

3.2.2 Flowchart Sistem

Untuk memperjelas alur dari sistem, maka akan digambarkan dalam bentuk *flowchart*. *Flowchart* menjelaskan rancangan urutan proses yang terjadi pada aplikasi.

dimulai dari proses pemasukan data, akses *database*, maupun proses lainnya hingga *output*. *Flowchart* proses aplikasi ditunjukkan dalam Gambar 24.



Gambar 2.5 Flowchart sistem

Penjelasan dari alur sistem flowchart adalah sebagai berikut:

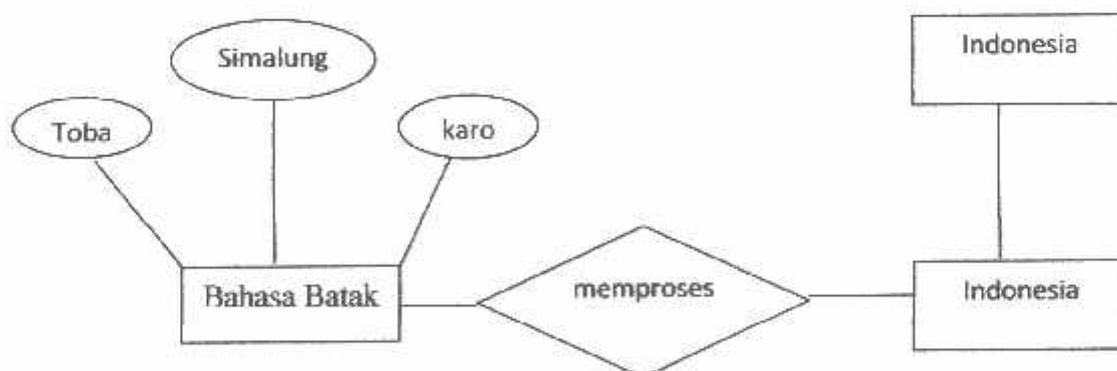
1. User atau pengguna menginputkan kata yang akan di cari artiannya.
2. Selanjutnya kata yang diinputkan oleh user tadi akan dicocokkan ke database, dalam hal ini proses pencocokan kata dilakukan menggunakan algoritma KNUTH-MORRIS-PRATT.

Keterangan:

- 1) User menginputkan kata yang akan di cari
- 2) Sistem akan menerima respon dari user dan akan melakukan proses.
- 3) Setelah berada pada bagian prosce maka akan diteruskan menuju database kamus untuk mencocokkan database dan *outputan* yang akan dihasilkan.
- 4) Setelah itu kembali akan menuju ke bagian proses pencarian data.
- 5) Kemudian diteruskan menuju user guna mengeluarkan *output* yang telah diproses.

3.2.4 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah sebuah konsep yang mendeskripsikan hubungan antara penyimpanan dan didasarkan pada persepsi dari sebuah dunia nyata yang terdiri dari sekumpulan objek, disebut relasi diantar objek-objek tersebut. Fungsi dari ERD dalam aplikasi kamus Batak-Indonesia adalah untuk memodelkan struktur data dan hubungan antar data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.



Gambar 2.6 DFD level 0

3.2.5 Perancangan Tabel Database

Berikut perancangan tabel – tabel yang digunakan pada aplikasi, pada tabel 3.1 merupakan tabel dari database aplikasi dimana semua data - data tersimpan di database internal. Tabel 3.2 merupakan tabel bookmark, tabel bookmark ini berisi data outputan yang telah disimpan oleh user untuk memudahkan jika kemudian hari user tidak ingin repot menginputkan data lagi.

Tabel 3.1 tabel database aplikasi

Field	Type	Length	keterangan
_id	Text	20	Primary
Indonesia - Batak	Text	100	
Batak – Indonesia	Text	100	
Terjemahan	Text	100	

Tabel 3.2 tabel bookmark aplikasi

Field	Type	Length	Keterangan
_id	Text	20	Primary
Hasil	Text	100	
Metode	Text	100	

3.2.3 Perancangan Form Aplikasi

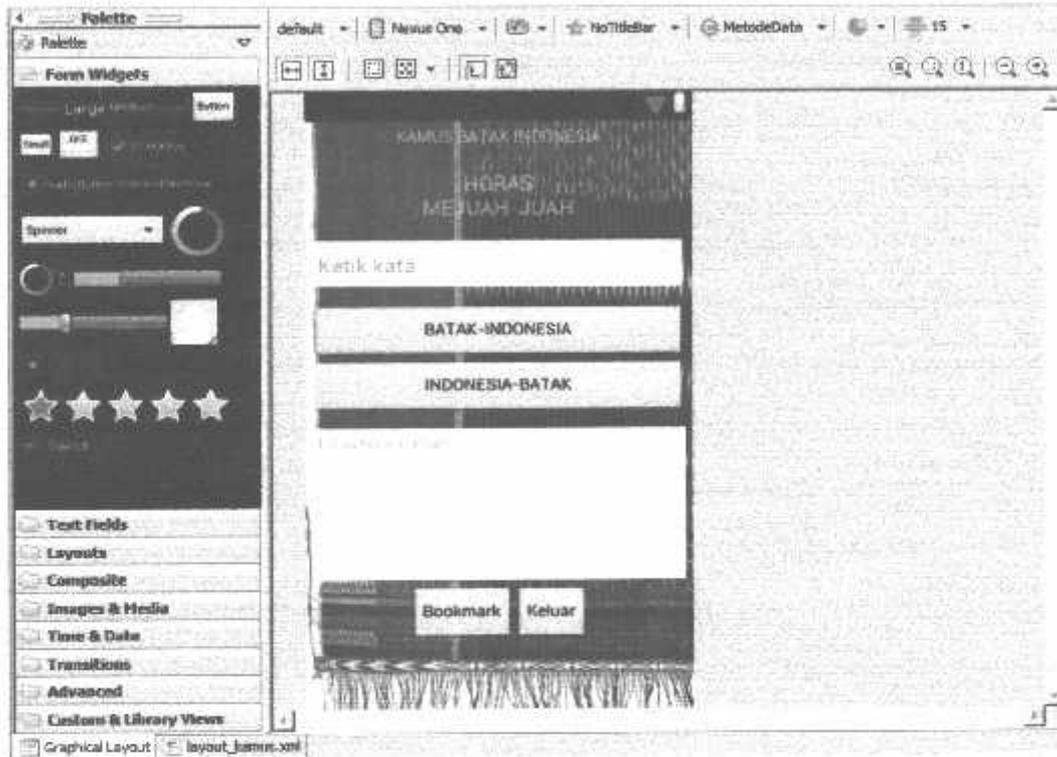
Berikut desain tampilan awal dari aplikasi, terdapat 6 buah menu pada gambaran awal aplikasi. Gambaran awal aplikasi bisa dilihat pada gambar

KAMUS BATAK INDONESIA	
HORAS MEJUAH-JUAH	
1	
2	
3	
4	
5	6

Gambar 3.2 perancangan form aplikasi

Keterangan: Dalam gambar struktur menu aplikasi kamus INDONESIA – BATAK ini terdapat 2 menu yaitu nomor 1 dan 4, nomor 1 sebagai tempat user untuk menginputkan kata yang ingin dicari sedangkan pada nomor 4 berfungsi sebagai outputan yang telah diproses oleh sistem. Nomor 2 sebagai button untuk menerjemahkan kata dari bahasa BATAK ke bahasa INDONESIA, dan untuk menerjemahkan kata dari bahasa INDONESIA ke bahasa BATAK buttonnya ada pada nomor 3. Untuk nomor 5 merupakan fitur bookmark dari hasil pencarian yang telah dilakukan oleh sistem. Untuk nomor 6 adalah button tombol keluar apabila user sudah selesai menggunakan aplikasi kamus INDONESIA – BATAK ini.

Berikut tampilan pembuatan form aplikasi menggunakan perangkat lunak eclipse maka tampilan dapat dilihat pada gambar



Gambar 3.3 perancangan form aplikasi pada celpipse

3.3 Analisa Sistem

3.3.1 Analisa Perangkat Keras (*Hardware*)

Pembuatan aplikasi kamus bahasa Indonesia – Batak ini menggunakan hardware antara lain:

1. Laptop Acer Aspire 4535
 - AMD Turion X2 RM-74 (2,2 GHz 1MB L2 cache)
 - ATI Radeon HD 3200 Graphics
 - 14" HD Acer CineCrystal LED LCD

- 250 HDD
2. Modem Venus VT-18
 3. Sony Ericsson Xperia X10i

Tabel 3.1 spesifikasi sony ericsson x10i

General	<i>2G Network</i>	<i>GSM 850 / 900 / 1800 / 1900</i>
	<i>3G Network</i>	<i>HSDPA 900 / 2100</i>
Display	<i>Type</i>	<i>TFT capacitive touchscreen, 65K colors</i>
	<i>Size</i>	<i>480 x 854 pixels, 4.0 inches (~245 ppi pixel density)</i>
Memory	<i>Internal</i>	<i>1 GB storage, 384 MB RAM</i>
	<i>Eksternal</i>	<i>microSD, up to 16GB</i>
Data	<i>GPRS</i>	<i>85.6 kbps</i>
	<i>EDGE</i>	<i>236.8 kbps</i>
	<i>Speed</i>	<i>HSDPA, 7.2 Mbps; HSUPA, 2 Mbps</i>
	<i>WLAN</i>	<i>Wi-Fi 802.11 b/g</i>
	<i>USB</i>	<i>microUSB v2.0</i>
Features	<i>OS</i>	<i>Android OS, v2.3 (Gingerbread)</i>
	<i>Chipset</i>	<i>Qualcomm QSD8250 Snapdragon</i>
	<i>CPU</i>	<i>1 GHz Scorpion</i>
	<i>GPU</i>	<i>Adreno 200</i>
	<i>GPS</i>	<i>Ada</i>
	<i>Java</i>	<i>Java MIDP emulator</i>
Battery	<i>Type</i>	<i>Li-Po 1500 mAh battery (BST-41)</i>
	<i>Stand-by</i>	<i>Up to 415 h (2G) / Up to 425 h (3G)</i>

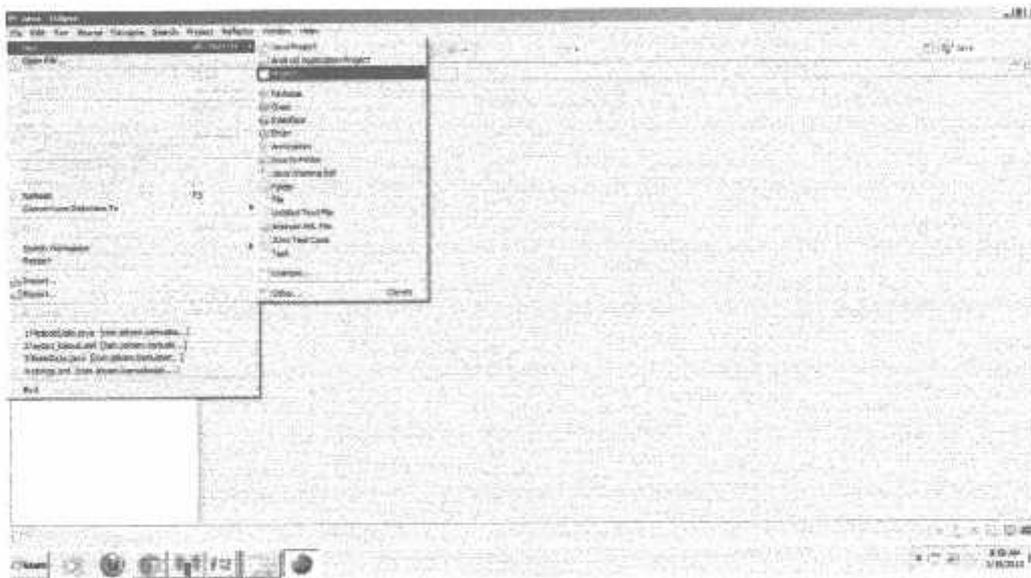
3.3.2 Analisa Perangkat Lunak (Software)

Selain *hardware*, dalam pembuatan aplikasi diperlukan beberapa *software* pendukung, yaitu:

1. Sistem Operasi Windows 7 Ultimate 32-bit
2. Eclipse Helios
3. JDK 6 Windows i586

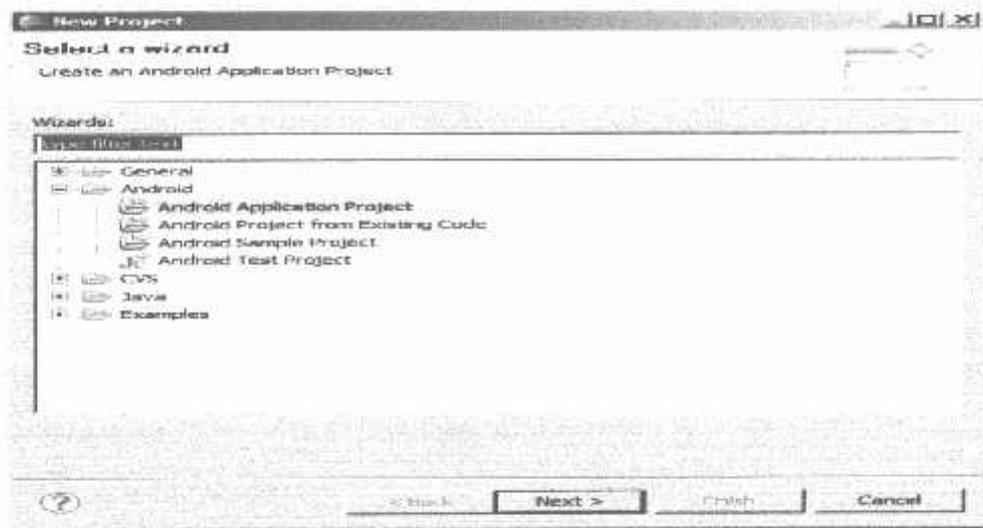
3.4 Memulai Pembuatan Project dengan Eclipse

Pembuatan Aplikasi ini dimulai dengan cara membuat lembar kerja baru pada Perangkat Lunak Eclipse dengan cara *klik menu File -> new -> project*. Tampilan proses tersebut dapat dilihat pada gambar 3.2.



Gambar 3.2 memulai project dengan eclipse

Setelah memilih *new project* kemudian akan muncul kotak dialog *New*, dan pilihlah *File -> Android Project* seperti pada gambar 3.3 dibawah ini.



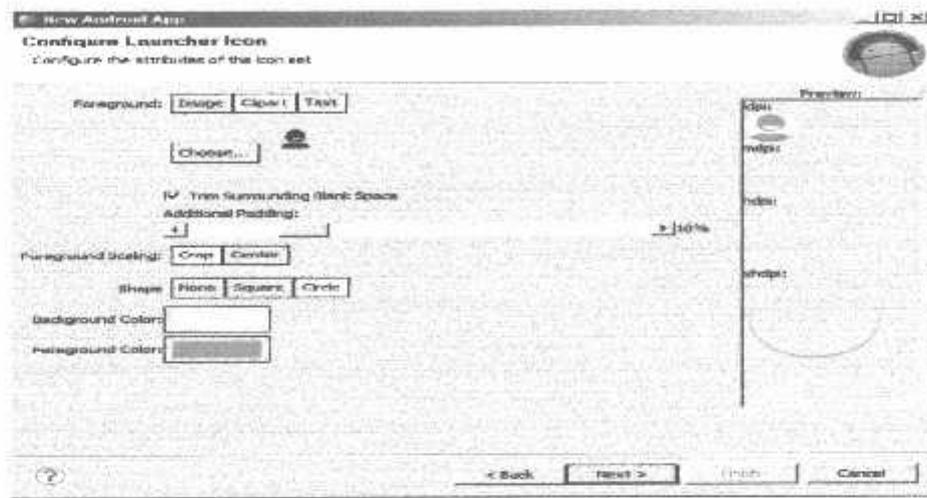
Gambar 3.3 new project pada eclipse

Lalu akan muncul tampilan seperti tampak pada gambar 3.4 yaitu *form New Android Application* dimana bagian ini berisi nama aplikasi baru, *project* baru dan SDK yang akan digunakan setelah diisi kemudian pilih *next*.



Gambar 3.4 form new android application

Proses selanjutnya akan tampak pada gambar 3.5 yaitu *form New Android App* dimana bagian ini berisi tentang konfigurasi atribut ,skala ukuran, ikon dll. Setelah itu dilanjutkan dengan *form Create Activity* dan *form New Blank Activity* lalu pilih *next*.



Gambar 3.5 form new android application

3.5 Database Sistem Pada Aplikasi Kamus

Basis data yang digunakan pada aplikasi kamus bahasa Indonesia-Batak berbasis android menggunakan algoritma Knuth-Morris-Pratt adalah basis data internal dimana basisdata ini menggunakan SQLite. Sama halnya pada database pada umumnya SQLite memiliki objek-objek seperti *table*, *view*, dan *index*. Perintah-perintah SQL-nya juga sangat mirip dengan yang biasa digunakan yaitu SELECT, INSERT, UPDATE, DELETE, dan sebagainya.

Dalam peletakan basis data SQLite ini berada pada satu *class* tersendiri dikarenakan untuk membedakan antara *class* yang berisikan fungsi-fungsi dari GUI (graphical user interface) dengan *class-class* lainnya. Di bawah ini cuplikan *sourcecode database* dari aplikasi kamus Indonesia-Batak menggunakan algoritma Knuth-Morris-Pratt.

```
public class BasisData extends SQLiteOpenHelper
```

didalam public class terdapat nama *database* dan *string* yaitu, *string Indonesia*, *string Keterangan*, *string input*, *string hasil*, *string method* dan *string ter_ket*

```
{
```

```
public static int DB_VERSION = 2;
```

```
private static final String DATABASE_NAME = "dbkamus.jeksen";
```

```
public static final String INDONESIA = "indonesia";
```

```
public static final String TER_KET = "ket";
```

```
public static final String INPUT = "input";
```

```
public static final String HASIL = "hasil";
```

```
public static final String METHOD = "metode";
```

Cangkupan yang dijadikan satu yang bertujuan memudahkan pengelompokan dari

```
public class datakamus extends SQLiteOpenHelper
```

```
public BasisData(Context context) {
```

```
super(context, DATABASE_NAME, null, DB_VERSION);
```

untuk menginisialisasikan database

```
public void createTable(SQLiteDatabase db) {
```

```
db.execSQL("DROP TABLE IF EXISTS kamus");
```

```
db.execSQL("CREATE TABLE IF NOT EXISTS kamus (" + "id  
INTEGER PRIMARY KEY AUTOINCREMENT, " + INDONESIA + " TEXT, " +  
TER_KET + " TEXT);");
```

```

        db.execSQL("CREATE TABLE IF NOT EXISTS bookmark (" + "id
INTEGER PRIMARY KEY AUTOINCREMENT, " + INPUT + " TEXT, " + HASH. |
" TEXT, " + METHOD + " TEXT");");
    }

```

Script di atas berfungsi sebagai kerangka tabel sebelum terisikan bagian-bagian seperti *INDONESIA* dan *TER_KET*.

```

public void generateData(SQLiteDatabase db) {
    ContentValues cv = new ContentValues();
    cv.put(INDONESIA, "ada");
    cv.put(TER_KET, ":\n" +
        ",TOBA = adong \n" +
        ",KARO = lit \n" |
        ",SIMALUNGUN = dong");
    if (checkData("ada", db))
        db.insert("kamus", INDONESIA, cv);

```

Script yang digunakan untuk mengisikan data kamus ke database internal

```

}

```

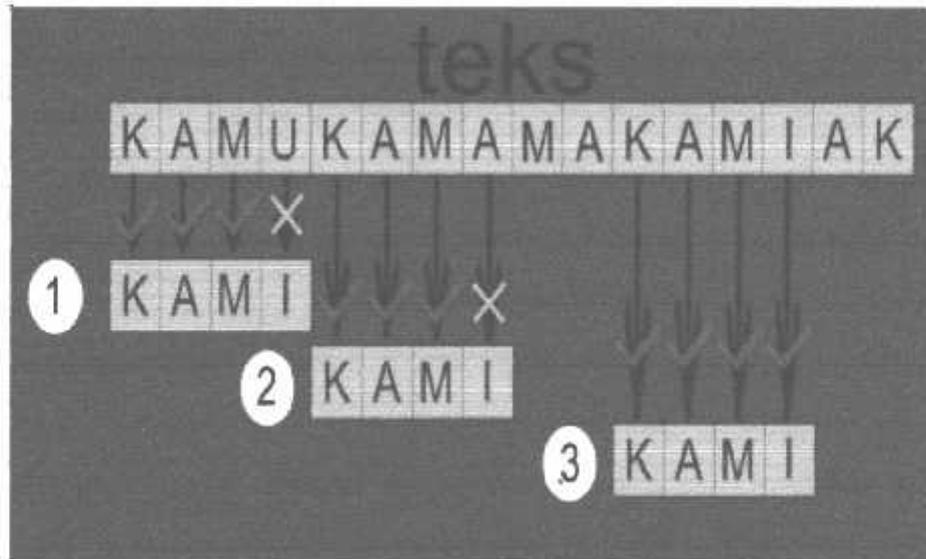
Kurung penutup untuk mengakhiri *class* yang telah dibuat.

3.6 Algoritma Knuth-Morris-Pratt

Secara sistematis, langkah-langkah yang dilakukan algoritma Knuth-Morris-Pratt pada saat mencocokkan string:

1. Algoritma Knuth-Morris-Pratt mulai mencocokkan pola pada awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter pola dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi:

2. Karakter di pola dan di teks yang dibandingkan tidak cocok (mismatch).
3. Semua karakter di pola cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini
4. Algoritma kemudian menggeser pola berdasarkan tabel teks, lalu mengulangi langkah 2 sampai pola berada di ujung teks



Gambar 3.2 Contoh pattern matching dengan KMP

Terdapat teks yang berisikan berisikan “K,A,M,U,K,A,M,A,M,A,K,A,M,I,A,K”. Dan akan dicocokkan dengan pattern yang telah di inputkan yaitu “K,A,M,I”

1. Pada langkah pertama pattern “K,A,M,I” dicocokkan dengan teks yang telah ada. Pada teks, indeks ke tiga tidak menemukan kecocokan dengan indeks ke tiga pada pattern.
2. Pencocokan dilanjutkan pada indeks ke empat pada teks. Semua menemui kecocokan kecuali indeks ke tiga pattern dengan indeks ke tujuh teks.

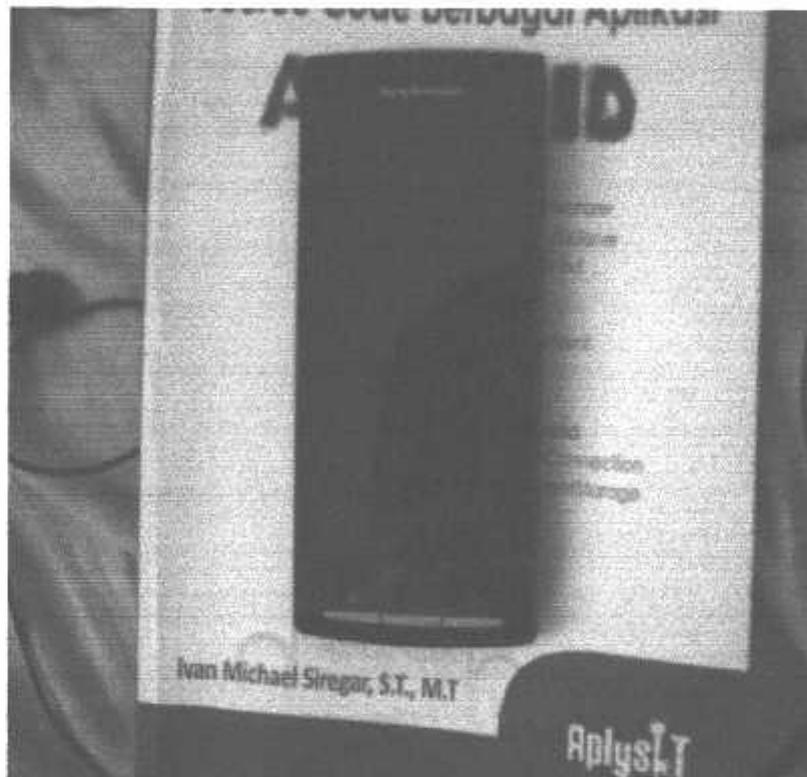
Dikarenakan indeks ke delapan dan Sembilan pada teks tidak memenuhi indeks pertama pada pattern maka pencarian dilakukan ke indeks ke sepuluh. Pada indeks ke-10,11,12,13 menemui kecocokan dengan pattern yang diinputkan.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan menjelaskan menjelaskan tentang implementasi dan uji coba terhadap "Aplikasi Kamus Bahasa Batak-Indonesia Berbasis Android Menggunakan Algoritma Knuth-Morris-Pratt".

4.1 Pengujian Aplikasi pada perangkat android

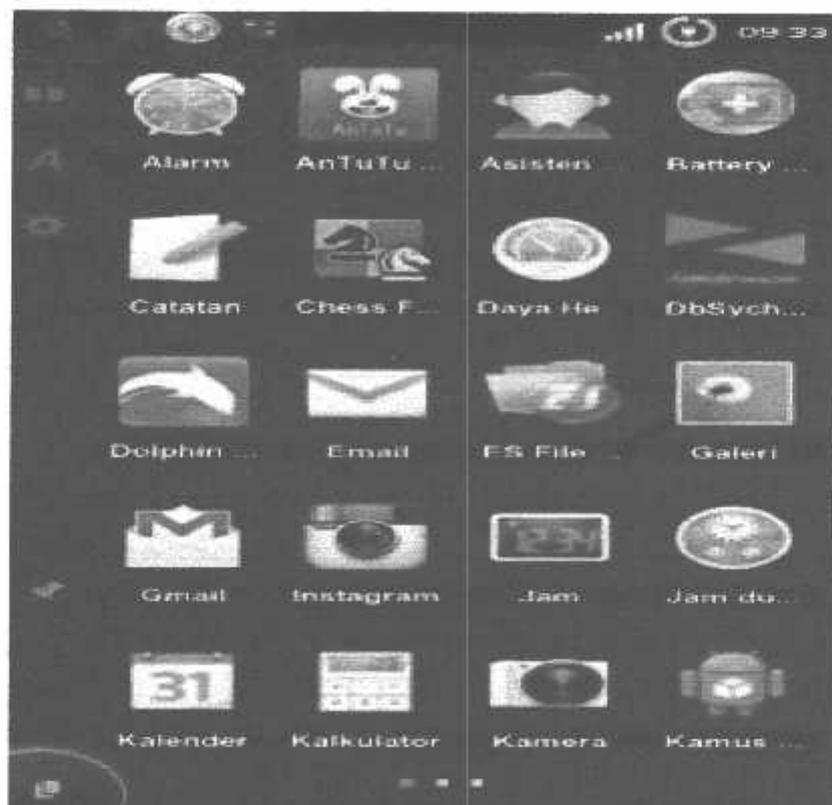
Perangkat android yang digunakan untuk pengujian ini adalah sony ericsson experia X10i



Gambar 4.2 perangkat android sony experia x10i

4.1.2 Tampilan aplikasi setelah diinstal di perangkat android

Pada gambar 4.3 aplikasi kamus bahasa BATAK – INDONESIA telah berhasil terinstal di perangkat android.



Gambar 4.3 tampilan aplikasi setelah diinstal di perangkat android

4.2.2 Tampilan awal aplikasi pertama kali

Pada gambar 4.4 tampilan awal terdapat background berupa kain ULOS, dimana kain ULOS ini adalah simbol dari adat suku BATAK yang bercorak merah kehitam hitaman dan beberapa tombol (button) yaitu tombol BATAK – INDONESIA, INDONESIA – BATAK, BOOKMARK, dan KELUAR. Dan juga

terdapa 2 buah textfied yaitu textfield inputan (masukan) dan textfield output (keluaran). Pada bagian atas terdapat tulisan Kamus Batak Indonesia dan kata sapaan yang biasa digunakan oleh suku batak yaitu HORAS dan MEJUAH – JUAH.

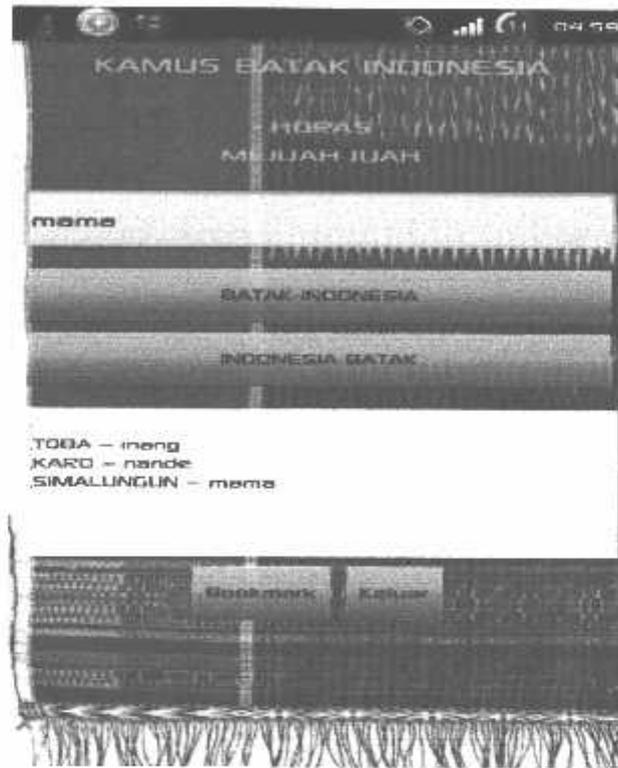


Gambar 4.4 tampilan awal aplikasi pertama kali digunakan

4.2.3 Tampilan aplikasi ketika dilakukan pencarian data

Menjelaskan proses pencarian data dengan cara menginputkan kata dari bahasa Indonesia ke Bahasa Batak dimana tampilan output adalah kata yang diinputkan tadi akan diterjemahkan ke dalam ketiga bahasa batak yang ada pada kamus. Kata yang diinputkan adalah “mama” setelah dilakukan proses penerjemahan dengan menekan tombol BATAK – INDONESIA .

Pada gambar 4.6 adalah hasil tampilan setelah proses pencarian selesai



Gambar 4.6 proses pencarian data pada aplikasi

Pada proses pencarian data pada gambar 4.6 diinputkan kata dalam bahasa Indonesia "mama" dan setelah diterjemahkan ke dalam bahasa batak maka akan menghasilkan outputan:

Toba: inang

Karo: nande

Simalungun: mama

Sedangkan pada gambar 4.7 dilakukan pencarian data dengan cara menginputkan salah satu kata dari hasil outputan bahasa Indonesia “mama”. Hasil dari outputan mama tadi digunakan sebagai inputan untuk mengetahui apakah hasil dari sesuai dengan pencarian kata sebelumnya.

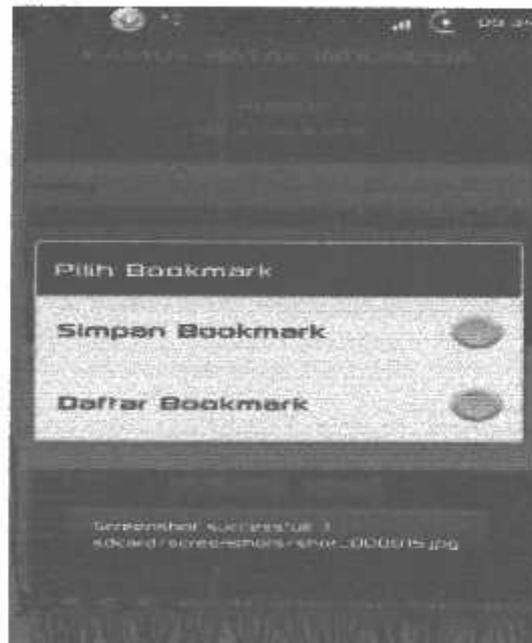


Gambar 4.7 proses pencarian data pada aplikasi

Pada gambar 4.7 dilakukan proses pencarian kata dengan cara menginputkan kata inang, yang merupakan hasil outputan dari inputan kata mama. Dari hasil penginputan kata “nande” didapat hasil outputan berupa “mama”, yang berarti hasil pencarian sesuai dengan pencarian pada proses sebelumnya. Oleh karena itu maka proses pencarian data dari bahasa BATAK – INDONESIA telah berhasil.

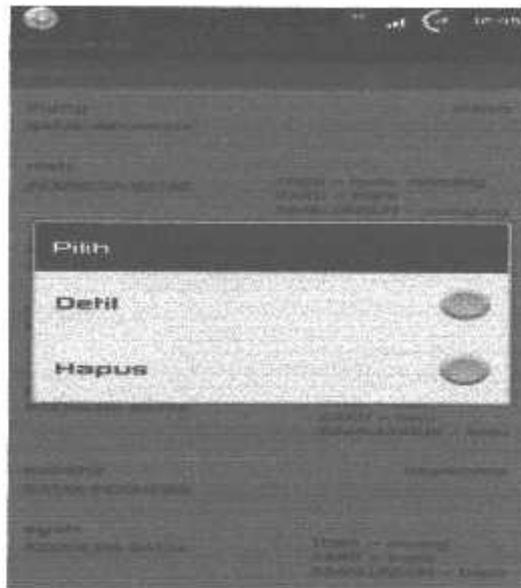
4.2.4 Pengujian bookmark

Pada aplikasi kamus bahasa INDONESIA – BATAK juga disertai fitur bookmark, fitur bookmark ini berfungsi dengan baik sebagai penyimpan data yang telah diterjemahkan apabila suatu saat user tidak ingin repot-repot lagi menginputkan kata yang telah dicari sebelumnya. Proses bookmark dapat dilihat pada gambar 4.8.



Gambar 4.8 proses bookmark

Pada gambar 4.8 terlihat proses bookmark terhadap hasil outputan dari kata inang dan bookmark tersebut juga tersedia menu yang memungkinkan pengguna ingin langsung menyimpan kata inputan maupun kata hasil keluaran



Gambar 4.9

Gambar 4.9 merupakan fitur edit pada bookmark, pada fitur ini tersedia dua pilihan, yaitu detail dan hapus, fungsi *radio button* detail adalah untuk mengecek kata yang tersimpan pada bookmark merupakan terjemahan dari bahasa yang mana dan fungsi *radio button* hapus adalah untuk menghapus kata yang tersimpan pada bookmark.

4.3 Hasil pengujian

4.3.1 Tabel Fungsionalitas Program

Proses ini dilakukan pada device sony ericsson xperia x10i yang menggunakan android versi 2.3 dengan lebar layar 4.0" dan device sony xperian tipo yang menggunakan android versi 4.0 dengan lebar layar 3.2"

Tabel 4.1 tabel fungsionalitas program

Prosedur	Hasil
Menampilkan menu utama aplikasi	Sesuai
Melakukan pencarian kata dari bahasa Indonesia ke Batak	Sesuai
Melakukan pencarian kata dari bahasa Batak ke Indonesia	Sesuai
Melakukan bookmark	Sesuai
Melakukan edit bookmark	Sesuai

Tabel 4.1 merupakan tabel dari pengujian aplikasi pada perangkat android yang digunakan oleh penulis, dimana pada hasil percobaan yang dilakukan oleh penulis aplikasi dapat berjalan dengan baik, semua prosedur yang telah ditetapkan oleh penulis dapat berjalan dengan baik pada perangkat android yang digunakan oleh penulis.

4.3.2 Tabel Pengujian User

Berikut tabel pengujian program yang telah dilakukan oleh beberapa user. Ada beberapa kriteria pertanyaan yang diajukan kepada user dengan penilaian seperti berikut:

Table 4.2 Pengujian User

No	Pertanyaan	Penilaian		
		B	C	K
1	Menurut anda apakah tampilan dari aplikasi ini sudah baik	6	2	
2	Bagaimana penilaian anda kemampuan aplikasi dalam melakukan pencarian kata	2	6	
3	Bagaimana penilaian anda terhadap fitur-fitur yang ada	4	4	
4	Bagaimana menurut anda tentang kelengkapan dan ketepatan aplikasi dalam menerjemahkan kata	2	4	2
5	Bagaimana menurut anda tentang kelengkapan dari jenis bahasa batak	7	1	

	yang ada pada kamus ini			
6	Apakah aplikasi ini sudah cukup memudahkan pengguna dalam belajar bahasa batak	7	1	
7	Apakah secara keseluruhan aplikasi ini sudah cukup baik menurut anda	4	2	2

Keterangan nilai:

B: Baik; C: Cukup; K: kurang

Keterangan:

1. Dari 8 orang user 75% user memberikan nilai baik dari tampilan kamus ini dan 25% memberikan nilai cukup.
 2. Untuk kemampuan aplikasi dalam melakukan pencarian kata 25% user memberikan nilai baik dan 75% user memberikan nilai cukup.
 3. Untuk penilaian terhadap fitur – fitur yang ada pada aplikasi kamus ini 50% user memberikan nilai baik dan 50% lagi memberikan nilai cukup.
 4. Untuk kelengkapan dan ketepatan aplikasi dalam menerjemahkan kata 25% user memberikan nilai cukup, 50% memberikan nilai baik dan 25% lagi memberikan nilai kurang.
 5. Untuk kelengkapan dari sub bahasa yang ada pada aplikasi kamus ini, 82,5% user memberikan nilai baik dan 12,5% lagi memberikan nilai cukup.
 6. Untuk kemudahan dalam belajar bahasa batak 87,5% user mengatakan baik dan 12,5% mengatakan cukup.
-

7. Untuk penilaian secara keseluruhan terhadap kamus 50% user memberikan nilai baik 25% memberikan nilai cukup dan 25% lagi memberikan nilai kurang.
-

BAB V PENUTUP

5.1 Kesimpulan

1. Aplikasi kamus ini berhasil melakukan penerjemahan kata dari INDONESIA – BATAK maupun dari BATAK – INDONESIA dengan database yang ada.
2. Setelah dilakukan pengujian oleh beberapa user, kamus ini cukup memudahkan pengguna dalam belajar bahasa Batak.
3. Metode KNUTH- MORRIS PRAT berhasil diterapkan pada kamus bahasa BATAK - INDONESIA.m,

5.2 Saran

1. Kedepannya aplikasi ini bisa lebih dikembangkan lagi dengan memperbanyak kata dalam database dan juga dengan penambahan dari bahasa batak lain yang belum sempat dimasukkan oleh penulis.
 2. Kamus ini juga perlu penambahan fitur yang bisa menerjemahkan tidak hanya kata-kata yang umum digunakan tapi juga berupa kata yang memiliki awalan dan imbuhan, peribahasa dan bahasa batak tradisional.
 3. Dikembangkan lebih baik lagi dengan penambahan dengan database eksternal untuk pengupdatean data apabila ada pengembang ingin menambahkan data-data yang baru.
-

Daftar Pustaka

1. Djuandi, Feri. 2011. *Program Database Sederhana di Android*
 2. Safaat, Nazarudin 2012, *Pemograman Aplikasi Android Mobile Smartphone dan Table PC Berbasis Android, Jakarta*
 3. Sinaga, Richad, 2007, *Bahasa Batak Toba untuk Pemula-Naposobulung, Dian Utama, Jakarta*
 4. Sinaga, Richad, 2007, *Kamus Batak Toba - Indonesia: Kosa kata, Istilah-istilah Adat, Ungkapan, Tamsil dan Peribahasa, Dian Utama, Jakarta*
 5. Siregar, Michael, Ivan 2011, *Membongkar Source Code Berbagai Aplikasi Android, Aplyist, Bandung*
 6. Zubeirsyah M. Hasyim, Zainal Abidin, Nurhayati Lubis, Bahren Umar Siregar, Farida Dahlan, T. Ayub Su, 2002, *Kamus bahasa Simalungun Indonesia Balai pustaka, Jakarta.*
 7. [http://id.wikipedia.org/wiki/Android_\(sistem_operasi\)](http://id.wikipedia.org/wiki/Android_(sistem_operasi)) (26 September)
 8. http://id.wikipedia.org/wiki/Algoritma_Knuth-Morris-Pratt (26 September)
 9. <http://kamusiana.com/index.php/list/26/.A.xhtml> (24 September.
 10. <http://kamus.karo.or.id/> (26 September 2012)
-

Lampiran 2 : Formulir Bimbingan Skripsi



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Jeksen Manuel Simarmata
NIM : 0818105
Jurusan : Teknik Informatika S-1
Judul : Implementasi Algoritma Knuth-Morris Prat pada kamus Indonesia –
Batak berbasis Android

No	Tanggal	Uraian	Paraf
1	18-07-2012	Revisi bab 1 dan bab 2	
2	13-11-2012	Makalah	
3	15-11-2012	Acc makalah	
4	22-11-2012	Bab 3 dan bab 4	
5	05-12-2012	Demo program	
6	12-12-2012	Bab 1 sampai bab 4	
7	15-12-2012	Acc bab 1 smapi bab 4	

Dosen Pembimbing

Joseph/Dedy Irawan, ST, MT
NIP. 1974041620005021002

Lampiran 3 : Formulir Bimbingan Skripsi



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

FORMULIR BIMBINGAN SKRIPSI

Nama : Jeksen Manuel Simarmata
NIM : 0818105
Jurusan : Teknik Informatika S-1
Judul : Implementasi Algoritma Knuth-Morris Prat pada kamus Indonesia –
Batak berbasis Android

No	Tanggal	Uraian	Paraf
1	05-07-2012	Revisi bab 1 dan bab 2	
2	13-11-2012	Makalah	
3	27-11-2012	Acc makalah	
4	22-11-2012	Bab 3 dan bab 4	
5	05-12-2012	Demo program	
6	12-12-2012	Bab 1 sampai bab 4	
7	15-12-2012	Acc bab 1 smapi bab 4	

Dosen Pembimbing

Yoseph Agus Pranoto ST
NIP.P. 1031000432

Lampiran 4 : Berita Acara Ujian Skripsi



INTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo Km. 2 Malang

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

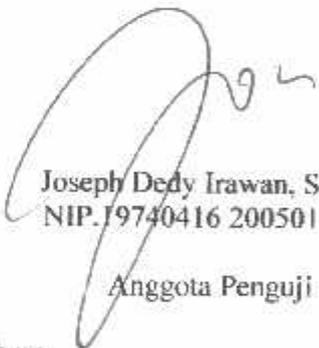
Nama : Jeksen Manuel Simarmata
NIM : 0818105
Jurusan : Teknik Informatika S-1
Judul : Implementasi Algoritma Knuth-Morris Prat pada kamus Indonesia -
Batak berbasis Android

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

Hari : Rabu
Tanggal : 20 Pebruari 2013
Nilai : A

Panitia Ujian Skripsi :

Ketua Majelis Penguji


Joseph Dedy Irawan, ST, MT
NIP.19740416 200501 1 002

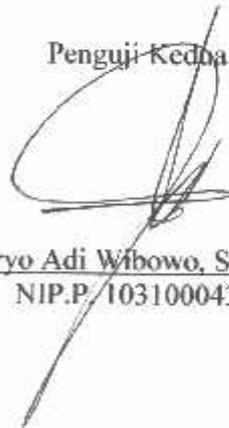
Anggota Penguji :

Penguji Pertama



Sandy Nataly Mantja, S.kom
NIP.P. 1030800418

Penguji Kedua



Suryo Adi Wibowo, ST, MT
NIP.P. 1031000438

Script untuk database

```
package com.jeksen.kamusbatak;

import java.util.ArrayList;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class BasisData extends SQLiteOpenHelper {

    public static int DB_VERSION = 2;

    private static final String DATABASE_NAME = "dbkamus.jeksen";
    public static final String INDONESIA = "indonesia";
    public static final String TER_KET = "ket";

    public static final String INPUT = "input";
    public static final String HASIL = "hasil";
    public static final String METHOD = "metode";

    // untuk inisialisasi database
    public BasisData(Context context) {
        super(context, DATABASE_NAME, null, DB_VERSION);
    }
}
```

```

// untuk membuat table kamus
public void createTable(SQLiteDatabase db) {
    db.execSQL("DROP TABLE IF EXISTS kamus");

    db.execSQL("CREATE TABLE IF NOT EXISTS kamus ("
        + "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
INDONESIA
        + " ITEXT, " + TER_KET + " TEXT );");

    db.execSQL("CREATE TABLE IF NOT EXISTS bookmark ("
        + "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
INPUT
        + " TEXT, " + HASIL + " TEXT, " + METHOD + " TEXT );");
}

@Override
public void onCreate(SQLiteDatabase db) {
    createTable(db);
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS kamus ");
    db.execSQL("DROP TABLE IF EXISTS bookmark ");

    createTable(db);
}

public Cursor getMatchingIndoBatak(String query, SQLiteDatabase db) {

```

```

        Cursor mCursor = db.query(false, "kamus", new String[] { "id", INDONESIA,
TER_KET },
                                INDONESIA + " LIKE ? ",
                                new String[] { "%" + query + "%" }, null, null, null, null);

        if (mCursor != null) {
            mCursor.moveToFirst();
            return mCursor;
        }

        return null;
    }

```

```

public Cursor getMatchingBatakIndo(String query, SQLiteDatabase db) {
    Cursor mCursor = db.query(false, "kamus", new String[] { "id", INDONESIA,
TER_KET },
                              TER_KET + " LIKE ? ",
                              new String[] { "%" + query + "%" }, null, null, null, null);

    if (mCursor != null) {
        mCursor.moveToFirst();
        return mCursor;
    }

    return null;
}

```

```

// untuk mengisi data ke kamus internal.
public void generateData(SQLiteDatabase db) {
    ContentValues cv = new ContentValues();

```

```
cv.put(INDONESIA, "mimpi");
cv.put(TER_KET, ":\n" +
        ",TOBA = nipi \n" +
        ",KARO = nipi \n" +
        ",SIMALUNGUN = nipi");
if (checkData("mimpi", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "minyak");
cv.put(TER_KET, ":\n" +
        ",TOBA = miak \n" +
        ",KARO = minak \n" +
        ",SIMALUNGUN = minak");
if (checkData("minyak", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "miskin");
cv.put(TER_KET, ":\n" +
        ",TOBA = pogos \n" +
        ",KARO = musil \n" +
        ",SIMALUNGUN = miskin");
if (checkData("miskin", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "mudah");
cv.put(TER_KET, ":\n" +
        ",TOBA = mura \n" +
        ",KARO = nukah \n" +
```

```

                                ",SIMALUNGUN = gampang");
if (checkData("mudah", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "mulut");
cv.put(TER_KET, ":\n" +
                                ",TOBA = baba \n" +
                                ",KARO = babah \n" +
                                ",SIMALUNGUN = babah");

if (checkData("mulut", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "musuh");
cv.put(TER_KET, ":\n" +
                                ",TOBA = alo \n" +
                                ",KARO = imbang \n" +
                                ",SIMALUNGUN = alo");

if (checkData("musuh", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "nafsu");
cv.put(TER_KET, ":\n" +
                                ",TOBA = pangimbung \n" +
                                ",KARO = nura \n" +
                                ",SIMALUNGUN = gijik");

if (checkData("nafsu", db))
db.insert("kamus", INDONESIA, cv);

```

```
cv.put(INDONESIA, "naik");
cv.put(TER_KET, ":\n" +
        ",TOBA = nakkok \n" +
        ",KARO = nangkih \n" +
        ",SIMALUNGUN = takkih");
if (checkData("naik", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nakal");
cv.put(TER_KET, ":\n" +
        ",TOBA = juguljogal \n" +
        ",KARO = gutil \n" +
        ",SIMALUNGUN = dajal");
if (checkData("nakal", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nangka");
cv.put(TER_KET, ":\n" +
        ",TOBA = pinasa \n" +
        ",KARO = nangka \n" +
        ",SIMALUNGUN = nakka");
if (checkData("nangka", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nanti");
cv.put(TER_KET, ":\n" +
        ",TOBA = annon \n" +
        ",KARO = kari \n" +
```

```
cv.put(INDONESIA, "nikah; menikah");
cv.put(TER_KET, ":\n" +
        ",TOBA = mangoli \n" +
        ",KARO = sereh \n" +
        ",SIMALUNGUN = moppo");
if (checkData("nikah; menikah", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nyanyi");
cv.put(TER_KET, ":\n" +
        ",TOBA = cndc \n" +
        ",KARO = benter \n" +
        ",SIMALUNGUN = doding");
if (checkData("nyanyi", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nyata");
cv.put(TER_KET, ":\n" +
        ",TOBA = nata \n" +
        ",KARO = tangkas \n" +
        ",SIMALUNGUN = takkas");
if (checkData("nyata", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "nyeri");
cv.put(TER_KET, ":\n" +
        ",TOBA = borit \n" +
```

```

",KARO = mesui \n" +
",SIMALUNGUN = borit");

if (checkData("nyeri", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "orang");
cv.put(TER_KET, ":\n" +
",TOBA = jolma; halalk \n" +
",KARO = urang \n" +
",SIMALUNGUN = halak; jolma");

if (checkData("orang", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "pacar");
cv.put(TER_KET, ":\n" +
",TOBA = hallet \n" +
",KARO = rondong \n" +
",SIMALUNGUN = hasoman");

if (checkData("pacar", db))
db.insert("kamus", INDONESIA, cv);

cv.put(INDONESIA, "pahat");
cv.put(TER_KET, ":\n" +
",TOBA = lotik \n" +
",KARO = tukak \n" +
",SIMALUNGUN = pahat");

if (checkData("pahat", db))
db.insert("kamus", INDONESIA, cv);

```

```
cv.put(INDONESIA, "pakaian");
cv.put(TER_KET, ":\n" +
        ",TOBA = abit \n" +
        ",KARO = tukak \n" +
        ",SIMALUNGUN = gijik");
if (checkData("pakaian", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "paku");
cv.put(TER_KET, ":\n" +
        ",TOBA = labang \n" +
        ",KARO = labang \n" +
        ",SIMALUNGUN = paku");
if (checkData("paku", db))
db.insert("kamus", INDONESIA, cv);
```

```
cv.put(INDONESIA, "palsu");
cv.put(TER_KET, ":\n" +
        ",TOBA = lipe \n" +
        ",KARO = lancung \n" +
        ",SIMALUNGUN = palsu");
if (checkData("palsu", db))
db.insert("kamus", INDONESIA, cv);
```

```
}
```

```

        ContentValues cv = new ContentValues();
        cv.put(INPUT, input);
        cv.put(HASIL, hasil);
        cv.put(METHOD, method);

        db.insert("bookmark", null, cv);
    }

    void deleteDataBookmark(String id, SQLiteDatabase db) {

        db.delete("bookmark", "id=?", new String[]{id});
    }

    void deleteAllBookmark(SQLiteDatabase db) {

        db.delete("bookmark", null, null);
    }

    public ArrayList<BookmarkBean> getAllBookmark(SQLiteDatabase db) {
        ArrayList<BookmarkBean> arrData = new ArrayList<BookmarkBean>();

        try {

            Cursor cur = db.query("bookmark", new String[]{ "id", INPUT, HASIL,
METHOD },
                null, null, null, null, " id DESC ");
            cur.moveToFirst();

            while (!cur.isAfterLast()) {

                BookmarkBean bean = new BookmarkBean();

```

```
        bean.setId(cur.getString(0));
        bean.setInput(cur.getString(1));
        bean.setHasil(cur.getString(2));
        bean.setMethod(cur.getString(3));

        arrData.add(bean);

        cur.moveToNext();
    }

    cur.close();
}
catch(Exception e) {
    e.printStackTrace();
}

return arrData;
}
}
```

Script untuk fitur bookmark

```
package com.jeksen.kamusbatak;

import java.util.ArrayList;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.TextView;

public class BookAdapter extends BaseAdapter {

    Context ctx;
    ArrayList<BookmarkBean> data;
    LayoutInflater inflater;

    public BookAdapter(Context ctx, ArrayList<BookmarkBean> data) {
        this.ctx = ctx;
        this.data = data;
        this.inflater = (LayoutInflater)
this.ctx.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    }

    @Override
    public int getCount() {
        // TODO Auto-generated method stub
        return data.size();
    }
}
```

```
}
```

```
@Override
```

```
public Object getItem(int arg0) {  
    // TODO Auto-generated method stub  
    return data.gct(arg0);  
}
```

```
}
```

```
@Override
```

```
public long getItemId(int position) {  
    // TODO Auto-generated method stub  
    return 0;  
}
```

```
}
```

```
@Override
```

```
public View getView(int position, View convertView, ViewGroup parent) {
```

```
    View v = inflater.inflate(R.layout.item_bookmark, null);
```

```
    BookmarkBean bean = (BookmarkBean) getItem(position);
```

```
    TextView id = (TextView) v.findViewById(R.id.bookId);
```

```
    TextView input = (TextView) v.findViewById(R.id.bookInput);
```

```
    TextView hasil = (TextView) v.findViewById(R.id.bookHasil);
```

```
    TextView method = (TextView) v.findViewById(R.id.bookMethod);
```

```
    id.setText(""+bean.getId());
```

```
    input.setText(""+bean.getInput());
```

```
    hasil.setText(""+bean.getHasil());
```

```
    method.setText(""+bean.getMethod());
```

```
return v;
```

```
}
```

```
}
```



Script untuk metode Knuth-Morris Pratt

```
package com.jeksen.kamusbatak;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MetodeData extends Activity {

    private SQLiteDatabase db = null;
    private Cursor kamusCursor = null;
    private BasisData basisdata = null;

    private TextView textket;
    private EditText textinput;
    private Button btnbatakindo, btnindobatak;
    private Button btnbookmark, btnonline, btnkeluar;

    private String method = "INDONESIA-BATAK";
```

```

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.layout_kamus);

    basisdata = new BasisData(this);
    db = basisdata.getWritableDatabase();

    if (!basisdata.checkDataAll(db))
        basisdata.generateData(db);

    textinput = (EditText) findViewById(R.id.textinput);
    textket = (TextView) findViewById(R.id.textket);

    btnindobatak = (Button) findViewById(R.id.btnindobatak);
    btnindobatak.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View arg0) {
            method = "INDONESIA-BATAK";
            translateIndoBatak();
        }
    });

    btnbatakindo = (Button) findViewById(R.id.btnbatakindo);
    btnbatakindo.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View arg0) {
            method = "BATAK-INDONESIA";

```

```

        translateBatakIndo();
    }
});

btnbookmark = (Button) findViewById(R.id.btnbookmark);
btnonline = (Button) findViewById(R.id.btnonline);
btnkeluar = (Button) findViewById(R.id.btnkeluar);

btnbookmark.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View arg0) {
        bookmarkClick();
    }
});

btnonline.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View arg0) {
        Intent i = new Intent(MetodeData.this, OnlineData.class);
        startActivity(i);
    }
});

btnkeluar.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View arg0) {
        finish();
    }
});

```

```

    });
}

String dataInput = "", dataHasil = "";
void translateIndoBatak() {

    dataInput = textinput.getText().toString().trim();
    textket.setText("");

    if (dataInput.length()<1) {
        Toast.makeText(this, "Data input kata kosong.",
Toast.LENGTH_LONG).show();
    }
    else {

        kamusCursor = basisdata.getMatchingIndoBatak(dataInput, db);
        if (kamusCursor==null) {
            Toast.makeText(this, "Data terjemahan kamus tidak ditemukan.",
Toast.LENGTH_LONG).show();
        }
        else {

            boolean check = false;
            kamusCursor.moveToFirst();
            while (!kamusCursor.isAfterLast()) {
                String ket = kamusCursor.getString(2);
                if (ket.trim().length()>0) {
                    textket.setText(ket+"\n\r");
                    check = true;
                }
            }
        }
    }
}

```

```

        kamusCursor.moveToNext();
    }

    if (check)
        Toast.makeText(this, "Data terjemahan ditemukan.",
Toast.LENGTH_LONG).show();
    else
        Toast.makeText(this, "Data terjemahan tidak ditemukan.",
Toast.LENGTH_LONG).show();

    }
}

dataHasil = textket.getText().toString().trim();

}

void translateBatakIndo() {

    dataInput = textinput.getText().toString().trim();
    textket.setText("");

    if (dataInput.length() < 1) {
        Toast.makeText(this, "Data input kata kosong.",
Toast.LENGTH_LONG).show();
        textket.setText("");
    }
    else {

        kamusCursor = basisdata.getMatchingBatakIndo(dataInput, db);
        if (kamusCursor == null) {

```

```

        Toast.makeText(this, "Data terjemahan kamus tidak ditemukan.",
Toast.LENGTH_LONG).show();
        textket.setText("");
    }
    else {
        boolean check = false;
        kamusCursor.moveToFirst();
        while (!kamusCursor.isAfterLast()) {
            String ket = kamusCursor.getString(1);
            if (ket.trim().length()>0) {
                textket.setText(ket+"\n\r");
                check = true;
            }

            kamusCursor.moveToNext();
        }

        if (check)
            Toast.makeText(this, "Data terjemahan ditemukan.",
Toast.LENGTH_LONG).show();
        else
            Toast.makeText(this, "Data terjemahan tidak ditemukan.",
Toast.LENGTH_LONG).show();
    }
}

    dataIhasil = textket.getText().toString().trim();
}

// -----
private void bookmarkClick() {

```

```

AlertDialog.Builder builder = new AlertDialog.Builder(this);
builder.setTitle("Pilih Bookmark");

final CharSequence[] choiceList = { "Simpan Bookmark", "Daftar Bookmark" };
int selected = -1; // does not select anything
builder.setSingleChoiceItems(choiceList, selected,
    new DialogInterface.OnClickListener() {

        @Override
        public void onClick(DialogInterface dialog, int which) {

            if (which==0) {
                if (dataInput.length()>0
                    && dataHasil.length()>0) {

                    basisdata.simpanDataBookmark(dataInput, dataHasil, method, db);
                    Toast.makeText(MetodeData.this,
                        "Data berhasil
disimpan..", Toast.LENGTH_LONG).show();
                }
                else
                    Toast.makeText(MetodeData.this,
                        "Data bookmark
masih kosong..", Toast.LENGTH_LONG).show();
            }

            if (which==1) {
                Intent i = new Intent(MetodeData.this,
BookmarkData.class);

```

```
        startActivity(i);
    }

    dialog.cancel();
}

});

AlertDialog alert = builder.create();
alert.show();
}

//-----
@Override
public void onBackPressed() {
    super.onBackPressed();
    finish();
}

@Override
protected void onDestroy() {
    super.onDestroy();

    if (db != null)
        db.close();
}
}
```