

**PENGEMBANGAN KAMUS BAHASA JAWA-BAHASA INDONESIA
MENGUNAKAN JAVA 2 MICRO EDITION (J2ME)**

SKRIPSI

**Diajukan kepada Fakultas Teknik
Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan
guna Memperoleh Gelar Sarjana Pendidikan Teknik Informatika**



**Oleh
Kartika Yudha Pratama
NIM 08520244042**


**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2013**

PERSETUJUAN

Skripsi yang berjudul “Pengembangan Kamus Bahasa Jawa Menggunakan Java 2 Micro Edition (J2ME)” yang disusun oleh Kartika Yudha Pratama, NIM 08520244042 ini telah diperiksa dan disetujui oleh pembimbing untuk diujikan.

Yogyakarta, 16 Januari 2013

Pembimbing Skripsi,



Adi Dewanto, M.Kom.
NIP. 19721228 200501 1 001

PERNYATAAN

Yang bertandatangan di bawah ini :

Nama : Kartika Yudha Pratama

NIM : 08520244042

Prodi : Pendidikan Teknik Informatika

Angkatan : 2008

Judul Skripsi : Pengembangan Kamus Bahasa Jawa Menggunakan Java 2 Micro
Edition (J2ME)

Menyatakan bahwa skripsi ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Demikian pernyataan ini dibuat dalam keadaan sadar dan tidak dipaksakan untuk digunakan sebagaimana mestinya.

Yogyakarta, 16 Januari 2013

Penulis



Kartika Yudha Pratama

NIM 08520244042

PENGESAHAN

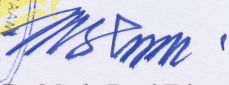
Skripsi yang berjudul "Pengembangan Kamus Bahasa Jawa Menggunakan Java 2 Micro Edition (J2ME)" yang disusun oleh Kartika Yudha Pratama, NIM 08520244042 ini telah dipertahankan di depan Dewan Penguji pada tanggal 1 Februari 2013 dan dinyatakan lulus.

DEWAN PENGUJI

Nama	Jabatan	Tanda Tangan	Tanggal
Adi Dewanto, M.Kom.	Ketua Penguji		21-2-2013
Drs. Djoko Santoso, M.Pd.	Sekretaris Penguji		21-2-2013
Drs. Totok Sukardiyono, M.T	Penguji Utama		21-2-2013

Yogyakarta, 21 Februari 2013
Fakultas Teknik
Dekan,




Dr. Moch. Bruri Triyono
NIP. 19560216 198603 1 003

MOTTO

"Sesuatu mungkin mendatangi mereka yang mau menunggu, namun hanya didapatkan oleh mereka yang bersemangat mengejarnya"
(Abraham Lincoln)

"Kawula mung saderma, mobah-mosik kersaning Hyang sukmo"
Lakukan yang kita bisa, setelahnya serahkan kepada Tuhan.
(Dian Ribut)

PERSEMBAHAN

Laporan Tugas Akhir Skripsi ini penulis persembahkan untuk :

- Bapak dan Ibu tercinta yang selalu memberikan kasih sayang, mendoakan, dan memberikan dukungan moril maupun materiil tanpa henti. Terimakasih.
- Kakak-kakak ku, Mas Anto, Mbak Lina, Mas Hendi, dan Mbak Dedeh serta keponakanku Nayaka Pradana Kusuma yang selalu menghiburku serta mendoakan dan memberikan dukungan moril maupun materiil.
- Anggi Sekar Rosalina yang selalu menemani, memberi inspirasi dan berjuang bersama dalam mengerjakan skripsi.
- Teman-teman di LIMUNY yang telah memberikan ilmu, kepercayaan, keceriaan, persahabatan, serta rasa kekeluargaan selama ini.
- Teman-teman GIFO '08 yang telah berjuang bersama menyelesaikan skripsi.
- Almamater ku, Pendidikan Teknik Informatika, Fakultas Teknik, Universitas Negeri Yogyakarta untuk ilmu dan pengetahuan yang ku peroleh selama duduk di bangku kuliah.

PENGEMBANGAN KAMUS BAHASA JAWA MENGUNAKAN JAVA 2 MICRO EDITION (J2ME)

Oleh
Kartika Yudha Pratama
NIM 08520244042

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan aplikasi kamus bahasa Jawa menggunakan Java 2 Micro Edition (J2ME), menguji kinerja aplikasi, serta mengetahui tingkat kelayakan aplikasi. Hasil akhir dari penelitian ini adalah produk aplikasi kamus bahasa Jawa berbasis J2ME yang telah teruji kinerja dan kelayakannya.

Penelitian ini merupakan Penelitian dan Pengembangan. Tahapan dalam penelitian ini meliputi tahap analisis, desain, pengembangan, implementasi, dan penilaian. Validasi dilakukan untuk mengetahui kelayakan aplikasi yang dikembangkan. Uji coba dilakukan di Fakultas Teknik Universitas Negeri Yogyakarta dengan melibatkan 30 orang mahasiswa yang memiliki *ponsel* yang mendukung teknologi J2ME. Instrumen penelitian menggunakan angket. Analisis data menggunakan teknik analisis statistik deskriptif kualitatif dan kuantitatif.

Hasil penelitian menunjukkan bahwa: (1) kualitas aplikasi kamus bahasa Jawa yang dikembangkan termasuk dalam kategori sangat layak dari segi konten dan layak dari segi *software*, hal ini berdasarkan prosentase penilaian ahli materi sebesar 80% dan pengujian aplikasi sesuai dengan *UTC for Java ME Application* dinyatakan layak dari aspek *reliability*, *functionality*, *efficiency*, dan *portability* setelah lolos dalam setiap tahapan pengujian aplikasi; (2) penilaian pengguna terhadap aplikasi dari aspek *usability* pada tahap implementasi diperoleh prosentase penilaian 76.53% dan termasuk dalam kategori sangat layak.

Kata kunci : aplikasi *ponsel*, bahasa Jawa, Java 2 Micro Edition, J2ME

KATA PENGANTAR



Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat serta hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir Skripsi berjudul **“Pengembangan Kamus Bahasa Jawa Menggunakan Java 2 Micro Edition (J2ME)”** tanpa aral rintangan yang berarti.

Dalam penyusunan Tugas Akhir Skripsi ini tentunya tidak lepas dari dukungan, bantuan, bimbingan, dan saran dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:


1. Prof. Dr. H. Rochmat Wahab, M.Pd., M.A. selaku Rektor Universitas Negeri Yogyakarta;
2. Dr. Moch. Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta;
3. Muhammad Munir, M.Pd. selaku Ketua Jurusan Pendidikan Teknik Elektronika;
4. Dr. Ratna Wardani selaku Ketua Prodi Pendidikan Teknik Informatika dan Dosen Pembimbing Akademik;
5. Handaru Jati, Ph.D. selaku Koordinator Tugas Akhir Skripsi;
6. Adi Dewanto, M.Kom. selaku Dosen Pembimbing Tugas Akhir Skripsi yang telah memberikan bimbingan dan dukungan dalam penyusunan skripsi ini;
7. Staff karyawan Fakultas Teknik dan jurusan Pendidikan Teknik Informatika yang telah membantu dalam hal administrasi selama ini;

8. Bapak, Ibuk, Mas Anto, Mbak Lina, Mas Hendi, Mbak Dedeh dan keponakanku Kaka tercinta yang senantiasa memberikan dukungan moril maupun materiil serta selalu mendoakan;
9. Anggi Sekar Rosalina terimakasih selalu memberikan dukungan, inspirasi, dan do'a yang tiada henti, terimakasih telah berjuang bersama mengerjakan skripsi dan terus berjuang bersama untuk masa depan;
10. Keluarga besar LIMUNY yang selalu memberikan bantuannya;
11. Bowo, Adit, Gunawan, Anggi, dan Faiq yang telah banyak membantu dalam proses pengembangan aplikasi, penelitian, dan penyusunan skripsi ini;
12. Teman-teman seperjuangan pendidikan teknik informatika angkatan 2008 khususnya kelas G dalam suka dukanya mengerjakan skripsi;
13. Semua pihak yang telah membantu penulis selama pembuatan skripsi ini.

Dengan sepenuh hati penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Untuk itu, saran dan kritik yang bersifat membangun penulis harapkan demi sempurnanya skripsi ini agar dapat memberikan sumbangsih dan bahan pemikiran bagi kita semua.

Akhir kata, semoga skripsi ini bermanfaat bagi kita untuk memperkaya ilmu dan wawasan di masa sekarang dan yang akan datang.

Yogyakarta, Januari 2013



Penulis

DAFTAR ISI

	hal
HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PERNYATAAN	iii
HALAMAN PENGESAHAN	iii
MOTTO	v
PERSEMBAHAN.....	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN.....	xviii
BAB I PENDAHULUAN	
A. Latar Belakang	1
B. Identifikasi Masalah	4
C. Batasan Masalah.....	5
D. Rumusan Masalah	5
E. Tujuan.....	5
F. Manfaat.....	6
BAB II KAJIAN PUSTAKA	
A. Deskripsi Teori.....	7

1. Pengembangan.....	7
2. Kamus	20
3. Bahasa	21
4. Java 2 Micro Edition (J2ME)	23
5. Object Oriented Programming (OOP)	23
6. Kualitas Perangkat Lunak	26
B. Penelitian Relevan.....	33
 BAB III METODOLOGI PENELITIAN	
A. Desain Penelitian.....	35
1. Tahap Analisis	36
2. Tahap Desain	37
3. Tahap Pengembangan	38
4. Tahap Implementasi.....	40
5. Tahap Penilaian	41
B. Definisi Operasional.....	43
C. Tempat dan Waktu Penelitian.....	43
D. Sampel Penelitian.....	44
E. Teknik Pengumpulan Data.....	45
1. Metode Tes	45
2. Metode Kuesioner atau Angket	46
F. Instrumen Penelitian.....	46

1. Instrumen Validasi Ahli <i>Software</i>	46
2. Instrumen Validasi Ahli Materi	48
3. Instrumen Penilaian Pengguna	48
G. Validitas dan Reliabilitas Instrumen	48
1. Validitas Instrumen	468
2. Reliabilitas Instrumen	51
H. Analisis Data	53
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	
A. Analisis	55
1. Analisis Isi Program	55
2. Analisis Pengguna	55
3. Analisis <i>Software</i>	55
4. Analisis <i>Hardware</i>	56
B. Desain	57
1. <i>Material Collecting</i>	57
2. <i>Flowchart</i>	57
3. Rancangan Antarmuka Pengguna	60
C. Pengembangan	63
1. Produksi Aplikasi <i>Mobile</i>	63
2. Validasi Ahli	83
3. Revisi	85

D. Implementasi.....	89
E. Penilaian.....	91
1. Penilaian Kelayakan Aplikasi oleh Para Ahli	91
2. Penilaian Pengguna Terhadap Aplikasi	91
3. Kelebihan, Kekurangan, dan Kendala Aplikasi	92
F. Pembahasan.....	93
BAB V KESIMPULAN DAN SARAN	
A. Kesimpulan	97
B. Saran	98
DAFTAR PUSTAKA	99
LAMPIRAN	103

DAFTAR TABEL

	hal
Tabel 1: Kisi-kisi Instrumen Validasi Ahli <i>Software</i>	47
Tabel 2: Kisi-kisi Instrumen Validasi Ahli Materi	48
Tabel 3: Kisi-kisi Instrumen Penilaian Pengguna	48
Tabel 4: Penjabaran Hasil Uji Validitas.....	51
Tabel 5: Tingkat Reliabilitas Instrumen.....	542
Tabel 6: Tabel Prosentase Kelayakan Aplikasi.....	54
Tabel 7: Hasil Perhitungan Validasi Ahli Materi	83
Tabel 8: Hasil Validasi Ahli <i>Software</i>	84
Tabel 9: Hasil Perhitungan Penilaian Pengguna	90
Tabel 10: Hasil Kelayakan Aplikasi Kamus Bahasa Jawa	96

DAFTAR GAMBAR

	hal
Gambar 1: Fase-fase Sekuensial Linier Model Pressman	11
Gambar 2: Fase-fase Sekuensial Linier Model Sommerville	11
Gambar 3: Model Pengembangan <i>Rapid Application Development</i> (RAD)....	12
Gambar 4: Model Pengembangan <i>Spiral</i>	16
Gambar 5: Aplikasi Kamus Bahasa Jawa (<i>Ngoko</i>)-Bahasa Indonesia Menggunakan <i>Visual Basic</i>	19
Gambar 6: Kualitas Internal dan Eksternal Model ISO 9126-1.....	27
Gambar 7: Model Pengembangan ADDIE.....	35
Gambar 8: Prosedur Pengembangan Aplikasi Kamus Bahasa Jawa.....	42
Gambar 9: Hasil Uji Validitas Menggunakan <i>SPSS 15.0</i>	50
Gambar 10: Hasil Uji Reliabilitas Menggunakan <i>SPSS 15.0</i>	53
Gambar 11: Kategori Kelayakan Aplikasi.....	54
Gambar 12: <i>Flowchart</i> Urutan Kerja dan Interaksi Antar Menu.....	58
Gambar 13: <i>Flowchart</i> Pencarian Kata	59
Gambar 14: Desain Instalasi <i>Database</i>	60
Gambar 15: Desain Menu Awal	61
Gambar 16: Desain Menu Bahasa Jawa-Bahasa Indonesia.....	61
Gambar 17: Desain Menu Bahasa Indonesia-Bahasa Jawa.....	62
Gambar 18: Desain Menu Petunjuk Penggunaan	62
Gambar 19: Desain Menu Tentang Aplikasi	63
Gambar 20: Istilah Tombol Pada <i>Ponsel</i>	64

	hal
Gambar 21: Tampilan Proses Instalasi <i>Database</i>	65
Gambar 22: Tampilan Menu Awal	66
Gambar 23: Menampilkan Kemungkinan Kata Yang Dicari	67
Gambar 24: Menampilkan Pilihan Menu ‘Cari’ dan ‘Cari Lagi’	67
Gambar 25: Menampilkan Arti Kata.....	68
Gambar 26: Menampilkan Pemberitahuan Kata yang Dicari Tidak Ditemukan.....	68
Gambar 27: Menampilkan Menu ‘Pelafalan’ dan ‘Cari Lagi’	68
Gambar 28: Menampilkan Ijin Menggunakan Jaringan.....	68
Gambar 29: Mengakses File Suara	69
Gambar 30: Proses Pelafalan Selesai	69
Gambar 31: Membatalkan Proses Pelafalan	69
Gambar 32: Menu Bahasa Indonesia-Bahasa Jawa	70
Gambar 33: Menu Petunjuk Penggunaan	70
Gambar 34: Menu Tentang Aplikasi.....	71
Gambar 35: Keluar Aplikasi.....	72
Gambar 36: Penambahan Menu Referensi	86
Gambar 37: Tampilan Menu Referensi Kamus	86
Gambar 38: Tampilan Proses Instalasi <i>Database</i>	86
Gambar 39: Tampilan <i>Title Page</i>	86
Gambar 40: Tampilan Menu Utama.....	87
Gambar 41: Tampilan Menu Cari Kata.....	87
Gambar 42: Tampilan Sub Menu Cari Kata.....	87

	hal
Gambar 43: Tampilan Sub Menu Cari Kata	87
Gambar 44: Tampilan Menu Cari Kata (Jawa-Indo)	88
Gambar 45: Menampilkan Arti Kata.....	88
Gambar 46: Kotak Dialog Pelafalan	88
Gambar 47: Proses Pelafalan	88
Gambar 48: Tampilan Menu Cari Kata (Indonesia-Jawa)	89
Gambar 49: Tampilan Menu Bantuan.....	89
Gambar 50: Tampilan Menu Tentang Aplikasi	89
Gambar 51: Kotak Dialog Keluar Aplikasi	89
Gambar 52: Diagram Batang Hasil Pengolahan Angket Penilaian Pengguna .	90

DAFTAR LAMPIRAN

	hal
Lampiran 1. Kelas <i>kamusJawaIndo.java</i>	104
Lampiran 2. Kelas <i>splashInstallRMS.java</i>	107
Lampiran 3. Kelas <i>threadInstall.java</i>	109
Lampiran 4. Kelas <i>listMenuAwal.java</i>	115
Lampiran 5. Kelas <i>formCariKata.java</i>	117
Lampiran 6. Kelas <i>streamWav.java</i>	132
Lampiran 7. Hasil <i>Black Box Testing</i>	138
Lampiran 8. Penilaian Validasi Ahli <i>Software</i>	140
Lampiran 9. Instrumen Validasi Ahli Materi.....	169
Lampiran 10. Perhitungan Validasi Ahli Materi.....	171
Lampiran 11. Instrumen Penilaian Pengguna	172
Lampiran 12. Perhitungan Penilaian Pengguna	175
Lampiran 13. Prodi dan <i>Ponsel</i> yang Digunakan Pengguna	177
Lampiran 14. Saran dari Pengguna	178
Lampiran 15. Spesifikasi <i>Nokia 5630 XpressMusic</i>	179
Surat-surat	
Lampiran 16. Surat Pengangkatan Pembimbing TA Skripsi.....	180
Lampiran 17. Surat Permohonan Ijin Penelitian dari Fakultas.....	181
Lampiran 18. Surat Permohonan Ijin Penelitian dari SETDA DIY	182
Lampiran 19. Surat Permohonan Ijin Penelitian dari BAPPEDA Sleman.....	183

BAB I PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi telekomunikasi dan informasi (*telematika*) saat ini semakin pesat. Berbagai macam *gadget* baru diluncurkan, salah satunya adalah *ponsel* beserta segala aplikasi yang ada di dalamnya yang dapat membantu segala aktivitas penggunanya. Seiring dengan perkembangan tersebut, membuat seseorang menginginkan segala sesuatunya dapat dilakukan menggunakan *ponsel*, misalkan saja untuk mengolah data-data perkantoran, peta navigasi suatu wilayah, Internet, ataupun untuk mengatur jadwal kegiatan sehari-hari. *Ponsel* memiliki peranan yang sangat penting bagi kehidupan bermasyarakat di era masa kini baik untuk sarana berkomunikasi maupun memanfaatkan segala aplikasi dan kecanggihan yang dimilikinya.

Perkembangan teknologi telekomunikasi dan informasi kurang diimbangi dengan pelestarian adat dan budaya Jawa. Beberapa pengembangannya yang ada saat ini terutama dalam bidang pemrograman masih lebih didominasi dengan pengenalan kebudayaan asing, terlihat dari banyaknya aplikasi kamus bahasa Inggris-bahasa Indonesia, media pembelajaran bahasa Korea, ataupun media pembelajaran bahasa Jepang dan belum banyak yang mengarah pada pelestarian bahasa Jawa sebagai wujud pelestarian budaya Jawa. Dampak buruk dari perkembangan ini salah satunya adalah semakin menurunnya eksistensi penggunaan bahasa Jawa oleh masyarakat Jawa itu sendiri hal ini ditandai oleh kaum muda mudi yang sekarang ini sudah mulai melupakannya, bahkan anak-

anak kecil sekarang ini sudah mulai kehilangan bahasa Jawa sebagai bahasa ibu mereka. Dra. Sri Harti Widyastuti, MSi, staf pengajar di Fakultas Bahasa dan Seni Universitas Negeri Yogyakarta mengatakan bahwa, sekarang ini bahasa Jawa sebagai bahasa percakapan sehari-hari sudah ditinggalkan. Berdasarkan penelitian menunjukkan banyak keluarga yang tidak lagi menggunakan bahasa Jawa sebagai bahasa ibu mereka dalam kehidupan rumah tangganya (Kompas, 2009).

Bahasa Jawa sebagai salah satu kebudayaan Jawa harus tetap dijaga kelestariannya baik itu oleh masyarakat Jawa sendiri pada khususnya dan masyarakat luas yang ingin mempelajari bahasa Jawa pada umumnya. Untuk mempermudah mempelajari bahasa Jawa dapat menggunakan kamus bahasa Jawa-bahasa Indonesia, akan tetapi kamus dengan bentuk konvensional memerlukan waktu cukup lama untuk mengartikan kata sehingga diperlukan sebuah kamus *digital* yang dapat mengartikan kata lebih cepat. Upaya yang dapat dilakukan salah satunya adalah dengan memanfaatkan teknologi telekomunikasi dan informasi dalam bidang pemrograman dengan membuat kamus bahasa Jawa-bahasa Indonesia. Sudah ada yang mengembangkan aplikasi kamus bahasa Jawa-bahasa Indonesia dengan menggunakan *Visual Basic*. *Visual Basic* merupakan bahasa pemrograman *visual* yang diaplikasikan pada komputer dengan sistem operasi *Windows*. Aplikasi kamus bahasa Jawa-bahasa Indonesia menggunakan *Visual Basic* ini masih memiliki beberapa kekurangan, di antaranya *Database* dari aplikasi ini masih sedikit, hanya dapat menerjemahkan dari bahasa Jawa ke bahasa Indonesia, dan aplikasi ini hanya dapat dijalankan melalui sebuah komputer, oleh karenanya aplikasi ini tidak bersifat *mobile* sehingga dirasa kurang

praktis, sedangkan di era masa kini masyarakat lebih senang menggunakan sesuatu yang bersifat *mobile/portable* dan hingga saat ini belum ada pengembangan aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* yang ditujukan untuk *ponsel*, oleh karena itu tercetuslah sebuah ide untuk mengembangkan sebuah aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* yang ditujukan untuk *ponsel* dengan menggunakan teknologi J2ME. Proses pengembangan aplikasi tidak hanya sebatas membuat sebuah aplikasi saja, namun harus diketahui unjuk kerja dari aplikasi selain itu juga perlu diketahui kelayakan aplikasi agar aplikasi tersebut memiliki kualitas yang bagus baik dari segi fungsi maupun konten aplikasi. Selain sebagai sarana belajar bahasa Jawa, aplikasi ini dapat juga digunakan untuk membantu berkomunikasi bagi mahasiswa atau pelajar maupun bagi masyarakat yang berasal dari luar Jawa dan berdomisili di wilayah Jawa Tengah, Yogyakarta, dan sebagian Jawa Timur. Aplikasi yang akan dikembangkan, menggunakan teknologi *Java 2 Micro Edition* (J2ME), merupakan salah satu kategori dari bahasa pemrograman Java yang digunakan pada perangkat-perangkat kecil dengan spesifikasi memori yang terbatas. Adapun alasan menggunakan teknologi J2ME adalah karena J2ME tidak membutuhkan spesifikasi *hardware* yang terlalu besar sehingga aplikasi yang dikembangkan dapat tetap berjalan pada spesifikasi minimum yang telah ditentukan.

Saat ini begitu banyak *ponsel* dengan berbagai macam spesifikasinya, salah satunya yaitu spesifikasi sistem operasi yang digunakannya, meskipun saat ini banyak bermunculan *ponsel* dengan sistem operasi Android yang tidak mendukung teknologi J2ME, akan tetapi *ponsel* dengan sistem operasi yang

mendukung teknologi Java masih terbilang cukup banyak, diantaranya adalah *ponsel* yang menggunakan sistem operasi Java, Symbian dan Blackberry. Beberapa *ponsel* pabrikan Nokia dan Blackberry masih mendukung teknologi J2ME serta beberapa dari Samsung dan Sony Ericsson dengan tipe lama, meskipun demikian tidak berarti sistem operasi yang tidak mendukung teknologi J2ME seperti Android tidak dapat menggunakan aplikasi ini, karena tersedia *emulator* untuk menjalankan aplikasi J2ME pada sistem operasi tersebut yaitu *JBlend* dan *JBed*.

Berbagai hal tersebut yang melatarbelakangi penyusunan Tugas Akhir Skripsi dengan judul “Pengembangan Kamus Bahasa Jawa-Bahasa Indonesia Menggunakan Java 2 Micro Edition (J2ME)”, dengan harapan melalui aplikasi ini dapat membantu seseorang mempelajari bahasa Jawa dengan lebih mudah, karena dapat digunakan kapanpun dan dimanapun dibutuhkan.

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, penulis menemui beberapa masalah, di antaranya:

1. Perkembangan teknologi telekomunikasi dan informasi kurang diimbangi dengan pelestarian adat dan budaya Jawa.
2. Bahasa Jawa sudah semakin jarang digunakan di era masa kini.
3. Kamus bahasa Jawa-bahasa Indonesia konvensional memerlukan waktu yang cukup lama untuk mengartikan kata.
4. Kamus bahasa Jawa-bahasa Indonesia berbasis *Visual Basic* tidak bersifat *mobile/portable*.

5. Belum ada pengembangan kamus bahasa Jawa-bahasa Indonesia versi *mobile* pada *ponsel* menggunakan J2ME.

C. Batasan Masalah

Agar penelitian yang dilakukan menjadi lebih terarah, maka penelitian ini dibatasi pada pengembangan kamus bahasa Jawa versi *mobile* pada *ponsel* menggunakan J2ME dengan spesifikasi *Mobile Information Device Profile* (MIDP) 2.0 atau sering juga disebut Java 2.0 dan konfigurasi *Connected Limited Device Configuration* (CLDC) 1.1.

D. Rumusan Masalah

Berdasarkan batasan masalah di atas maka penelitian ini akan membahas permasalahan:

1. Bagaimana mengembangkan sebuah aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* pada *ponsel* dengan menggunakan J2ME?
2. Bagaimana unjuk kerja dari aplikasi kamus bahasa Jawa versi *mobile* pada *ponsel* menggunakan J2ME?
3. Bagaimana kelayakan dari aplikasi kamus bahasa Jawa versi *mobile* pada *ponsel* menggunakan J2ME?

E. Tujuan

Tujuan dari penelitian ini kaitannya dengan rumusah masalah di atas adalah:

1. Menghasilkan aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* pada *ponsel* menggunakan J2ME.

2. Mengetahui unjuk kerja dari aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* pada *ponsel* menggunakan J2ME.
3. Mengetahui tingkat kelayakan dari aplikasi kamus bahasa Jawa-bahasa Indonesia versi *mobile* pada *ponsel* menggunakan J2ME.

F. Manfaat

Manfaat yang didapat dari penelitian ini adalah sebagai berikut:

- **Manfaat praktis**

- a. Bagi dunia teknologi, dengan terciptanya aplikasi ini, diharapkan dapat mendorong pengembang-pengembang aplikasi yang lain untuk membuat aplikasi-aplikasi yang mengarah pada pelestarian adat dan budaya bangsa.
- b. Bagi dunia pendidikan, selain dapat digunakan untuk masyarakat umum, aplikasi ini juga dapat digunakan sebagai media belajar bahasa jawa bagi pendidik maupun peserta didik.
- c. Bagi pengguna, aplikasi kamus bahasa Jawa menggunakan J2ME diharapkan dapat membantu pengguna dalam berkomunikasi menggunakan bahasa Jawa.
- d. Bagi penulis, hasil dari penelitian ini dapat menjadi sarana untuk belajar mengembangkan aplikasi pada perangkat mobile berbasis J2ME.

- **Manfaat teoritis**

Hasil penelitian ini diharapkan dapat menjadi referensi guna penelitian lebih lanjut yang berkaitan dengan aplikasi kamus Bahasa Jawa menggunakan J2ME.

BAB II KAJIAN PUSTAKA

A. Deskripsi Teori

1. Pengembangan

Pengembangan dapat diartikan sebagai menciptakan sesuatu baik itu berupa ide, gagasan, ataupun produk. Dapat juga diartikan sebagai peningkatan ide, gagasan, atau produk yang sudah ada untuk meningkatkan kemampuannya. Suharso dan Retnoningsih (2005) dalam Aprianti (2011:8) mendefinisikan pengembangan sebagai suatu proses untuk menjadikan sesuatu (pengetahuan, pikiran, dan sebagainya) agar menjadi bertambah sempurna. Pengembangan adalah pengembangan berarti memperdalam dan memperluas pengetahuan yang telah ada (Sugiyono, 2009:3).

Menurut Undang-Undang Republik Indonesia No. 18 Tahun 2002, pengembangan adalah kegiatan ilmu pengetahuan dan teknologi yang bertujuan memanfaatkan kaidah dan teori ilmu pengetahuan yang telah terbukti kebenarannya untuk meningkatkan fungsi, manfaat, dan aplikasi ilmu pengetahuan dan teknologi yang telah ada atau menghasilkan teknologi baru.

Dapat disimpulkan bahwa pengembangan merupakan proses memperdalam dan memperluas pengetahuan yang telah ada untuk mencapai tujuan yang diinginkan agar lebih sempurna dari pada sebelumnya, serta bertujuan untuk memanfaatkan kaidah dan teori ilmu pengetahuan yang telah terbukti kebenarannya untuk meningkatkan fungsi, manfaat, dan aplikasi ilmu pengetahuan dan teknologi yang telah ada atau menghasilkan teknologi baru.

a. Pengembangan Perangkat Lunak

Pressman(2002:10) mengemukakan gambaran mengenai perangkat lunak sebagai (1) perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan. (2) struktur data yang memungkinkan program memanipulasi informasi secara proporsional, dan (3) dokumen yang menggambarkan operasi dan kegunaan program.

Pengembangan perangkat lunak adalah suatu proses dimana kebutuhan pemakai diterjemahkan menjadi produk perangkat lunak. Proses ini mencakup aktivitas penerjemahan kebutuhan pemakai menjadi kebutuhan perangkat lunak, transformasi kebutuhan perangkat lunak menjadi desain, penerapan desain menjadi kode program, uji coba kode program, dan instalasi serta pemeriksaan kebenaran perangkat lunak untuk operasional.

Berdasarkan pengertian tersebut, dapat dikatakan secara umum proses pengembangan perangkat lunak meliputi tahap-tahap:

- 1) Menentukan apa yang harus dikerjakan oleh perangkat lunak dalam satu rentang waktu tertentu.
- 2) Mendefinisikan bagaimana perangkat lunak dibuat, mencakup arsitektur perangkat lunaknya, antarmuka internal, algoritma.
- 3) Penerapan (coding) dan pengujian unit-unit program.
- 4) Integrasi dan pengujian modul-modul program.
- 5) Validasi perangkat lunak secara keseluruhan (pengujian system).

Ada beberapa model pengembangan perangkat lunak yang dapat digunakan untuk mengembangkan aplikasi, yaitu:

1) *Waterfall* atau *Sekuensial Linier*

Model pertama untuk proses pengembangan perangkat lunak dan merupakan salah satu model yang dipakai secara luas untuk pengembangan sistem yang praktis adalah model *waterfall* atau disebut juga model *sekuensial linier*.

Sekuensial linier mengusulkan sebuah pendekatan kepada pengembangan perangkat lunak yang sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan. Dimodelkan setelah siklus rekayasa konvensional, model sekuensial linier melingkupi aktivitas-aktivitas sebagai berikut (Pressman, 2002:38):

a) Rekayasa dan pemodelan sistem/informasi

Sebuah perangkat lunak merupakan bagian dari sebuah sistem ataupun perangkat lunak yang lain, oleh karena itu, pada aktivitas ini kegiatan dilakukan dengan mengumpulkan kebutuhan pada tingkat sistem dengan sejumlah kecil analisis serta desain tingkat puncak.

b) Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak, untuk memahami sifat program yang dibangun, perekayasa perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk kerja dan antarmuka (interface) yang diperlukan.

c) Desain

Proses desain menerjemahkan syarat/kebutuhan perangkat lunak yang telah didefinisikan sebelumnya ke dalam sebuah representasi perangkat lunak

yang dapat diperkirakan demi menjamin kualitas sebelum memulai pengkodean. Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural.

d) Generasi kode

Proses menerjemahkan desain ke dalam bentuk mesin yang dapat dibaca.

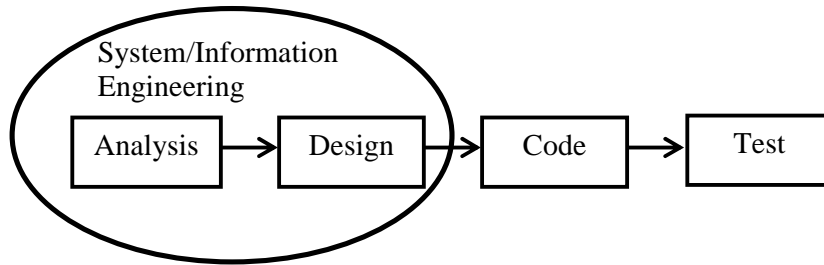
e) Pengujian

Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

f) Pemeliharaan

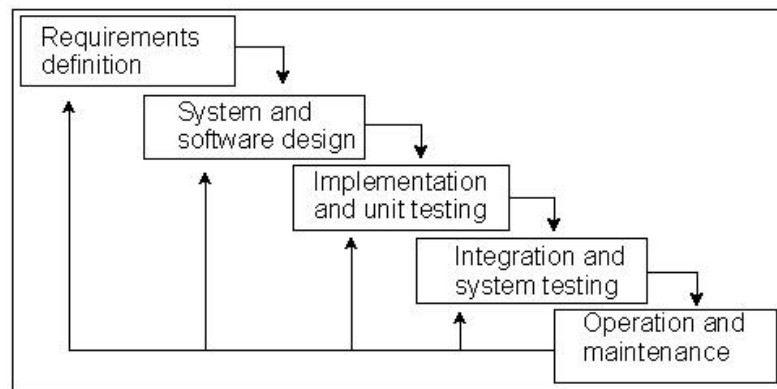
Perubahan akan terjadi pada setiap perangkat lunak, hal ini dikarenakan perubahan-perubahan di dalam lingkungan eksternalnya, misalnya perubahan yang diperlukan sebagai akibat dari perangkat *peripheral* atau sistem operasi yang baru. Pemeliharaan perangkat lunak mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.

Fase-fase dalam Sekuensial Linier Model menurut referensi Pressman (Pressman, 2002: 37) :



Gambar 1: Fase-fase Sekuensial Linier Model Pressman

Fase-fase dalam Sekuensial Linier Model menurut referensi Sommerville (Somerville, 2003: 43) :

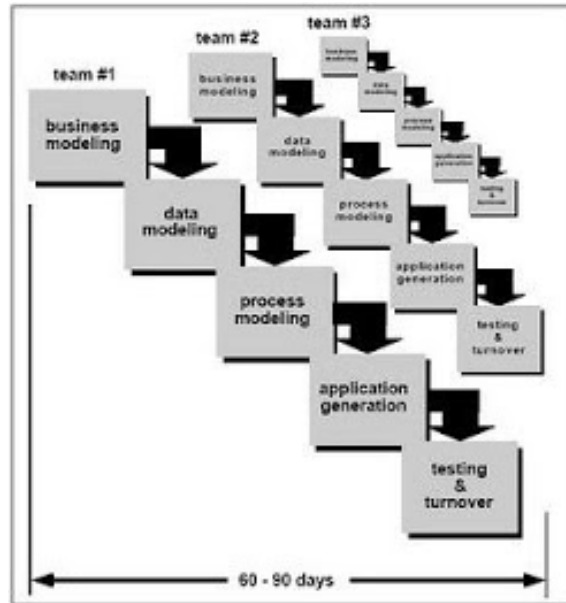


Gambar 2: Fase-fase Sekuensial Linier Model Sommerville

2) *Rapid Application Development (RAD)*

Rapid Application Development (RAD) adalah sebuah model proses perkembangan software sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier di mana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen. Jika kebutuhan dipahami dengan baik, proses RAD memungkinkan tim pengembangan menciptakan

“sistem fungsional yang utuh” dalam periode waktu yang sangat pendek (kira-kira 60 sampai 90 hari). Berikut ini adalah gambar dari model pengembangan RAD:



Gambar 3: Model Pengembangan *Rapid Application Development* (RAD)

Berikut ini penjelasan dari tahapan pada model pengembangan RAD:

a) *Business modeling*

Aliran informasi di antara fungsi–fungsi bisnis dimodelkan dengan suatu cara untuk menjawab pertanyaan–pertanyaan mengenai informasi apa yang mengendalikan proses bisnis, informasi apa yang di munculkan, siapa yang memunculkannya, ke mana informasi itu pergi, dan siapa yang memprosesnya.

b) *Data modeling*

Aliran informasi yang didefinisikan sebagai bagian dari fase bussiness modelling disaring ke dalam serangkaian objek data yang dibutuhkan untuk menopang bisnis tersebut. Karakteristik (disebut atribut) masing–masing objek diidentifikasi dan hubungan antara objek – objek tersebut didefinisikan.

c) *Prosess modelling*

Aliran informasi yang didefinisikan di dalam fase data modeling ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali sebuah objek data.

d) *Aplication generation*

RAD mengasumsikan pemakaian teknik generasi ke empat. Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memkai lagi komponen program yang ada (pada saat memungkinkan) atau menciptakan komponen yang bisa dipakai lagi (bila perlu). Pada semua kasus, alat-alat bantu otomatis dipakai untuk memfasilitasi konstruksi perangkat lunak.

e) *Testing and turnover*

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen program telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tetapi komponen baru harus di uji dan semua *interface* harus dilatih secara penuh.

3) *Prototyping*

Model pengembangan ini menggunakan sistem informasi yang menggambarkan hal-hal penting dari sistem informasi yang akan datang. *Prototype* sistem informasi bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dimodifikasi kembali, dikembangkan, ditambahkan atau digabungkan dengan sistem informasi yang lain bila perlu.

Siklus dalam model ini melalui tiga proses yaitu:

- a) Pengumpulan kebutuhan: developer dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Detail kebutuhan mungkin tidak dibicarakan disini, pada awal pengumpulan kebutuhan.
- b) Perancangan : perancangan dilakukan cepat dan rancangan mewakili semua aspek software yang diketahui, dan rancangan ini menjadi dasar pembuatan prototype.
- c) Evaluasi prototype: klien mengevaluasi prototype yang dibuat dan digunakan untuk memperjelas kebutuhan software

4) *Spiral Model*

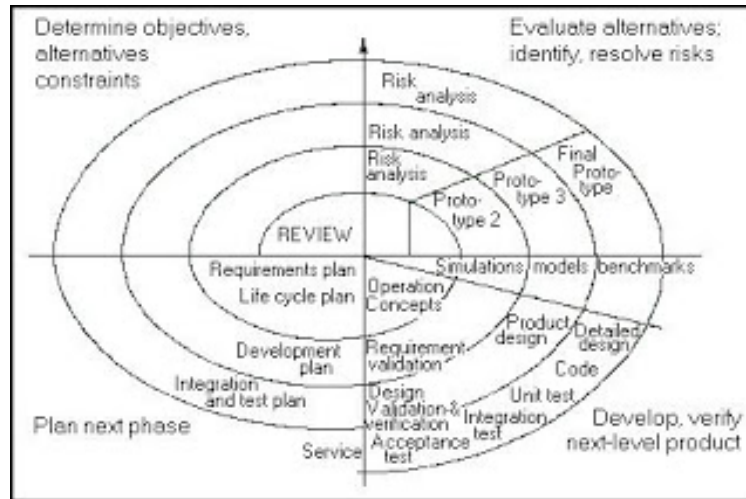
Model ini cukup baru ditemukan,yaitu pada tahun 1988 oleh Barry Boehm. Spiral adalah salah satu bentuk evolusi yang menggunakan metode iterasi natural yang dimiliki oleh model prototyping dan digabungkan dengan aspek sistematis yang dikembangkan model *waterfall*.

Spiral model dibagi menjadi beberapa framework aktivitas, yang disebut dengan task regions. Kebanyakan aktivitas-aktivitas tersebut dibagi antara tiga sampai enam aktivitas. Berikut adalah aktivitas-aktivitas yang dilakukan dalam spiral model:

- a) *Customer communication*, aktivitas yang dibutuhkan untuk membangun komunikasi yang efektif antara developer dengan user / customer terutama mengenai kebutuhan dari customer.

- b) *Planning*, aktivitas perencanaan ini dibutuhkan untuk menentukan sumberdaya, perkiraan waktu pengerjaan, dan informasi lainnya yang dibutuhkan untuk pengembangan software.
- c) *Analysis risk*, aktivitas analisis resiko ini dijalankan untuk menganalisis baik resiko secara teknikal maupun secara manajerial. Tahap inilah yang mungkin tidak ada pada model proses yang juga menggunakan metode iterasi, tetapi hanya dilakukan pada spiral model.
- d) *Engineering*, aktivitas yang dibutuhkan untuk membangun 1 atau lebih representasi dari aplikasi secara teknikal.
- e) *Construction & Release*, aktivitas yang dibutuhkan untuk develop software, testing, instalasi dan penyediaan user / costumer support seperti training penggunaan software serta dokumentasi seperti buku manual penggunaan software.
- f) *Customer evaluation*, aktivitas yang dibutuhkan untuk mendapatkan feedback dari user / customer berdasarkan evaluasi mereka selama representasi software pada tahap engineering maupun pada implementasi selama instalasi software pada tahap construction and release.

Berikut ini gambar dari model *Spiral* secara garis besar:



Gambar 4: **Model Pengembangan Spiral**

5) *Incremental Model*

Incremental model dalam rekayasa perangkat lunak, menerapkan rekayasa perangkat lunak perbagian, hingga menghasilkan perangkat lunak yang lengkap. Proses membangun berhenti jika produk telah mencapai seluruh fungsi yang diharapkan.

Adapun beberapa tahapan yang ada pada model *incremental* dimana tahapan-tahapan tersebut dilakukan secara berurutan. Setiap bagian yang sudah selesai dilakukan testing, dikirim ke pemakai untuk langsung dapat digunakan.

Tahapan *Incremental Model* adalah :

- a) *Requirement*
- b) *Specification*
- c) *Architecture Design*

6) ADDIE Model

ADDIE merupakan singkatan dari *Analysis, Design, Development or Production, Implementation or Delivery and Evaluations*. Menurut langkah-langkah pengembangan produk. Model ini dapat digunakan untuk berbagai macam bentuk pengembangan produk seperti model, strategi pembelajaran, metode pembelajaran, media dan bahan ajar.

Model ADDIE dikembangkan oleh Dick and Carry untuk merancang sistem pembelajaran (Endang Mulyatiningsih, 2012:5). Menurut Benny A. Pribadi (2009:125) model pengembangan ADDIE terdiri atas 5 langkah pokok, yaitu *Analysis, Design, Development, Implementation, dan Evaluation*. Tahapan penelitian tersebut adalah sebagai berikut.

a) *Analisis*

Kegiatan utama pada tahap ini adalah menganalisis perlunya pengembangan media/metode pembelajaran baru dan menganalisis kelayakan dan syarat-syarat pengembangan media pembelajaran baru. Pengembangan media pembelajaran baru diawali oleh adanya masalah dalam media pembelajaran yang sudah diterapkan. Setelah analisis masalah perlunya pengembangan media pembelajaran baru, peneliti juga perlu menganalisis kelayakan dan syarat-syarat pengembangan media pembelajaran baru tersebut.

b) *Design*

Kegiatan ini merupakan proses sistematis yang dimulai dari menetapkan tujuan belajar, merancang skenario atau kegiatan belajar mengajar, merancang perangkat pembelajaran, merancang materi pembelajaran dan alat

evaluasi hasil belajar. Rancangan media/metode pembelajaran ini masih bersifat konseptual dan akan mendasari proses pengembangan berikutnya.

c) *Development*

Development dalam model ADDIE berisi kegiatan realisasi rancangan produk. Dalam tahap desain, telah disusun kerangka konseptual media/metode pembelajaran yang baru. Dalam tahap pengembangan, kerangka yang masih konseptual tersebut direalisasikan menjadi produk yang siap diimplementasikan.

d) *Implementation*

Pada tahap ini diimplementasikan media/metode yang telah dikembangkan pada situasi yang nyata yaitu di kelas. Selama implementasi, rancangan media/metode yang telah dikembangkan diterapkan pada kondisi yang sebenarnya. Setelah penerapan, dilakukan evaluasi awal untuk memberi umpan balik pada penerapan media/metode berikutnya.

e) *Evaluation*

Evaluasi dilakukan dalam dua bentuk yaitu evaluasi formatif dan sumatif. Evaluasi formatif dilaksanakan pada akhir setiap tahap sedangkan evaluasi sumatif dilakukan setelah kegiatan berakhir secara keseluruhan. Revisi dibuat sesuai dengan hasil evaluasi atau kebutuhan yang belum dapat dipenuhi oleh media/metode tersebut.

Dari berbagai model pengembangan perangkat lunak tersebut, model ADDIE dirasa paling tepat untuk mengembangkan aplikasi kamus bahasa Jawa-bahasa Indonesia ini karena model ini cukup sederhana namun tetap kompleks pada setiap tahapan-tahapannya.

b. Pengembangan Kamus

Pengembangan kamus merupakan proses untuk meningkatkan fungsi dengan menerapkan ilmu pengetahuan dan teknologi untuk menambah nilai manfaat dari kamus. Kamus konvensional dengan bentuk buku dirasa masih kurang efektif, karena memerlukan waktu yang cukup lama untuk menerjemahkan arti dari suatu kata, sehingga diperlukan pengembangan dari kamus konvensional menjadi kamus *digital*. Salah satu pengembangan kamus konvensional menjadi kamus *digital* adalah aplikasi kamus bahasa Jawa-bahasa Indonesia yang dikembangkan oleh Feriawan Agung Nugroho yang menggunakan *Visual Basic* sebagai basis dari pembuatan aplikasi. Aplikasi ini hanya dapat menerjemahkan kata dari bahasa Jawa (*ngoko*) ke bahasa Indonesia. Aplikasi ini juga dapat menambah kosakata baru ataupun menghapus kosakata yang sudah ada. *Database* dari aplikasi juga masih cukup sedikit yaitu 1883 kosakata. Berikut ini tampilan dari aplikasi:



Gambar 5: Aplikasi Kamus Bahasa Jawa (*Ngoko*)-Bahasa Indonesia Menggunakan *Visual Basic*

2. Kamus

Kamus Besar Bahasa Indonesia versi *daring* (dalam jaringan), mendefinisikan kamus sebagai (1) buku acuan yg memuat kata dan ungkapan, biasanya disusun menurut abjad berikut keterangan tentang makna, pemakaian, atau terjemahannya; (2) buku yg memuat kumpulan istilah atau nama yg disusun menurut abjad beserta penjelasan tentang makna dan pemakaiannya.

Kata kamus diserap dari bahasa Arab, *qamus* (), dengan bentuk jamaknya *qawamis*. Kata Arab itu sendiri berasal dari kata Yunani (*okeanos*) yang berarti *samudera* (Ahmad, 2002). Sejarah kata itu jelas memperlihatkan makna dasar yang terkandung dalam kata kamus, yaitu wadah pengetahuan, khususnya pengetahuan bahasa, yang tidak terhingga dalam dan luasnya.

Suryawinata (2003:28) menggolongkan kamus berdasarkan bahasa yang digunakan, yakni kamus ekabahasa, kamus dwibahasa, dan kamus aneka bahasa. Kamus ekabahasa adalah kamus yang hanya menggunakan satu bahasa saja, contohnya adalah Kamus Besar Bahasa Indonesia, *Oxford Advanced Dictionary*, dan Bausastra Jawa. Kamus dwibahasa menggunakan dua bahasa, contohnya adalah Kamus Indonesia-Jawa seperti yang digunakan dalam kamus ini. Kamus aneka bahasa berisi padanan kata atau keterangan tentang arti kata di dalam dua bahasa atau lebih. Contohnya adalah kamus bahasa Inggris-Indonesia-Arab. Perlu diperhatikan bahwa kamus dwibahasa dan aneka bahasa tidak selalu sesuai untuk mencari makna suatu kata karena makna kata di dalam satu bahasa tidak sama benar dengan makna kata dalam bahasa lain.

Berdasarkan beberapa pengertian tersebut, maka dapat disimpulkan bahwa kamus merupakan sumber penyedia informasi dengan cara menterjemahkan atau mengartikan bahasa asing ke dalam bahasa yang dikenal oleh penggunanya dan aplikasi kamus bahasa Jawa-bahasa Indonesia yang dikembangkan merupakan jenis kamus dwibahasa.

3. Bahasa

Bahasa dapat dikatakan sebagai sarana untuk dapat berkomunikasi, hal ini diperkuat oleh Kridalaksana (2001:21) yang mendefinisikan bahasa sebagai sistem lambang bunyi yang arbitrer yang digunakan oleh anggota suatu masyarakat untuk bekerjasama, berinteraksi, dan mengidentifikasi diri.

Wibowo (2003:3) menyatakan bahwa bahasa adalah sistem symbol bunyi bermakna dan berartikulasi (dihasilkan oleh alat ucap), yang bersifat arbitrer dan konvensional, yang dipakai sebagai alat berkomunikasi oleh sekelompok manusia untuk melahirkan perasaan dan pikiran.

Berdasarkan pengertian tersebut dapat disimpulkan bahwa bahasa merupakan alat berkomunikasi dalam bentuk sistem lambing bunyi yang arbitrer yang digunakan oleh sekelompok masyarakat untuk berinteraksi.

a. Bahasa Indonesia

Bahasa Indonesia adalah bahasa resmi Republik Indonesia dan bahasa persatuan bangsa Indonesia sifatnya sebagai bahasa pengantar bagi masyarakat Indonesia yang berbeda etnis maupun bahasa ibu. Bahasa Indonesia diresmikan penggunaannya setelah Proklamasi Kemerdekaan Indonesia, tepatnya sehari sesudahnya, bersamaan dengan mulai berlakunya konstitusi.

Bahasa Indonesia yang akan digunakan dalam pembuatan program kamus ini sudah disesuaikan dengan ejaan yang disempurnakan (EYD) sesuai dengan Kamus Besar Bahasa Indonesia (KBBI) dan juga bentuk penulisan setiap katanya sama dengan penulisan kamus yang ada pada umumnya.

b. Bahasa Jawa

Menurut Harimurti Kridalaksana (2001:xxx) dalam bukunya yang berjudul “Wiwara: Pengantar Bahasa dan Kebudayaan Jawa”, bahasa Jawa adalah bahasa ibu yang digunakan oleh masyarakat yang berasal dari wilayah Jawa Tengah dan sebagian besar Jawa Timur. Luasnya wilayah dan kendala geografis menyebabkan bahasa Jawa dapat dikelompokkan berdasarkan letak geografisnya, yakni dialek Jawa Timur, dialek Jawa Tengah, dialek Pesisir Utara, dialek Pesisir Selatan, dan dialek perbatasan Jawa Tengah dan Jawa Barat (dialek Cirebon dan dialek Banten). Meski demikian, yang dianggap Bahasa Jawa Baku adalah bahasa Jawa yang digunakan di wilayah Yogyakarta dan Surakarta.

Bahasa Jawa mengenal *unggah-ungguh basa*, yaitu variasi penggunaan bahasa Jawa. Secara umum variasi ini terdiri dari *ngoko* (kasar), *krama madya* (biasa), *krama inggil* (halus). Penggunaan dari variasi ini sendiri tergantung pada lawan bicara dan statusnya, status bisa dilihat berdasarkan usia maupun posisi sosialnya. Bahasa Jawa yang digunakan dalam pembuatan aplikasi kamus *mobile* ini, bentuk penulisannya sama dengan penulisan kamus yang ada pada umumnya.

Bahasa Jawa yang akan digunakan dalam pembuatan program kamus ini merujuk pada Kamus Lengkap Jawa-Indonesia Indonesia-Jawa yang disusun oleh Drs. Sudarmanto.

4. Java 2 Micro Edition (J2ME)

J2ME merupakan salah satu kategori bahasa pemrograman Java yang ditujukan untuk perangkat-perangkat dengan kapasitas memori terbatas, ukuran layar yang kecil, bandwidth yang rendah, dan menggunakan daya yang kecil dibandingkan dengan perangkat-perangkat desktop pada umumnya. Alasan penggunaan J2ME sebagai basis dari aplikasi ini adalah karena J2ME tidak membutuhkan spesifikasi *hardware* yang terlalu besar sehingga aplikasi yang dikembangkan dapat tetap berjalan pada spesifikasi minimum yang telah ditentukan. J2ME juga masih dapat berjalan pada beberapa *platform* yang berbeda

5. Object Oriented Programming (OOP)

Pemrograman berorientasi objek atau yang lebih lazim dikenal dengan Object Oriented Programming (OOP) adalah inti dari pemrograman java. Semua program java merupakan objek. Seperti yang telah diketahui, semua program komputer terdiri atas kode dan data. Kode yang dimaksud di sini adalah proses logis program (misalnya fungsi), sedangkan data dapat berupa variabel maupun konstanta. Selanjutnya, kode dan data tersebut secara konseptual diatur sedemikian rupa sehingga dapat bekerja sama untuk menghasilkan program keluaran sesuai dengan proses yang diinginkan. Pada model pemrograman tradisional atau prosedural atau terstruktur(disebut process-oriented model), semua data dan kode digabung menjadi satu bagian dalam satu program. Untuk program-program sederhana yang hanya membutuhkan beberapa baris kode, penggunaan model ini tentu tidak begitu menjadi masalah. Permasalahan baru akan timbul pada saat program tersebut berkembang ke arah yang lebih besar dan

kompleks, yang mungkin membutuhkan ratusan bahkan ribuan baris kode. Alasan semacam inilah yang menjadi benih diciptakannya model pemrograman berorientasi objek. Pada model ini, komponen-komponen penyusun program secara konseptual akan dipecah menjadi bagian-bagian tersendiri yang disebut dengan objek.

Beberapa ciri dari pemrograman berorientasi objek adalah abstraksi, pembungkusan, pewarisan, dan polimorfisme.

a. Abstraksi

Abstraksi adalah pengabstrakan atau penyembunyian kerumitan dari suatu proses. Contoh abstraksi dalam model pemrograman berorientasi objek, misalkan saja seorang programmer atau pengembang program tidak perlu tahu secara detail bagaimana membuat sebuah tombol, yang perlu diketahui adalah bagaimana cara menggunakan objek tombol yang sudah ada (baik data/properti maupun *method*-nya), dalam kasus ini si pembuat objek telah mengabstraksikan semua proses yang berkaitan dengan pembuatan tombol.

b. Pembungkusan

Sebenarnya proses pembungkusan itu sendiri merupakan cara atau mekanisme untuk melakukan abstraksi. Dalam pemrograman tradisional, proses semacam ini sering dinamakan penyembunyian informasi. Dalam melakukan pembungkusan kode dan data di dalam Java, terdapat tiga tingkat akses, yaitu:

1) Tingkat akses *private*

Saat mendeklarasikan data dan method menggunakan tingkat akses *private*, maka data dan method tersebut hanya dapat diakses oleh kelas yang

memilikinya saja. Ini berarti data dan method tersebut tidak boleh diakses atau digunakan oleh kelas-kelas lain yang terdapat di dalam program.

2) Tingkat akses *protected*

Data dan method dengan tingkat akses *protected* dapat diakses oleh kelas yang memilikinya dan juga oleh kelas-kelas yang memiliki hubungan turunan.

3) Tingkat Akses *public*

Data dan method yang bersifat *public* dapat diakses oleh semua kelas di dalam program.

c. Pewarisan

Pewarisan pada dasarnya adalah mengizinkan objek-objek pada suatu kelas untuk mengakses data-data serta fungsi-fungsi yang termuat dalam kelas yang lebih dahulu terdefinisi tanpa harus melakukan definisi kembali. Ini memungkinkan untuk menciptakan kelas baru yang merupakan perluasan atau spesialisasi dari kelas lain yang telah ada. Kelas yang baru merupakan turunan dari kelas dasar.

d. Polimorfisme

Polimorfisme adalah kemampuan suatu objek untuk mengungkap banyak hal melalui satu cara yang. Sebagai contoh, terdapat kelas A yang diturunkan menjadi kelas B, C, dan D, dengan konsep polimorfisme, kita dapat menjalankan kumpulan method yang terdapat pada kelas B, C, dan D hanya dari objek yang diinstansiasi dengan kelas A.

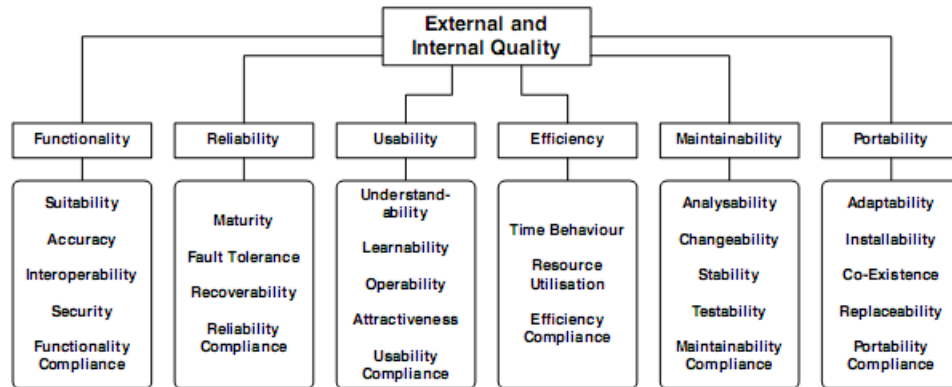
6. Kualitas Perangkat Lunak

Kualitas merupakan atribut dari sesuatu yang dapat diukur sesuai dengan standar yang telah ditentukan. Pressman (2002:217) mengemukakan kualitas yang didefinisikan oleh *American Heritage Dictionary*, bahwa kata kualitas sebagai “sebuah karakteristik atau atribut dari sesuatu” yang dapat diukur – sesuatu yang dapat dibandingkan dengan standar yang sudah diketahui, seperti panjang, warna, sifat kelistrikan, kelunakan, dan sebagainya, akan tetapi pada perangkat lunak, yang sebagian besar merupakan entitas intelektual, lebih menantang untuk dikarakterisasi daripada objek fisik.

Kualitas memegang peranan penting dalam pengembangan perangkat lunak, yaitu untuk menjamin layak tidaknya sebuah aplikasi untuk dipublikasikan, bagi para pengembang perangkat lunak sudah pasti kualitas yang tinggi menjadi tujuan utama dalam proses pengembangan perangkat lunak. Kualitas perangkat lunak diukur berdasarkan kriteria-kriteria tertentu sesuai dengan standar yang digunakan.

a. ISO 9126-1

ISO 9126-1 merupakan standar internasional dari evaluasi produk perangkat lunak. Kualitas perangkat lunak berdasarkan ISO 9126-1 terbagi menjadi 2 yakni kualitas internal dan eksternal. ISO 9126-1 menggunakan model yang sama namun dengan matrik yang berbeda. Model ISO 9126-1 secara umum terbagi menjadi enam karakteristik yaitu, *functionality*, *reliability*, *usability*, *efficiency*, *maintainability*, dan *portability*, seperti yang ditunjukkan pada gambar dibawah ini,



Gambar 6: **Kualitas Internal dan Eksternal Model ISO 9126-1**

Functionality, kemampuan menutupi fungsi produk perangkat lunak yang menyediakan kepuasan kebutuhan user. *Reliability*, kemampuan perangkat lunak untuk perawatan dengan level performansi. *Usability*, kemampuan yang berhubungan dengan penggunaan perangkat lunak. *Efficiency*, kemampuan yang berhubungan dengan sumber daya fisik yang digunakan ketika perangkat lunak dijalankan. *Maintainability*, kemampuan yang dibutuhkan untuk membuat perubahan perangkat lunak. *Portability*, kemampuan yang berhubungan dengan kemampuan perangkat lunak yang dikirim ke lingkungan berbeda.

b. *Unified Testing Criteria for Java ME Application*

Unified Testing Criteria (UTC) for Java ME application merupakan seperangkat kriteria pengujian yang digunakan untuk menguji sebuah aplikasi berbasis J2ME. UTC disusun oleh beberapa anggota dari *Unified Testing Initiative (UTI)* dan merupakan kriteria yang disetujui oleh *Java Verified*, di mana *Java Verified* inilah yang mengatur perizinan dan garansi pada sebuah aplikasi J2ME yang tentunya melewati suatu proses pengujian dan verifikasi melalui *Java Verified Submissions portal*, jika sebuah aplikasi memenuhi kriteria pengujian,

maka aplikasi tersebut mendapatkan tanda persetujuan dari *Java Verified* dan bisa dipublikasikan dengan status terpercaya, namun jika tidak memenuhi kriteria bukan berarti aplikasi tersebut tidak diperbolehkan untuk dipublikasikan hanya saja statusnya menjadi aplikasi yang tidak terpercaya dan *Java Verified* tidak memberikan garansi apapun jika aplikasi tersebut tidak dapat berjalan atau bahkan menimbulkan kerugian pada perangkat yang diinstal. UTC mengusulkan pengorganisasian *test case* yang terbagi menjadi sepuluh kategori yang berbeda, yaitu:

1. *Application Characteristics*, informasi mengenai aplikasi yang ditujukan untuk membantu penguji dalam melakukan pengujian.
2. *Stability*, fokus pada kestabilan aplikasi terhadap perangkat, dalam hal ini adalah *ponsel*.
3. *Application Launch*, sekali aplikasi dijalankan harus mulai dan berhenti dengan benar, hal ini berhubungan dengan *ponsel* dan aplikasi lain yang ada pada *ponsel* tersebut.
4. *User Interface Requirements*, tujuannya bukan sepenuhnya terfokus pada bagaimana mendesain sebuah *user interface*, tetapi lebih kepada memberi panduan atau petunjuk mengenai aplikasi tersebut.
5. *Localization*, aplikasi yang akan disebarluaskan ke masyarakat luas, lebih dari daerah asalnya harus menyediakan perubahan format bahasa, alfabet, tanggal, mata uang, dan sebagainya.

6. *Functionality*, fitur dokumentasi diterapkan pada aplikasi dan bekerja seperti yang diharapkan. Sumber informasi dapat berupa *user manuals*, dokumentasi khusus dari aplikasi, dan dokumentasi *online*.
7. *Connectivity*, jika sebuah aplikasi menggunakan komunikasi melalui jaringan atau internet maka harus didemonstrasikan kemampuannya dalam berkomunikasi menggunakan jaringan atau internet dengan benar. Aplikasi juga harus mampu menangani permasalahan jaringan dan permasalahan pada server.
8. *Personal Information Management*, ketika aplikasi mengakses informasi user maka informasi ini harus digunakan dengan cara yang tepat dan bukan untuk merusaknya.
9. *Security*, mencatat segala permasalahan keamanan saat pengujian pada aplikasi.
10. *Retesting*, tes khusus yang ditujukan hanya untuk menguji ulang, jika pada tes sebelumnya menemui kesalahan.

Sebuah aplikasi tidak selamanya menggunakan semua fitur yang ada pada J2ME misalnya seperti fitur komunikasi dan izin untuk pengaksesan data user, jika aplikasi seperti ini harus melalui pengujian dengan menggunakan semua kriteria yang ada tentunya ada kemungkinan satu atau beberapa kategori tes tidak bisa diterapkan, oleh karena itu UTC membagi tes menjadi dua yakni *normal tests (full criteria tests)* dan *simple application tests*. Izin pengaksesan ini dapat dilihat pada file *manifest* dan *Java Application Descriptor (JAD)*.

c. Aspek dan kriteria penilaian media pembelajaran

Aplikasi ini berisi kosakata dan terjemahannya baik itu dari bahasa Jawa-bahasa Indonesia maupun dari bahasa Indonesia-bahasa Jawa, oleh karena itu kebenaran materi merupakan hal yang perlu diperhatikan dan dinilai. Teori yang digunakan adalah teori yang diadaptasi dari kriteria pengembangan media pembelajaran oleh Wahono (2006), Dikmenum (2008: 2-3) dalam Prayoga (2011:34) yang meliputi:

- 1) Aspek umum
 - a) Kreatif dan inovatif (baru, luwes, menarik, cerdas, unik, dan tidak asal beda),
 - b) Komunikatif (mudah dipahami serta menggunakan bahasa yang baik, benar, dan efektif),
 - c) Unggul (memiliki kelebihan dibanding multimedia pembelajaran lain ataupun dengan cara konvensional).
- 2) Aspek Substansi Materi
 - a) Kebenaran materi secara teori dan konsep,
 - b) Ketepatan penggunaan istilah sesuai bidang keilmuan,
 - c) Kedalaman materi,
 - d) Aktualitas.
- 3) Aspek pembelajaran
 - a) Kejelasan tujuan pembelajaran (realistis dan terstruktur),
 - b) Relevansi tujuan pembelajaran dengan Kurikulum/SK/KD,
 - c) Kesesuaian antara materi, media dan evaluasi dengan tujuan pembelajaran,
 - d) Sistematika yang runut, logis, dan jelas,

- e) Interaktivitas,
 - f) Penumbuhan motivasi belajar,
 - g) Kontekstualitas,
 - h) Kelengkapan dan kualitas bahan bantuan belajar,
 - i) Kejelasan uraian materi, pembahasan, contoh, simulasi, latihan,
 - j) Relevansi dan konsistensi alat evaluasi,
 - k) Konsistensi evaluasi dengan tujuan pembelajaran,
 - l) Pemberian umpan balik terhadap latihan dan hasil evaluasi.
- 4) Aspek rekayasa perangkat lunak
 - a) Efektif dan efisien dalam pengembangan maupun penggunaan,
 - b) Reliabilitas (kehandalan),
 - c) Maintainabilitas (dapat dipelihara/dikelola dengan mudah),
 - d) Usabilitas (mudah digunakan dan sederhana dalam pengoperasian),
 - e) Ketepatan pemilihan jenis aplikasi/multimedia/*tool* untuk pengembangan.
 - f) Kompatibilitas (dapat diinstalasi dan dijalankan di berbagai *hardware* dan *software* yang ada),
 - g) Pemaketan multimedia pembelajaran secara terpadu dan mudah dalam eksekusi,
 - h) Dokumentasi multimedia pembelajaran yang lengkap meliputi: petunjuk instalasi (jelas, singkat, lengkap), penggunaan, *troubleshooting* (jelas, terstruktur, dan antisipatif), desain program (jelas dan menggambarkan alur kerja program),

- i) Reusabilitas (sebagian atau seluruh multimedia pembelajaran dapat dimanfaatkan kembali untuk mengembangkan multimedia pembelajaran lain).
- 5) Aspek komunikasi visual
 - a) Komunikatif: unsur visual dan audio mendukung materi ajar, agar mudah diterima oleh siswa,
 - b) Kreatif: visualisasi diharapkan disajikan secara unik dan tidak klise (sering digunakan), agar menarik perhatian,
 - c) Sederhana: visualisasi tidak rumit, agar tidak mengurangi kejelasan isi materi ajar dan mudah diingat,
 - d) *Unity*: menggunakan bahasa visual dan audio yang harmonis, utuh, dan senada, agar materi ajar dipersepsi secara utuh (komprehensif),
 - e) Penggambaran objek dalam bentuk *image* (citra) baik realistik maupun simbolik,
 - f) Pemilihan warna yang sesuai, agar mendukung kesesuaian antara konsep kreatif dan topik yang dipilih,
 - g) Tipografi (*font* dan susunan huruf), untuk memvisualisasikan bahasa verbal agar mendukung isi pesan, baik secara fungsi keterbacaan maupun fungsi psikologisnya,
 - h) Tata letak (*layout*): peletakan dan susunan unsur-unsur visual terkendali dengan baik, agar memperjelas peran dan hirarki masing-masing unsur tersebut,

- i) Unsur visual bergerak (animasi dan/atau *movie*), animasi dapat dimanfaatkan untuk mensimulasikan materi ajar dan *movie* untuk mengilustrasikan materi secara nyata,
- j) Navigasi yang familiar dan konsisten agar efektif dalam penggunaannya,
- k) Unsur audio (dialog, monolog, narasi, ilustrasi musik, dan *sound/special effect*) sesuai dengan karakter topic dan dimanfaatkan untuk memperkaya imajinasi.

B. Penelitian Relevan

Hasil penelitian yang dilakukan Lenny Ike C. M. (tahun tidak diketahui) terhadap aplikasi kamus Inggris-Indonesia berbasis J2ME menunjukkan bahwa aplikasi ini dapat membantu mempermudah untuk menerjemahkan kata dan membantu *user* dalam belajar bahasa Inggris.

Hasil penelitian yang dilakukan Indres Deswita (2009) terhadap aplikasi kamus *Indonesia-English-Italy* pada perangkat mobile dengan J2ME yang telah diujicobakan pada beberapa ponsel yang mendukung Java dari berbagai merek dan model ponselnya menunjukkan bahwa aplikasi ini dapat bekerja dengan baik pada ponsel, namun kecepatan program berjalan tergantung dari memori ponsel tersebut, semakin besar memori yang tersedia maka semakin cepat pula jalannya program kamus mobile ini, begitu pula sebaliknya.

Aplikasi kamus bahasa Jawa (*ngoko*)-bahasa Indonesia yang dikembangkan oleh Feriawan Agung Nugroho, menggunakan *Visual Basic* sebagai basis dari pembuatan aplikasi. Aplikasi ini hanya dapat menerjemahkan kata dari bahasa Jawa (*ngoko*) ke bahasa Indonesia. Aplikasi ini juga dapat

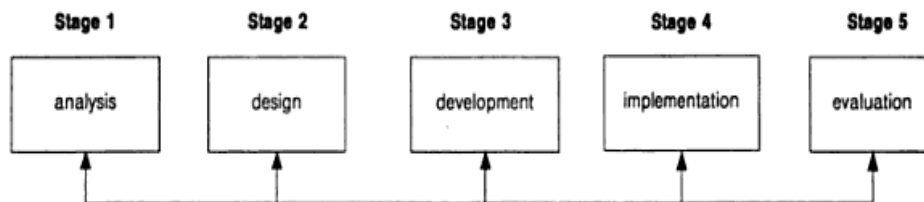
menambah kosakata baru ataupun menghapus kosakata yang sudah ada. *Database* dari aplikasi juga masih cukup sedikit yaitu 1883 kosakata. Karena aplikasi ini hanya dapat digunakan pada komputer maka aplikasi ini kurang praktis dan tidak bersifat *mobile*.

BAB III METODOLOGI PENELITIAN

A. Desain Penelitian

Penelitian ini secara umum bertujuan untuk mengembangkan suatu produk yaitu aplikasi kamus bahasa Jawa berbasis J2ME, sehingga metode penelitian yang digunakan adalah metode penelitian dan pengembangan (*Research and Development*). Menurut Borg dan Gall (1983) dalam Setyosari (2010:194) penelitian penembangan adalah suatu proses yang dipakai untuk mengembangkan dan memvalidasi produk pendidikan. Setiap tahapan dalam penelitian dan pengembangan dilakukan dengan mengacu pada tahapan sebelumnya. Sugiyono (2012:298) mengungkapkan langkah-langkah dalam penelitian dan pengembangan yang digambarkan sebagai berikut :

Model pengembangan yang digunakan pada penelitian ini yaitu menggunakan model ADDIE yang dikembangkan oleh Ellington dan Aris (2000 : 12). Tahapan-tahapan dalam model ADDIE digambarkan sebagai berikut:



Gambar 7: Model Pengembangan ADDIE

Tahapan-tahapan di atas dijabarkan sebagai berikut:

1. Tahap Analisis

Tahap analisis bertujuan untuk mengumpulkan informasi yang sesuai dengan kebutuhan pengembangan aplikasi *mobile* yang dilakukan. Tahap analisis meliputi analisis isi program, pengguna, *software* dan *hardware*.

a. Analisis isi program

Analisis ini dilakukan untuk menelaah tujuan, mendapatkan informasi mengenai konten atau materi yang tepat untuk diterapkan dalam aplikasi *mobile* yang dikembangkan.

b. Analisis pengguna

Analisis kebutuhan pengguna diperlukan agar pengembangan aplikasi tersebut tepat sasaran dan dapat difungsikan dengan baik sebagai kamus bahasa Jawa yang layak digunakan.

c. Analisis *software*

Tahap analisis ini dilakukan untuk mengetahui *software* apa saja yang dapat mendukung pengembangan aplikasi *mobile*. Analisis *software* yang dimaksud disini terbagi menjadi dua, yang pertama *software* yang digunakan dalam pembuatan program dan yang kedua adalah *software* yang digunakan *ponsel* untuk menjalankan aplikasi kamus bahasa Jawa tersebut, dalam hal ini adalah sistem operasi dari *ponsel* tersebut.

d. Analisis *hardware*

Analisis kebutuhan *hardware* dilakukan untuk mengetahui syarat minimal sebuah perangkat ditinjau dari segi *hardware* yang digunakan untuk

mengembangkan aplikasi *mobile* dan dari segi *hardware* yang digunakan untuk pengaplikasian aplikasi *mobile*, agar dapat mengakomodasi pengembangan dan penggunaan aplikasi *mobile* tersebut, sehingga baik dalam pengembangan maupun pada saat penggunaan aplikasi *mobile* dapat berjalan lancar dalam pengoperasiannya.

2. Tahap Desain

Tahap desain atau perancangan dilakukan untuk mempermudah *programer* dalam menerjemahkan desain ke dalam bahasa pemrograman. Tahap ini merupakan perancangan aplikasi berdasarkan penelaahan pada tahap analisis. Perancangan dilakukan dalam empat tahapan yaitu :

a. *Material collecting*

Material collecting adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang dikerjakan.

b. Pembuatan *flowchart*

Flowchart adalah bagan yang terdiri dari simbol-simbol tertentu yang menunjukkan langkah-langkah suatu prosedur atau program.

c. Rancangan antarmuka pengguna

Sebelum sebuah aplikasi dibuat dan di-*coding* kan, perlu dibuat rancangan antarmuka pengguna, hal ini dilakukan dengan tujuan untuk mempermudah dan menghemat waktu proses produksi.

3. Tahap Pengembangan

a. Produksi aplikasi *mobile*

Tahap ini merupakan tahap dimana mulai dilakukan pengembangan aplikasi *mobile* sesuai dengan desain yang telah dibuat. Proses produksi meliputi pembuatan antarmuka, *coding*, *testing*, dan *deploying*.

1) Pembuatan antarmuka

Pembuatan antarmuka dilakukan untuk mengimplementasikan desain yang telah dibuat pada tahapan sebelumnya ke dalam aplikasi yang akan dikembangkan.

2) *Coding*

Coding merupakan penerjemahan hasil rancangan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan.

3) *Testing*

Testing dilakukan untuk mengecek apakah aplikasi dan fungsi-fungsi di dalamnya sudah berjalan dengan baik sesuai dengan yang diharapkan atau masih terdapat permasalahan atau *error* sehingga perlu dilakukan perbaikan. Metode pengujian yang digunakan dalam tahap ini adalah *black box*. Pengujian *black box* berusaha menemukan kesalahan yang terdapat pada aplikasi dengan kategori : (1) fungsi-fungsi yang tidak benar atau hilang, (2) kesalahan *interface*, (3) kesalahan dalam struktur data atau akses *database* eksternal, (4) kesalahan kinerja, dan (5) inisialisasi dan kesalahan terminasi.

4) *Deploying*

Deploying adalah proses pengiriman aplikasi yang telah dibuat pada IDE ke perangkat *mobile/ponsel* atau ke sebuah *server* untuk proses pengujian atau proses implementasi selanjutnya. Netbeans 7.2.1 mendukung beberapa metode *deployment*, yaitu:

- *Device Anywhere Deployment*, *website* untuk pengembang aplikasi *mobile* professional yang mengizinkan untuk melakukan *remote deploy MIDlet suites* ke berbagai merek *ponsel* yang sedang tren di pasaran.
- *File Copy*, memindahkan *MIDlet suites file* (*.jar* dan *.jad*) ke lokasi yang telah ditentukan pada perangkat lokal.
- *File Transfer Protocol* (FTP), menggunakan *file transfer protocol* untuk memindahkan file *MIDlet suite* ke sebuah *remote server* melalui Internet.
- *Nokia Mobile Devices*, mentransfer file *MIDlet suite* ke sebuah *Nokia Mobile device* dari IDE menggunakan aplikasi *Nokia PC Suite*.
- *Ricoh Devices*, mentransfer aplikasi *CDC* ke sebuah perangkat *Ricoh* menggunakan *HTTP POST* atau *SD card*
- *Secure Copy* (SCP, SFTP), membuat sebuah koneksi yang aman dan melakukan enkripsi data sebelum melewati data melalui klien dan server.
- *Sony Ericsson Phone (locally attached device)*, mentransfer file *MIDlet suite* ke sebuah *ponsel Sony Ericsson*. Metode *deployment* ini juga memiliki fitur untuk menjalankan dan menemukan kesalahan pada aplikasi yang berjalan pada *ponsel* dari IDE (harus memiliki *Sony Ericsson (On-Device Debug) emulator* yang telah terinstall untuk menggunakan pilihan ini).

- *Web-based Distribution WebAuthoring and Versioning (WebDAV)*, memindahkan file *Midlet suite* menggunakan protokol *Web-based Distribution WebAuthoring and Versioning (WebDAV)*, sebuah set ekstensi untuk protokol *HTTP* yang mengizinkan untuk bekerja secara berkolaborasi dengan pengembang yang lain untuk mengubah dan mengatur file-file pada *remote web servers*.
- *Windows Mobile Devices*, mentransfer file *MIDlet suite* ke sebuah perangkat *Windows Mobile* dari IDE menggunakan *ActiveSync* dan *Sun's CLDC/MIDP Virtual Machine*.

b. Validasi Ahli

Pada saat produk awal berupa aplikasi *mobile* jadi, selanjutnya akan dilakukan pengujian validasi dari ahli *software* dan ahli materi. Metode pengujian yang digunakan adalah *alpha testing*. Validasi dilakukan untuk mengetahui tingkat kelayakan aplikasi *mobile* yang dikembangkan berdasarkan teori yang telah dipaparkan pada kajian teori.

c. Revisi

Revisi dilakukan jika setelah divalidasi diperlukan perbaikan, kemudian setelah aplikasi *mobile* telah dinyatakan layak oleh ahli *software* dan ahli materi, maka akan dilanjutkan ke tahap implementasi.

4. Tahap Implementasi

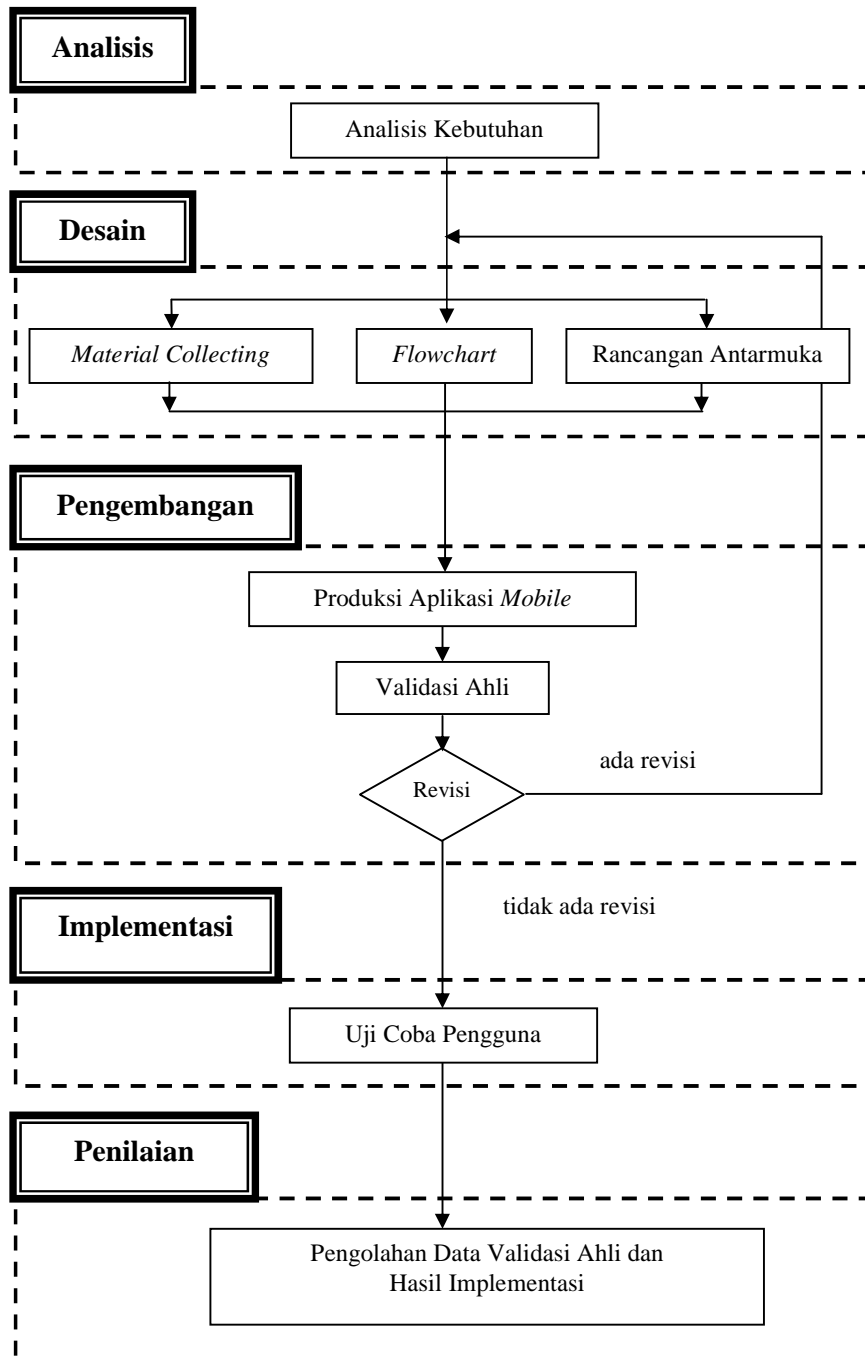
Implementasi adalah langkah nyata untuk menerapkan aplikasi *mobile* yang telah dikembangkan pada kondisi yang sebenarnya dengan cara dipasang

atau diset sedemikian rupa sesuai dengan peran atau fungsinya agar bisa diujicobakan.

Pada tahap implementasi akan dilakukan uji coba kepada pengguna yaitu mahasiswa yang memiliki *ponsel* yang mendukung teknologi J2ME. Tahap ini berupa uji coba untuk mengukur bagaimana tanggapan dan penilaian pengguna setelah menggunakan aplikasi *mobile* yang telah dikembangkan, ditinjau dari beberapa aspek dengan menggunakan angket penilaian. Metode yang digunakan adalah *beta testing*.

5. Tahap Penilaian

Penilaian didapatkan dari hasil validasi ahli *software* dan ahli materi, serta uji coba pada pengguna. Hal ini dilakukan untuk mengetahui tingkat kelayakan aplikasi *mobile* yang telah dikembangkan serta tanggapan dan penilaian pengguna setelah menggunakan aplikasi *mobile*, selain itu juga dipaparkan kelebihan, kekurangan, dan kendala aplikasi *mobile*.



Gambar 8: Prosedur Pengembangan Aplikasi Kamus Bahasa Jawa

B. Definisi Operasional

Definisi operasional dalam penelitian ini antara lain :

1. Pengembangan meliputi tahap analisis, desain, pengembangan, implementasi, dan penilaian.
2. Kamus merupakan sumber penyedia informasi dengan cara menterjemahkan atau mengartikan bahasa asing ke dalam bahasa yang dikenal oleh penggunanya.
3. Bahasa Indonesia adalah bahasa resmi Republik Indonesia dan bahasa persatuan bangsa Indonesia sifatnya sebagai bahasa pengantar bagi masyarakat Indonesia yang berbeda etnis maupun bahasa ibu.
4. Bahasa Jawa adalah bahasa ibu yang digunakan oleh masyarakat yang berasal dari wilayah Jawa Tengah, D.I. Yogyakarta dan sebagian besar Jawa Timur.
5. Kelayakan yaitu menilai layak tidaknya aplikasi pembelajaran bahasa Jepang tingkat dasar dengan melihat hasil uji kelayakan melalui ahli media, ahli materi, dan pengguna.

C. Tempat dan Waktu Penelitian

Penelitian yang dilakukan adalah pengujian oleh ahli *software* dan ahli materi pada bulan November 2012, serta pengujian oleh responden pada bulan Desember 2012. Ahli *software* adalah dosen Program Studi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta dan ahli materi adalah dosen Program Studi Pendidikan Bahasa Jawa Universitas Negeri Yogyakarta. Responden penelitian ini adalah mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta.

D. Sampel Penelitian

Sesuai dengan pendapat Suharsimi Arikunto (2002: 112) yang menjelaskan apabila subyeknya kurang dari 100 lebih baik diambil semua sehingga penelitiannya merupakan penelitian populasi. Tetapi, jika jumlah subjeknya besar, dapat diambil antara 10-15% atau 20-25% atau lebih, tergantung setidak-tidaknya dari:

1. Kemampuan peneliti dilihat dari waktu, tenaga dan dana.
2. Sempit luasnya wilayah pengamatan dari setiap subjek, karena hal ini menyangkut banyak sedikitnya data.
3. Besar kecilnya resiko yang ditanggung oleh peneliti, untuk penelitian yang risikonya besar, tentu saja jika sampel besar, hasilnya akan lebih baik.

Roscoe dalam buku *Research Methods For Business* (1982:253) memberikan saran-saran tentang ukuran sampel untuk penelitian di antaranya adalah ukuran sampel yang layak dalam penelitian adalah antara 30 sampai dengan 500 (Sugiyono, 2010:74).

Berdasarkan saran tersebut maka penulis mengambil ukuran sampel sebanyak 34 orang. Teknik Sampling menggunakan *Nonprobability Sampling*. *Nonprobability Sampling* adalah teknik sampling yang memberi peluang atau kesempatan tidak sama bagi setiap unsur atau anggota populasi untuk dipilih menjadi sampel. Teknik Sampling yang akan penulis tindak lanjuti meliputi Sampling Kuota dan Sampling *Purposive*.

1. Sampling kuota

Sampling kuota adalah teknik untuk menentukan sampel dari populasi yang mempunyai ciri-ciri tertentu sampai jumlah (kuota) yang diinginkan. Ukuran sampel yang dipakai adalah sebanyak 30 orang. Sampel adalah mahasiswa yang memiliki *ponsel* dengan sistem operasi yang mendukung teknologi J2ME.

2. Sampling *purposive*

Sampling *purposive* adalah teknik penentuan sampel dengan pertimbangan tertentu. Misalnya sumber data adalah orang yang ahli dibidangnya. Ukuran sampel yang dipakai adalah sebanyak empat orang, yaitu tiga orang ahli *software* dan seorang ahli materi.

E. Teknik Pengumpulan Data

Menurut Arikunto (1992: 190-200) ada beberapa teknik pengumpulan data yang digunakan untuk suatu penelitian, antara lain: (1) Metode Tes; (2) Metode Kuesioner atau Angket; (3) Metode Wawancara; (4) Metode Observasi; dan (5) Metode Dokumentasi. Teknik pengumpulan data dalam penelitian ini adalah:

1. Metode Tes

Metode tes digunakan untuk mengukur ada atau tidaknya serta besarnya kemampuan objek yang diteliti (Arikunto, 2010:266). Objek disini berarti Kamus Bahasa Jawa yang penulis kembangkan. Tes dilakukan dengan cara memberikan instrumen tes disertai dengan pedoman pelaksanaan secara jelas sesuai dengan *UTC for Java ME Application*. Responden yang dilibatkan adalah tiga ahli *software*.

2. Metode Kuesioner atau Angket

Metode angket dilakukan dengan cara memberi beberapa pernyataan tertulis kepada responden. Angket yang digunakan adalah angket tertutup dan disertai saran. Responden yang dilibatkan dalam pengambilan data adalah seorang ahli materi dan pengguna kamus.

F. Instrumen Penelitian

Instrumen yang digunakan dalam penelitian ini adalah tes dan angket (kuesioner). Teknik pengumpulan data dengan tes digunakan untuk memvalidasi aplikasi oleh ahli *software* yaitu dengan menggunakan serangkaian *test-case* sesuai dengan *UTC for Java ME Application*. Teknik pengumpulan data dengan angket dilakukan pada *end-user*/pengguna dan seorang ahli materi, untuk *end-user* menggunakan instrumen angket berdasarkan aspek *usability* yang digunakan untuk mendapatkan umpan balik berupa data opini individu dari *end-user* dan untuk ahli materi menggunakan angket yang penulis buat sendiri berdasarkan teori aspek substansi materi untuk mengukur kualitas aplikasi *mobile* dari segi konten/materi.

1. Instrumen Validasi Ahli Software

Instrumen validasi ahli *software* menggunakan serangkaian *test-case* sesuai dengan *UTC for Java ME Application* yang telah disesuaikan dengan kriteria kualitas perangkat lunak menurut ISO 9126-1 meliputi *functionality*, *reliability*, *efficiency* dan *portability*. Instrumen ini tetap menggunakan bahasa Inggris dan tidak diterjemahkan ke dalam bahasa Indonesia dengan alasan untuk

menghindari terjadinya kesalahan pentafsiran. Berikut ini kisi-kisi instrumen validasi ahli *software*:

Tabel 1: **Kisi-kisi Instrumen Validasi Ahli *Software***

No.	Aspek		Test Case
	ISO 9126-1	UTC	
1	<i>Reliability -> Fault Tolerance</i>	<i>Application Stability</i>	<i>The application must not crash or freeze at any time while running on the device.</i>
2	<i>Portability -> Installability</i>	<i>Application Installation</i>	<i>The application must install via OTA (Over The Air).</i>
3	<i>Efficiency -> Time Behaviour</i>	<i>Application Start Up</i>	<i>Application must start properly in 25s.</i>
4	<i>Portability -> Adaptability</i>	<i>Graphic Clarity</i>	<i>All graphics and animations displayed must be readable and clear to the user.</i>
5	<i>Functionality -> Interoperability</i>	<i>UI Consistency</i>	<i>The user interface of the application must be consistent throughout the application.</i>
6	<i>Portability -> Adaptability</i>	<i>Technical Text Errors</i>	<i>The text in the application must be clear and readable. The application must be free of technical text display issues such as: Text cut off/Text overlapping.</i>
7	<i>Reliability -> Recoverability</i>	<i>External Incoming Communication-Voice Call</i>	<i>The application can handle incoming communications.</i>
8	<i>Reliability -> Recoverability</i>	<i>External Incoming Communication-SMS</i>	<i>The application can handle incoming communications.</i>
9	<i>Functionality -> Suitability</i>	<i>Main Menu Requirements</i>	<i>The main functionalities of Exit, Help and About are easily available through the main menu.</i>
10	<i>Efficiency -> Time Behaviour</i>	<i>The Speed Of Application In Use</i>	<i>The application works in the device it was targeted for. It is usable on the device. The speed of the application is acceptable to the purpose of the application and must not alter the user experience by being uncontrollable.</i>
11	<i>Reliability -> Fault Tolerance</i>	<i>Unexpected User Behaviour</i>	<i>The application must be able to handle unexpected user behavior, for example erroneous actions and multiple key presses.</i>

2. Instrumen Validasi Ahli Materi

Instrumen validasi ahli materi berdasarkan aspek dan kriteria penilaian pembelajaran yang telah dijelaskan pada kajian teori. Kisi-kisi instrumen dijelaskan pada tabel berikut.

Tabel 2: **Kisi-kisi Instrumen Validasi Ahli Materi**

Aspek	Indikator	No. Butir
Substansi Materi	Kebenaran Materi Secara Teori dan Konsep	1,2,3,4,5,6,7

3. Instrumen Penilaian Pengguna

Instrumen penilaian pengguna menggunakan angket *IBM Computer Usability Satisfaction Questionnaires* untuk mengukur kualitas aplikasi ditinjau dari kriteria *usability*.

Instrumen ini diterjemahkan dari bahasa Inggris ke dalam bahasa Indonesia dengan alasan tidak semua responden memiliki kemampuan berbahasa Inggris yang baik. Berikut ini kisi-kisi instrumen penilaian pengguna:

Tabel 3: **Kisi-kisi Instrumen Penilaian Pengguna**

	Aspek	No. Butir
<i>Over All</i>	<i>System Usefulness</i>	1,2,3,4,5,6,7,8
	<i>Information Quality</i>	9,10,11,12,13,14,15
	<i>Interface Quality</i>	16,17,18
	<i>Over All</i>	1-19

G. Validitas dan Reliabilitas Instrumen

1. Validitas Intrumen

Validitas adalah keadaan yang menggambarkan tingkat instrumen yang bersangkutan mampu mengukur apa yang diukur (Arikunto, 2007:167). Menurut

Nursalam (dalam Sakti & Astria, 2012) validitas adalah suatu ukuran yang menunjukkan tingkat kevalidan atau kesahihan suatu instrumen. Jadi dapat disimpulkan bahwa validitas merupakan standar ukur yang menunjukkan tingkat ketepatan alat ukur, baik berupa tes atau suatu instrumen. Instrumen dikatakan valid jika dapat digunakan untuk mengukur apa yang seharusnya diukur.

Pengujian validitas instrumen yang dilakukan pada penelitian ini menggunakan pengujian validitas konstruksi (*construct validity*), untuk menguji validitas konstruksi dapat digunakan pendapat para ahli. Dalam hal ini instrumen terlebih dahulu dikonstruksi tentang aspek-aspek yang akan diukur berlandaskan teori tertentu, lalu instrumen dikonsultasikan dengan ahli. Setelah pengujian konstruksi dari ahli maka diteruskan dengan uji coba instrumen pada sampel dari mana populasi diambil (Sugiyono, 2010:352). Pada penelitian ini pengujian validitas konstruksi pada tahap uji coba instrumen dilakukan dengan analisis item menggunakan korelasi *Pearson Product Moment* (Riduwan, 2010) :

$$r_{hitung} = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{\{n\sum x^2 - (\sum x)^2\}\{n\sum y^2 - (\sum y)^2\}}}$$

Keterangan :

r_{hitung} = Koefisien korelasi

n = Jumlah responden

x = Jumlah skor item

y = Jumlah skor total

xy = Jumlah perkalian skor item dan skor total

x^2 = Jumlah kuadrat skor item

y^2 = Jumlah kuadrat skor total

Untuk mengetahui valid tidaknya sebuah instrumen, hasil perhitungan akan dibandingkan dengan table *r Product Moment* dengan taraf signifikansi 5% atau taraf kepercayaan 95%. Parameter dari hasil uji r_{hitung} adalah besarnya koefisien korelasi *Pearson Product Moment* antara 0.0 sampai 1 dan dikatakan valid jika besarnya r_{hitung} lebih besar dari r_{tabel} ($r_{hitung} > r_{tabel}$).

Pada penelitian ini pengujian validitas instrumen dilakukan dengan mengambil sampel berjumlah 15 orang ($N = 15$) dan taraf signifikansi sebesar 5%. Adapun r_{tabel} untuk $N = 15$ dan taraf signifikansi 5% adalah 0.514. Analisis item yang dilakukan dengan bantuan aplikasi *SPSS 15.0* menghasilkan data sebagai berikut:

Item-Total Statistics				
	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Cronbach's Alpha if Item Deleted
Item1	69.6000	101.257	.570	.935
Item2	69.3333	96.524	.684	.933
Item3	70.0667	100.781	.751	.932
Item4	69.8667	102.267	.641	.934
Item5	70.1333	104.124	.586	.935
Item6	69.8000	98.029	.785	.931
Item7	69.9333	94.924	.572	.939
Item8	70.2667	105.495	.577	.935
Item9	70.2000	96.457	.747	.931
Item10	71.4667	99.981	.688	.933
Item11	69.7333	98.352	.728	.932
Item12	70.3333	103.095	.645	.934
Item13	69.4000	103.686	.606	.935
Item14	69.8000	97.600	.734	.932
Item15	69.2000	102.600	.694	.933
Item16	69.6667	102.952	.556	.935
Item17	70.2667	106.067	.522	.936
Item18	69.8000	98.457	.758	.931
Item19	69.5333	101.981	.662	.933

Gambar 9: Hasil Uji Validitas Menggunakan *SPSS 15.0*

Berdasarkan hasil uji validitas tersebut dapat dijabarkan perbandingan **r hitung** yang diperoleh terhadap **r tabel** dengan N = 15 dan taraf signifikansi 5 % untuk tiap item sebagai berikut:

Tabel 4: Penjabaran Hasil Uji Validitas

No. Item Instrumen	Koefisien Korelasi (r_{xy} hitung)	r_{xy} tabel	Keterangan
1	0.570	0.514	Valid
2	0.684	0.514	Valid
3	0.751	0.514	Valid
4	0.641	0.514	Valid
5	0.586	0.514	Valid
6	0.785	0.514	Valid
7	0.572	0.514	Valid
8	0.577	0.514	Valid
9	0.747	0.514	Valid
10	0.688	0.514	Valid
11	0.728	0.514	Valid
12	0.645	0.514	Valid
13	0.606	0.514	Valid
14	0.734	0.514	Valid
15	0.694	0.514	Valid
16	0.556	0.514	Valid
17	0.522	0.514	Valid
18	0.758	0.514	Valid
19	0.662	0.514	Valid

Sumber : Data primer yang diolah

2. Reliabilitas Instrumen

Reliabilitas adalah karakter lain dari evaluasi. Reliabilitas juga dapat diartikan sama dengan konsistensi atau keajegan (Sukardi dalam Binham, 2012). Bila terdapat kesamaan data dalam waktu yang berbeda, hasil penelitian dapat dikatakan reliabel. Sugiyono (2012:121) berpendapat bahwa instrumen yang reliabel bila digunakan beberapa kali untuk mengukur obyek yang sama, akan menghasilkan data yang sama.

Pengujian reliabilitas instrumen ini menggunakan rumus koefisien reliabilitas *Alpha Cronbach* (Arikunto, 2007:180) :

$$r_{11} = \left[\frac{k}{k-1} \right] \left[1 - \frac{(\sum \sigma_b^2)}{\sigma_t^2} \right]$$

Keterangan :

r_{11} = Reliabilitas instrumen

k = Banyaknya item pertanyaan (banyak soal)

$\sum \sigma_b^2$ = Jumlah varian skor tiap-tiap item

σ_t^2 = Varian total

Hasil perhitungan selanjutnya dibandingkan pada tabel berikut untuk mengetahui reliabilitas instrumen berdasarkan klasifikasi dari Sugiyono (2010 : 231) sebagai berikut :

Tabel 5: **Tingkat Reliabilitas Instrumen**

Interval Koefisien	Tingkat Hubungan
0.00 – 0.199	Sangat rendah
0.20 – 0.399	Rendah
0.40 – 0.599	Sedang
0.60 – 0.799	Kuat
0.80 – 1.00	Sangat Kuat

Uji reliabilitas yang dilakukan dengan bantuan aplikasi *SPSS 15.0* memperoleh nilai reliabilitas (*Cronbach'a Alpha*) sebesar 0.937 yang dapat dilihat pada gambar berikut :

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
.937	.942	19

Gambar 10: Hasil Uji Reliabilitas dengan SPSS 15.0

Jika diukur berdasarkan tabel 5, hasilnya adalah **sangat kuat**. Jadi dapat disimpulkan bahwa instrumen penelitian ini reliabel dan masuk dalam kategori reliabilitas sangat kuat.

H. Analisis Data

Data yang didapat dari serangkaian *test case* untuk ahli *software* adalah data kualitatif yaitu pernyataan *pass* (lolos)/*passed with exception* (lolos dengan pengecualian) dan *fail* (gagal). Sesuai dengan *UTC for Java ME Application, test-case* ini tidak memerlukan sistem skoring karena semua pengujian berbobot sama, untuk dapat dikatakan layak dan memenuhi kriteria perangkat lunak ISO 9126-1, maka aplikasi harus lolos pada setiap pengujian.

Data yang telah dikumpulkan dari angket validasi untuk ahli materi dan penilaian pengguna adalah data kualitatif, karena setiap poin pernyataan dibagi ke dalam kategori sangat tidak layak, tidak layak, cukup layak, layak, dan sangat layak, untuk menghitungnya maka data terlebih dahulu ditransformasikan menjadi data kuantitatif sesuai dengan bobot skor yaitu satu, dua, tiga, empat, dan lima.

Data yang telah ditransformasikan kemudian dilakukan perhitungan *rating scale* dengan rumus dari Sugiyono (2009:99) dalam Prayoga (2011:41) sebagai berikut:

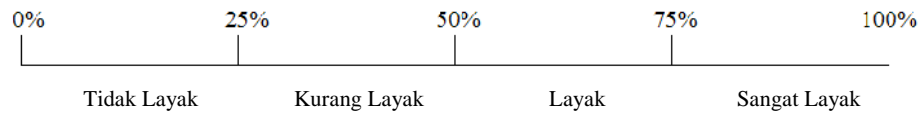
$$P = \frac{\text{Skor hasil pengumpulan data}}{\text{Skor ideal}} \times 100\%$$

Keterangan :

P = angka prosentase

Skor ideal = skor tertinggi tiap butir x jumlah responden x jumlah butir.

Selanjutnya tingkat validasi materi dan factor *usability* dari aplikasi kamus bahasa Jawa dalam penelitian ini digolongkan dalam empat kategori dengan menggunakan skala yang diadaptasi dari Gonia (2009:50):



Gambar 11: **Kategori Kelayakan Aplikasi**

Bagan *rating scale* bila dijelaskan dalam bentuk tabel adalah sebagai berikut (Gonia, 2009: 50):

Tabel 6: **Tabel Prosentase Kelayakan Aplikasi**

Skor Prosentase (%)	Interpretasi
0% - 24,99%	Tidak Layak
25% - 44,99%	Kurang Layak
50% - 74,99%	Layak
75% - 100%	Sangat Layak

Data penelitian yang bersifat kualitatif seperti komentar dan saran dijadikan dasar dalam merevisi aplikasi.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

A. Analisis

1. Analisis Isi Program

Pada tahap ini, penelitian diawali dengan studi literatur untuk mendapatkan informasi yang dibutuhkan dalam pengembangan aplikasi. Studi literatur dilakukan dengan mencari kamus bahasa Jawa-bahasa Indonesia dan bahasa Indonesia-bahasa Jawa yang tepat digunakan untuk materi dari aplikasi *mobile* ini.

2. Analisis Pengguna

Aplikasi kamus bahasa Jawa berbasis Java 2 Micro Edition (J2ME) ini ditujukan untuk masyarakat luas pada umumnya dan mahasiswa/pelajar yang berasal dari luar Jawa yang berdomisili di Jawa Tengah dan D.I.Y pada khususnya untuk memudahkan dalam mempelajari bahasa Jawa agar dapat membantu sebagai sarana alat bantu komunikasi dalam kehidupan sehari-hari

3. Analisis *Software*

a. *Software* pembuatan program

Software yang digunakan untuk mengembangkan aplikasi kamus bahasa Jawa menggunakan J2ME ini adalah:

1) Netbeans 7.2.1

Netbeans 7.2.1 digunakan sebagai *Integrated Development Environment* (IDE), yaitu sebuah *tool* dalam pemrograman untuk menulis, mengompilasi, mencari kesalahan, dan menyebarkan program dalam

pengembangan aplikasi ini. Bahasa pemrograman yang digunakan yaitu bahasa pemrograman java.

2) Corel Draw X5

Corel Draw X5 merupakan *tool* yang digunakan untuk menggambar vektor. *Tool* ini digunakan untuk membuat *icon-icon* yang dibutuhkan dalam aplikasi ini.

3) Audio Recorder

Audio recorder ini digunakan untuk merekam suara pelafalan yang nantinya akan diunggah ke *website* dan selanjutnya diakses oleh pengguna melalui fitur dalam aplikasi ini. Format output yang dihasilkan dapat berupa WAV, MP3, dan OGG. Format output yang digunakan dalam aplikasi ini adalah WAV, karena sebagian besar *ponsel* mendukung tipe *file* ini.

b. Software pada ponsel

Software yang dimaksud adalah sistem operasi pada *ponsel* yang digunakan untuk menjalankan aplikasi ini, oleh karena aplikasi ini berbasis J2ME, maka *ponsel* yang digunakan adalah *ponsel* dengan sistem operasi yang mendukung teknologi Java (Java 2.0), di antaranya adalah *Symbian*, *Java*, dan *Blackberry*. *Ponsel* yang digunakan dalam pengembangan adalah *Nokia 5630 XpressMusic* dengan sistem operasi *Symbian* dan mendukung Java MIDP 2.1.

4. Analisis Hardware

a. Spesifikasi hardware pengembang

Pengembang disini adalah penulis sebagai pengembang aplikasi. *Hardware* yang digunakan untuk mengembangkan aplikasi ini adalah *notebook*

dengan prosesor *intel core i3*, RAM 2 GB, VGA *Ati Radeon*, dan *harddisk* 320 GB, dengan spesifikasi ini *software-software* yang digunakan untuk mengembangkan aplikasi ini dapat berjalan lancar.

b. Spesifikasi *hardware ponsel*

Spesifikasi *hardware ponsel* yang digunakan untuk *testing* dari aplikasi ini adalah *ponsel Nokia 5630 XpressMusic* dengan ukuran layar 240 x 320 *pixels*, memori internal 60 MB, RAM 128 MB, CPU 600 MHz ARM 11, dengan spesifikasi ini aplikasi dapat berjalan lancar, tidak terlalu pelan dan tidak terlalu cepat. Adapun spesifikasi *hardware ponsel* minimal yang dapat digunakan untuk menjalankan aplikasi dengan baik adalah *ponsel* dengan ukuran layar 240 x 320 *pixels* dengan memori internal lebih dari 2 MB dan didukung dengan memori eksternal, karena aplikasi ini berukuran 882 KB dan setelah diinstal ukuran aplikasi ini menjadi kurang lebih 1 MB.

B. Desain

1. Material Collecting

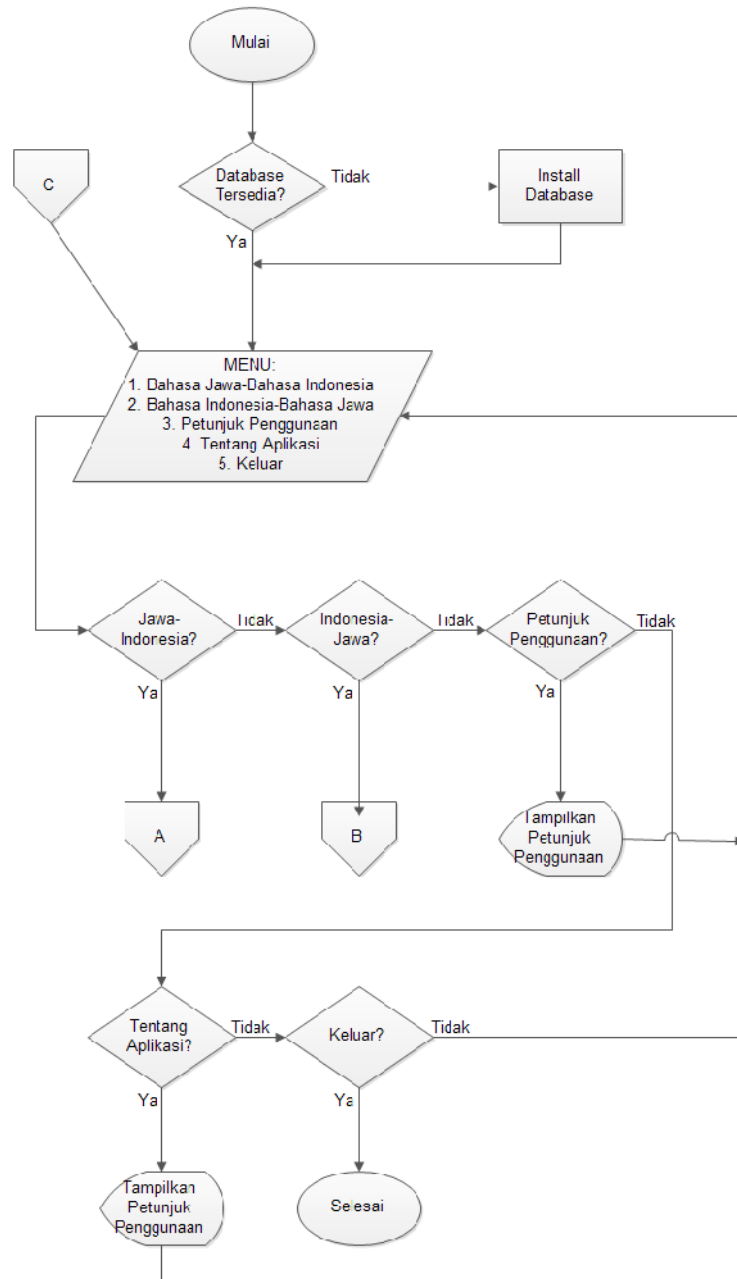
Pada tahap ini bahan-bahan yang terkait dengan materi sebagai konten dari aplikasi dikumpulkan. Bahan-bahan tersebut antara lain, kamus bahasa Jawa-bahasa Indonesia, bahasa Indonesia-bahasa Jawa karangan Drs. Sudarmanto, majalah bahasa jawa *Djaka Lodang*, dan buku pelajaran bahasa Jawa.

2. Flowchart

Flowchart atau diagram alir digunakan untuk menggambarkan langkah-langkah kerja dari sistem yang dibuat sehingga memudahkan dalam proses pengembangan aplikasi pembelajaran ini.

a. *Flowchart* sistem bagian satu

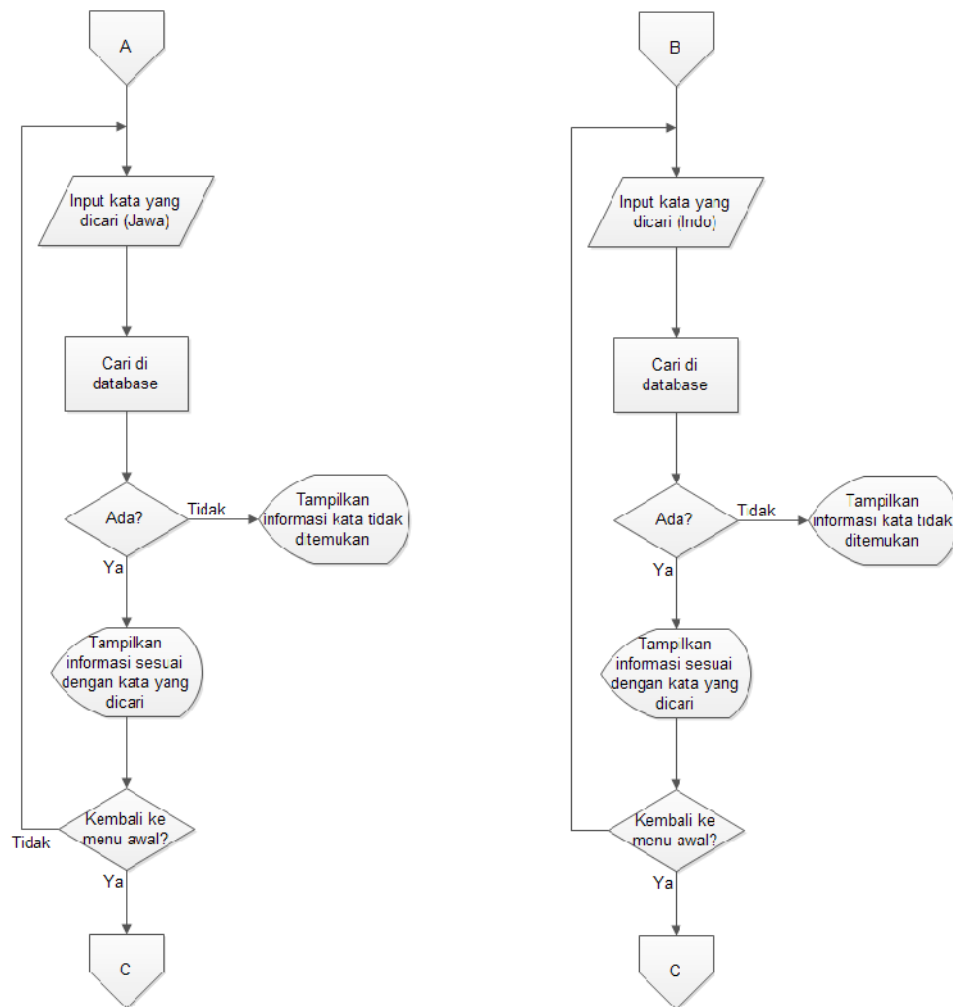
Pada flowchart sistem bagian satu menggambarkan bagaimana urutan kerja dan interaksi antar menu dari kamus bahasa Jawa ini.



Gambar 12: *Flowchart* Urutan Kerja dan Interaksi Antar Menu

b. *Flowchart* sistem bagian dua

Flowchart sistem bagian dua menggambarkan alur pencarian kata, karena proses pencarian kata dari bahasa Jawa ke bahasa Indonesia dan sebaliknya merupakan suatu proses yang sama.



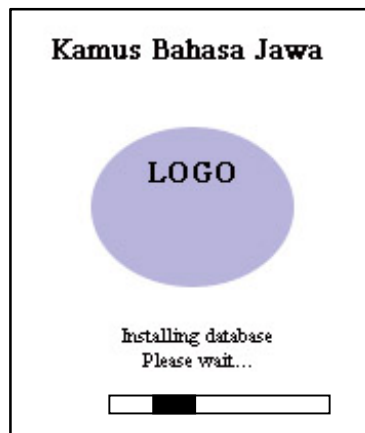
Gambar 13: *Flowchart* Pencarian Kata

3. Rancangan Antarmuka Pengguna

Flowchart yang telah dibuat selanjutnya digunakan sebagai acuan untuk merancang antarmuka pengguna. Berikut ini adalah desain antarmuka beserta penjelasannya untuk kamus bahasa Jawa ini:

a. Proses instalasi *database*

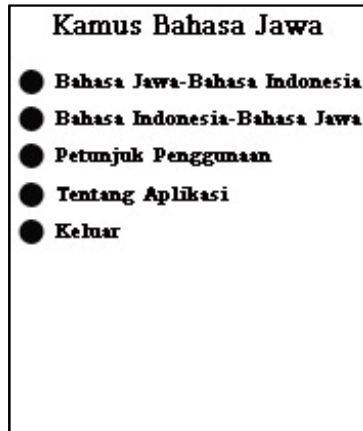
Pada saat kamus bahasa Jawa ini dijalankan untuk pertama kalinya, proses yang dijalankan adalah menginstall *database* dengan tampilan antarmuka berupa judul/nama aplikasi, logo aplikasi, *progress bar*, dan *label* pemberitahuan.



Gambar 14: **Desain Instalasi Database**

b. Menu utama

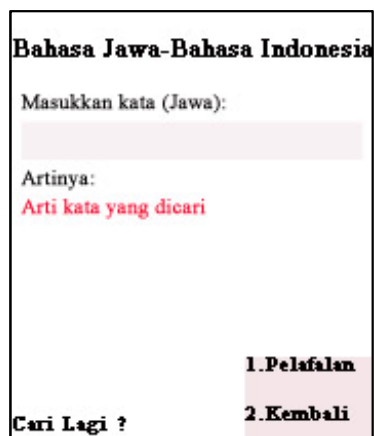
Menu ini adalah menu pokok dari aplikasi. Menu ini menggunakan *list* sebagai navigasi menuju sub-menu lain yang tersedia, pada menu ini tidak disediakan tombol *exit*/keluar untuk keluar dari aplikasi namun fungsi *exit*/keluar untuk keluar dari aplikasi disediakan pada menu ini dengan nama menu 'Keluar'. List ini berisi *icon* dan nama sub-menu. Berikut ini adalah gambar dari rancangan antarmuka menu utama:



Gambar 15: Desain Menu Awal

c. Bahasa Jawa-Bahasa Indonesia

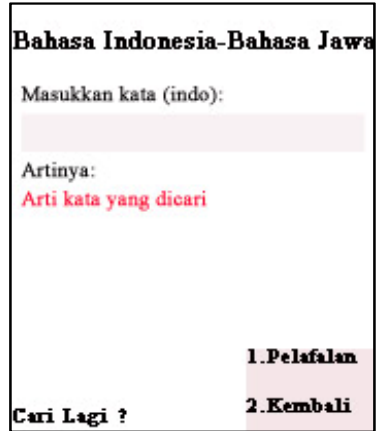
Menu ini berisi *title*/judul sub-menu, *label* masukan, *textfield*, *label* arti, dan tiga buah *command*. Berikut ini adalah gambar dari rancangan antarmuka sub-menu bahasa Jawa-bahasa Indonesia:



Gambar 16: Desain Menu Bahasa Jawa-Bahasa Indonesia

d. Bahasa Indonesia-Bahasa Jawa

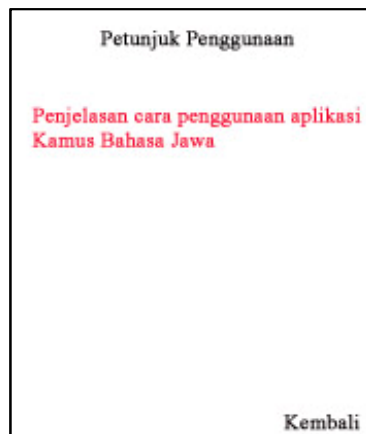
Menu ini memiliki komponen yang sama dengan menu bahasa Indonesia-bahasa Jawa. Berikut ini adalah gambar dari rancangan antarmuka sub-menu bahasa Indonesia-bahasa Jawa:



Gambar 17: Desain Menu Bahasa Indonesia-Bahasa Jawa

e. Petunjuk Penggunaan

Menu ini berisi *title*/judul dari sub-menu, sebuah *textarea*, dan sebuah *command* untuk kembali ke menu utama. Berikut ini adalah gambar dari rancangan antarmuka petunjuk penggunaan:

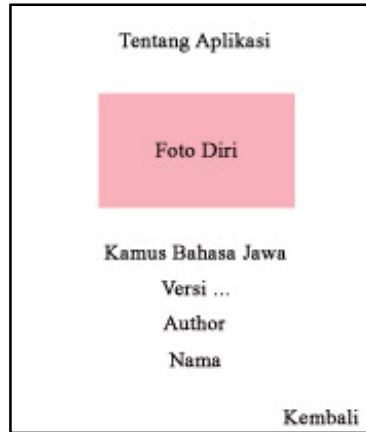


Gambar 18: Desain Menu Petunjuk Penggunaan

f. Tentang Aplikasi

Menu ini berisi *title*/judul sub-menu, sebuah *image* untuk foto diri sebagai identitas, empat *label* untuk nama aplikasi, versi aplikasi, nama *developer*,

dan nama penulis sebagai pengembang aplikasi, kemudian sebuah *command* untuk kembali ke menu utama.



Gambar 19: **Desain Menu Tentang Aplikasi**

C. Pengembangan

1. Produksi Aplikasi *Mobile*

Di bawah ini adalah gambar desain *ponsel* dengan keterangan istilah pada *ponsel* yang digunakan pada pembahasan kali ini,



Gambar 20: **Istilah Tombol Pada Ponsel**

a. Pembuatan antarmuka

Pembuatan antarmuka untuk aplikasi ini dilakukan dengan menggunakan IDE Netbeans 7.2.1. Netbeans menyediakan dua cara untuk membuat sebuah antarmuka yaitu yang pertama secara *visual* dengan *drag and drop* komponen yang sudah disediakan dan yang kedua secara *coding*. Pada pengembangan aplikasi ini cara yang digunakan adalah *coding*, yaitu dengan menuliskan serangkaian kode program hingga membentuk tampilan program yang

diinginkan. Pembuatan antarmuka ini didasarkan pada rancangan antarmuka yang telah dibuat pada tahap sebelumnya. Antarmuka yang telah dibuat tidak sepenuhnya sesuai dengan rancangan sebelumnya, hal ini dikarenakan keterbatasan proses *coding* yang dilakukan. J2ME mengenal dua macam pembuatan antarmuka yaitu dengan *canvas* dan *screen*. Aplikasi ini menggunakan antarmuka *screen* dalam pengembangannya.

1) Proses instalasi *database*

Aplikasi dijalankan untuk yang pertama kali setelah proses instalasi akan memunculkan tampilan pemasangan *database* berupa RMS (*Record Management Store*) yang berisi kosakata baik dalam bahasa Jawa maupun dalam bahasa Indonesia, agar aplikasi dapat digunakan dengan lancar tanpa ada kesalahan, maka pengguna harus menunggu hingga proses ini selesai. Berikut ini adalah gambar proses instalasi *database*:



Gambar 21: **Tampilan Proses Instalasi *Database***

2) Menu utama

Menu utama ini akan muncul setelah *database* sudah terinstal. Menu ini berupa *list* dan untuk memilihnya pengguna hanya perlu menggerakkan kursor ke atas atau ke bawah sesuai dengan menu yang dipilih dan kemudian pilih dengan menekan tombol *fire key*. Pada menu ini terdapat lima menu yaitu:

- a) Bahasa Jawa-Bahasa Indonesia
- b) Bahasa Indonesia-Bahasa Jawa
- c) Petunjuk Penggunaan
- d) Tentang Aplikasi
- e) Keluar

Berikut ini adalah gambar menu utama:



Gambar 22: Tampilan Menu Awal

3) Menu bahasa Jawa-bahasa Indonesia (Jawin)

Sesuai dengan judulnya, menu ini ditujukan untuk melakukan pencarian kata dalam bahasa Jawa. Fitur-fitur yang terdapat pada menu ini yaitu arti, pelafalan, dan cari lagi. Fitur arti digunakan untuk mengartikan kata yang

dimaksud, dilakukan dengan cara jika kata yang dimaksud sudah muncul dalam kemungkinan kata dicari yang ditampilkan maka pengguna tinggal memilih kata tersebut dengan menekan tombol *fire key* maka akan memunculkan arti dari kata yang dimaksud, jika kata yang dimaksud tidak tersedia di *database* maka akan muncul pemberitahuan bahwa kata tidak ditemukan. Fitur pelafalan terdapat pada menu *right soft button* pada *ponsel*, untuk menggunakannya, saat pengguna sudah memilih kata yang dimaksud kemudian tekan tombol *right soft key* dan pilih ‘pelafalan’, ijinan *ponsel* untuk terhubung Internet dan tunggu hingga proses selesai, untuk menggagalkan atau menghentikan proses pelafalan pilih ‘gagalkan’ dengan menekan *left soft key*. Fitur cari lagi digunakan jika pengguna ingin *reset* hasil pencarian. Untuk kembali ke menu utama, pengguna tinggal memilih ‘kembali’. Berikut ini adalah gambar-gambar dari menu bahasa Jawa-bahasa Indonesia beserta fiturnya:



Gambar 23: Menampilkan Kemungkinan Kata yang Dicari



Gambar 24: Menampilkan Pilihan Menu ‘Cari’ dan ‘Cari Lagi’



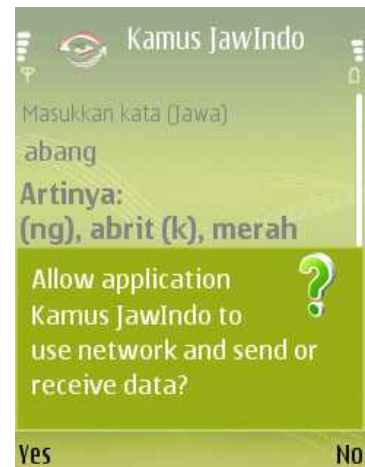
Gambar 25: **Menampilkan Arti Kata**



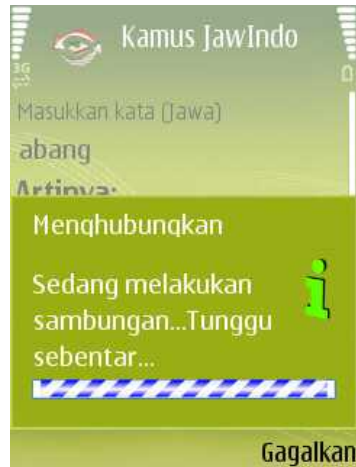
Gambar 26: **Menampilkan Pemberitahuan Kata yang Dicari Tidak Ditemukan**



Gambar 27: **Memilih Pilihan Menu 'Pelafalan' Untuk Melafalkan Kata yang Dicari**



Gambar 28: **Menampilkan Ijin Menggunakan Jaringan**



Gambar 29: Mengakses File Suara



Gambar 30: Proses Pelafalan Selesai



Gambar 31: Membatalkan Proses Pelafalan

4) Menu bahasa Indonesia-bahasa Jawa (Injaw)

Sesuai dengan judulnya, menu ini ditujukan untuk melakukan pencarian kata dalam bahasa Indonesia. Fitur yang terdapat dalam menu ini juga sama dengan yang ada pada menu Bahasa Jawa-Bahasa Indonesia. Berikut ini adalah gambar dari menu bahasa Indonesia-bahasa Jawa (Injaw):



Gambar 32: **Tampilan Menu Bahasa Indonesia-Bahasa Jawa**

5) Menu petunjuk penggunaan

Menu ini berisi penjelasan cara penggunaan aplikasi ditujukan untuk membantu pengguna, untuk kembali ke menu utama pilih menu 'kembali' pada menu yang tersedia di *right soft button*. Berikut ini adalah gambar dari menu petunjuk penggunaan:



Gambar 33: **Menu Petunjuk Penggunaan**

6) Menu tentang aplikasi

Sesuai dengan *UTC for Java ME Application* pada menu ini harus terdapat nama aplikasi (*MIDlet*), versi sebenarnya dari aplikasi, dan nama developer. Pilih menu ‘kembali’ pada menu yang tersedia di *right soft button* untuk kembali ke menu utama. Berikut ini adalah gambar dari menu tentang aplikasi:



Gambar 34: Menu Tentang Aplikasi

7) Menu keluar aplikasi

Menu ini akan memunculkan peringatan jika pengguna akan keluar dari aplikasi. Berikut ini gambar dari peringatan jika pengguna akan keluar dari aplikasi:



Gambar 35: **Keluar Aplikasi**

b. *Coding*

Coding merupakan proses penerjemahan desain/rancangan aplikasi baik itu antarmuka maupun fitur-fitur/fungsi-fungsi yang ada di dalamnya ke dalam bahasa pemrograman untuk selanjutnya di-*compile* agar menjadi sebuah aplikasi yang dapat digunakan seperti yang sudah direncanakan sebelumnya. Aplikasi ini memiliki kelas yang memiliki fungsi sendiri-sendiri dan beberapa dapat digunakan oleh kelas yang lain. Berikut ini dijelaskan *method-method* penting yang terdapat pada aplikasi ini, untuk *coding* lengkapnya dapat dilihat pada lampiran 1 sampai dengan 6.

1) Kelas *kamusJawaIndo.java*

Kelas ini sebagai *main MIDlet* pada aplikasi kamus bahasa Jawa ini. MIDlet merupakan aplikasi yang dibuat dengan menggunakan J2ME dengan standar MIDP. Dapat dikatakan *MIDlet* merupakan kelas utama yang harus ada dalam setiap *coding* menggunakan J2ME. Kelas *MIDlet* memiliki tiga *method* yang harus ada, yaitu *method startApp()* yang diproses pertama kali saat aplikasi

dijalankan, *pauseApp()* digunakan untuk memberikan *state pause/berhenti*, dan *destroyApp()* diproses untuk keluar dari aplikasi. Beberapa *vendor ponsel* tidak mengaktifkan fitur *pauseApp()*. Berikut ini adalah *method startApp()* yang digunakan untuk menjalankan aplikasi dan kode program lengkapnya dapat dilihat pada lampiran 1.

```
public void startApp() throws MIDletStateChangeException {
    try {
        if (!started) {
```

Untuk membuat *database* jika sudah tersedia atau membuat *database* jika belum tersedia

```
        dbControl.openRS();
        int countIndo = 0, countJawa = 0;
        countIndo = dbControl.dbIndo.getNumRecords();
        countJawa = dbControl.dbJawa.getNumRecords();
        if ((countJawa == 0 && countIndo == 0)) {
```

Jika *database* masih kosong langsung install

```
            installRMS = new splashInstallRMS(this);
            installRMS.show();
        } else if ((countJawa>0&&countJawa < 2224) || (countIndo>=0 && countIndo <
1613)) {
```

Jika *database* tidak terinstall dengan sempurna, beri pemberitahuan

```
            alYa = new Command("Ya", Command.OK, 1);
            alTidak = new Command("Tidak", Command.EXIT, 1);
            alert = new Alert("Peringatan","Database tidak terpasang dengan benar. Pilih
\u0027Ya\u0027 memasang database ulang dan tunggu hingga proses selesai. Pilih
\u0027Tidak\u0027 untuk keluar dari aplikasi!",null,AlertType.WARNING);
            alert.addCommand(alYa);
            alert.addCommand(alTidak);
            alert.setCommandListener(this);
            display.setCurrent(alert);
        } else {
```

Jika *database* sudah tersedia langsung masuk menu utama

```
            menuAwal = new listMenuAwal(this);
            menuAwal.show();
        }
        started=true;
    } else {
    }
} catch (IOException ex) {
    ex.printStackTrace();
```

```

    } catch (RecordStoreNotOpenException ex) {
        ex.printStackTrace();
    }
}

```

2) Kelas *splashInstallRMS.java*

Kelas ini akan menampilkan proses instalasi *database*. Terdapat *method* penting di sini yang memanggil kelas yang lain, yaitu *method startInstall()* yang dijalankan ketika kelas ini dipanggil. Kode program lengkapnya dapat dilihat pada lampiran 2. Berikut ini *method startInstall()*:

```
private void startInstall() throws RecordStoreNotOpenException, IOException {
```

Membuat sebuah timer untuk menjalankan proses instalasi

```
timer = new Timer();
```

Membuat sebuah objek dari kelas *threadInstall.java*

```
timerTaskOne = new threadInstall(sIns, gauge, m);
```

Menjalankan proses instalasi

```
timer.schedule(timerTaskOne, 1,1);
```

```
timersRunning = true;
```

```
}
```

3) Kelas *threadInstall.java*

Kelas ini digunakan pada saat aplikasi dijalankan pertama kali untuk memasang *database*. Kelas ini digunakan oleh kelas *splashInstall.java*. Cara kerja dari kelas ini adalah dengan membaca file teks *indojaw.txt* (lampiran 7) yang berisi kosakata bahasa Indonesia beserta artinya dalam bahasa Jawa dan *jawindo.txt* (lampiran 8) yang berisi kosakata bahasa Jawa beserta artinya dalam bahasa Indonesia menggunakan *method writeDataJawIn()* dan *writeDataInJaw()* kemudian kosakata ditampung dalam variabel bertipe data *Vector* yaitu, *vectorData1* untuk kosakata bahasa Jawa, *vectorData2* untuk kosakata artinya dalam bahasa Indonesia dan *vectordata3* untuk kosakata bahasa Indonesia,

vectorData4 untuk kosakata artinya dalam bahasa Jawa, setelah itu data dalam *Vector* tersebut dimasukan ke dalam *RecordStore* sebagai *database*-nya, berikut ini beberapa *method* yang digunakan dan untuk kode program lengkapnya dapat dilihat pada lampiran 3.

Method writeDataJawIn() ini cara kerjanya sama dengan *method writeDataInJaw()*

```
public void writeDataJawIn() {
```

Method splitC digunakan untuk mengambil baris per baris yang berisi kata dalam bahasa Jawa beserta artinya dalam bahasa Indonesia pada file teks *jawindo.txt* yang kemudian ditampung dalam sebuah variabel *kata* bertipe data array *String*

```
String[] kata = splitC(file("/text/jawindo.txt"), '\n');
```

Melakukan perulangan sebanyak baris yang ada pada file teks *jawindo.txt*

```
for(int i=0;i<kata.length;i++) {  
    String kataTrans = kata[i];  
    int len = kataTrans.length();
```

Melakukan perulangan sebanyak huruf per baris yang ada pada file teks *jawindo.txt*

```
for(int j=0;j<len;j++) {
```

Memisahkan kosakata bahasa Jawa dengan artinya (dipisahkan dengan karakter ':')

```
    if(kataTrans.charAt(j)==':') {  
        vectorData1.addElement(kataTrans.substring(0,j));  
        vectorData2.addElement(kataTrans.substring(j+1,len));  
        maxVector++;  
        maxJawIndo++;  
    }  
}  
}
```

4) Kelas *listMenuAwal.java*

Kelas ini digunakan untuk menampilkan menu utama. *Method* yang dibahas pada kelas ini adalah *commandAction(Command c, Displayable d)*, untuk kode program lengkapnya dapat dilihat pada lampiran 4, berikut ini penjelasan *method*:

Command c merupakan *command* yang telah di-*register*-kan oleh kelas ini yaitu *list* itu sendiri, sehingga saat *command* tersebut dipilih proses yang ada dalam *command* tersebut akan dijalankan. *Displayable d* digunakan untuk menangani tampilan.

```
public void commandAction(Command c, Displayable d) {
```

Jika *d* adalah *list* (*listMenuAwal*) maka dapatkan index pertama (sebagai *Command*) dari *list* tersebut dan jalankan proses yang ada. Index pertama dimulai dari nol.

```
    if (d==this) {
        if (getSelectedIndex() == 0) {
            code = 1;
            m.menuCariKata = new formCariKata(m);
            m.menuCariKata.setTextField("Masukkan kata (Jawa)", null, 30, 0);
            m.menuCariKata.append(m.menuCariKata.getTF());
            tittle = jawa;
            m.menuCariKata.show();
        } else if (getSelectedIndex() == 1) {
            code = 2;
            m.menuCariKata = new formCariKata(m);
            m.menuCariKata.setTextField("Masukkan kata (Indo)", null, 30, 0);
            m.menuCariKata.append(m.menuCariKata.getTF());
            tittle = indo;
            m.menuCariKata.show();
        } else if (getSelectedIndex() == 2) {
            m.menuBantuan = new formBantuan(m);
            m.menuBantuan.show();
        } else if (getSelectedIndex() == 3) {
            m.menuTentangAplikasi = new formTentangAplikasi(m);
            m.menuTentangAplikasi.show();
        } else if (getSelectedIndex() == 4) {
            alert = new Alert("Peringatan!", "Yakin mau keluar?", null,
AlertType.CONFIRMATION);
            alert.addCommand(alYes);
            alert.addCommand(alNo);
            alert.setCommandListener(this);
            m.display.setCurrent(alert);
        }
    } else if (d==alert) {
        if (c==alYes) {
            try {
                m.destroyApp(true);
            } catch (MIDletStateChangeException ex) {
                ex.printStackTrace();
            }
        } else if (c==alNo) {
            try {
                m.destroyApp(false);
            } catch (MIDletStateChangeException ex) {
                ex.printStackTrace();
            }
        }
    }
}
```

```

    }
    alert.removeCommand(alYes);
    alert.removeCommand(alNo);
    alert.setTimeout(1);
    alert.setCommandListener(null);
  }
}
}
}

```

5) Kelas *formCariKata.java*

Kelas ini memuat fitur utama dari aplikasi yaitu pencarian kata, untuk memudahkan dan mempercepat proses pencarian kata digunakan proses *indexing* yang terdapat pada *method indexingJawa()* dan *indexingIndo()* yang dapat dilihat di lampiran 4. *Method* yang dibahas di sini adalah *method searchNsort()* yang memiliki parameter *dicari*, *database*, *indexAwal*, dan *indexAkhir*. Logika dari *method* untuk pencarian ini adalah dengan membaca karakter pertama dan kedua (jika ada) pada *input* yang diberikan, kemudian melakukan *indexing* sesuai dengan karakter tersebut, hasil dari proses *indexing* digunakan untuk melakukan pencarian, hal ini dapat mempercepat waktu pencarian, untuk lebih jelasnya berikut ini kode program dari *method searchNsort()*, dan untuk kode program lengkapnya dapat dilihat pada lampiran 5.

String dicari adalah variabel untuk menampung kata yang akan dicari, *RecordStore database* adalah variabel untuk menampung *database* yang digunakan, *int indexAwal* dan *indexAkhir* adalah variabel untuk menampung *index* dari *method indexingJawa()* dan *indexingIndo()*.

```

public void searchNsort(String dicari, RecordStore database, int indexAwal, int
indexAkhir) {
    byte[] byteInputData = new byte[1000];
    ByteArrayInputStream inputStream = new
ByteArrayInputStream(byteInputData);
    DataInputStream inputDataStream = new DataInputStream(inputStream);

```

Variabel *sbuff* digunakan untuk menampung kata dari *recordstore*.

```

StringBuffer sbuff = new StringBuffer();

```

Counter untuk index awal *substring*, *countArr* untuk index awal *array* yang digunakan untuk menampung hasil pencarian.

```
int counter = 0, countArr=0;
```

Mendapatkan panjang dari kata yang dicari.

```
int lenDicari = dicari.length();  
String tmpSelection, tmpKata, compTmpKata;
```

Menyeleksi *record* di dalam *recordstore*.

```
try {  
for (int recId=indexAwal;recId<=indexAkhir;recId++) {  
database.getRecord(recId, byteInputData, 0);
```

Buffering database.

```
sbuff.append(inputDataStream.readUTF());  
tmpSelection = sbuff.toString();
```

Proses menampung per kata dalam *StringBuffer* dg variabel *tmpKata* (menampilkan kata satu persatu).

```
tmpKata = tmpSelection.substring(counter, tmpSelection.length());  
if (lenDicari<=tmpKata.length()){
```

Mengambil *substring* dari *tmpKata* untuk melakukan pencarian, variabel *compTmpKata* adalah per kata awal dari *record*.

```
compTmpKata = tmpKata.substring(0, lenDicari);
```

Proses melakukan pencarian, jika pada *recordstore* mengandung kata yang dicari maka diproses untuk menampilkan hasil

```
if (compTmpKata.equals(dicari)){
```

Reset DataStream sebelumnya, jika hal ini tidak dilakukan maka *DataStream* sebelumnya akan tetap ada dan proses pencarian *error*.

```
inputDataStream.reset();
```

Memunculkan kemungkinan kata lain yang dimaksud sebanyak yang diinginkan.

```
for (int j=recId;j<=recId+4;j++){  
if (j<=database.getNumRecords()) {
```

Mendapatkan *record* dari *recordstore* dimulai dari *record* ke-*j*.

```
database.getRecord(j, byteInputData, 0);
```

Menampung kata asal (kata yang ingin diterjemahkan).

```
String kata = inputDataStream.readUTF().toString();
```

Menampung kata terjemahan.

```
String terjemahan = inputDataStream.readUTF().toString();
```

Index untuk menemukan apakah kata asal mengandung kata yang dicari atau tidak, index akan bernilai -1 jika kata asal tidak mengandung kata yang dicari.

```
int containIdx = kata.indexOf(dicari);
int containIdxT = terjemahan.indexOf(dicari);
if (containIdx!=-1) {
    arrKata[countArr]=kata;
    arrTerjemahan[countArr]=terjemahan;
    if1 = false;
    if2 = false;
} else {
    break;
}
countArr++;
inputDataStream.reset();
}
}

buatStringItem(countArr);
break;
}
else
    if2 = true;
}
counter = tmpSelection.length();
inputDataStream.reset();
}
} catch (Exception e) {
    if (tfInput.size()!=0) {
```

Jika kata yang dicari tidak ditemukan dalam *recordstore* maka menampilkan pemberitahuan bahwa kata tidak ditemukan.

```
if (e.toString().indexOf("javax.microedition.rms.InvalidRecordIDException")!=-1){
    this.append("Kata tidak ditemukan");
} else
    System.out.println(e.toString());
}
}
}
```

6) Kelas *streamWav.java*

Kelas ini digunakan untuk melakukan proses pelafalan ketika pengguna menggunakan fitur pelafalan. Proses pelafalan ini dilakukan dengan cara aplikasi mengakses file suara yang telah diunggah pada *website* penulis kemudian melakukan beberapa tahapan hingga akhirnya suara dapat dilafalkan dari aplikasi.

Proses pelafalan ini dilakukan menggunakan sebuah *thread*, berikut ini penjelasan *method-method* yang digunakan untuk melakukan proses pelafalan dan untuk kode program lengkapnya dapat dilihat pada lampiran 6.

```
public void run() {
    if (readyC=true) {
        try {
```

Alamat *website* untuk mengakses file suara yang telah diunggah.

```
url="http://belajarjawa.host22.com/audiojawin/"+file+".wav";
HttpConnection conn = (HttpConnection)
Connector.open(url,Connector.READ_WRITE);
isConnected=true;
if (conn.getResponseCode()==conn.HTTP_OK) {
    alert.setString("Menghubungkan ke "+ conn.getHost().toString()+"\nTunggu
sebenantar...");
    InputStream is = conn.openInputStream();
    setIS(is);
    if (readyP==true) {
        alert.removeCommand(cmdCancel);
        alert.addCommand(cmdStop);
        alert.setCommandListener(this);
        isUnreal=true;
```

Method untuk membuat sebuah *player*, *player* ini yang digunakan untuk memainkan sebuah file *audio* pada sebuah *MIDlet*.

```
pStarted();
isPlayed=false;
} else
    alAborted();
} else if (conn.getResponseCode()!=conn.HTTP_OK) {
    alert.setType(AlertType.ERROR);
    alert.setTitle("Kesalahan");
    alert.setString("File tidak ditemukan di server!");
    alert.setCommandListener(null);
    alert.setTimeout(2000);
}
conn.close();
```

Penanganan jika terjadi kesalahan pada proses mengeksekusi file *audio*.

```
} catch (MediaException ex) {
    isConnected=false;
    alert.setTitle("Kesalahan");
    alert.setType(AlertType.ERROR);
    alert.setString("Format file suara tidak didukung");
    alert.setCommandListener(null);
    alert.setTimeout(Alert.FOREVER);
```

Penanganan jika terjadi kesalahan pada proses koneksi ke *website* file *audio*.

```
    } catch (IOException ex) {
        alert.setTitle("Kesalahan");
        alert.setType(AlertType.ERROR);
        alert.setString("Pastikan ponsel anda terhubung Internet!");
        alert.setCommandListener(null);
        alert.setTimeout(Alert.FOREVER);
    }
    } else {
        alAborted();
    }
}
```

Method untuk membuat *Player* untuk memainkan file *audio* bertipe *wav*.

```
public void createP() throws IOException, MediaException, InterruptedException {
    isReal=true;
}
```

Menampilkan status bahwa aplikasi sedang membuat sebuah *Player* untuk memainkan file pelafalan.

```
    alert.setString("Creating player...");
    player = Manager.createPlayer(getIS(), "audio/x-wav");
    player.addPlayerListener(this);
}
```

Method untuk memperoleh lokasi sumber media yang sudah diidentifikasi pada saat membuat *Player*. Proses ini memerlukan waktu beberapa saat.

```
public void realizeP() throws IOException, MediaException, InterruptedException {
    isPref=true;
    alert.setString("Realizing player...");
    player.realize();
}
```

Pada tahap ini *Player* sudah memiliki sumber media yang siap untuk dimainkan. Tahap ini belum memainkan media, tetapi masih tahap pengisian konten ke *buffer* objek *Player*.

```
public void prefetchP() throws IOException, MediaException, InterruptedException {
    isStart=true;
    alert.setString("Prefetching player...");
    vc = (VolumeControl) player.getControl("VolumeControl");
    if (vc != null) {
        vc.setLevel(100);
    }
    player.prefetch();
}
```

Pada tahap ini objek *Player* mulai memainkan media. Objek *Player* kembali masuk ke tahap *Prefetched* jika *method stop()* dipanggil atau jika objek *Player* telah mencapai akhir dari media.

```
void startP() throws MediaException {  
    int dur = (int) player.getDuration() / 1000;  
    alert.setTitle("Pelafalan");  
    alert.setString("Starting player...");  
}
```

Memulai *Player*.

```
player.start();  
}
```

c. *Testing*

Tahap *testing* diawali dengan menjalankan aplikasi melalui emulator yang sudah tersedia pada Netbeans 7.2.1. Tes dilakukan untuk mengetahui apakah fungsi-fungsi yang terdapat dalam aplikasi setelah proses *coding* dapat berjalan dengan benar dan untuk menemukan kekurangan atau kesalahan yang harus diperbaiki, jika dalam pengujian masih belum sesuai dan terdapat permasalahan-permasalahan, maka akan dilakukan perbaikan baik terhadap objek maupun fungsi hingga diperoleh hasil yang sesuai dengan yang diharapkan. Pengujian pada tahapan ini dilakukan dengan metode *black box testing*. Hasil pengujian yang dilakukan dapat dilihat pada lampiran 9.

d. *Deploying*

Metode yang digunakan untuk mentransfer file aplikasi (*MIDlet suite*) ke *ponsel* atau sebuah *server* untuk proses pengujian atau proses implementasi yaitu *File Copy*, dilakukan dengan cara memindahkan *MIDlet suites file* (*.jar* dan *.jad*) ke lokasi yang telah ditentukan pada perangkat lokal. File tersebut kemudian dipindah ke *ponsel* melalui koneksi *bluetooth* untuk diinstal secara *offline*. File tersebut juga diunggah pada *website* penulis di www.belajarjawa.host22.com untuk selanjutnya di-*download* dari *ponsel* untuk melakukan instalasi secara OTA (*Over The Air*).

2. Validasi Ahli

Aplikasi yang telah dikembangkan selanjutnya dilakukan pengujian *alpha* untuk mengukur tingkat kelayakan dari aplikasi. *Alpha Testing* merupakan pengujian yang dilakukan pada sisi pengembang perangkat lunak oleh seorang pelanggan dalam lingkungan yang terkontrol, yang dimaksud terkontrol adalah pengembang masih dapat mengikuti proses pengujian tersebut. Pengembang disini berarti penulis, sedangkan pelanggan adalah ahli *software* dan ahli materi. Pengujian oleh ahli *software* dilakukan dengan mengikuti serangkaian *test case* dari *UTC for Java ME Applications*, sedangkan untuk ahli materi dilakukan dengan mengisi angket yang telah dibuat dan memberikan umpan balik yang selanjutnya digunakan untuk penilaian dan perbaikan kamus selanjutnya. Validasi ahli atau yang biasa dikenal dengan istilah *expert judgement* ini dilakukan untuk mengetahui kelayakan aplikasi yang telah dikembangkan.

a. Validasi ahli materi

Kelayakan aplikasi pembelajaran dari segi konten dilakukan oleh seorang ahli materi yaitu dosen Pendidikan Bahasa Jawa Fakultas Bahasa dan Seni Universitas Negeri Yogyakarta. Aspek yang dinilai adalah substansi materi. Hasil dari perhitungan penilaian ahli materi digambarkan sebagai berikut:

Tabel 7: **Hasil Perhitungan Validasi Ahli Materi**

Aspek	Indikator	Total nilai	Prosentase	Kategori
Substansi Materi	Kebenaran Materi Secara Konsep dan Teori	28	80 %	Sangat Layak

Perhitungan untuk kelayakan aplikasi kamus bahasa Jawa yang telah dikembangkan oleh validasi ahli materi secara lengkap terdapat pada angket

dalam lampiran 10, dari hasil perhitungan didapatkan prosentase kelayakan aplikasi dilihat dari aspek substansi materi berdasarkan penilaian ahli materi sebesar 80% dan masuk ke dalam kategori **sangat layak**.

Ahli materi juga memberikan saran untuk perbaikan aplikasi sebagai berikut:

- 1) Petunjuk penggunaan perlu sistematis agar lebih mudah digunakan.
- 2) Berikan tambahan menu “Referensi” terutama sumber pustaka kamus yang digunakan.
- 3) Kamus perlu ditambah rujukannya.

Saran tersebut digunakan sebagai perbaikan, sebelum dilakukan pengujian *beta*.

b. Validasi ahli *software*

Kelayakan dari segi *software* dilakukan oleh tiga orang dosen yang bidang kajiannya berkaitan dengan *software*. Validasi dilakukan dengan metode tes, untuk mengetahui kemampuan aplikasi tersebut berdasarkan *UTC for Java ME Application* yang kemudian disesuaikan dengan kriteria ISO 9126-1. Tes yang dilakukan, menghasilkan temuan sebagai berikut:

Tabel 8: **Hasil Validasi Ahli Software**

No.	Kriteria ISO 9126-1	Pengujian (UTC)	Hasil
1.	<i>Reliability</i>	<i>Application Stability</i>	Lolos
		<i>Unexpected User Behaviour</i>	Lolos
		<i>External Incoming Communication-Voice Call</i>	Lolos
		<i>External Incoming Communication-SMS</i>	Lolos
2.	<i>Portability</i>	<i>Application Installation</i>	Lolos
		<i>Graphic Clarity</i>	Lolos

Tabel Lanjutan.

No.	Kriteria ISO 9126-1	Pengujian (UTC)	Hasil
2.	<i>Portability</i>	<i>Technical Text Errors</i>	Lolos
3.	<i>Efficiency</i>	<i>Application Start Up</i>	Lolos
		<i>The Speed Of Application In Use</i>	Lolos
4.	<i>Fucntionality</i>	<i>UI Consistency</i>	Lolos
		<i>Main Menu Requirements</i>	Lolos

Keterangan penilaian validasi ahli *software* dapat dilihat pada lampiran 10. Ahli *software* juga memberikan saran untuk perbaikan aplikasi sebagai berikut:

- 1) Perlu ditambah *title page* yang berisi judul aplikasi dan identitas pengembang.
- 2) Antarmuka aplikasi dibuat lebih berwarna agar lebih menarik.

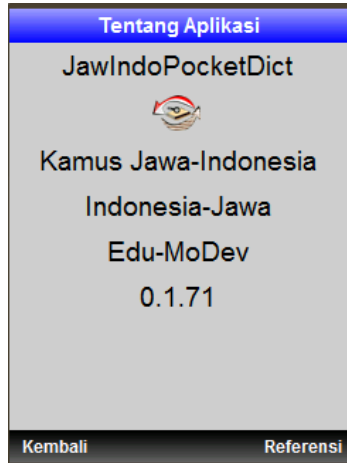
Saran tersebut digunakan sebagai perbaikan, sebelum dilakukan pengujian *beta*.

3. Revisi

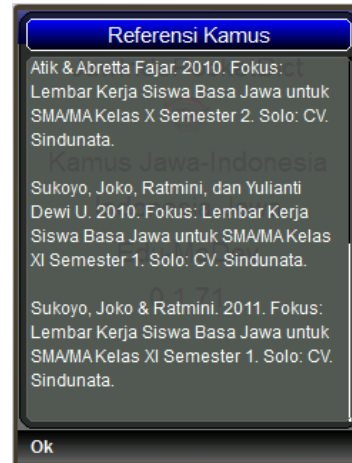
Berdasarkan validasi ahli materi dan ahli *software* terdapat beberapa hal yang harus diperbaiki walau aplikasi pembelajaran ini telah dinyatakan layak. Berikut ini tampilan aplikasi yang telah dilakukan perbaikan.

a. Perbaikan dari ahli materi

Berdasarkan saran dari ahli materi maka dilakukan perbaikan pada aplikasi ini, yaitu dengan menambahkan menu Referensi pada menu Tentang Aplikasi. Berikut ini tampilan program setelah dilakukan perbaikan.



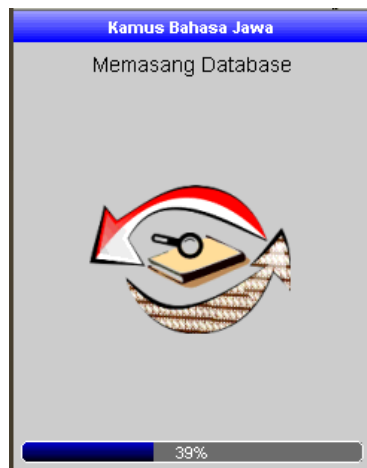
Gambar 36: **Penambahan Menu Referensi**



Gambar 37: **Tampilan Menu Referensi Kamus**

b. Perbaikan dari ahli *software*

Berdasarkan saran dari ahli *software* maka dilakukan perbaikan pada aplikasi ini, yaitu dengan menambahkan *title page* dan antarmuka dibuat lebih menarik dengan pemberian warna pada komponen antarmuka aplikasi. Berikut ini tampilan program setelah dilakukan perbaikan.



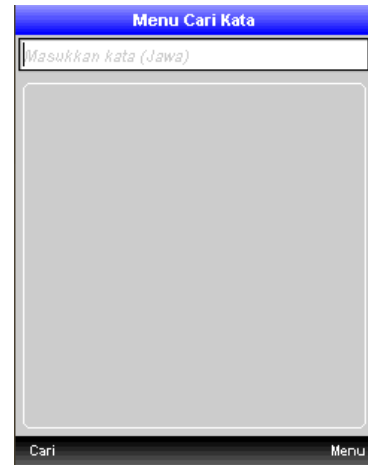
Gambar 38: **Tampilan Proses Instalasi Database**



Gambar 39: **Tampilan Title Page**



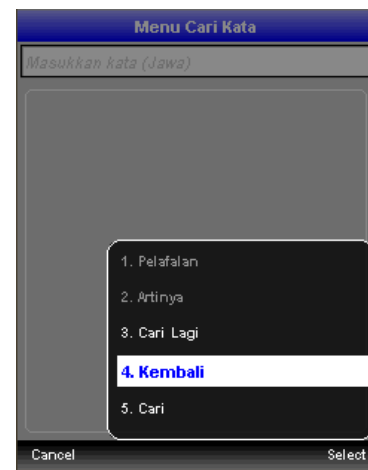
Gambar 40: **Tampilan Menu Utama**



Gambar 41: **Tampilan Menu Cari Kata**



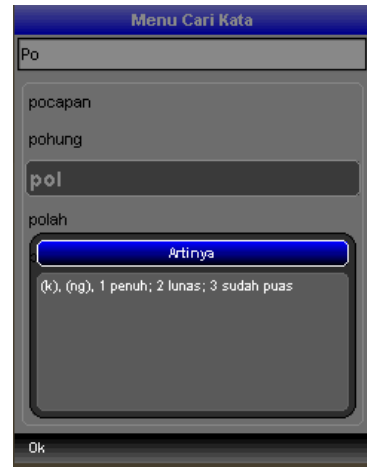
Gambar 42: **Tampilan Sub-Menu Dari Menu Cari Kata**



Gambar 43: **Tampilan Sub-Menu Dari Menu Cari Kata**



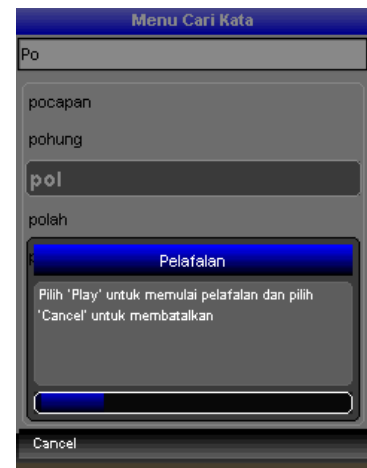
Gambar 44: Tampilan Menu Cari Kata (Jawa-Indo)



Gambar 45: Menampilkan Arti Kata yang Dimaksud



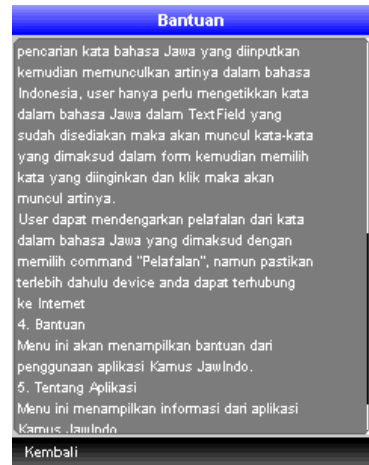
Gambar 46: Kotak Dialog Pelafalan



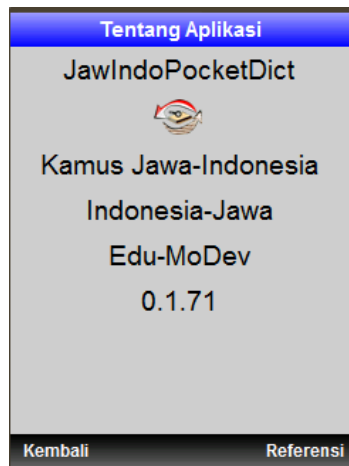
Gambar 47: Proses Pelafalan



Gambar 48: Tampilan Menu Cari Kata (Indo-Jawa)



Gambar 49: Tampilan Menu Bantuan



Gambar 50: Tampilan Menu Tentang Aplikasi



Gambar 51: Kotak Dialog Keluar Aplikasi

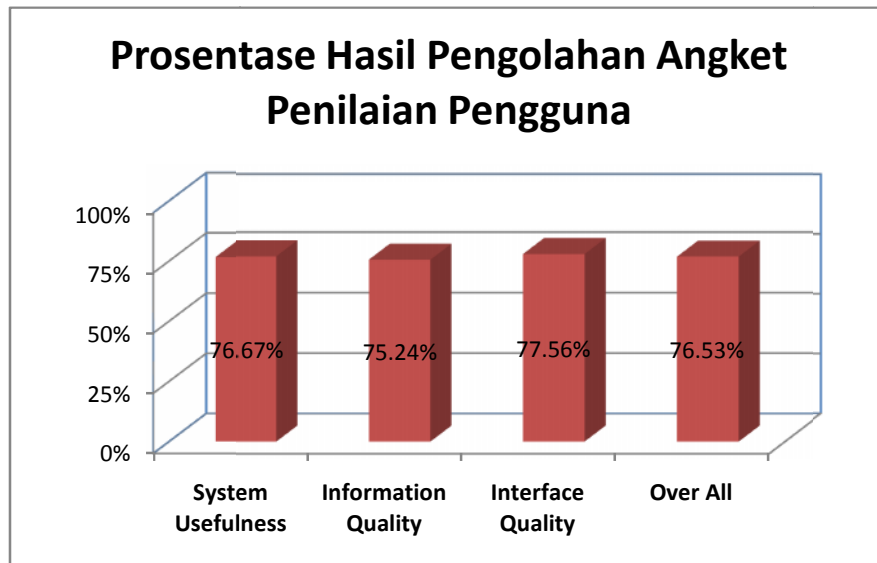
D. Implementasi

Implementasi aplikasi diawali dengan instalasi aplikasi pada *ponsel* yang dimiliki responden dengan cara mengirimkan *master* aplikasi berupa file bertipe *.jar* dari *ponsel* peneliti melalui media *Bluetooth* untuk kemudian diinstal pada *ponsel* responden tersebut. Proses instalasi tidak memerlukan waktu yang lama, setelah aplikasi terinstal, responden kemudian mencoba untuk

menggunakan aplikasi tersebut. Proses uji coba aplikasi tidak memerlukan waktu yang terlalu lama, setelah responden merasa cukup mencoba aplikasi tersebut kemudian responden diberi angket penilaian yang telah peneliti persiapkan untuk diisi sesuai dengan pendapat responden mengenai aplikasi tersebut. Kriteria yang dinilai adalah *usability* yang mencakup aspek *system usefulness*, *information quality*, *interface quality*, dan *over all*. Dari angket yang telah dibagikan, diperoleh data yang disajikan pada tabel berikut:

Tabel 9: Hasil Perhitungan Penilaian Pengguna

No.	Aspek	Prosentase	Kategori
1.	<i>System Usefulness</i>	76.67%	Sangat Layak
2.	<i>Information Quality</i>	75.24%	Sangat Layak
3.	<i>Interface Quality</i>	77.56%	Sangat Layak
4.	<i>Over All</i>	76.53%	Sangat Layak



Gambar 52: Diagram Batang Hasil Pengolahan Angket Penilaian Pengguna

Diagram batang di atas menunjukkan perolehan data dari responden yang telah diolah dan menunjukkan bahwa ditinjau dari aspek *system usefulness* diperoleh hasil penilaian dengan prosentase 76.67% dan masuk kategori **sangat layak**, dari aspek *information quality* diperoleh hasil penilaian prosentase 75.24% kategori **sangat layak**, dan dari aspek *interface quality* diperoleh hasil penilaian dengan prosentase 77.56% dan masuk kategori **sangat layak**. Secara keseluruhan, prosentase penilaian oleh pengguna pada aplikasi kamus bahasa Jawa menggunakan J2ME adalah 76.53% dan masuk kategori **sangat layak**, ini ditunjukkan oleh aspek *over all*.

E. Penilaian

1. Penilaian Kelayakan Aplikasi oleh Para Ahli

Berdasarkan penilaian ahli *software* pada tahap validasi ahli setelah melakukan serangkaian *test-case* yang menunjukkan hasil bahwa aplikasi ini lolos dalam setiap tahapan pengujian dan dinyatakan layak dari kriteria *functionality*, *reliability*, *efficiency*, dan *portability*.

Berdasarkan penilaian ahli materi pada tahap validasi ahli dan setelah mendapatkan hasil dari pengolahan data angket ahli materi, diperoleh prosentase penilaian sebesar 80% dan termasuk kategori sangat layak untuk aspek substansi materi.

2. Penilaian Pengguna Terhadap Aplikasi

Berdasarkan hasil penilaian pengguna terhadap aplikasi yang dilakukan pada tahap implementasi diperoleh prosentase penilaian sebesar 76.53% dan

termasuk kategori sangat layak untuk keempat aspek yang diujikan yaitu aspek *system usefulness*, *information quality*, *interface quality*, dan *over all*.

3. Kelebihan, Kekurangan, dan Kendala Aplikasi

a. Kelebihan

- 1) Aplikasi kamus bahasa Jawa ini mudah digunakan dengan fitur dan tampilannya yang cukup simpel.
- 2) Aplikasi ini mampu melafalkan kata dalam bahasa Jawa agar pengguna dapat memahami dan dapat melafalkannya dengan benar, khususnya bagi pengguna yang berasal dari luar Jawa.
- 3) Sifatnya yang *portable* karena diinstal pada *ponsel* membuat pengguna dapat menggunakannya kapanpun dan dimanapun saat dibutuhkan.
- 4) Meskipun dikembangkan dengan J2ME, aplikasi ini dapat diinstal pada *platform* ataupun *ponsel* yang tidak memiliki fitur J2ME dengan menggunakan *emulator*, misalkan saja untuk *ponsel Android*, dapat menggunakan aplikasi *JBed* sebagai *emulator* untuk menjalankan aplikasi ini.

b. Kekurangan

- 1) Berdasarkan penilaian dari responden, kosakata pada aplikasi kamus bahasa Jawa ini masih kurang, terutama untuk fitur pencarian kata dari bahasa Indonesia ke bahasa Jawa.
- 2) Antarmuka aplikasi kamus bahasa Jawa ini masih kurang menarik.
- 3) Antarmuka aplikasi kamus bahasa Jawa ini dapat berubah jika resolusi layar tidak sesuai dengan *ponsel* yang digunakan untuk mengembangkan aplikasi.
- 4) Belum ada fitur aplikasi untuk tambah, hapus, ataupun edit kosakata.

- 5) Kemampuan *multi-platform* aplikasi kamus bahasa Jawa ini masih kurang.
 - 6) Ukuran *master* aplikasi cukup besar.
- c. Kendala
- 1) Pemberian audio untuk pelafalan setiap kata yang ada pada materi cukup sulit, karena peneliti harus merekam setiap kata bahasa Jawa yang ada kemudian mengunggahnya ke *website* peneliti agar dapat diakses pengguna, hal ini karena belum ada *text-to-speech* untuk bahasa Jawa.
 - 2) Terbatasnya pengetahuan dan kemampuan peneliti pada saat ini mengenai pengolahan data pada *Record Store Management* (RMS) yang digunakan sebagai *database* aplikasi.

F. Pembahasan

Penelitian ini secara umum bertujuan untuk mengembangkan sebuah aplikasi kamus bahasa Jawa berbasis J2ME yang dapat diinstal pada *ponsel* pengguna yang mendukung teknologi J2ME agar dapat membantu pengguna untuk berkomunikasi, khususnya bagi mahasiswa yang berasal dari luar Jawa dan sedang berdomisili di Jawa (Jawa Tengah, DIY, Jawa Timur). Metode yang digunakan adalah metode penelitian dan pengembangan (*R&D*) yang meliputi tahapan analisis, desain, pengembangan, implementasi, dan penilaian.

Aplikasi kamus bahasa Jawa menggunakan J2ME ini dikembangkan menggunakan IDE Netbeans 7.2.1. Materi yang disajikan berupa pencarian kata dan pelafalan kata. Pencarian kata digunakan untuk mencari dan mengartikan kata dari bahasa Jawa ke bahasa Indonesia maupun dari bahasa Indonesia ke bahasa Jawa. Pelafalan kata digunakan untuk melafalkan kata dalam bahasa Jawa.

Aplikasi yang telah dikembangkan diuji dengan menggunakan teknik pengujian *black box* untuk mengetahui kesalahan fungsi yang terdapat pada aplikasi. Kesalahan fungsi yang terjadi selama pengujian *black box* kemudian diperbaiki hingga mencapai hasil yang diharapkan, yaitu aplikasi dapat berjalan tanpa ada kesalahan. Aplikasi yang telah melalui pengujian *black box* kemudian masuk pada tahap *deploying* dengan menggunakan metode *file copy* untuk memindahkan *MIDlet Suite* berupa file berekstensi *.jar* dan *.jad* yang selanjutnya digunakan sebagai *master* aplikasi untuk diunggah pada *website* peneliti dan diinstal pada *ponsel* secara OTA. Proses selanjutnya yaitu aplikasi divalidasi oleh ahli *software* dan ahli materi sebelum aplikasi diujicobakan pada pengguna.

Validasi ahli *software* dilakukan untuk mengetahui kelayakan aplikasi yang dikembangkan berdasar pada kriteria ISO 9126-1 yang meliputi *functionality*, *reliability*, *efficiency*, dan *portability*. Hasil validasi yang dilakukan oleh tiga ahli *software* menunjukkan aplikasi ini lolos pada setiap tahap pengujian dengan menggunakan *test-case* dari *UTC for Java ME Application* dan aplikasi ini dinyatakan layak berdasarkan kriteria ISO 9126-1 yang telah disebutkan tersebut.

Validasi ahli materi dilakukan untuk mengetahui kelayakan aplikasi yang dikembangkan dari segi konten aplikasi. Hasil validasi yang dilakukan oleh seorang ahli materi berdasarkan aspek substansi materi diperoleh prosentase sebesar 80% dan termasuk dalam kategori sangat layak.

Berdasarkan penilaian ahli tersebut, maka aplikasi ini layak digunakan dari sisi fungsionalitas aplikasi dan sangat layak dari sisi konten aplikasi. Ahli

software dan ahli materi juga memberikan saran untuk perbaikan aplikasi sebelum diujicobakan pada pengguna.

Aplikasi yang telah melewati tahap revisi berdasarkan saran dari para ahli kemudian masuk ke tahap implementasi, yaitu tahap pengujian oleh pengguna yang sebenarnya. Uji coba dilakukan di Fakultas Teknik Universitas Negeri Yogyakarta pada tiga puluh mahasiswa yang digunakan sebagai sampel. Teknik yang digunakan untuk menentukan sampel yaitu teknik sampling kuota yaitu dengan memilih mahasiswa dengan ciri-ciri tertentu sampai kuota terpenuhi. Data yang telah didapat dari angket pengguna kemudian diolah dan diperoleh prosentase penilaian aspek *system usefulness* sebesar 76.67%, aspek *information quality* sebesar 75.24%, aspek *interface quality* sebesar 77.56%, dan aspek *over all* yang mewakili kriteria *usability* secara keseluruhan yaitu sebesar 76.53%. Berdasarkan hasil penilaian untuk kriteria *usability* tersebut menunjukkan aplikasi ini termasuk dalam kategori sangat layak dilihat dari aspek *system usefulness*, *information quality*, *interface quality*, dan *over all*.

Berdasarkan hasil penilaian ahli *software*, ahli materi dan pengguna, dapat disimpulkan bahwa aplikasi kamus bahasa Jawa yang dikembangkan dinilai layak oleh hampir seluruh responden. Berikut tabel hasil kelayakan aplikasi kamus bahasa Jawa menggunakan J2ME yang dinilai oleh ahli *software*, ahli materi, dan pengguna.

Tabel 10: Hasil Kelayakan Aplikasi Kamus Bahasa Jawa Menggunakan
J2ME

No.	Responden	Penilaian	Kategori
1.	Ahli <i>Software</i>	Lolos <i>test-case UTC for Java ME Application</i>	Layak
2.	Ahli Materi	80%	Sangat Layak
3.	Pengguna	76.53%	Sangat Layak

BAB V KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan tahap penelitian dan pengembangan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut :

1. Proses pengembangan aplikasi kamus bahasa Jawa-bahasa Indonesia menggunakan J2ME melalui lima tahapan, yaitu (1) tahap analisis, (2) desain aplikasi, (3) pengembangan aplikasi, (4) implementasi dengan melakukan uji coba aplikasi kepada pengguna, dan (5) penilaian terhadap produk aplikasi yang dikembangkan. Dari hasil pengembangan dapat diketahui aplikasi kamus bahasa Jawa-bahasa Indonesia menggunakan J2ME ini memiliki beberapa keunggulan dibandingkan dengan aplikasi kamus bahasa Jawa (ngoko)-bahasa Indonesia menggunakan *Visual Basic* yang sudah ada di antaranya yaitu (1) *database* yang lebih lengkap dengan 3837 kosakata, (2) mampu melafalkan kosakata dalam bahasa Jawa, (3) bersifat *mobile*, dan (4) dapat mengartikan kata dari bahasa Jawa ke bahasa Indonesia dan dari bahasa Indonesia ke bahasa Jawa.
2. Penilaian unjuk kerja aplikasi kamus bahasa Jawa menggunakan J2ME dilakukan oleh peneliti, ahli *software*, dan ahli materi. Penilaian oleh peneliti dilakukan dengan metode pengujian *black box testing* dan menunjukkan bahwa fungsi dari setiap komponen program dapat berjalan dengan benar, tidak terdapat kesalahan pada antarmuka aplikasi, dan navigasi pada aplikasi berfungsi sesuai dengan yang diharapkan. Ahli *software* dan ahli materi

memberikan penilaian baik terhadap unjuk kerja aplikasi kamus bahasa Jawa menggunakan J2ME.

3. Ahli *software* menyatakan aplikasi kamus bahasa Jawa menggunakan J2ME yang telah dikembangkan dinyatakan layak untuk digunakan, setelah lolos pada setiap tahapan pengujian *UTC for Java ME Application*. Berdasarkan penilaian dari ahli materi dengan prosentase 80% dan penilaian pengguna dengan prosentase 76.53%, aplikasi kamus bahasa Jawa menggunakan J2ME yang telah dikembangkan dinyatakan sangat layak untuk digunakan.

B. Saran

Dari hasil penelitian, terdapat beberapa saran yang dapat dijadikan masukan bagi pengembangan aplikasi selanjutnya, yaitu :

1. Perlu dikembangkan lagi aplikasi kamus bahasa Jawa yang *multiplatform*.
2. Aplikasi dapat dibuat lebih menarik lagi, baik dari segi antarmuka maupun fitur yang terdapat di dalamnya, misalkan saja dapat ditambah permainan acak kata dengan kosakata bahasa Jawa.
3. *Database* kosakata dapat di-*update* secara berkala melalui Internet.
4. Dapat ditambah fitur *add*, *edit*, atau *delete* pada aplikasi, agar pengguna dapat melakukan modifikasi kosakata pada *database*.
5. Perlu dikembangkan *text-to-speech* dalam bahasa Jawa untuk lebih memudahkan pengembangan aplikasi-aplikasi yang berhubungan dengan bahasa Jawa.

DAFTAR PUSTAKA

- Aprianti, Tery. (2011). Pengembangan Tes Pada Pembelajaran Problem Solving Materi Kelarutan dan Hasil Kali Kelarutan. *Skripsi S1*. Bandung: Program Studi Pendidikan Kimia Universitas Pendidikan Indonesia.
- Arikunto, Suharsimi. (1992). *Prosedur Penelitian Suatu Pendekatan Praktis*. Jakarta: PT. Renika Cipta.
- _____. (2007). *Manajemen Penelitian*. Jakarta : Rineka Cipta
- _____. (2010). *Prosedur Penelitian Suatu Pendekatan Praktik* (Edisi Revisi 2010.). Jakarta: Rineka Cipta.
- Atik & Abretta Fajar. (2010). *Fokus: Lembar Kerja Siswa Basa Jawa untuk SMA/MA Kelas X Semester 2*. Solo: CV. Sindunata.
- Benny A Pribadi. 2009. *Model Desain Sistem Pembelajaran*. Jakarta:PT. Dian Rakyat.
- Endang Mulyatiningsih.2012.*Modul kuliah pengembangan Model Pembelajaran UNY*.
- Binham. (2012). *Validitas dan Reliabilitas Instrumen Evaluasi*. Diakses dari <http://binham.wordpress.com/2012/01/07/validitas-reliabilitas-instrumen-evaluasi/> pada tanggal 7 Mei 2012, jam 08.15 WIB.
- Budi dkk. (2007). “Pemrograman Berorientasi Obyek Dengan J2ME”, <http://mtamim.files.wordpress.com/2010/05/bukupab.pdf>. Diunduh pada 2 Oktober 2011.
- Dewan Riset Nasional. (2002). Undang-Undang Republik Indonesia Nomor 18 Tahun 2002 Tentang Sistem Nasional Penelitian, Pengembangan, dan Penerapan Ilmu Pengetahuan dan Teknologi.
- Dikmenum. (2008). “Lomba Pembuatan Multimedia Pembelajaran SMA Tahun 2008”. <http://www.dikmenum.go.id/data/File/f2008/berita/KAL/BAHAN%20LEAFLET%20LPMP%20SMA%202008.pdf>. Diunduh pada tanggal 12 April 2012.
- Ellington, Henry & Aris, Baharuddin. (2000). *A Practical Guide to Instructional Design*. Malaysia : Mufakat Jaya Percetakan.

- Gonia, M Firdaus. (2009). Pengembangan Multimedia Interaktif untuk Assesmen pada Pembelajaran Pembiasaan Cahaya. *Skripsi SI*. Bandung: Program Studi Pendidikan Ilmu Komputer Fakultas Matematika dan Pengetahuan Alam Universitas Pendidikan Indonesia.
- Hermawan, Benny. (2004). *Menguasai JAVA 2 & Object Oriented Programming*. Yogyakarta: ANDI OFFSET.
- Ibrahim bin Ahmad. (2002). *Perkamusan Melayu: Suatu Pengenalan*. Kuala Lumpur. Dewan Bahasa dan Pustaka.
- Indres Deswita. (2009). Aplikasi Kamus Indonesia-English-Italy Pada Perangkat Mobile Dengan J2ME. Penulisan Ilmiah. Universitas Gunadarma.
- Kompas. (2009). “Memprihatinkan Nasib Penggunaan Bahasa Jawa”, <http://www1.kompas.com/printnews/xml/2009/05/17/06032772/Memprihatinkan.Nasib.Penggunaan.Bahasa.Jawa>. Diakses pada tanggal 13 Desember 2011.
- Harimurti Kridalaksana, dkk. (2001). *Wiwara Pengantar Bahasa dan Kebudayaan Jawa*. Jakarta: PT. Gramedia Pustaka Utama.
- Lenny Ike C. M. (Tahun Tidak Diketahui). Kamus Inggris-Indonesia Berbasis J2ME.
- Lewis, J.R. (1995). “IBM Computer Usability Satisfication Questionnaires: Psycometric Evaluation and Instruction For Use”. *International Journal of Human-Computer Interaction*. pp. 57-58.
- Moehtar dkk. (2006). *Panjebar Semangat*. Surabaya: PT Percetakan Penyebar Semangat.
- Mulyono, Slamet. (2008). *Kamus Pepak Basa Jawa*. Jakarta: PT. Buku Kita.
- Nugroho, Eddy Prasetyo, dkk. (2009). *Rekayasa Perangkat Lunak*. Bandung: Telkom Polytechnic Bandung.
- Prayoga, Yoga Rizki. (2011). Pengembangan Multimedia Pembelajaran Permainan Ular Tangga Pada Materi Bangun Ruang di Sekolah Dasar. *Skripsi SI*. Bandung: Program Studi Ilmu Komputer Fakultas Matematika dan Pengetahuan Alam Universitas Pendidikan Indonesia.
- Pressman, Roger S. (2002). *Rekayasa Perangkat Lunak: Pendekatan Praktisi* (Buku 1). Yogyakarta: Andi.

- Purwodarsono, Abdullah dkk. (2008). *Djaka Lodang*. Yogyakarta: PT. Djaka Lodang Pers.
- Pusat Bahasa Departemen Pendidikan Nasional. (2008). "Kamus Besar Bahasa Indonesia", <http://pusatbahasa.kemdiknas.go.id/kbbi/>. Diakses pada tanggal 13 Desember 2011.
- Riduwan. (2010). *Belajar Mudah Penelitian untuk Guru, Karyawan, dan Peneliti Pemula*. Bandung : Alfabeta.
- Sakti, Galuh Fitriana & Astria, Anita. (2012). Pengaruh Pola Asuh Orangtua Demokratis Terhadap Prestasi Belajar Mahasiswa Esa Unggul Jurusan Psikologi 2011. Diakses dari <http://galuhsakti.blog.esaunggul.ac.id/category/uncategorized/> pada tanggal 20 Desember 2012.
- Seffah, Ahmed dkk. (2006). "Usability Measurement and Metrics: A Consolidated Model". *Software Qual J XXIV*. pp. 159-178.
- Setyosari, Punaji. (2010). *Metode Penelitian Pendidikan dan Pengembangan*. Jakarta: Kencana Prenada Media Group
- Sommerville, Ian. (2003). *Rekayasa Perangkat Lunak* (Edisi 6.). Jakarta: Erlangga
- Sudarmanto. (2011). *Kamus Lengkap Bahasa Jawa (Jawa-Indonesia, Indonesia-Jawa)*. Semarang:Widya Karya
- Sugiyono. (2009). *Metode Penelitian Pendidikan (Pendekatan Kuantitatif, Kualitatif, dan R&D)*. Bandung:CV.ALFABETA.
- _____. (2010). *Metode Penelitian Pendidikan (Pendekatan Kuantitatif, Kualitatif, dan R&D)*. Bandung:CV.ALFABETA.
- _____. (2012). *Metode Penelitian Pendidikan (Pendekatan Kuantitatif, Kualitatif, dan R&D)*. Bandung:CV.ALFABETA.
- Suharso dan Retnoningsih, A. (2005). *Kamus Besar Bahasa Indonesia Edisi Lux*. Semarang: CV Widya Karya.
- Sukoyo, Joko, Ratmini, dan Yulianti Dewi U. (2010). *Fokus: Lembar Kerja Siswa Basa Jawa untuk SMA/MA Kelas XI Semester 1*. Solo: CV. Sindunata.
- Suryawinata, Zuchridin dan Sugeng Hariyanto. (2003). *Translation: Bahasan Teori dan Penuntun Praktis Menerjemahkan*. Yogyakarta: Kanisius.

Wibowo, Wahyu. (2003). *Manajemen Bahasa: Pengorganisasian Karangan Pragmatik dalam Bahasa Indonesia untuk Mahasiswa dan Praktisi Bisnis*. Jakarta: PT Gramedia Pustaka Utama

LAMPIRAN

Lampiran 1. Kelas *kamusJawaIndo.java*

```
package jawIndoSource;
import java.io.IOException;
import java.util.Timer;
import java.util.TimerTask;
import javax.microedition.lcdui.*;
import javax.microedition.midlet.MIDlet;
import javax.microedition.midlet.MIDletStateChangeException;
import javax.microedition.rms.RecordStoreNotOpenException;
public class kamusJawaIndo extends MIDlet implements CommandListener {
    public Display display;
    public listMenuAwal menuAwal;
    public formCariKata menuCariKata;
    public formBantuan menuBantuan;
    public formTentangAplikasi menuTentangAplikasi;
    public splashInstallRMS installRMS;
    public loadingScreen loading;
    public Alert alertClass, alert, alPause;
    public static boolean isSafeToExit;
    private Image img;
    private DatabaseControl dbControl;
    private Command alYa, alTidak;
    private boolean started = false;
    private Timer timer;
    private TimerTask task;
    private static int TimerInterval=1000;
    private Thread bgThread;

    public kamusJawaIndo() {
        display = Display.getDisplay(this);
        dbControl = new DatabaseControl();
    }

    public void startApp() throws MIDletStateChangeException {
        try {
            if (!started) {
                dbControl.openRS();
                int countIndo = 0, countJawa = 0;
                countIndo = dbControl.dbIndo.getNumRecords();
                countJawa = dbControl.dbJawa.getNumRecords();
                if ((countJawa == 0 && countIndo == 0)) {
                    //kalo rms masih kosong langsung install
                    installRMS = new splashInstallRMS(this);
                    installRMS.show();
                } else if ((countJawa>0&&countJawa < 2224) || (countIndo>=0 && countIndo < 1613)) {
                    //kalo rms tidak terinstall dengan sempurna, beri pemberitahuan
                    alYa = new Command("Ya", Command.OK, 1);
                    alTidak = new Command("Tidak", Command.EXIT, 1);
                    alert = new Alert("Peringatan","Database tidak terpasang dengan benar. Pilih
\u0027Ya\u0027 untuk memasang database ulang dan tunggu hingga proses selesai. Pilih
\u0027Tidak\u0027 untuk keluar dari aplikasi!",null,AlertType.WARNING);
                    alert.addCommand(alYa);
                    alert.addCommand(alTidak);
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

        alert.setCommandListener(this);
        display.setCurrent(alert);
    } else {
        //kalo semua ud beres munculkan loading screen
        loading = new loadingScreen(this);
        loading.show();
    }
    started=true;
    } else {
    }
    } catch (IOException ex) {
        ex.printStackTrace();
    } catch (RecordStoreNotOpenException ex) {
        ex.printStackTrace();
    }
    }
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) throws MIDletStateChangeException {
    if (unconditional == true) {
        if (dbControl.isIndoOpen==true&&dbControl.isJawaOpen==true) {
            dbControl.closeRS();
        }
        notifyDestroyed();
    }
}

public void commandAction(Command c, Displayable d) {
    if (d==alert) {
        if (c==alTidak) {
            System.out.println("ini Keluar");
            alert.setCommandListener(null);
            try {
                destroyApp(true);
            } catch (MIDletStateChangeException ex) {
                ex.printStackTrace();
            }
        } else if (c==alYa) {
            dbControl.closeRS();
            dbControl.deleteRS();
            try {
                installRMS = new splashInstallRMS(this);
                installRMS.show();
            } catch (RecordStoreNotOpenException ex) {
                ex.printStackTrace();
            } catch (IOException ex) {
                ex.printStackTrace();
            }
        }
    }
}
}
}

```

Lampiran 2. Kelas *splashInstallRMS.java*

```
package jawIndoSource;
import javax.microedition.lcdui.*;
import java.io.*;
import java.util.Timer;
import javax.microedition.rms.RecordStoreNotOpenException;
public class splashInstallRMS extends Form {
    private kamusJawaIndo m;
    private ImageItem imItem;
    private Timer timer;
    private threadInstall timerTaskOne;
    private Image img;
    private Gauge gauge;
    private StringItem si, sIns;
    private boolean isSafeToExit, timersRunning;
    private DatabaseControl dbControl;

    public splashInstallRMS(final kamusJawaIndo m) throws RecordStoreNotOpenException,
    IOException {
        super("Kamus JawaIndo Mobile");
        this.m = m;
        dbControl = new DatabaseControl();
        try {
            img = Image.createImage("/icons/aplikasi.png");
        } catch(IOException e) {}
        imItem = new ImageItem(null, img, ImageItem.LAYOUT_CENTER, null);
        isSafeToExit = false;
        si = new StringItem(null, "Kamus Jawa-Indonesia\nIndonesia-Jawa");
        sIns = new StringItem(null, "Sedang Memasang Database.");
        gauge = new Gauge(null, false, Gauge.CONTINUOUS_RUNNING, Gauge.INDEFINITE);
        si.setLayout(3);
        sIns.setLayout(3);
        gauge.setLayout(gauge.LAYOUT_EXPAND);
        append(si);
        append("\n");
        append(imItem);
        append("\n");
        append(gauge);
        startInstall();
    }
    public void show(){
        try {
            dbControl.openRS();
            if (dbControl.dbJawa.getNumRecords()==0&&dbControl.dbIndo.getNumRecords()==0) {
                m.display.setCurrent(this);
            } else {
                Alert alert = new Alert("Peringatan", "Database tidak terpasang dengan benar, install
ulang aplikasi dapat memperbaikinya.", null, AlertType.INFO);
                alert.setTimeout(Alert.FOREVER);
                m.display.setCurrent(alert);
                m.menuAwal.show();
            }
        }
    }
}
```

Lampiran 3. Kelas *threadInstall.java*

```
package jawIndoSource;
import java.io.ByteArrayOutputStream;
import java.io.DataOutputStream;
import java.io.InputStream;
import java.util.TimerTask;
import java.util.Vector;
import javax.microedition.lcdui.Gauge;
import javax.microedition.lcdui.StringItem;
import javax.microedition.rms.RecordStore;
import javax.microedition.rms.RecordStoreNotOpenException;
public class threadInstall extends TimerTask{
    private kamusJawaIndo m;
    private Gauge gaugeInstall;
    private StringItem sIns;
    private int lastID, maxVector=0, num=0, max=0, i=0, maxJawIndo=0, maxJaw=0, persen;
    Vector recordIDs = new Vector();
    public RecordStore dbJawa=null, dbIndo=null;
    static final String rsJawa = "dbJawa", rsIndo = "dbIndo";
    private Vector vectorData1= new Vector(), vectorData2= new Vector(), vectorData3= new
    Vector(), vectorData4= new Vector();
    private DatabaseControl dbControl;
    private boolean cekInput;
    public threadInstall(StringItem ins, Gauge gauge, final kamusJawaIndo m){
        cekInput = true;
        this.gaugeInstall=gauge;
        this.m=m;
        this.sIns=ins;
        openRS();
        gaugeInstall.setLabel("Sedang memasang database!");
        gaugeInstall.setLayout(3);
        try {
            if (cekIsiRMS() == false) {
                writeDataJawIn();
                writeDataInJaw();
                this.max = getMaxVector();
                this.maxJaw = getMaxJaw();
                int maxIndo = max-maxJaw;
            }
        } catch (RecordStoreNotOpenException ex) {
            ex.printStackTrace();
        }
        gaugeInstall.setMaxValue(max);
    }
    public void run() {
        try {
            if (num < max) {
                if (cekInput==true) {
                    byte[] outputRecord;
                    ByteArrayOutputStream outputStream = new ByteArrayOutputStream();
                    DataOutputStream outputDataStream = new DataOutputStream(outputStream);
                    if (vectorData1.size()==vectorData2.size()) {
                        if (i < vectorData1.size()) {
                            outputDataStream.writeUTF(vectorData1.elementAt(i).toString());
                        }
                    }
                }
            }
        }
    }
}
```

```

        outputStream.writeUTF(vectorData2.elementAt(i).toString());
        outputStream.flush();
        outputRecord = outputStream.toByteArray();
        dbJawa.addRecord(outputRecord, 0, outputRecord.length);
        outputStream.reset();
        i++;
        num++;
        gaugeInstall.setValue(num);
    }
    if (num>maxJaw-1) {
        i = 0;
        outputStream.close();
        outputStream.close();
        cekInput=false;
    }
} else {
    byte[] outputRecord;
    ByteArrayOutputStream outputStream = new ByteArrayOutputStream();
    DataOutputStream outputStreamData = new DataOutputStream(outputStream);
    if (vectorData3.size()==vectorData4.size()) {
        if (i < vectorData3.size()) {
            outputStreamData.writeUTF(vectorData3.elementAt(i).toString());
            outputStreamData.writeUTF(vectorData4.elementAt(i).toString());
            outputStreamData.flush();
            outputRecord = outputStream.toByteArray();
            dbIndo.addRecord(outputRecord, 0, outputRecord.length);
            outputStream.reset();
            i++;
            num++;
            gaugeInstall.setValue(num);
        }
    }
} else {
    gaugeInstall.setLabel("Selesai.");
    this.cancel();
    m.menuAwal = new listMenuAwal(m);
    m.menuAwal.show();
}
} catch (Exception e) {}
}
public String file(String path) {
    InputStream iniDb = getClass().getResourceAsStream(path);
    StringBuffer sb = new StringBuffer();
    try {
        int charsJawa;
        while ((charsJawa = iniDb.read()) != -1) {
            sb.append((char) charsJawa);
        }
        return sb.toString();
    } catch (Exception e) {}
    return null;
}
public static String[] splitC(String s, char separator) {

```

```

Vector v = new Vector();
for (int ini = 0, end = 0; ini < s.length(); ini = end + 1) {
    end = s.indexOf(separator, ini);
    if (end == -1) {
        end = s.length();
    }
    String st = s.substring(ini, end).trim();
    if (st.length() > 0) {
        v.addElement(st);
    } else {
        v.addElement("null");
    }
}
String temp[] = new String[v.size()];
v.copyInto(temp);
return temp;
}
public void writeDataJawIn() {
    String[] kata = splitC(file("/text/jawindo.txt"), '\n');
    for(int i=0;i<kata.length;i++) {
        String kataTrans = kata[i];
        int len = kataTrans.length();
        for(int j=0;j<len;j++) {
            if(kataTrans.charAt(j)==':') {
                vectorData1.addElement(kataTrans.substring(0,j));
                vectorData2.addElement(kataTrans.substring(j+1,len));
                maxVector++;
                maxJawIndo++;
            }
        }
    }
}
public void writeDataInJaw() {
    String[] kata = splitC(file("/text/indojaw.txt"), '\n');
    for(int i=0;i<kata.length;i++) {
        String kataTrans = kata[i];
        int len = kataTrans.length();
        for(int j=0;j<len;j++) {
            if(kataTrans.charAt(j)==':') {
                vectorData3.addElement(kataTrans.substring(0,j));
                vectorData4.addElement(kataTrans.substring(j+1,len));
                maxVector++;
            }
        }
    }
}
public void openRS() {
    try {
        dbJawa = RecordStore.openRecordStore(rsJawa, true);
        dbIndo = RecordStore.openRecordStore(rsIndo, true);
    } catch (Exception e) {
        db(e.toString());
    }
}
}

```

```

public boolean cekIsiRMS() throws RecordStoreNotOpenException {
    if (dbJawa.getNumRecords()==0&&dbIndo.getNumRecords()==0)
        return false;
    else
        return true;
}
public void writeRecords(Vector outputData1, Vector outputData2, RecordStore database) {
    try {
        if (database.getNumRecords() == 0) {
            byte[] outputRecord;
            ByteArrayOutputStream outputStream = new ByteArrayOutputStream();
            DataOutputStream outputStream2 = new DataOutputStream(outputStream);
            if (outputData1.size()==outputData2.size()) {
                for (int i = 0; i < outputData1.size(); i++) {
                    outputStream2.writeUTF(outputData1.elementAt(i).toString());
                    outputStream2.writeUTF(outputData2.elementAt(i).toString());
                    outputStream2.flush();
                    outputRecord = outputStream.toByteArray();
                    database.addRecord(outputRecord, 0, outputRecord.length);
                    recordIDs.addElement(new Integer(++lastID));
                    outputStream.reset();
                }
                outputStream.close();
                outputStream2.close();
                System.out.println(recordIDs.size());
            }
        }
    } catch (Exception e) {
        db(e.toString());
    }
}
public int getMaxVector() {
    return maxVector;
}
public int getMaxJaw() {
    return maxJawIndo;
}
private void db(String str) {
    System.err.println("Msg: " + str);
}
}

```

Lampiran 4. Kelas *listMenuAwal.java*

```
package jawIndoSource;
import java.io.IOException;
import javax.microedition.lcdui.*;
import javax.microedition.midlet.MIDletStateChangeException;
public class listMenuAwal extends List implements CommandListener {
    public String tittle;
    public int code;
    private kamusJawaIndo m;
    private String jawa = "Jawa-Indo", indo = "Indo-Jawa";
    private Image img;
    private DatabaseControl dbControl;
    private Alert alert;
    private Command alYes = new Command("Ya", Command.EXIT, 1);
    private Command alNo = new Command("Tidak", Command.CANCEL, 2);
    public listMenuAwal(final kamusJawaIndo m){
        super("Kamus JawIndo Mobile", IMPLICIT);
        this.m = m;
        dbControl = new DatabaseControl();
        append("Jawa-Indo",getImg("/icons/Jawa-Indo.png"));
        append("Indo-Jawa",getImg("/icons/Indo-Jawa.png"));
        append("Bantuan",getImg("/icons/bantuan.png"));
        append("Tentang Aplikasi", getImg("/icons/tentang aplikasi.png"));
        append("Keluar",getImg("/icons/exit.png"));
        setCommandListener(this);
    }
    public Image getImg(String pathIm) {
        try {
            img = Image.createImage(pathIm);
        } catch(IOException e) {}
        return img;
    }
    public void show() {
        m.display.setCurrent(this);
    }
    public void commandAction(Command c, Displayable d) {
        if (d==this) {
            if (getSelectedIndex() == 0) {
                code = 1;
                m.menuCariKata = new formCariKata(m);
                m.menuCariKata.setTextField("Masukkan kata (Jawa)", null, 30, 0);
                m.menuCariKata.append(m.menuCariKata.getTF());
                tittle = jawa;
                m.menuCariKata.show();
            } else if (getSelectedIndex() == 1) {
                code = 2;
                m.menuCariKata = new formCariKata(m);
                m.menuCariKata.setTextField("Masukkan kata (Indo)", null, 30, 0);
                m.menuCariKata.append(m.menuCariKata.getTF());
                tittle = indo;
                m.menuCariKata.show();
            } else if (getSelectedIndex() == 2) {
                m.menuBantuan = new formBantuan(m);
                m.menuBantuan.show();
            }
        }
    }
}
```


Lampiran 5. Kelas *formCariKata.java*

```
package jawIndoSource;
import java.io.ByteArrayInputStream;
import java.io.DataInputStream;
import javax.microedition.lcdui.*;
import javax.microedition.rms.RecordStore;
import javax.microedition.rms.RecordStoreNotOpenException;
public class formCariKata extends Form implements ItemStateListener, CommandListener,
ItemCommandListener{
    private kamusJawaIndo m;
    private Thread threadWav;
    private TextField tfInput;
    private boolean flag, isSafeToClose, if1, if2, pelafalan;
    private StringItem result[]= new StringItem[5], trans;
    private String label, text, input, suara;
    public Alert alert, alTrans;
    private Gauge gauge, inCari;
    private String[] arrKata=new String[5], arrTerjemahan=new String[5];
    private char charInput1, charInput2;
    private int pJg, type, code, idCase;
    private int[] idxJawa, idxIndo, id;
    private boolean avPlay, isCmdExist, avail, playStat, jawa;
    private DatabaseControl dbControl;
    private Command cmdBack = new Command("Menu Awal",Command.BACK,1);
    private Command cmdFind = new Command("Cari",Command.SCREEN,1);
    private Command cmdCariLagi = new Command("Cari Lagi",Command.BACK,1);
    private Command cmdPlay = new Command("Pelafalan",Command.OK,1);
    private Command cmdCancel = new Command("Cancel",Command.CANCEL,1);
    public formCariKata(final kamusJawaIndo m) {
        super("Cari Kata");
        this.m = m;
        dbControl = new DatabaseControl();
        addCommand(cmdBack);
        addCommand(cmdCariLagi);
        setCommandListener(this);
        setItemStateListener(this);
    }
    public void show() {
        m.display.setCurrent(this);
    }
    public void itemStateChanged(Item item) {
        if (item == tfInput) {
            clearForm();
            int awalJ=0,akhirJ=0,awalI=0,akhirI=0;
            dbControl.openRS();
            if (m.menuAwal.code==1) {
                pelafalan = true;
                jawa = true;
                idxJawa = indexingJawa();
                awalJ = idxJawa[0];
                akhirJ = idxJawa[1];
                searchNsort(tfInput.getString().toLowerCase(), dbControl.dbJawa, awalJ, akhirJ);
            } else if (m.menuAwal.code==2) {
                pelafalan = false;
            }
        }
    }
}
```

```

        jawa = false;
        idxIndo = indexingIndo();
        awalI = idxIndo[0];
        akhirI = idxIndo[1];
        searchNsort(tfInput.getString().toLowerCase(), dbControl.dbIndo, awalI, akhirI);
    } else
        clearForm();
    if (play()==true) {
        System.out.println(pelafalan);
        this.addCommand(cmdPlay);
        isCmdExist=true;
    } else if (isCmdExist=true) {
        this.removeCommand(cmdPlay);
    }
    if (if2 == true) {
        if (this.size()>=1) {
            this.append("Kata Tidak Ditemukan");
        }
    }
}

public boolean invalidInput() {
    boolean invalid;
    if (((int)'z'<charInput1||charInput1<(int)'A')&&tfInput.getString()!=""){
        invalid=true;
    } else
        invalid=false;
    return invalid;
}

public void buatStringItem(int jmlSI) {
    for (int idxSI=0;idxSI<jmlSI;idxSI++){
        result[idxSI] = new StringItem(null,arrKata[idxSI]+"\\n",1);
    }
    for (int i=0;i<result.length;i++) {
        this.append(result[i]);
        result[i].setDefaultCommand(cmdFind);
        result[i].setItemCommandListener(this);
    }
    if (jmlSI<5) {
        for (int j=this.size();j>jmlSI+1;j--) {
            this.delete(jmlSI+1);
        }
    }
}

public void commandAction(Command c, Displayable d) {
    String cLabel = c.getLabel().toLowerCase();
    if(cLabel.equals("menu awal")) {
        m.menuAwal = new listMenuAwal(m);
        m.menuAwal.show();
    } else if (cLabel.equals("cari lagi")) {
        clearForm();
        tfInput.setString("");
        if (isCmdExist=true) {
            this.removeCommand(cmdPlay);
        }
    }
}

```

```

    } else if (cLabel.equals("pelafalan")) {
        alert = new Alert("Menghubungkan", "Sedang melakukan sambungan...Tunggu
sebentar...", null, AlertType.INFO);
        Gauge gauge = new Gauge(null, false, Gauge.INDEFINITE,
Gauge.INCREMENTAL_UPDATING);
        alert.setTimeout(alert.FOREVER);
        alert.addCommand(cmdCancel);
        alert.setCommandListener(this);
        gauge.setValue(2);
        alert.setIndicator(gauge);
        m.display.setCurrent(alert, this);
        streamWav wav = new streamWav();
        if (jawa=true)
            suara = tfInput.getString();
        else
            suara = this.get(1).toString();
        wav.setWav(suara.toLowerCase(), alert);
        alert.setCommandListener(null);
        alert.removeCommand(cmdCancel);
        wav.playWav();
    } else if (cLabel.equals(cmdCancel)) {
        alert.setTitle("Peringatan");
        alert.setString("Tunggu hingga proses koneksi selesai!");
        alert.setType(AlertType.WARNING);
    }
}
}
public void clearForm() {
    if (this.size()!=1) {
        for (int count = 1;this.size()!=1;count++)
            this.delete(1);
    }
}
public void setTextField(String label, String text, int pjg, int type){
    this.label = label;
    this.text = text;
    this.pjg = pjg;
    this.type = type;
}
public TextField getTF() {
    tfInput = new TextField(label, text, pjg, type);
    return tfInput;
}
public Alert alert(String label, String string){
    alert = new Alert(label, string, null, AlertType.INFO);
    alert.setTimeout(Alert.FOREVER);
    return alert;
}
public void prosesInput() {
    input = tfInput.getString().toLowerCase();
    if (input.length()==1) {
        charInput1 = input.charAt(0);
    } else if(input.length(>1) {
        charInput1 = input.charAt(0);
        charInput2 = input.charAt(1);
    }
}

```

```

}
public int[] indexingJawa() {
    int[] index = new int[2];
    index[0]=0;
    index[1]=0;
    prosesInput();
    if (input.length()!=0) {
        switch (charInput1) {
            case 'a':
                if (isAv()==false) {
                    index[0] = 1;
                    index[1] = 5;
                    break;
                } else if (isAv()==true) {
                    if ((int)'n'<=charInput2&&charInput2<=(int)'y') {
                        index[0] = 61;
                        index[1] = 129;
                        break;
                    } else {
                        index[0] = 1;
                        index[1] = 60;
                        break;
                    }
                }
            case 'b':index[0] = 130;index[1] = 217;break;
            case 'c':index[0] = 218;index[1] = 274;break;
            case 'd':index[0] = 275;index[1] = 335;break;
            case 'e': index[0] = 361; index[1] = 382; break;
            case 'é': index[0] = 336; index[1] = 352; break;
            case 'è': index[0] = 353; index[1] = 360; break;
            case 'g':
                if (isAv()==false) {
                    index[0] = 383;
                    index[1] = 387;
                    break;
                } else if (isAv()==true) {
                    if ((int)'i'<=charInput2&&charInput2<=(int)'u'){
                        index[0] = 451;
                        index[1] = 512;
                        break;
                    } else {
                        index[0] = 383;
                        index[1] = 450;
                        break;
                    }
                }
            case 'h': index[0] = 513; index[1] = 518; break;
            case 'i': index[0] = 519; index[1] = 566; break;
            case 'j': index[0] = 567; index[1] = 633; break;
            case 'k':
                if (isAv()==false) {
                    index[0] = 634;
                    index[1] = 638;
                    break;
                } else if (isAv()==true) {

```

```

    if (charInput2==(int)'e'){
        index[0] = 692;
        index[1] = 770;
        break;
    } else if ((int)'i'<=charInput2&&charInput2<=(int)'u' ) {
        index[0] = 785;
        index[1] = 861;
        break;
    } else if ((int)'è'<=charInput2&&charInput2<=(int)'é' ) {
        index[0] = 771;
        index[1] = 784;
        break;
    } else {
        index[0] = 634;
        index[1] = 691;
        break;
    }
}
case 'l':
    if (isAv()==false) {
        index[0] = 862;
        index[1] = 866;
        break;
    } else if (isAv()==true) {
        if ((int)'i'<=charInput2&&charInput2<=(int)'u'){
            index[0] = 922;
            index[1] = 962;
            break;
        } else {
            index[0] = 862;
            index[1] = 921;
            break;
        }
    }
case 'm':
    if (isAv()==false) {
        index[0] = 963;
        index[1] = 967;
        break;
    } else if (isAv()==true) {
        if ((int)'a'<=charInput2&&charInput2<=(int)'e'){
            index[0] = 963;
            index[1] = 1062;
            break;
        } else {
            index[0] = 1063;
            index[1] = 1137;
            break;
        }
    }
case 'n': index[0] = 1138; index[1] = 1199; break;
case 'o': index[0] = 1200; index[1] = 1214; break;
case 'p':
    if (isAv()==false) {
        index[0] = 1215;

```

```

        index[1] = 1219;
        break;
    } else if (isAv()==true) {
        if (charInput2==(int)'a'){
            index[0] = 1215;
            index[1] = 1287;
            break;
        } else if (charInput2==(int)'e') {
            index[0] = 1288;
            index[1] = 1317;
            break;
        } else if ((int)'i'<=charInput2&&charInput2<=(int)'l') {
            index[0] = 1318;
            index[1] = 1367;
            break;
        } else {
            index[0] = 1368;
            index[1] = 1442;
            break;
        }
    }
}
case 'r':
    if (isAv()==false) {
        index[0] = 1443;
        index[1] = 1447;
        break;
    } else if (isAv()==true) {
        if ((int)'i'<=charInput2&&charInput2<=(int)'u'){
            index[0] = 1503;
            index[1] = 1544;
            break;
        } else {
            index[0] = 1443;
            index[1] = 1502;
            break;
        }
    }
}
case 's':
    if (isAv()==false) {
        index[0] = 1545;
        index[1] = 1549;
        break;
    } else if (isAv()==true) {
        if (charInput2==(int)'a') {
            index[0] = 1545;
            index[1] = 1646;
            break;
        } else if ((int)'i'<=charInput2&&charInput2<=(int)'w'){
            index[0] = 1739;
            index[1] = 1838;
            break;
        } else {
            index[0] = 1647;
            index[1] = 1738;
            break;
        }
    }
}

```

```

    }
  }
  case 't':
    if (isAv()==false) {
      index[0] = 1839;
      index[1] = 1843;
      break;
    } else if (isAv()==true) {
      if ((int)'h'<=charInput2&&charInput2<=(int)'u'){
        index[0] = 1918;
        index[1] = 1989;
        break;
      } else {
        index[0] = 1839;
        index[1] = 1917;
        break;
      }
    }
  }
  case 'u': index[0] = 1990; index[1] = 2071; break;
  case 'w':
    if (isAv()==false) {
      index[0] = 2072;
      index[1] = 2076;
      break;
    } else if (isAv()==true) {
      if ((int)'i'<=charInput2&&charInput2<=(int)'u'){
        index[0] = 2161;
        index[1] = 2210;
        break;
      } else {
        index[0] = 2072;
        index[1] = 2160;
        break;
      }
    }
  }
  case 'y': index[0] = 2211; index[1] = 2224; break;
}
}
return index;
}
public int[] indexingIndo() {
  int[] index = new int[2];
  index[0]=0;
  index[1]=0;
  prosesInput();
  if (input.length()!=0) {
    switch (charInput1) {
      case 'a':
        if (isAv()==false) {
          index[0] = 1;
          index[1] = 5;
          break;
        } else if (isAv()==true) {
          if ((int)'m'<=charInput2&&charInput2<(int)'p') {
            index[0] = 57;

```

```

        index[1] = 126;
        break;
    } else if ((int)'p'<=charInput2&&charInput2<=(int)'y'){
        index[0] = 127;
        index[1] = 186;
        break;
    } else {
        index[0] = 1;
        index[1] = 56;
        break;
    }
}
}
case 'b':
    if (isAv()==false) {
        index[0] = 187;
        index[1] = 191;
        break;
    } else if (isAv()==true) {
        if ((int)'e'<=charInput2&&charInput2<(int)'l' {
            index[0] = 267;
            index[1] = 347;
            break;
        } else if ((int)'l'<=charInput2&&charInput2<=(int)'u'){
            index[0] = 348;
            index[1] = 423;
            break;
        } else {
            index[0] = 187;
            index[1] = 266;
            break;
        }
    }
}
case 'c':
    if (isAv()==false) {
        index[0] = 424;
        index[1] = 428;
        break;
    } else if (isAv()==true) {
        if ((int)'e'<=charInput2&&charInput2<=(int)'u' {
            index[0] = 481;
            index[1] = 561;
            break;
        } else {
            index[0] = 424;
            index[1] = 480;
            break;
        }
    }
}
case 'd':
    if (isAv()==false) {
        index[0] = 562;
        index[1] = 566;
        break;
    } else if (isAv()==true) {
        if ((int)'o'<=charInput2&&charInput2<=(int)'u' {

```

```

        index[0] = 643;
        index[1] = 697;
        break;
    } else {
        index[0] = 562;
        index[1] = 642;
        break;
    }
}
case 'e': index[0] = 698; index[1] = 773; break;
case 'f': index[0] = 774; index[1] = 799; break;
case 'g': index[0] = 800; index[1] = 848; break;
case 'h': index[0] = 849; index[1] = 903; break;
case 'i': index[0] = 904; index[1] = 925; break;
case 'j': index[0] = 926; index[1] = 979; break;
case 'k': index[0] = 980; index[1] = 1072; break;
case 'l': index[0] = 1073; index[1] = 1127; break;
case 'm':
    if (isAv()==false) {
        index[0] = 1128;
        index[1] = 1132;
        break;
    } else if (isAv()==true) {
        if (charInput2==(int)'e'){
            index[0] = 1174;
            index[1] = 1221;
            break;
        } else {
            index[0] = 1128;
            index[1] = 1173;
            break;
        }
    }
case 'n': index[0] = 1222; index[1] = 1236; break;
case 'o': index[0] = 1237; index[1] = 1242; break;
case 'p':
    if (isAv()==false) {
        index[0] = 1243;
        index[1] = 1247;
        break;
    } else if (isAv()==true) {
        if (charInput2==(int)'e'){
            index[0] = 1287;
            index[1] = 1347;
            break;
        } else {
            index[0] = 1243;
            index[1] = 1286;
            break;
        }
    }
case 'r': index[0] = 1348; index[1] = 1374; break;
case 's': index[0] = 1375; index[1] = 1448; break;
case 't':
    if (isAv()==false) {

```

```

        index[0] = 1449;
        index[1] = 1453;
        break;
    } else if (isAv()==true) {
        if ((int)'i'<=charInput2&&charInput2<=(int)'u'){
            index[0] = 1525;
            index[1] = 1561;
            break;
        } else {
            index[0] = 1449;
            index[1] = 1524;
            break;
        }
    }
    case 'u': index[0] = 1562; index[1] = 1587; break;
    case 'w': index[0] = 1588; index[1] = 1607; break;
    case 'y': index[0] = 1608; index[1] = 1613; break;
}
}
return index;
}

public void searchNsort(String dicari, RecordStore database, int indexAwal, int indexAkhir) {
    byte[] byteInputData = new byte[1000];
    ByteArrayInputStream inputStream = new ByteArrayInputStream(byteInputData);
    DataInputStream inputDataStream = new DataInputStream(inputStream);
    StringBuffer sbuff = new StringBuffer();
    int counter = 0, countArr=0;
    int lenDicari = dicari.length();
    String tmpSelection, tmpKata, compTmpKata;
    try {
        for (int recId=indexAwal;recId<=indexAkhir;recId++) {
            database.getRecord(recId, byteInputData, 0);
            sbuff.append(inputDataStream.readUTF());
            tmpSelection = sbuff.toString();
            tmpKata = tmpSelection.substring(counter, tmpSelection.length());
            if (lenDicari<=tmpKata.length()){
                compTmpKata = tmpKata.substring(0, lenDicari);
                if (compTmpKata.equals(dicari)){
                    inputDataStream.reset();
                    for (int j=recId;j<=recId+4;j++){
                        if (j<=database.getNumRecords()) {
                            database.getRecord(j, byteInputData, 0);
                            String kata = inputDataStream.readUTF().toString();
                            String terjemahan = inputDataStream.readUTF().toString();
                            int containIdx = kata.indexOf(dicari);
                            int containIdxT = terjemahan.indexOf(dicari);
                            if (containIdx!=-1) {
                                arrKata[countArr]=kata;
                                arrTerjemahan[countArr]=terjemahan;
                                if1 = false;
                                if2 = false;
                            } else {
                                break;
                            }
                        }
                    }
                    countArr++;
                }
            }
        }
    }
}

```


Lampiran 6. Kelas *streamWav.java*

```
package jawIndoSource;
import java.io.IOException;
import java.io.InputStream;
import java.util.Timer;
import javax.microedition.io.Connector;
import javax.microedition.io.HttpConnection;
import javax.microedition.lcdui.Alert;
import javax.microedition.lcdui.AlertType;
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Displayable;
import javax.microedition.media.Manager;
import javax.microedition.media.MediaException;
import javax.microedition.media.Player;
import javax.microedition.media.PlayerListener;
import javax.microedition.media.control.VolumeControl;
public class streamWav implements Runnable, CommandListener, PlayerListener {
    private kamusJawaIndo m;
    private Thread threadPlay;
    public static Player player;
    private Alert alert;
    private VolumeControl vc;
    private String folder, file;
    private Timer timer;
    private boolean isPlayed, isConnected, isStop, readyP, readyC;
    private boolean isUnreal, isReal, isPref, isStart, isClose, var;
    private int count=0;
    private HttpConnection conn;
    private InputStream is;
    private Command cmdCancel = new Command("Gagalkan",Command.BACK,1);
    private Command cmdStop = new Command("Stop",Command.OK,1);
    private Command cmdDone = new Command("Selesai", Command.OK,1);
    private int status;
    private String url;
    private StringBuffer sbOut,sbHasil;
    private InputStream inputStream;
    public void setWav(String setFile, Alert alert) {
        this.alert = alert;
        StringBuffer tmpFile=new StringBuffer(setFile);
        String output;
        if (setFile.indexOf(" ")!=-1) {
            for (int i=0;i<tmpFile.length();i++) {
                if (tmpFile.charAt(i)==' ') {
                    tmpFile.deleteCharAt(i);
                }
                if (tmpFile.toString().indexOf(" ")==-1){
                    file = tmpFile.toString();
                    break;
                }
            }
        }
        } else
        this.file = setFile;
    }
}
```

```

public void playWav() {
    threadPlay = new Thread(this);
    alert.addCommand(cmdCancel);
    alert.setCommandListener(this);
    threadPlay.start();
    readyP=true;
    readyC=true;
    isStop=false;
    var=true;
}
public void run() {
    if (readyC=true) {
        try {
            url="http://belajarjawa.host22.com/audiojawa/"+file+".wav";
HttpConnection conn = (HttpConnection) Connector.open(url,Connector.READ_WRITE);
            isConnected=true;
            if (conn.getResponseCode()==conn.HTTP_OK) {
                alert.setString("Menghubungkan ke "+ conn.getHost().toString()+"\nTunggu sebentar...");
                InputStream is = conn.openInputStream();
                setIS(is);
                if (readyP==true) {
                    alert.removeCommand(cmdCancel);
                    alert.addCommand(cmdStop);
                    alert.setCommandListener(this);
                    isUnreal=true;
                    pStarted();
                    isPlayed=false;
                } else
                    alAborted();
            } else if (conn.getResponseCode()!=conn.HTTP_OK) {
                alert.setType(AlertType.ERROR);
                alert.setTitle("Kesalahan");
                alert.setString("File tidak ditemukan di server!");
                alert.setCommandListener(null);
                alert.setTimeout(2000);
            }
            conn.close();
        } catch (MediaException ex) {
            isConnected=false;
            alert.setTitle("Kesalahan");
            alert.setType(AlertType.ERROR);
            alert.setString("Format file suara tidak didukung");
            alert.setCommandListener(null);
            alert.setTimeout(Alert.FOREVER);
        } catch (IOException ex) {
            alert.setTitle("Kesalahan");
            alert.setType(AlertType.ERROR);
            alert.setString("Pastikan ponsel anda terhubung Internet!");
            alert.setCommandListener(null);
            alert.setTimeout(Alert.FOREVER);
        } catch (InterruptedException ex) {
            System.out.println("ini error InterruptedException");
            ex.printStackTrace();
        } catch (Exception ex) {
            System.out.println("error yang g tertangani= "+ex.toString());
        }
    }
}

```

```

    }
    } else {
        alAborted();
    }
}
public void commandAction(Command c, Displayable d) {
    if (c==cmdCancel) {
        if (isConnected=true) {
            readyP=false;
        } else {
            isConnected=false;
            try {
                conn.close();
            } catch (IOException ex) {
                ex.printStackTrace();
            }
        }
        alAborted();
    } else if (c==cmdStop) {
        if (isPlayed=true) {
            try {
                defPlayer();
            } catch (MediaException ex) {
                ex.printStackTrace();
            }
            alDone();
        } else {
            alDone();
        }
    }
}
public void setIS(InputStream inStream) {
    this.inputStream = null;
    this.inputStream = inStream;
}
public InputStream getIS() {
    return inputStream;
}
public void alAborted() {
    alert.setTitle("Gagal");
    alert.setType(AlertType.ERROR);
    alert.setString("Digagalkan pengguna");
    alert.removeCommand(cmdCancel);
    alert.setCommandListener(null);
    alert.setTimeout(Alert.FOREVER);
}
void alDone() {
    alert.setTitle("Selesai");
    alert.setType(AlertType.INFO);
    alert.setString("Pelafalan selesai");
    alert.removeCommand(cmdStop);
    alert.setCommandListener(null);
    alert.setTimeout(Alert.FOREVER);
}
}






```

```

public void createP() throws IOException, MediaException, InterruptedException {
    isReal=true;
    alert.setString("Creating player...");
    player = Manager.createPlayer(getIS(), "audio/x-wav");
    player.addPlayerListener(this);
}
public void realizeP() throws IOException, MediaException, InterruptedException {
    isPref=true;
    alert.setString("Realizing player...");
    player.realize();
}
public void prefetchP() throws IOException, MediaException, InterruptedException {
    isStart=true;
    alert.setString("Prefetching player...");
    vc = (VolumeControl) player.getControl("VolumeControl");
    if (vc != null) {
        vc.setLevel(100);
    }
    player.prefetch();
}
void startP() throws MediaException {
    int dur = (int) player.getDuration() / 1000;
    alert.setTitle("Pelafalan");
    alert.setString("Starting player...");
    player.start();
}
void pStarted() throws IOException, MediaException, InterruptedException {
    if (isUnreal==var) {
        createP();
    }
    if (isReal==var) {
        realizeP();
    }
    if (isPref==var) {
        prefetchP();
    }
    if (isStart==var) {
        startP();
    }
    } else
        System.out.println(var);
    } else
        System.out.println(var);
    } else
        System.out.println(var);
    } else
        System.out.println(var);
    var=true;
}
public void playerUpdate(Player player, String event, Object eventData) {
    if (event.equals(PlayerListener.END_OF_MEDIA)) {
        alDone();
    }
}
}
}
}

```

Lampiran 7. Hasil *Black Box Testing*

No.	Test Case	Fungsi yang dirancang	Output
1	Pengguna mengeksekusi aplikasi kamus bahasa Jawa.	Aplikasi kamus bahasa Jawa muncul dan menampilkan proses instalasi <i>database</i> dan setelah selesai segera menampilkan menu utama.	OK
2	Pengguna memilih menu Jawa-Indo 	Menuju menu cari kata (jawa)	OK
3	Pengguna memilih menu Indo-Jawa 	Menuju menu cari kata (indo)	OK
4	Pengguna memilih menu Bantuan 	Menuju menu bantuan	OK
5	Pengguna memilih menu Tentang Aplikasi 	Menuju menu tentang aplikasi	OK
6	Pengguna memilih keluar 	Memunculkan kotak dialog konfirmasi untuk keluar program atau tidak	OK
7	Pengguna memilih “ya” pada kotak dialog keluar	Keluar dari aplikasi	OK
8	Pengguna memilih “tidak” pada kotak dialog keluar	Menampilkan kembali menu utama	OK
Menu Jawa-Indo			
9	Pengguna memasukkan kata yang ingin dicari	Menampilkan kemungkinan kata yang dimaksud	OK
10	Pengguna mengarahkan kursor <i>keypad</i> pada kemungkinan kata yang muncul	Kemungkinan kata yang muncul dapat dipilih	OK
11	Pilih kata yang dimaksud dengan menekan tombol tengah kursor (<i>fire key</i>) atau pilih tombol “cari”	Menampilkan arti dari kata yang dimaksud	OK
12	Pilih tombol “pelafalan”	Muncul kotak dialog pelafalan dan proses pelafalan berjalan	OK
13	Pilih tombol “gagalkan” pada kotak dialog pelafalan	Pelafalan dibatalkan dan kembali menampilkan menu cari kata	OK
14	Pilih tombol “cari lagi”	Merest hasil pencarian	OK
15	Tekan tombol tengah kursor (<i>fire key</i>) atau pilih tombol “cari” ketika masukan masih kosong	Menampilkan kotak dialog bahwa masukan masih kosong	
16	Pengguna memasukkan sembarang masukan	Menampilkan pemberitahuan kata tidak ditemukan	OK

Menu Indo-Jawa

17	Pengguna memasukkan kata yang ingin dicari	Menampilkan kemungkinan kata yang dimaksud	OK
18	Pengguna mengarahkan kursor <i>keypad</i> pada kemungkinan kata yang muncul	Kemungkinan kata yang muncul dapat dipilih	OK
19	Pilih kata yang dimaksud dengan menekan tombol tengah kursor (<i>fire key</i>) atau pilih tombol “cari”	Menampilkan arti dari kata yang dimaksud	OK
20	Pilih tombol “cari lagi”	Merest hasil pencarian	OK
21	Tekan tombol tengah kursor (<i>fire key</i>) atau pilih tombol “cari” ketika masukan masih kosong	Menampilkan kotak dialog bahwa masukan masih kosong	

Lampiran 8. Penilaian Validasi Ahli Software

LEMBAR UJI KELAYAKAN AHLI SOFTWARE

Berilah tanda centang (✓) pada pilihan *PASS*, *FAIL*, *PASSED WITH EXCEPTION* yang disediakan sesuai dengan penilaian untuk pengujian *mobile application* sebagai hasil dari skripsi: **“PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)”** yang disusun oleh **Kartika Yudha Pratama**.

PS1	Permission Statements	
<u>Test Description</u> The application should not ask for any permissions.		
<u>Steps to conduct the test</u> 1. Open the JAD and the manifest 2. Observe that there are no permission statements		<u>Expected Result</u> -There are no MIDlet-Permissions statements in the JAD and manifest
<u>Notes</u> Any application that merits an exception should then be subjected to test SA1		<u>Exceptions</u> 1. Application is a demo which connects to a site accessed by a browser to download full version 2. Application connects to a server to provide in-application advertising 3. Application has no other purpose than to launch a browser session
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

ST1	Application stability	
<u>Full Description</u> The application must not crash or freeze at any time while running on the device.		
<u>Steps to conduct the test</u> 1. Observe the application behaviour during the testing		<u>Expected Result</u> -The application must not stop the user experience unexpectedly without any user input.
<u>Notes</u> -During any time of the testing observe the application behaviour -The report must indicate if the error can be reproduced or not		<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

AL1		Application Installation	
<u>Test Description</u> The application must install via OTA			
<u>Steps to conduct the test</u> 1. Open the browser application of the device 2. Type the URL of the application JAD file 3. Connect to the typed URL 4. Accept the installation of the application		<u>Expected Result</u> -The application installs to the device -The icon for the application can be found from the device	
<u>Notes</u> If errors occur at installation time, corresponding messages must be reported by the test house in the test report.		<u>Exceptions</u> If the device does not display the icon, then the user must be able to start the application using other means.	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

AL2		Application start up	
<u>Test Description</u> Application must start properly in 25s.			
<u>Steps to conduct the test</u> 1. Find the application icon and select it 2. "Press a button" on the device to launch the application 3. Observe the application launch In the timeline defined 4. The application should have displayed a main menu or interactive menu such as language selection screen where the use of the application can be started 5. Use some of the application features		<u>Expected Result</u> -The application starts in 25s or less, this is the time between steps 2 and 4 -No error messages are displayed -The application appears to function properly	
<u>Notes</u> If launch time errors occur, corresponding messages must be reported by the test house in the test report. This test does not take into consideration the different screens displayed between the "button press" and the display of the main menu of the application. For example branded splash screen.		<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

UI1	Graphic clarity	
<u>Full Description</u>		
All graphics and animations displayed must be readable and clear to the user.		
<u>Steps to conduct the test</u>	<ol style="list-style-type: none"> 1. Launch application in target language 2. Check graphics appearing in <ol style="list-style-type: none"> f) Splash/Title/Logo/Loading Screen g) Main Menu and all its subsidiary menus h) Help/Instructions Screen(s) i) About screen j) Application Pause Menu and all its subsidiary menus (if present) 3. Repeat steps 1 and 2 for each language version of the game 	<u>Expected Result</u>
<ul style="list-style-type: none"> -The application must utilise the full screen size available to them, applicable to the target device. - If device's screen orientation can be changed during application execution (e.g. from portrait to landscape mode) the application must adapt its appearance accordingly, so that the requirement above is still met - There should be no event in the defined areas of the application that would prevent the user from understanding the functionality of the application. <p>For example: a graphical display issue including but not necessarily limiting to the following: overlapping graphics, colour conflict, images truncated and/or displayed incorrectly</p>		
<u>Notes</u>	<ul style="list-style-type: none"> -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application. -Definition of full screen may vary from device manufacturer to manufacturer. For example, the status bar at the top of the screen may remain during full screen mode display. - This test should be run in Portrait mode for all devices. For devices which support Landscape mode, Steps 1 and 2 should be run a second time in this mode. 	<u>Exceptions</u>
<ul style="list-style-type: none"> - Step 2(a) is omitted where the application does not have this screen. 		
<p>PASS <input type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input checked="" type="checkbox"/></p>		

UI2	UI consistency
<u>Full Description</u>	
The user interface of the application must be consistent throughout the application	
<u>Steps to conduct the test</u> 1. Start the application 2. Use the application in the following areas: f) Splash/Title/Logo/Loading Screen g) Main Menu and all its subsidiary menus h) Help/Instructions Screen(s) i) About screen j) Application Pause Menu and all its subsidiary menus (if present) 3. Observe the consistency of: h) Common series of actions i) Action sequences j) Terms k) Layouts l) Soft key definitions m) Use of vibration n) Sounds	<u>Expected Result</u> -The actions are sequenced in the same way throughout the application -The application uses the same terms for the same things throughout the application -The soft key functionality is the same throughout the application (for example "Back" is always set for the right soft key) -The vibration is used for similar cases -The same sound is not used for different purposes -Two commands with different title must not execute the same action (for example close and exit both close the application) -Two different actions must not be named with a same title (for example exit must be used to exit the application to the devices and not to exit the application to the Main Menu, back could be used instead) -There are no menu orphans -The menu items open the functionality or option which is specified in the menu (for example selecting settings will open settings and not help)
<u>Notes</u> - Observe the consistency of the application through the testing -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application.	<u>Exceptions</u>
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

LO4	Technical text errors	
<p><u>Full Description</u> The text in the application must be clear and readable. The application must be free of technical text display issues such as: Text cut off / Text overlapping.</p>		
<p><u>Steps to conduct the test</u> 1. Launch application in target language 2. Check text appearing in</p> <ul style="list-style-type: none"> f) Splash/Title/Logo/Loading Screen g) Main Menu and all its subsidiary menus h) Help/Instructions Screen(s) i) About screen j) Application Pause Menu and all its subsidiary menus (if present) 	<p><u>Expected Result</u> - All text located in the specified areas is displayed without technical display issues that prevent legibility.</p>	
<p><u>Notes</u> -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application -All text in each target language is displayed without corruption, distortion or other display problems. Examples may include:</p> <ul style="list-style-type: none"> g) Menu item text labels incorrectly aligned with cursor h) Button text label over-running the button area i) Text over-running other bounded text display areas (e.g. speech bubbles, user interface elements etc) j) Text not wrapping at the edge of the screen resulting in words being cut off k) Multiple pieces of text overlapping each other l) Text must not be cut horizontally <p>This text will autodestruct in 5 seconds</p> <p style="text-align: center;">-></p> <p>This text will autodestruct in 5 seconds</p>	<p><u>Exceptions</u> -</p>	
<p>PASS <input checked="" type="checkbox"/> FAIL <input checked="" type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/></p>		

✱

FN2		External incoming communication – voice call	
<u>Full Description</u> The application can handle incoming communications			
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: d) Main menu e) Application in use f) In use pause state (if applicable) 3. Make an incoming call to the device 4. Observe the application behaviour		<u>Expected Result</u> -When the incoming communication enters the device the application goes into pause state, after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at	
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.		<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application)	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

FN3		External incoming communication – SMS	
<u>Full Description</u> The application can handle incoming communications			
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: d) Main menu e) Application in use f) In use pause state (if applicable) 3. Send a SMS to the device 4. Observe the application behaviour		<u>Expected Result</u> -When the incoming communication enters the device the application must at least respect one of the following: a) Go into pause state, after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at b) Give a visual or audible notification -The application must not crash or hang.	
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.		<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application) -Panasonic X400, X60	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

FN8	Main menu requirements	
<u>Full Description</u> The main functionalities of Exit, Help and About are easily available through the main menu		
<u>Steps to conduct the test</u> 1. Start the application 2. Open the main menu 3. Check that Exit, Help and About are available 4. Check that Help displays the help information 5. Check that the help includes: aims of the application, use of keys (for example for games) and the descriptions of the application features. 6. Check that Exit menu item exits the application 7. Check the information on the About and compare it to the JAD and JAR's manifest file information		<u>Expected Result</u> -The main menu includes Exit, About and Help -Exit, About and Help both work as expected, without any error messages -About must include: d) Developer name e) Application name f) The exact version number of the application -It's consistent with the information found in the JAD file and JAR's manifest as follows: d) Developer name: MIDlet-Vendor e) Application name: MIDlet-Name f) The version number: MIDlet-Version
<u>Notes</u>		<u>Exceptions</u> -The About can be included in the Help menu - 'About' and 'Help' are not required for an application that simply launches a browser session.
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>

FN10	The speed of application in use	
<u>Full Description</u> The application works in the device it was targeted for. It is usable on the device. -The speed of the application is acceptable to the purpose of the application and must not alter the user experience by being uncontrollable.		
<u>Steps to conduct the test</u> 1. Use the application 2. Observe how fast the application is to use and if it is, too slow or too fast in its operation 3. If the application behaviour is uncontrollable due to its speed please report such findings		<u>Expected Result</u> -The application is usable on the device -The speed of the application is good enough for the application usage, i.e. the application frame rate must remain adequate and must not compromise the application usage and therefore prevent the user to progress normally
<u>Notes</u> -The developer / publisher is expected to test the entire application. For example play through the entire game. The test house will only conduct representative sample test of the application in different areas if possible, for a 15 minutes period only.		<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>

FN12	Unexpected user behaviour	
<u>Full Description</u> The application must be able to handle unexpected user behaviour, for example erroneous actions and multiple key presses.		
<u>Steps to conduct the test</u> 1. Start the application 2. Press 2-5 different handset buttons simultaneously when: e) The application is loading f) On the main menu g) The application is in use h) The application is in pause state	<u>Expected Result</u> -The application does not crash or freeze, but functions as expected.	
<u>Notes</u> -If you press the handset override button ('end' key, red key, etc. depending on the manufacturer) the application will exit and that is not the purpose of this test -Please note that the Menu key in Series 60 devices will take the application to background -Some applications may not have pause	<u>Exceptions</u> -	
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

SA1	Simple application – Connection exception	
<u>Full Description</u> For applications which merit an exception in test PS1, the connection part of the application must be tested to ensure it connects to the correct service.		
<u>Steps to conduct the test</u> 1. Start the application 2. Select the option to cause a connection 3. Check the connection is successful	<u>Expected Result</u> -The application connects to the required service as appropriate. For demo applications, check the upgrade process works. For applications with in-application advertising, check an advertisement is delivered and displayed. For applications that only launch a browser, check that the browser launches and connects.	
<u>Notes</u>	<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/> FAIL <input checked="" type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

Saran :

- Jeda di coba pada kta Simulasi CG625, perlu perbaikan pada saat incoming call / voice / text. Expires signal suspended / pause
- Halp perlu dipekerja lagi
- Pelang pada awal tampilan perlu di menentu
- Untuk pengembangan lanjut perlu di buat menu setting / pengaturan

Validator



Signature

LEMBAR UJI KELAYAKAN AHLI SOFTWARE

Berilah tanda centang (✓) pada pilihan yang disediakan sesuai dengan penilaian untuk pengujian *mobile application* sebagai hasil dari skripsi: "PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)" yang disusun oleh Kartika Yudha Pratama.

PS1	Permission Statements
<u>Test Description</u> The application should not ask for any permissions.	
<u>Steps to conduct the test</u> 1. Open the JAD and the manifest 2. Observe that there are no permission statements	<u>Expected Result</u> -There are no MIDlet-Permissions statements in the JAD and manifest
<u>Notes</u> Any application that merits an exception should then be subjected to test SA1	<u>Exceptions</u> 1. Application is a demo which connects to a site accessed by a browser to download full version 2. Application connects to a server to provide in-application advertising 3. Application has no other purpose than to launch a browser session
PASS <input checked="" type="checkbox"/>	FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>

ST1	Application stability
<u>Full Description</u> The application must not crash or freeze at any time while running on the device.	
<u>Steps to conduct the test</u> 1. Observe the application behaviour during the testing	<u>Expected Result</u> -The application must not stop the user experience unexpectedly without any user input.
<u>Notes</u> -During any time of the testing observe the application behaviour -The report must indicate if the error can be reproduced or not	<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/>	FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>

AL1		Application Installation	
<u>Full Description</u> The application must install via OTA			
<u>Steps to conduct the test</u> 1. Open the browser application of the device 2. Type the URL of the application JAD file 3. Connect to the typed URL 4. Accept the installation of the application		<u>Expected Result</u> -The application installs to the device -The icon for the application can be found from the device	
<u>Notes</u> If errors occur at installation time, corresponding messages must be reported by the test house in the test report.		<u>Exceptions</u> If the device does not display the icon, then the user must be able to start the application using other means.	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

AL2		Application start up	
<u>Test Description</u> Application must start properly in 25s.			
<u>Steps to conduct the test</u> 1. Find the application icon and select it 2. "Press a button" on the device to launch the application 3. Observe the application launch in the timeline defined 4. The application should have displayed a main menu or interactive menu such as language selection screen where the use of the application can be started 5. Use some of the application features		<u>Expected Result</u> -The application starts in 25s or less, this is the time between steps 2 and 4 -No error messages are displayed -The application appears to function properly	
<u>Notes</u> If launch time errors occur, corresponding messages must be reported by the test house in the test report. This test does not take into consideration the different screens displayed between the "button press" and the display of the main menu of the application. For example branded splash screen.		<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

UI1	Graphic clarity		
<u>Full Description</u>			
All graphics and animations displayed must be readable and clear to the user.			
<p><u>Steps to conduct the test</u></p> <ol style="list-style-type: none"> 1. Launch application in target language 2. Check graphics appearing in <ol style="list-style-type: none"> a) Splash/Title/Logo/Loading Screen b) Main Menu and all its subsidiary menus c) Help/Instructions Screen(s) d) About screen e) Application Pause Menu and all its subsidiary menus (if present) 3. Repeat steps 1 and 2 for each language version of the game 	<p><u>Expected Result</u></p> <ul style="list-style-type: none"> -The application must utilise the full screen size available to them, applicable to the target device. - If device's screen orientation can be changed during application execution (e.g. from portrait to landscape mode) the application must adapt its appearance accordingly, so that the requirement above is still met - There should be no event in the defined areas of the application that would prevent the user from understanding the functionality of the application. <p>For example: a graphical display issue including but not necessarily limiting to the following: overlapping graphics, colour conflict, images truncated and/or displayed incorrectly</p>	<p><u>Notes</u></p> <ul style="list-style-type: none"> -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application. -Definition of full screen may vary from device manufacturer to manufacturer. For example, the status bar at the top of the screen may remain during full screen mode display. - This test should be run in Portrait mode for all devices. For devices which support Landscape mode, Steps 1 and 2 should be run a second time in this mode. 	<p><u>Exceptions</u></p> <ul style="list-style-type: none"> - Step 2(a) is omitted where the application does not have this screen.
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>			

UI12	UI consistency	
<u>Full Description</u>		
The user interface of the application must be consistent throughout the application		
<u>Steps to conduct the test</u>	<u>Expected Result</u>	
<ol style="list-style-type: none"> 1. Start the application 2. Use the application in the following areas: <ol style="list-style-type: none"> a) Splash/Title/Logo/Loading Screen b) Main Menu and all its subsidiary menus c) Help/Instructions Screen(s) d) About screen e) Application Pause Menu and all its subsidiary menus (if present) 3. Observe the consistency of: <ol style="list-style-type: none"> a) Common series of actions b) Action sequences c) Terms d) Layouts e) Soft key definitions f) Use of vibration g) Sounds 	<ul style="list-style-type: none"> -The actions are sequenced in the same way throughout the application -The application uses the same terms for the same things throughout the application -The soft key functionality is the same throughout the application (for example "Back" is always set for the right soft key) -The vibration is used for similar cases -The same sound is not used for different purposes -Two commands with different title must not execute the same action (for example close and exit both close the application) -Two different actions must not be named with a same title (for example exit must be used to exit the application to the devices and not to exit the application to the Main Menu, back could be used instead) -There are no menu orphans -The menu items open the functionality or option which is specified in the menu (for example selecting settings will open settings and not help) 	
<u>Notes</u>	<u>Exceptions</u>	
<ul style="list-style-type: none"> - Observe the consistency of the application through the testing -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application. 		
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

LO4	Technical text errors	
<p><u>Full Description</u> The text in the application must be clear and readable. The application must be free of technical text display issues such as: Text cut off / Text overlapping.</p>		
<p><u>Steps to conduct the test</u></p> <ol style="list-style-type: none"> 1. Launch application in target language 2. Check text appearing in <ol style="list-style-type: none"> a) Splash/Title/Logo/Loading Screen b) Main Menu and all its subsidiary menus c) Help/Instructions Screen(s) d) About screen e) Application Pause Menu and all its subsidiary menus (if present) 	<p><u>Expected Result</u></p> <p>- All text located in the specified areas is displayed without technical display issues that prevent legibility.</p>	
<p><u>Notes</u></p> <p>-The test house will perform the test as specified above.</p> <p>-The developer must ensure that this requirement is fulfilled throughout the application</p> <p>-All text in each target language is displayed without corruption, distortion or other display problems. Examples may include:</p> <ol style="list-style-type: none"> a) Menu item text labels incorrectly aligned with cursor b) Button text label over-running the button area c) Text over-running other bounded text display areas (e.g. speech bubbles, user interface elements etc) d) Text not wrapping at the edge of the screen resulting in words being cut off e) Multiple pieces of text overlapping each other f) Text must not be cut horizontally <p>This text will autodestruct in 5 seconds</p> <p style="text-align: center;">-></p> <p>This text will autodestruct in 5 seconds</p>	<p><u>Exceptions</u></p> <p>-</p>	
<p>PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/></p>		

FN2	External incoming communication – voice call	
<u>Full Description</u> The application can handle incoming communications		
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: a) Main menu b) Application in use c) In use pause state (if applicable) 3. Make an incoming call to the device 4. Observe the application behaviour		<u>Expected Result</u> -When the incoming communication enters the device the application goes into pause state. after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.		<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application)
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

FN3	External incoming communication – SMS	
<u>Full Description</u> The application can handle incoming communications		
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: a) Main menu b) Application in use c) In use pause state (if applicable) 3. Send a SMS to the device 4. Observe the application behaviour		<u>Expected Result</u> -When the incoming communication enters the device the application must at least respect one of the following: a) Go into pause state, after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at b) Give a visual or audible notification -The application must not crash or hang.
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.		<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application) -Panasonic X400, X60
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

FN8	Main menu requirements	
<u>Full Description</u> The main functionalities of Exit, Help and About are easily available through the main menu		
<u>Steps to conduct the test</u>	<ol style="list-style-type: none"> 1. Start the application 2. Open the main menu 3. Check that Exit, Help and About are available 4. Check that Help displays the help information 5. Check that the help includes: aims of the application, use of keys (for example for games) and the descriptions of the application features. 6. Check that Exit menu item exits the application 7. Check the information on the About and compare it to the JAD and JAR's manifest file information 	<u>Expected Result</u> -The main menu includes Exit, About and Help -Exit, About and Help both work as expected, without any error messages -About must include: <ol style="list-style-type: none"> a) Developer name b) Application name c) The exact version number of the application -It's consistent with the information found in the JAD file and JAR's manifest as follows: <ol style="list-style-type: none"> a) Developer name: MIDlet-Vendor b) Application name: MIDlet-Name c) The version number: MIDlet-Version
<u>Notes</u>		<u>Exceptions</u> -The About can be included in the Help menu -'About' and 'Help' are not required for an application that simply launches a browser session.
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

FN10	The speed of application in use	
<u>Full Description</u> The application works in the device it was targeted for. It is usable on the device. -The speed of the application is acceptable to the purpose of the application and must not alter the user experience by being uncontrollable.		
<u>Steps to conduct the test</u>	<ol style="list-style-type: none"> 1. Use the application 2. Observe how fast the application is to use and if it is, too slow or too fast in it's operation 3. If the application behaviour is uncontrollable due to it's speed please report such findings 	<u>Expected Result</u> -The application is usable on the device -The speed of the application is good enough for the application usage, i.e. the application frame rate must remain adequate and must not compromise the application usage and therefore prevent the user to progress normally
<u>Notes</u>	-The developer / publisher is expected to test the entire application. For example play through the entire game. The test house will only conduct representative sample test of the application in different areas if possible, for a 15 minutes period only.	<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

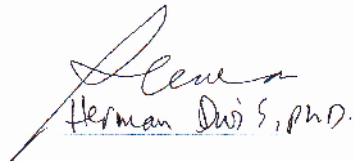
FN12	Unexpected user behaviour	
<u>Full Description</u> The application must be able to handle unexpected user behaviour, for example erroneous actions and multiple key presses.		
<u>Steps to conduct the test</u> 1. Start the application 2. Press 2-5 different handset buttons simultaneously when: a) The application is loading b) On the main menu c) The application is in use d) The application is in pause state		<u>Expected Result</u> -The application does not crash or freeze, but functions as expected.
<u>Notes</u> -If you press the handset override button ('end' key, red key, etc. depending on the manufacturer) the application will exit and that is not the purpose of this test -Please note that the Menu key in Series 60 devices will take the application to background -Some applications may not have pause		<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

SA1	Simple application – Connection exception	
<u>Full Description</u> For applications which merit an exception in test PS1, the connection part of the application must be tested to ensure it connects to the correct service.		
<u>Steps to conduct the test</u> 1. Start the application 2. Select the option to cause a connection 3. Check the connection is successful		<u>Expected Result</u> -The application connects to the required service as appropriate. For demo applications, check the upgrade process works. For applications with in-application advertising, check an advertisement is delivered and displayed. For applications that only launch a browser, check that the browser launches and connects.
<u>Notes</u>		<u>Exceptions</u>
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

Saran :

- perlu ditambah title page yg berisi: judul aplikasi
dan identitas pengembang (jangan pakai foto)

Validator B/11 2012


Herman Dwi S, PhD.

LEMBAR UJI KELAYAKAN AHLI SOFTWARE

Berilah tanda centang (✓) pada pilihan *PASS*, *FAIL*, *PASSED WITH EXCEPTION* yang disediakan sesuai dengan penilaian untuk pengujian *mobile application* sebagai hasil dari skripsi: **“PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)”** yang disusun oleh **Kartika Yudha Pratama**.

PS1	Permission Statements	
<u>Test Description</u> The application should not ask for any permissions.		
<u>Steps to conduct the test</u> 1. Open the JAD and the manifest 2. Observe that there are no permission statements	<u>Expected Result</u> -There are no MIDlet-Permissions statements in the JAD and manifest	
<u>Notes</u> Any application that merits an exception should then be subjected to test SA1	<u>Exceptions</u> 1. Application is a demo which connects to a site accessed by a browser to download full version 2. Application connects to a server to provide in-application advertising 3. Application has no other purpose than to launch a browser session	
PASS <input checked="" type="checkbox"/>	FAIL <input type="checkbox"/>	PASSED WITH EXCEPTION <input type="checkbox"/>

ST1	Application stability	
<u>Full Description</u> The application must not crash or freeze at any time while running on the device.		
<u>Steps to conduct the test</u> 1. Observe the application behaviour during the testing	<u>Expected Result</u> -The application must not stop the user experience unexpectedly without any user input.	
<u>Notes</u> -During any time of the testing observe the application behaviour -The report must indicate if the error can be reproduced or not	<u>Exceptions</u> -	
PASS <input checked="" type="checkbox"/>	FAIL <input type="checkbox"/>	PASSED WITH EXCEPTION <input type="checkbox"/>

ST1		Application stability	
<u>Full Description</u> The application must not crash or freeze at any time while running on the device.			
<u>Steps to conduct the test</u> 1. Observe the application behaviour during the testing		<u>Expected Result</u> -The application must not stop the user experience unexpectedly without any user input.	
<u>Notes</u> -During any time of the testing observe the application behaviour -The report must indicate if the error can be reproduced or not		<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/>	PASSED WITH EXCEPTION <input type="checkbox"/>

AL1		Application Installation	
<u>Test Description</u> The application must install via OTA			
<u>Steps to conduct the test</u> 1. Open the browser application of the device 2. Type the URL of the application JAD file 3. Connect to the typed URL 4. Accept the installation of the application		<u>Expected Result</u> -The application installs to the device -The icon for the application can be found from the device	
<u>Notes</u> If errors occur at installation time, corresponding messages must be reported by the test house in the test report.		<u>Exceptions</u> If the device does not display the icon, then the user must be able to start the application using other means.	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/>	PASSED WITH EXCEPTION <input type="checkbox"/>

AL2		Application start up	
<u>Test Description</u> Application must start properly in 25s.			
<u>Steps to conduct the test</u> 1. Find the application icon and select it 2. "Press a button" on the device to launch the application 3. Observe the application launch In the timeline defined 4. The application should have displayed a main menu or interactive menu such as language selection screen where the use of the application can be started 5. Use some of the application features		<u>Expected Result</u> -The application starts in 25s or less, this is the time between steps 2 and 4 -No error messages are displayed -The application appears to function properly	
<u>Notes</u> If launch time errors occur, corresponding messages must be reported by the test house in the test report. This test does not take into consideration the different screens displayed between the "button press" and the display of the main menu of the application. For example branded splash screen.		<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/>		FAIL <input type="checkbox"/>	PASSED WITH EXCEPTION <input type="checkbox"/>

UI1	Graphic clarity	
<u>Full Description</u>		
All graphics and animations displayed must be readable and clear to the user.		
<u>Steps to conduct the test</u>	<ol style="list-style-type: none"> 1. Launch application in target language 2. Check graphics appearing in <ol style="list-style-type: none"> f) Splash/Title/Logo/Loading Screen g) Main Menu and all its subsidiary menus h) Help/Instructions Screen(s) i) About screen j) Application Pause Menu and all its subsidiary menus (if present) 3. Repeat steps 1 and 2 for each language version of the game 	<u>Expected Result</u>
		<ul style="list-style-type: none"> -The application must utilise the full screen size available to them, applicable to the target device. - If device's screen orientation can be changed during application execution (e.g. from portrait to landscape mode) the application must adapt its appearance accordingly, so that the requirement above is still met - There should be no event in the defined areas of the application that would prevent the user from understanding the functionality of the application. <p>For example: a graphical display issue including but not necessarily limiting to the following: overlapping graphics, colour conflict, images truncated and/or displayed incorrectly</p>
<u>Notes</u>	<ul style="list-style-type: none"> -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application. -Definition of full screen may vary from device manufacturer to manufacturer. For example, the status bar at the top of the screen may remain during full screen mode display. - This test should be run in Portrait mode for all devices. For devices which support Landscape mode, Steps 1 and 2 should be run a second time in this mode. 	<u>Exceptions</u>
		<ul style="list-style-type: none"> - Step 2(a) is omitted where the application does not have this screen.
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

UI2

UI consistency

Full Description

The user interface of the application must be consistent throughout the application

Steps to conduct the test

1. Start the application
2. Use the application in the following areas:
 - f) Splash/Title/Logo/Loading Screen
 - g) Main Menu and all its subsidiary menus
 - h) Help/Instructions Screen(s)
 - i) About screen
 - j) Application Pause Menu and all its subsidiary menus (if present)
3. Observe the consistency of:
 - h) Common series of actions
 - i) Action sequences
 - j) Terms
 - k) Layouts
 - l) Soft key definitions
 - m) Use of vibration
 - n) Sounds

Expected Result

- The actions are sequenced in the same way throughout the application
- The application uses the same terms for the same things throughout the application
- The soft key functionality is the same throughout the application (for example "Back" is always set for the right soft key)
- The vibration is used for similar cases
- The same sound is not used for different purposes
- Two commands with different title must not execute the same action (for example close and exit both close the application)
- Two different actions must not be named with a same title (for example exit must be used to exit the application to the devices and not to exit the application to the Main Menu, back could be used instead)
- There are no menu orphans
- The menu items open the functionality or option which is specified in the menu (for example selecting settings will open settings and not help)

Notes

- Observe the consistency of the application through the testing
- The test house will perform the test as specified above.
- The developer must ensure that this requirement is fulfilled throughout the application.

Exceptions

PASS



FAIL



PASSED WITH EXCEPTION



LO4	Technical text errors	
<p><u>Full Description</u> The text in the application must be clear and readable. The application must be free of technical text display issues such as: Text cut off / Text overlapping.</p>		
<p><u>Steps to conduct the test</u> 1. Launch application in target language 2. Check text appearing in</p> <ul style="list-style-type: none"> f) Splash/Title/Logo/Loading Screen g) Main Menu and all its subsidiary menus h) Help/Instructions Screen(s) i) About screen j) Application Pause Menu and all its subsidiary menus (if present) 	<p><u>Expected Result</u> - All text located in the specified areas is displayed without technical display issues that prevent legibility.</p>	
<p><u>Notes</u> -The test house will perform the test as specified above. -The developer must ensure that this requirement is fulfilled throughout the application -All text in each target language is displayed without corruption, distortion or other display problems. Examples may include:</p> <ul style="list-style-type: none"> g) Menu item text labels incorrectly aligned with cursor h) Button text label over-running the button area i) Text over-running other bounded text display areas (e.g. speech bubbles, user interface elements etc) j) Text not wrapping at the edge of the screen resulting in words being cut off k) Multiple pieces of text overlapping each other l) Text must not be cut horizontally <p>This text will autodestruct in 5 seconds</p> <p style="text-align: center;">-></p> <p>This text will autodestruct in 5 seconds</p>	<p><u>Exceptions</u> -</p>	
<p>PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/></p>		

FN2	External incoming communication – voice call	
<u>Full Description</u> The application can handle incoming communications		
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: d) Main menu e) Application in use f) In use pause state (if applicable) 3. Make an incoming call to the device 4. Observe the application behaviour	<u>Expected Result</u> -When the incoming communication enters the device the application goes into pause state, after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at	
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.	<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application)	
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

FN3	External incoming communication – SMS	
<u>Full Description</u> The application can handle incoming communications		
<u>Steps to conduct the test</u> 1. Start the application 2. In the following locations of the application: d) Main menu e) Application in use f) In use pause state (if applicable) 3. Send a SMS to the device 4. Observe the application behaviour	<u>Expected Result</u> -When the incoming communication enters the device the application must at least respect one of the following: a) Go into pause state, after the user exits the communication, the application presents the user with a continue option or is continued automatically from the point it was suspended at b) Give a visual or audible notification -The application must not crash or hang.	
<u>Notes</u> -The developer is encouraged to use the available APIs the pause and continue methods. -S60 devices may close the Java application if there is not enough RAM.	<u>Exceptions</u> -Not required for applications where the immediate user intervention is not needed (for example timer application) -Panasonic X400, X60	
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

FN8 Main menu requirements	
<u>Full Description</u> The main functionalities of Exit, Help and About are easily available through the main menu	
<u>Steps to conduct the test</u> 1. Start the application 2. Open the main menu 3. Check that Exit, Help and About are available 4. Check that Help displays the help information 5. Check that the help includes: aims of the application, use of keys (for example for games) and the descriptions of the application features. 6. Check that Exit menu item exits the application 7. Check the information on the About and compare it to the JAD and JAR's manifest file information	<u>Expected Result</u> -The main menu includes Exit, About and Help -Exit, About and Help both work as expected, without any error messages -About must include: d) Developer name e) Application name f) The exact version number of the application -It's consistent with the information found in the JAD file and JAR's manifest as follows: d) Developer name: MIDlet-Vendor e) Application name: MIDlet-Name f) The version number: MIDlet-Version
<u>Notes</u>	<u>Exceptions</u> -The About can be included in the Help menu - 'About' and 'Help' are not required for an application that simply launches a browser session.
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

FN10 The speed of application in use	
<u>Full Description</u> The application works in the device it was targeted for. It is usable on the device. -The speed of the application is acceptable to the purpose of the application and must not alter the user experience by being uncontrollable.	
<u>Steps to conduct the test</u> 1. Use the application 2. Observe how fast the application is to use and if it is, too slow or too fast in it's operation 3. If the application behaviour is incontrollable due to it's speed please report such findings	<u>Expected Result</u> -The application is usable on the device -The speed of the application is good enough for the application usage, i.e. the application frame rate must remain adequate and must not compromise the application usage and therefore prevent the user to progress normally
<u>Notes</u> -The developer / publisher is expected to test the entire application. For example play through the entire game. The test house will only conduct representative sample test of the application in different areas if possible, for a 15 minutes period only.	<u>Exceptions</u> -
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>	

FN12	Unexpected user behaviour	
<u>Full Description</u> The application must be able to handle unexpected user behaviour, for example erroneous actions and multiple key presses.		
<u>Steps to conduct the test</u> 1. Start the application 2. Press 2-5 different handset buttons simultaneously when: e) The application is loading f) On the main menu g) The application is in use h) The application is in pause state	<u>Expected Result</u> -The application does not crash or freeze, but functions as expected.	
<u>Notes</u> -If you press the handset override button ('end' key, red key, etc. depending on the manufacturer) the application will exit and that is not the purpose of this test -Please note that the Menu key in Series 60 devices will take the application to background -Some applications may not have pause	<u>Exceptions</u> -	
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

SA1	Simple application – Connection exception	
<u>Full Description</u> For applications which merit an exception in test PS1, the connection part of the application must be tested to ensure it connects to the correct service.		
<u>Steps to conduct the test</u> 1. Start the application 2. Select the option to cause a connection 3. Check the connection is successful	<u>Expected Result</u> -The application connects to the required service as appropriate. For demo applications, check the upgrade process works. For applications with in-application advertising, check an advertisement is delivered and displayed. For applications that only launch a browser, check that the browser launches and connects.	
<u>Notes</u>	<u>Exceptions</u>	
PASS <input checked="" type="checkbox"/> FAIL <input type="checkbox"/> PASSED WITH EXCEPTION <input type="checkbox"/>		

Saran :

Warna background file dan icon sebaiknya dapat diganti.
ganti supaya dalam dipalmi pd tlp yg smat dapat menarik.
Tetapi dalam dipalmi pd tlp yg hanya mode teks saja sudah
cukup.

Validator



Totok Sukardiyono, M.T.

Lampiran 9. Penilaian Validasi Ahli Materi

INSTRUMEN UNTUK AHLI MATERI

PENGEMBANGAN KAMUS BAHASA JAWA-BAHASA INDONESIA MENGUNAKAN JAVA 2 MICRO EDITION (J2ME)

A. CHECK LIST

Petunjuk Pengisian:

1. Mohon dengan hormat bantuan dan ketersediaan Ibu untuk menjawab seluruh pertanyaan yang ada.
2. Berilah tanda *check* (✓) pada kolom sesuai dengan pendapat Ibu berdasarkan keadaan yang sebenarnya.
3. Jawaban diberikan pada kolom yang sudah disediakan, dengan penilaian:
STL = Sangat Tidak Layak
TL = Tidak Layak
CL = Cukup Layak
L = Layak
SL = Sangat Layak
4. Adapun skala penilaian tersebut memiliki nilai sebagai berikut: sangat tidak layak (1), tidak layak (2), cukup layak (3), layak (4), sangat layak (5).

No.	Indikator	Penilaian				
		STL	TL	CL	L	SL
1	Kejelasan petunjuk penggunaan			✓		
2	Kejelasan isi materi					✓
3	Ketertelusuran kata yang dicari				✓	
4	Kebenaran penulisan kata				✓	
5	Kebenaran arti pada setiap kata				✓	
6	Kebenaran pelafalan dalam bahasa Jawa				✓	
7	Kejelasan pelafalan dalam bahasa Jawa				✓	

B. Saran

1. Petunjuk penggunaan perlu sistematis agar lebih mudah digunakan
2. Berikan tambahan menu "referensi" terutama sumber ...
... pustaka kamus yang digunakan
3. Kamus perlu ditambah nyukannya

Validator



Nurhidayati, M. Hum

NIP. 19780610 200112 2 002

Lampiran 10. Perhitungan Validasi Ahli Materi

Skor Prosentase	Interpretasi
0% - 24,99%	Tidak Layak
25% - 49,99%	Kurang Layak
50% - 74,99%	Layak
75% - 100%	Sangat Layak

No	Ahli Materi	Aspek						
		Substansi Materi						
		1	2	3	4	5	6	7
1	Nurhidayati, M. Hum.	3	5	4	4	4	4	4
Skor Pengumpulan Data		28						

$$P = \frac{\textit{Skor hasil pengumpulan data}}{\textit{Skor ideal}} \times 100\%$$

$$= \frac{28}{35} \times 100\%$$

$$= 80\% \quad (\text{Sangat Layak})$$

Lampiran 11. Angket Penilaian Pengguna
ANGKET UJI KELAYAKAN USER

Nama :

Prodi :

Merek dan Tipe Ponsel :

Berilah tanda centang () pada pilihan yang sesuai dengan keyakinan masing-masing untuk penelitian skripsi “Pengembangan Kamus Bahasa Jawa Menggunakan Java 2 Micro Edition (J2ME)” yang disusun oleh Kartika Yudha Pratama.

Keterangan :

1 = Sangat Tidak Layak

4 = Layak

2 = Tidak Layak

5 = Sangat Layak

3 = Cukup Layak

No.	Butir Penilaian	1	2	3	4	5
1	Saya dapat menggunakan antarmuka aplikasi kamus bahasa Jawa-bahasa Indonesia ini dengan mudah					
2	Saya dapat menggunakan fitur pelafalan kosa kata dalam bahasa Jawa pada menu Jawa-Indonesia dengan mudah					
3	Saya dapat menemukan kata yang dicari dengan mudah.					
4	Saya dapat menerjemahkan kata baik dari bahasa Jawa ke bahasa Indonesia maupun dari bahasa Indonesia ke bahasa Jawa dengan mudah.					
5	Saya dapat mempelajari penggunaan aplikasi kamus bahasa Jawa-bahasa Indonesia ini dengan mudah.					
6	Saya dapat menghemat waktu untuk menemukan kata yang dicari dengan menggunakan aplikasi kamus bahasa Jawa-bahasa Indonesia ini.					
7	Saya dapat menghemat waktu dalam mengartikan kata dengan menggunakan aplikasi kamus bahasa Jawa-bahasa Indonesia ini.					

No	Butir Penilaian	1	2	3	4	5
8	Aplikasi kamus bahasa Jawa-bahasa Indonesia ini tidak <i>hang</i> /rusak pada saat digunakan.					
9	Ketika terjadi eror, aplikasi kamus bahasa Jawa-bahasa Indonesia ini sudah memberitahukan cara memperbaikinya dengan jelas.					
10	Ketika saya membuat kesalahan, saya dapat dengan mudah dan cepat menjalankan kembali aplikasi kamus bahasa Jawa-bahasa Indonesia ini.					
11	Kamus bahasa Jawa-bahasa Indonesia ini menyediakan menu petunjuk yang membantu pengguna baru (pengguna awam) dalam menggunakannya.					
12	Petunjuk penggunaan aplikasi kamus bahasa Jawa-bahasa Indonesia pada menu Bantuan ini sudah jelas.					
13	Menu-menu yang ada pada aplikasi kamus bahasa Jawa-bahasa Indonesia ini mudah dimengerti.					
14	Arti kata yang ditampilkan sudah jelas.					
15	Pelafalan kosakata bahasa Jawa pada menu Jawa-Indonesia sudah jelas.					
16	Menu-menu yang ada pada aplikasi kamus bahasa Jawa-bahasa Indonesia ini sudah tersusun dengan jelas.					
17	Aplikasi kamus bahasa Jawa-bahasa Indonesia ini memiliki semua fungsi dan kemampuan sesuai dengan yang saya harapkan.					
18	Antarmuka aplikasi kamus bahasa Jawa-bahasa Indonesia ini sudah cukup menarik.					
19	Secara keseluruhan, saya dapat menggunakan aplikasi kamus bahasa Jawa-bahasa Indonesia ini dengan mudah sehingga membantu dalam menerjemahkan kata.					

Saran :

.....

.....

.....

.....

Responden

“PETUNJUK PENGGUNAAN APLIKASI JAWINDO *POCKET DICT*”

Aplikasi ini ditujukan untuk membantu pengguna dalam menerjemahkan kata dari Bahasa Jawa ke Bahasa Indonesia maupun dari Bahasa Indonesia ke Bahasa Jawa. Kosakata dalam kamus ini dibatasi pada kata-kata yang biasa digunakan dalam kehidupan sehari-hari bersumber pada majalah bahasa Jawa "Djaka Lodang" dan buku pelajaran bahasa Jawa.

1. Bahasa Jawa – Bahasa Indonesia

Menu ini digunakan untuk mengartikan kata bahasa Jawa yang dicari ke dalam bahasa Indonesia.

Masukkan kata yang ingin dicari pada kotak masukan yang telah disediakan, tidak perlu mengetikkan secara lengkap, ketikkan beberapa huruf kemudian pilih menu "Cari" maka aplikasi akan menampilkan kemungkinan kata yang dicari, selanjutnya pengguna tinggal memilih kata yang ingin diartikan dengan memilih menu pada tombol sebelah kanan ponsel dan pilih "Artinya" atau bisa juga dengan menekan tombol tengah ponsel (jika tersedia). Pengguna dapat mendengarkan pelafalan dari kata dalam bahasa Jawa yang dimaksud dengan memilih menu "Pelafalan", namun pastikan terlebih dahulu ponsel anda dapat terhubung dengan Internet, Pelafalan kosakata dalam menu "Pelafalan" masih terbatas.

2. Bahasa Indonesia – Bahasa Jawa

Menu ini digunakan untuk mengartikan kata bahasa Indonesia yang dicari ke dalam bahasa Jawa.

Masukkan kata yang ingin dicari pada kotak masukan yang telah disediakan, tidak perlu mengetikkan secara lengkap, ketikkan beberapa huruf kemudian pilih menu "Cari" maka aplikasi akan menampilkan kemungkinan kata yang dicari, selanjutnya pengguna tinggal memilih kata yang ingin diartikan dengan memilih menu pada tombol sebelah kanan ponsel dan pilih "Artinya" atau bisa juga dengan menekan tombol tengah ponsel (jika tersedia). Menu "Pelafalan" tidak tersedia pada menu bahasa Jawa – bahasa Indonesia.

3. Bantuan

Menu ini akan menampilkan bantuan dari penggunaan aplikasi kamus bahasa Jawa – bahasa Indonesia.

4. Tentang Aplikasi

Menu ini menampilkan informasi dari aplikasi kamus bahasa Jawa – bahasa Indonesia.

5. Keluar

Menu ini untuk keluar dari aplikasi kamus bahasa Jawa – bahasa Indonesia.

Lampiran 12. Perhitungan Penilaian Pengguna

No	Nama	System Usefulness								Information Quality							Interface Quality			Over All	Total
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
1	Nur Cahya	5	5	4	4	4	3	2	3	4	3	5	4	5	5	5	4	3	4	4	76
2	Candra	5	3	4	4	4	5	5	4	3	4	4	3	5	3	4	5	4	5	4	78
3	Wahyu Sulisty N.	4	3	4	3	4	4	4	4	4	3	4	3	4	3	4	4	3	4	3	68
3	Rosyid S	4	4	3	4	4	4	5	4	3	3	5	4	5	4	4	5	4	4	5	78
5	Resti	3	3	4	3	3	4	4	4	3	4	4	4	4	4	3	4	4	4	3	69
6	Indah Setyo Putri	4	4	3	4	4	4	2	3	3	2	4	3	4	3	5	4	4	3	4	67
7	Dewi Murniawati	4	4	4	4	3	5	5	4	5	3	3	4	5	5	5	5	4	4	4	80
8	Riana Putri	3	3	3	3	4	4	4	4	4	3	4	3	4	3	5	4	4	3	4	69
9	Ayu Siti E	4	5	3	3	3	4	4	3	3	2	3	4	4	3	4	3	3	3	4	65
10	Rosiani	5	5	3	3	3	3	3	4	3	1	3	3	4	4	4	4	3	4	4	66
11	Ari Purnomo Aji	3	5	4	5	4	4	4	3	3	3	4	4	4	5	5	4	3	5	4	76
12	Adi Setiawan	3	3	4	5	2	5	5	4	4	4	5	3	4	3	4	4	3	5	5	75
13	Agustina Dwi W	3	4	3	3	3	4	4	3	4	3	3	4	4	4	3	4	4	4	4	68
14	Anggi	4	4	3	3	4	5	5	5	2	3	4	3	4	4	3	4	3	4	4	71
15	Arief Budi H	4	5	4	4	5	5	4	4	4	3	5	4	5	5	5	4	3	5	5	83
16	Ahmad F	4	5	3	4	3	4	4	3	3	3	4	4	4	3	4	4	3	3	3	68
17	Aris Setyo Wibowo	5	5	3	3	3	4	5	4	4	1	3	4	5	3	4	4	3	4	4	71
18	Mukhlas Fajar P	5	5	5	5	4	4	5	4	4	3	5	4	5	5	5	5	4	5	5	87
19	Novianto	3	4	3	3	2	4	4	3	3	3	3	3	4	3	3	4	3	4	3	62
20	Dwi Sudarsono	5	5	4	4	4	4	5	4	4	3	5	4	5	4	5	5	4	5	4	83
21	Anggi	5	5	4	3	3	5	5	3	5	5	5	5	5	3	5	5	5	3	3	82
22	Tegar Pratama P	3	4	4	4	4	4	5	3	2	2	4	3	3	3	5	3	4	3	5	68
23	Anindita Radita N.	5	5	4	4	4	4	5	4	4	3	5	4	5	5	5	3	4	5	5	83
24	Nurul Farida	3	1	3	3	3	2	1	3	2	1	3	2	4	3	4	3	3	3	3	50
25	Fery Ferial N.E.C.	5	5	4	4	4	5	5	4	5	5	4	5	5	4	5	5	5	5	5	89
26	Pram S.A.	3	4	3	3	3	3	3	3	3	2	3	3	4	3	3	4	3	3	4	60
27	Omar M.A.A.	5	5	5	5	4	5	2	3	5	2	5	3	4	5	5	5	4	4	5	81
28	Rini S.	2	1	2	2	2	4	4	3	2	3	3	2	3	3	2	3	1	3	3	48
29	Taufik Hendarto	5	5	4	4	4	5	5	4	5	3	5	3	5	4	5	5	4	4	5	84
30	Nike	5	5	4	3	3	4	5	3	4	2	5	4	5	3	5	4	4	4	4	76
TOTAL		920								790							349			122	2181

Prosentase Aspek *System Usefulness*

$$\begin{aligned} P &= \frac{920}{5 \times 30 \times 8} \times 100 \% \\ &= \frac{920}{1200} \times 100 \% \\ &= \mathbf{76,67\%} \quad (\mathbf{Sangat Layak}) \end{aligned}$$

Prosentase Aspek *Informatin Quality*

$$\begin{aligned} P &= \frac{790}{5 \times 30 \times 7} \times 100 \% \\ &= \frac{790}{1050} \times 100 \% \\ &= \mathbf{75,24\%} \quad (\mathbf{Sangat Layak}) \end{aligned}$$

Prosentase Aspek *Interface Quality*

$$\begin{aligned} P &= \frac{349}{5 \times 30 \times 3} \times 100 \% \\ &= \frac{349}{450} \times 100 \% \\ &= \mathbf{77,56\%} \quad (\mathbf{Sangat Layak}) \end{aligned}$$

Prosentase Aspek *Over All* (keseluruhan kriteria *Usability*)

$$\begin{aligned} P &= \frac{1313+1164+479+168}{5 \times 30 \times 19} \times 100 \% \\ &= \frac{2181}{2850} \times 100 \% \\ &= \mathbf{76,53\%} \quad (\mathbf{Sangat Layak}) \end{aligned}$$

Lampiran 13. Prodi dan Ponsel yang Digunakan Pengguna

No.	Nama	Prodi	Merek dan Tipe Ponsel
1	Nur Cahya	PT. Elektronika	Spony Ericsson W880i
2	Candra	PT. Informatika	Sony 5630
3	Wahyu Sulisty N.	PT. Elektronika	Blackberry 8530
3	Rosyid S	PT. Informatika	Blackberry Curve 9330
5	Resti	PT. Elektronika	Nokia C3
6	Indah Setyo Putri	PT. Elektronika	Nokia 6120c
7	Dewi Murniawati	PT. Elektronika	Nokia E71
8	Riana Putri	PT. Elektronika	Blackberry 8520
9	Ayu Siti E	PT. Elektronika	Blackberry 8520
10	Rosiani	PT. Informatika	Nokia N70
11	Ari Purnomo Aji	PT. Informatika	Nokia 7610
12	Adi Setiawan	PT. Informatika	Samsung Galaxy Ace
13	Agustina Dwi W	PT. Informatika	Sony S312
14	Anggita	PT. Informatika	Nokia 5630
15	Arief Budi H	PT. Elektronika	Nokia 6120c
16	Ahmad F	PT. Informatika	Nokia 3230
17	Aris Setyo Wibowo	PT. Informatika	Nokia N70
18	Mukhlas Fajar P	PT. Elektronika	Nokia N73
19	Novianto	PT. Informatika	Nokia 2600c
20	Dwi Sudarsono	PT. Informatika	Nokia 2600c
21	Anggi	PT. Informatika	Nokia Asha 305
22	Tegar Pratama P	PT. Informatika	Nokia 5630
23	Anindita Radita N.	PT. Informatika	Nokia N70
24	Nurul Farida	PT. Elektronika	Samsung Galaxy Ace II
25	Fery Ferial N.E.C.	PT. Elektronika	Sony Ericsson K550i
26	Pram S.A.	PT. Informatika	Nokia N73
27	Omar M.A.A.	PT. Informatika	Nokia 5630
28	Rini S.	PT. Elektronika	Nokia E71
29	Taufik Hendarto	PT. Elektronika	Nokia E71
30	Nike	PT. Informatika	Nokia E71

Lampiran 14. Saran dari Pengguna

No.	Nama	Saran
1	Nur Cahya	-
2	Candra	-
3	Wahyu Sulisty N.	-
4	Rosyid S	-
5	Resti	-
6	Indah Setyo Putri	Aplikasi disebarakan ke mahasiswa luar Jawa yang kurang faham dengan bahasa Jawa, agar lebih membantu mereka dalam berkomunikasi menggunakan bahasa Jawa.
7	Dewi Murniawati	Kamus ini sangat membantu untuk orang yang bukan asli orang Jawa tapi baru belajar bahasa Jawa dilengkapi bukan untuk Jawa saja.
8	Riana Putri	Sangat cocok digunakan untuk mahasiswa yang bukan asli jogja sehingga membantu mereka dalam berkomunikasi di kawasan jogja.
9	Ayu Siti E	-
10	Rosiani	Lengkapi kata-kata yang belum ada dalam aplikasi.
11	Ari Purnomo Aji	Akan lebih bagus lagi dikembangkan ke semua platform.
12	Adi Setiawan	-
13	Agustina Dwi W	Masih ada kata-kata Indonesia yang belum ada di kamus, akan lebih baik lagi jika database ditambah.
14	Anggita	-
15	Arief Budi H	Cukup mudah untuk menggunakan, perbanyak kosakata, bahasa Jawa ada yang halus dan kasar, jadi bias ditambah pada pilihan menu
16	Ahmad F	-
17	Aris Setyo Wibowo	Mungkin lebih banyak kosakata lagi, karena dengan lebih banyak kosakata maka program tersebut akan lebih dapat membantu lagi.
18	Mukhlas Fajar P	Aplikasi yang membantu masyarakat luar Jogja yang tidak/kurang paham bahasa Jawa, namun kata dapat dilengkapi lagi.
19	Novianto	Pilihan menu lebih diperjelas, kosakata lebih diperbanyak, keterangan tambahan lebih diperjelas.
20	Dwi Sudarsono	Dibuat multiplatform.
21	Anggi	Diperjelas pada kosakata Jawanya, mana yang termasuk ngoko dan karma.
22	Tegar Pratama P	Jumlah kata perlu ditambah.
23	Anindita Radita N.	-
24	Nurul Farida	Database kosakat diperbanyak.
25	Fery Ferial N.E.C.	-
26	Pram S.A.	Antarmuka aplikasi sebenarnya sudah baik tapi masih kurang menyenangkan.
27	Omar M.A.A.	-
28	Rini S.	-
29	Taufik Hendarto	-
30	Nike	Beberapa kata ada yang belum tersedia di aplikasi, lebih menarik jika ditambah tema di aplikasi

Lampiran 15. Spesifikasi *Nokia 5630 XpressMusic*

General	: 2G Network	GSM 850/900/1800/1900
	: 3G Network	HSDPA 2100/900
	: SIM	Mini-SIM
	: Announced	2009, February
	: Status	Available, Released 2009, June
Body	: Dimension	112 x 46 x 12 mm (4.41 x 1.81 x 0.47 in)
	: Weight	83 g (2.93 oz)
Display	: Type	TFT, 16M colors
	: Size	240 x 320 pixels, 2.2 inches
Sound	: Alert types	Vibration; Downloadable polyphonic, MP3 ringtones
	: Loudspeaker	Yes
	: 3.5 mm Jack	Yes
Memory	: Card slot	Dedicated music keys microSD, up to 16 GB, 4 GB included
	: internal	60 MB storage, 128 MB RAM
Data	: GPRS	Class 32
	: EDGE	Class 32
	: Speed	HSDPA, 10.2 Mbps; HSUPA, 2 Mbps
	: WLAN	Wi-Fi 802.11 b/g, UPnP technology
	: Bluetooth	Yes, v2.0 with A2DP
Camera	: USB	Yes, microUSB v2.0
	: Primary	3.15 MP, 2048 x 1536 pixels, enhanced fixed focus, LED flash, check quality
	: Video	Yes
Features	: Secondary	VGA videocall camera
	: OS	Symbian OS, S60 rel. 3.2
	: CPU	600 MHz ARM 11
	: Messaging	SMS, MMS, Email
	: Browser	WAP 2.0/xHTML
	: Radio	Stereo FM radio with RDS
	: GPS	No
: Java	Yes, MIDP 2.1	

(sumber: www.gsmarena.com)

Lampiran 16. Surat Pengangkatan Pembimbing TA Skripsi

**KEPUTUSAN DEKAN
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 249/ELK/Q-IXI/2011
TENTANG
PENGANGKATAN PEMBIMBING TUGAS AKHIR SKRIPSI
BAGI MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

Menimbang : 1. Bahwa sehubungan dengan telah dipenuhi syarat untuk penulisan Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta, perlu diangkat pembimbing.
2. Bahwa untuk keperluan dimaksud perlu ditetapkan dengan Keputusan Dekan.

Mengingat : 1. Undang-undang Nomor 20 tahun 2003.
2. Peraturan Pemerintah RI Nomor 60 tahun 1999.
3. Keputusan Presiden RI: a. Nomor 95 tahun 1999; b. 305/M tahun 1999.
4. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 274/O/1999.
5. Keputusan Mendiknas RI Nomor 003/O/2001.
6. Keputusan Rektor UNY Nomor : 1160/UN24/KP/2011

MEMUTUSKAN

Menetapkan

Pertama : Mengangkat Pembimbing Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta sebagai berikut :


Nama Pembimbing : Adi Dewanto, M.Kom
Bagi mahasiswa :
Nama/No.Mahasiswa : Kartika Yudha Pratama / 08520244042
Jurusan/ Prodi : Pendidikan Teknik Elektronika / Pendidikan Teknik Informatika

Kedua : Dosen pembimbing disertai tugas membimbing penulisan Tugas Akhir Skripsi sesuai dengan Pedoman Tugas Akhir Skripsi.

Ketiga : Keputusan ini berlaku sejak ditetapkan

Keempat : Segala sesuatu akan diubah dan dibetulkan sebagaimana mestinya apabila di kemudian hari ternyata terdapat kekeliruan dalam Keputusan ini.

Ditetapkan : di Yogyakarta
Pada tanggal : 17 November 2011
Dekan



Dr. Moch. Bruni Triyoro
NIP. 195602161986031003

Tembusan Yth :

1. Pembantu Dekan I, II, III FT UNY
2. Ketua Jurusan Pendidikan Teknik Elektronika
3. Ka Bag. Tata Usaha FT UNY
4. Yang bersangkutan


Lampiran 17. Surat Permohonan Ijin Penelitian dari Fakultas

12/09/2012 14:16:49



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS NEGERI YOGYAKARTA
FAKULTAS TEKNIK

Alamat : Kampus Karangmalang, Yogyakarta, 55281
Telp. (0274) 586168 psw. 276, 289, 292 (0274) 586734 Fax. (0274) 586734
website : <http://ft.uny.ac.id> e-mail: ft@uny.ac.id - teknik@uny.ac.id



Definisi No. : 26C 0052

Nomor : 2998/UN34.15/PL/2012 19 September 2012
Lamp. : 1 (satu) bendel
Hal : Permohonan Ijin Penelitian

Yth,

1. Gubernur Provinsi DIY c.q. Ka. Biro Administrasi Pembangunan Setda Provinsi DIY
2. Bupati Sleman c.q. Kepala Bappeda Kabupaten Sleman
3. Kepala Dinas Pendidikan, Pemuda dan Olahraga Propinsi DIY
4. Kepala Dinas Pendidikan Kabupaten Sleman
5. KEPALA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

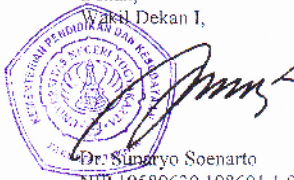
Dalam rangka pelaksanaan Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul **"PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)"**, bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:

No.	Nama	NIM	Jurusan/Prodi	Lokasi Penelitian
1	Kartika Yudha P	08520244042	Pend. Teknik Informatika - S1	FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

Dosen Pembimbing/Dosen Pengampu : Adi Dewanto, M.Kom.
NIP : 19721228 200501 1 001

Adapun pelaksanaan penelitian dilakukan mulai tanggal 19 September 2012 sampai dengan selesai.
Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.

Dekan,
Wakil Dekan I,



Dr. Sunaryo Soenarto
NIP 19580630 198601 1 001

Tembusan:
Ketua Jurusan

08520244042 No 966

Lampiran 18. Surat Permohonan Ijin Penelitian dari SETDA DIY



PEMERINTAH PROVINSI DAERAH ISTIMEWA YOGYAKARTA SEKRETARIAT DAERAH

Kompleks Kepatihan, Danurejan, Telepon (0274) 562811 - 562814 (Hunting)
YOGYAKARTA 55213

SURAT KETERANGAN / IJIN

070/7916/N/6/2012

Membaca Surat : Dekan Fak. Teknik UNY Nomor : 2593/UN.34.5/PL/2012
Tanggal : 19 September 2012 Perihal : Permohonan Ijin Penelitian

- Mengingat :
1. Peraturan Pemerintah Nomor 41 Tahun 2006, tentang Perizinan bagi Perguruan Tinggi Asing, Lembaga Penelitian dan Pengembangan Asing, Badan Usaha Asing dan Orang Asing dalam melakukan Kegiatan Penelitian dan Pengembangan di Indonesia;
 2. Peraturan Menteri Dalam Negeri Nomor 33 Tahun 2007, tentang Pedoman penyelenggaraan Penelitian dan Pengembangan di Lingkungan Departemen Dalam Negeri dan Pemerintah Daerah;
 3. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 37 Tahun 2008, tentang Rincian Tugas dan Fungsi Satuan Organisasi di Lingkungan Sekretariat Daerah dan Sekretariat Dewan Perwakilan Rakyat Daerah;
 4. Peraturan Gubernur Daerah Istimewa Yogyakarta Nomor 18 Tahun 2009 tentang Pedoman Pelayanan Perizinan Rekomendasi Pelaksanaan Survei, Penelitian, Pendataan, Pengembangan, Pengkajian, dan Studi Lapangan di Daerah Istimewa Yogyakarta.

DIJINKAN untuk melakukan kegiatan survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan kepada:

Nama : KARTIKA YULCHA PRATAMA NIP/NIM : 08520244042
Alamat : Karangmalang, Yogyakarta
Judul : PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)
Lokasi : - Kota/Kab. KOTA YOGYAKARTA
Waktu : 25 September 2012 s/c 25 Desember 2012

Dengan Ketentuan

1. Menyerahkan surat keterangan/ijin survei/penelitian/pendataan/pengembangan/pengkajian/studi lapangan *) dari Pemerintah Provinsi DIY kepada Bupati/Walikota melalui institusi yang berwenang mengeluarkan ijin dimaksud;
2. Menyerahkan soft copy rasi penelitiannya baik kepada Gubernur Daerah Istimewa Yogyakarta melalui Biro Administrasi Pembangunan Setda Provinsi DIY dalam compact disk (CD) maupun mengunggah (upload) melalui website adbang.jogjaprov.go.id dan menunjukkan cetakan asli yang sudah disahkan dan dibubuhi cap institusi;
3. Ijin ini hanya dipergunakan untuk keperluan ilmiah, dan pemegang ijin wajib mentaati ketentuan yang berlaku di lokasi kegiatan;
4. Ijin penelitian dapat diperpanjang maksimal 2 (dua) kali dengan menunjukkan surat ini kembali sebelum berakhir waktunya setelah mengajukan perpanjangan melalui website adbang.jogjaprov.go.id;
5. Ijin yang diberikan dapat dicabut sewaktu-waktu apabila pemegang ijin ini tidak memenuhi ketentuan yang berlaku.

Dikeluarkan di Yogyakarta

Pada tanggal 25 September 2012

A.n Sekretaris Daerah

Asisten Perencanaan dan Pembangunan

Kepala Biro Administrasi Pembangunan


Joko Wuryentoro, M.Si
NIP. 12580108 198502 011

Tembusan :

1. Yth. Gubernur Daerah Istimewa Yogyakarta (sebagai laporan);
2. Rektor Universitas Negeri Yogyakarta
3. Dekan Fak. Teknik UNY
4. Yang Berangkutan

Lampiran 19. Surat Permohonan Ijin Penelitian dari BAPPEDA Sleman



PEMERINTAH KABUPATEN SLEMAN
BADAN PERENCANAAN PEMBANGUNAN DAERAH

Jalan Parasma Nomor 1 Beran, Tridadi, Sleman, Yogyakarta 55511
Telepon (0274) 868800, Faksimile (0274) 868800
Website: slemankab.go.id, E-mail: bappeda@slemankab.go.id

SURAT IZIN

Nomor : 070 / Bappeda / 2670 / 2012

**TENTANG
PENELITIAN**

KEPALA BADAN PERENCANAAN PEMBANGUNAN DAERAH

Dasar : Keputusan Bupati Sleman Nomor : 55/Kep.KDH/A/2005 tentang Izin Kuliah Kerja Nyata, Praktek Kerja Lapangan, dan Penelitian.
Menunjuk : Surat dari Dekan, Wakil Dekan I, Fak. Teknik UNY
Nomor : 2998/UN34.15/PL/2012
Hal : Izin Penelitian
Tanggal : 19 September 2012

MENGIZINKAN :

Kepada :
Nama : KARTIKA YUDHA P
No.Mhs/NIM/NIP/NIK : 08520244042
Program/Tingkat : S1
Instansi/Perguruan Tinggi : Universitas Negeri Yogyakarta
Alamat instansi/Perguruan Tinggi : Kampus Karangmalang Yogyakarta
Alamat Rumah : Perum. Madukoro Asri E-3 Rt 3 Rw 06 Kelurahan Bumiroso Wcnosobo
No. Telp / HP : 085 227052004
Untuk : Mengadakan Penelitian / Pra Survey / Uji Validitas / PKL dengan judul **PENGEMBANGAN KAMUS BAHASA JAWA MENGGUNAKAN JAVA 2 MICRO EDITION (J2ME)**
Lokasi : Fak. Teknik UNY
Waktu : Selama 3 bulan mulai tanggal: 27 September 2012 s/d 27 Desember 2012

Dengan ketentuan sebagai berikut :

1. *Wajib melapor diri kepada Pejabat Pemerintah setempat (Camat/ Kepala Desa) atau Kepala Instansi untuk mendapat petunjuk seperlunya.*
2. *Wajib menjaga tata tertib dan menaati ketentuan-ketentuan setempat yang berlaku.*
3. *Izin tidak disalahgunakan untuk kepentingan-kepentingan di luar yang direkomendasikan.*
4. *Wajib menyampaikan laporan hasil penelitian berupa 1 (satu) CD format PDF kepada Bupati diserahkan melalui Kepala Badan Perencanaan Pembangunan Daerah.*
5. *Izin ini dapat dibatalkan sewaktu-waktu apabila tidak dipembi ketentuan-ketentuan di atas.*

Demikian ijin ini dikeluarkan untuk digunakan sebagaimana mestinya, diharapkan pejabat pemerintah/non pemerintah setempat memberikan bantuan seperlunya.

Setelah selesai pelaksanaan penelitian Saudara wajib menyampaikan laporan kepada kami 1 (satu) bulan setelah berakhirnya penelitian.

Dikeluarkan di Sleman

Pada Tanggal : 27 September 2012

a.n. Kepala Badan Perencanaan Pembangunan Daerah

Sekretaris
u.b.
Kepala Bidang Pengendalian dan Evaluasi

Dra. SUCI IRIANI SINURAYA, M.Si, M.M
Pembina, IV/a
NIP 19630112 198903 2 003

Tembusan :

1. Bupati Sleman (sebagai laporan)
2. Kepala Kantor Kesatuan Bangsa Kab. Sleman
3. Kepala Dinas Dikpora Kab. Sleman
4. Kabid. Sosial Bappeda Kab. Sleman
5. Camat Depok
6. Dekan Fak. Teknik-UNY
7. Yang Bersangkutan