

**PENGEMBANGAN APLIKASI PENERJEMAH BAHASA INDONESIA-
BAHASA JAWA PADA PLATFORM BERBASIS ANDROID**

TUGAS AKHIR SKRIPSI

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta
untuk Memenuhi Sebagian Persyaratan
Guna Memperoleh Gelar Sarjana Pendidikan



Oleh:
Iman Santoso
NIM 10520244057

**PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2017**

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

PENGEMBANGAN APLIKASI PENERJEMAH BAHASA INDONESIA- BAHASA JAWA PADA PLATFORM BERBASIS ANDROID

Disusun oleh:

Iman Santoso

NIM 10520244057

telah memenuhi syarat dan disetujui oleh Dosen Pembimbing untuk dilaksanakan
Ujian Tugas Akhir Skripsi bagi yang bersangkutan.

Yogyakarta, 28 Juli 2017

Mengetahui,
Ketua Program Studi
Pendidikan Teknik Informatika,

Handaru Jati, S.T., M.M., M.T., Ph.D.
NIP. 19740511 199903 1 002

Disetujui,
Dosen Pembimbing,

Totok Sukardiyono, M.T.
NIP. 19670930 199303 1 005

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Iman Santoso

NIM : 10520244057

Program Studi : Pendidikan Teknik Informatika

Judul TAS : Pengembangan Aplikasi Penerjemah Bahasa
Indonesia-Bahasa Jawa Pada Platform Berbasis
Android

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.

Yogyakarta, 27 Juli 2017

Penulis,



Iman Santoso

NIM. 10520244057

HALAMAN PENGESAHAN

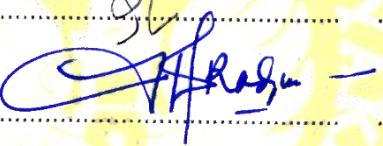
Tugas Akhir Skripsi

PENGEMBANGAN APLIKASI PENERJEMAH BAHASA INDONESIA- BAHASA JAWA PADA PLATFORM BERBASIS ANDROID

Disusun Oleh:
Iman Santoso
NIM 10520244057

telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi
Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta
pada tanggal 03 Agustus 2017

TIM PENGUJI

Nama/Jabatan	Tanda Tangan	Tanggal
Totok Sukardiyono, M.T. Ketua Penguji/Pembimbing		18 - 08 - 2017
Sigit Pambudi, M.Eng Sekretaris		18 - 08 - 2017
Dr. Eko Marpanaji Penguji		18 - 08 - 2017

Yogyakarta, Agustus 2017
Fakultas Teknik Universitas Negeri Yogyakarta
Dekan,



Dr. Widarto, M.Pd.
NIP. 19631230 198812 1 001

MOTTO

"Karena sesungguhnya bersama setiap kesulitan ada kemudahan. Sesungguhnya bersama setiap kesulitan ada kemudahan" — (Surat Al-Insyirah:5-6)

"Jadikanlah sabar dan shalat sebagai penolongmu..." — (Surat Al-Baqarah:45)

"Setiap ada awal pasti ada akhir, setiap masalah pasti ada solusi" — Anonim

"Yesterday is history, tomorrow is a mystery, today is a gift of God, which is why we call it the present" — Kung Fu Panda Film

"Tidak ada cara yang terbaik, tetapi selalu ada cara yang lebih baik" — Anonim

HALAMAN PERSEMBAHAN

Karya ini kupersembahkan untuk:

Bapak dan Ibu tercinta, yang selalu memberikan doa dan dukungan serta pengertiannya hingga selesaiya karya ini.

Mbak Pristi, Mas Ias, dan Mas Hape, terima kasih atas dukungan serta bantuan dalam penyelesaian karya ini.

Teman-teman “Geng Cacing”, yang selalu memberikan keceriaan di setiap kumpul bersama.

Teman-teman kelas G PTI 2010, terima kasih atas kebersamaannya serta jasa kalian yang tidak dapat tergantikan.

PENGEMBANGAN APLIKASI PENERJEMAH BAHASA INDONESIA-BAHASA JAWA PADA PLATFORM BERBASIS ANDROID

Oleh:

Iman Santoso
NIM. 10520244057

ABSTRAK

Tujuan penelitian ini dirancang untuk: (1) mengembangkan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa* pada *platform* Android, dan (2) mengetahui kualitas aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa berdasarkan standar pengujian kualitas perangkat lunak ISO 25010 meliputi aspek *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi.

Metode yang digunakan adalah *Research and Development* (R&D) dengan menggunakan model pengembangan *waterfall*. Model pengembangan *waterfall* meliputi: (1) analisis, (2) desain, (3) implementasi, dan (4) pengujian. Sumber data/subjek penelitian melibatkan tiga ahli *software* untuk pengujian *functionality suitability*, dua ahli materi, 30 responden siswa SMK Muhammadiyah 2 Yogyakarta untuk pengujian *usability*, dan dokumentasi terhadap aplikasi untuk pengujian *performance efficiency, compatibility, dan reliability*. Teknik pengumpulan data berupa wawancara, observasi dan angket atau kuesioner.

Hasil dari penelitian ini adalah: (1) pengembangan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa* pada *platform* Android menggunakan *Android Studio*, (2) hasil pengujian kualitas aplikasi diperoleh nilai *functional suitability* 100% (Sangat Layak), *performance efficiency* menunjukkan skala kualitas "Layak", *compatibility* diperoleh skala kualitas "Sangat Layak", *usability* diperoleh nilai 80,04% (Layak), *reliability* diperoleh nilai 100% dengan skala kualitas "Sangat Layak", dan pengujian materi diperoleh nilai 100% (Sangat Layak).

Kata kunci: *aplikasi, penerjemah, Bahasa Jawa, Android, ISO 25010*

KATA PENGANTAR

Puji syukur kehadirat Allah Subhanahu Wa Ta'ala atas berkat rahmat dan karunia-Nya, Tugas Akhir Skripsi dalam rangka untuk memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan dengan judul "Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform Berbasis Android" dapat disusun sesuai harapan. Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan dan arahan dari berbagai pihak. Berkennaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada yang terhormat:

1. Totok Sukardiyono, M.T. selaku Dosen Pembimbing TAS yang telah banyak memberikan semangat, dorongan, dan bimbingan selama penyusunan Tugas Akhir Skripsi ini.
2. Muhammad Izzuddin Mahali, M.Cs., Nurkhamid, Ph.D., Apriyani, S.Pd., Elvina Kartikasari, S.Pd., Nur Rahmad Suhendra, Arijati Faipkah, S.Pd., Yuniarti, S.Pd., selaku validator dalam penelitian TAS yang memberikan saran/masukan perbaikan sehingga penelitian TAS dapat terlaksana sesuai dengan tujuan.
3. Totok Sukardiyono, M.T., Sigit Pambudi, M.Eng., Dr. Eko Marpanaji, selaku Ketua Penguji, Sekretaris, dan Penguji yang memberikan koreksi perbaikan secara komprehensif terhadap TAS ini.
4. Dr. Fatchul Arifin, S.T., M.T. dan Handaru Jati, S.T., M.M., M.T., Ph.D., selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Informatika dan Ketua Program Studi Pendidikan Teknik Informatika beserta dosen dan staf yang

telah memberikan bantuan dan fasilitas selama proses penyusunan pra proposal sampai dengan selesainya TAS ini.

5. Dr. Widarto, M.Pd., selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi.
6. Muhammin, S.Ag., M.Pd., selaku Kepala SMK Muhammadiyah 2 Yogyakarta yang telah memberikan izin dan bantuan dalam pelaksanaan Tugas Akhir Skripsi.
7. Wakil Kepala Sekolah, Guru, Staff, dan siswa SMK Muhammadiyah 2 Yogyakarta yang telah memberi bantuan dalam pelaksanaan Tugas Akhir Skripsi.
8. Semua pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan di sini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Akhirnya, semoga segala bantuan yang telah diberikan semua pihak di atas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah Subhanahu Wa Ta'ala dan Tugas Akhir Skripsi ini menjadi informasi yang bermanfaat bagi pembaca atau pihak lain yang membutuhkan.

Yogyakarta, 27 Juli 2017

Penulis,



Iman Santoso

NIM. 10520244057

DAFTAR ISI

	Halaman
HALAMAN SAMPUL	i
LEMBAR PERSETUJUAN	ii
SURAT PERNYATAAN	iii
HALAMAN PENGESAHAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR	xv
DAFTAR LAMPIRAN.....	xvii
BAB I PENDAHULUAN.....	1
A. Latar Belakang Masalah.....	1
B. Identifikasi Masalah	5
C. Batasan Masalah.....	6
D. Rumusan Masalah.....	6
E. Tujuan Penelitian.....	7
F. Manfaat Penelitian	7
1. Manfaat Praktis	7
2. Manfaat Teoritis	8
BAB II KAJIAN PUSTAKA	9
A. Kajian Teori.....	9

1. Aplikasi.....	9
2. Penerjemah	10
a. Pengertian Penerjemahan	10
b. Proses Penerjemahan	11
3. Bahasa Jawa	13
a. Ragam <i>Ngoko</i>	14
b. Ragam <i>Krama</i>	15
4. Android	17
5. <i>Stemming</i>	20
a. <i>Stemming</i> Nazief dan Adriani.....	21
b. <i>Stemming</i> Bahasa Jawa	26
6. Afiksasi	27
a. Afiksasi Bahasa Indonesia	29
b. Afiksasi Bahasa Jawa	31
7. Model Pengembangan.....	35
a. Pengembangan Perangkat Lunak.....	35
b. Model <i>Waterfall</i>	37
8. Kualitas Perangkat Lunak	38
B. Hasil Penelitian yang Relevan	45
C. Kerangka Pikir	47
D. Pertanyaan Penelitian.....	48
BAB III METODE PENELITIAN.....	49
A. Model Pengembangan.....	49
B. Prosedur Pengembangan.....	50
1. Analisis	50
2. Desain.....	51

a. Desain <i>Unified Modelling Language</i> (UML)	51
b. Desain Sistem	59
c. Desain <i>Interface</i>	68
3. Implementasi	70
4. Pengujian	70
C. Tempat dan Waktu Penelitian	71
D. Sumber Data / Subjek Penelitian.....	71
E. Teknik Pengumpulan Data.....	72
1. Wawancara.....	72
2. Observasi.....	72
3. Angket atau Kuesioner.....	72
F. Instrumen Penelitian.....	73
1. Instrumen <i>Functionality Suitability</i>	74
2. Instrumen <i>Perfomance Efficiency</i>	74
3. Instrumen <i>Compatibility</i>	74
4. Instrumen <i>Usability</i>	76
5. Instrumen <i>Reliability</i>	78
6. Instrumen Uji Materi.....	78
G. Teknik Analisis Data.....	79
1. Aspek <i>Functionality Suitability, Compatibility, Usability, Reliability</i> , dan Uji Materi.....	79
2. Aspek <i>Performance Efficiency</i>	80
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	81
A. Hasil Penelitian.....	81
1. Tahap Analisis.....	81
a. Analisis Fungsional	81

b. Analisis Nonfungsional	81
2. Tahap Desain.....	83
3. Tahap Implementasi.....	83
a. Implementasi <i>Layout Interface</i> Aplikasi	84
b. Implementasi Logika Pemrograman	87
4. Tahap Pengujian	91
a. Pengujian <i>Functionality Suitability</i>	92
b. Pengujian <i>Performance Efficiency</i>	95
c. Pengujian <i>Compatibility</i>	100
d. Pengujian <i>Usability</i>	105
e. Pengujian <i>Reliability</i>	107
f. Pengujian Aspek Materi.....	108
B. Pembahasan	110
BAB V SIMPULAN DAN SARAN.....	113
A. Simpulan.....	113
B. Saran.....	114
DAFTAR PUSTAKA	116
LAMPIRAN	120

DAFTAR TABEL

	Halaman
Tabel 1. Kombinasi Awalan dan Akhiran yang tidak diizinkan	23
Tabel 2. Aturan Pemenggalan Prefiks (Sumber: Santosa, 2016)	23
Tabel 3. Persamaan Afiks Bahasa Jawa dan Bahasa Indonesia (Sumber: Krishandini, 2011).....	27
Tabel 4. Perbandingan Model Kualitas Perangkat Lunak.....	39
Tabel 5. Model Kualitas Produk ISO 25010	40
Tabel 6. Definisi Aktor	52
Tabel 7. Definisi <i>Use Case</i>	52
Tabel 8. Skenario <i>Use Case</i> Memulai Aplikasi.....	53
Tabel 9. Skenario <i>Use Case</i> Terjemahan	53
Tabel 10. Skenario <i>Use Case</i> Bantuan.....	54
Tabel 11. Skenario <i>Use Case</i> Info Pengembang	54
Tabel 12. Skenario <i>Use Case</i> Keluar	55
Tabel 13. Skala Guttman	73
Tabel 14. Skala Likert.....	73
Tabel 15. Kisi-kisi Instrumen <i>Functionality Suitability</i>	74
Tabel 16. Instrumen <i>Compatibility</i>	75
Tabel 17. Kisi-kisi Instrumen <i>Usability</i>	76
Tabel 18. Instrumen <i>Usability</i>	76
Tabel 19. Kisi-kisi Instrumen Uji Materi.....	78
Tabel 20. Skala Persentase Kelayakan	79
Tabel 21. Hasil Pengujian <i>Functionality Suitability</i>	92
Tabel 22. Analisis Data Pengujian <i>Functionality Suitability</i>	94
Tabel 23. Hasil Pengujian Aspek <i>Co-existence</i>	100
Tabel 24. Hasil Pengujian Berbagai Sistem Operasi dan Jenis Perangkat.....	103
Tabel 25. Analisis Data Pengujian Berbagai Sistem Operasi dan Jenis Perangkat.....	105
Tabel 26. Hasil Pengujian <i>Usability</i>	106
Tabel 27. Ringkasan Hasil Pengujian <i>Reliability</i>	108
Tabel 28. Analisis Data Hasil Pengujian Aspek Materi	109
Tabel 29. Ringkasan Hasil Pengujian Aplikasi	111

DAFTAR GAMBAR

	Halaman
Gambar 1. Statistik Kepemilikan <i>Smartphone</i> di Indonesia (DI Marketing, 2016)	3
Gambar 2. Statistik Penggunaan Smartphone di Indonesia (DI Marketing, 2016)	4
Gambar 3. Proses penerjemahan menurut Nida dan Taber.....	12
Gambar 4. Distribusi perangkat berdasarkan versi Android (Developers, 2017)	18
Gambar 5. Prosedur Pengembangan Model <i>Waterfall</i>	37
Gambar 6. Model Pengembangan <i>Waterfall</i>	49
Gambar 7. <i>Use Case</i> Aplikasi Ukara Jawa.....	51
Gambar 8. <i>Sequence Diagram</i> Memulai Aplikasi	56
Gambar 9. <i>Sequence Diagram</i> Terjemahan.....	56
Gambar 10. <i>Sequence Diagram</i> Bantuan.....	57
Gambar 11. <i>Sequence Diagram</i> Info Pengembang	58
Gambar 12. <i>Sequence Diagram</i> Keluar.....	58
Gambar 13. <i>Flowchart</i> pemilihan bahasa terjemahan	59
Gambar 14. <i>Flowchart</i> pencarian arti kata.....	60
Gambar 15. <i>Flowchart</i> pencarian arti kalimat.....	61
Gambar 16. <i>Flowchart</i> algoritma <i>stemming</i> Nazief dan Adriani.....	62
Gambar 17. <i>Flowchart</i> algoritma <i>stemming</i> Nazief dan Adriani (Lanjutan).....	63
Gambar 18. <i>Flowchart</i> algoritma <i>stemming</i> Bahasa Jawa	64
Gambar 19. <i>Flowchart</i> proses afiksasi.....	66
Gambar 20. Desain Perancangan Basis Data	67
Gambar 21. Desain <i>Interface Splash Screen</i>	68
Gambar 22. Desain <i>Interface</i> Halaman Utama	68
Gambar 23. Desain <i>Interface</i> Halaman Terjemahan	69
Gambar 24. Desain <i>Interface</i> Halaman Bantuan	69
Gambar 25. Desain <i>Interface</i> Halaman Info Pengembang.....	70
Gambar 26. Potongan <i>script</i> implementasi pada <i>ukarajawa_search_layout</i>	84
Gambar 27. Implementasi <i>layout interface</i> <i>splash screen</i>	85
Gambar 28. Implementasi <i>layout interface</i> Halaman Utama.....	85

Gambar 29. Implementasi <i>layout interface</i> Halaman Terjemahan.....	86
Gambar 30. Implementasi <i>layout interface</i> Halaman Bantuan	86
Gambar 31. Implementasi <i>layout interface</i> Halaman Info Pengembang	87
Gambar 32. <i>Package file</i> Java pada Aplikasi Ukara Jawa.....	88
Gambar 33. Potongan Implementasi Algoritma <i>Stemming</i> Nazief Adriani	89
Gambar 34. Potongan Hasil Implementasi Algoritma <i>Stemming Ngoko</i>	90
Gambar 35. Potongan Hasil Implementasi Algoritma <i>Stemming Krama</i>	90
Gambar 36. Potongan Hasil Implementasi Logika Pemrograman Afiksasi Bahasa.....	91
Gambar 37. Hasil Pengujian <i>Launching Time</i>	95
Gambar 38. Hasil Pengujian <i>Rendering Time</i>	96
Gambar 39. Hasil Pengujian <i>CPU Usage</i>	97
Gambar 40. <i>CPU Usage</i> pada perangkat Samsung Galaxy S III	98
Gambar 41. Rata-rata Hasil Pengujian <i>CPU Usage</i>	98
Gambar 42. Hasil Pengujian <i>Resource Utilization Memory</i>	99
Gambar 43. Daftar Perangkat Pengujian <i>Compatibility</i>	102
Gambar 44. Sampel Pengujian Perangkat Samsung Galaxy S4.....	102
Gambar 45. Hasil Pengujian <i>Reliability</i>	107

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Surat Keputusan Pengangkatan Pembimbing Skripsi.....	121
Lampiran 2. Surat Izin Penelitian	122
Lampiran 3. Surat Rekomendasi Izin Penelitian PDM	123
Lampiran 4. Surat Permohonan Validasi Instrumen TAS.....	124
Lampiran 5. Surat Keterangan Validasi Instrumen TAS	126
Lampiran 6. Hasil Validasi Instrumen TAS.....	128
Lampiran 7. Sampel Kuesioner Instrumen <i>Functionality Suitability</i>	130
Lampiran 8. Sampel Kuesioner Instrumen Uji Materi.....	133
Lampiran 9. Sampel Kuesioner Instrumen <i>Usability</i>	140
Lampiran 10. Kartu Bimbingan.....	143

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Bahasa Jawa merupakan salah satu bahasa daerah yang digunakan masyarakat Jawa di Indonesia untuk berkomunikasi. Selain sebagai alat komunikasi, Bahasa Jawa juga merupakan salah satu warisan kebudayaan sehingga perlu dilestarikan agar tetap menjadi bagian dari kekayaan budaya Indonesia. Pada Kongres Bahasa Jawa VI yang dilaksanakan pada tanggal 8 sampai 12 November 2016 di Hotel Inna Garuda Yogyakarta dicetuskan jalur upaya pelestarian Bahasa Jawa dapat dilaksanakan melalui lembaga pendidikan baik formal maupun informal, lembaga pemerintahan, serta komunitas yang bergerak dalam bidang kejawaan (Harian Jogja, 2016). Sejalan dengan hal itu, Gubernur Provinsi Daerah Istimewa Yogyakarta telah menetapkan Pergub No 64 tahun 2013 supaya Bahasa Jawa menjadi muatan lokal wajib yang diadakan pada jenjang sekolah/madrasah (Krjogja, 2016).

Perlunya pelestarian Bahasa Jawa dikarenakan Bahasa Jawa mencerminkan budaya Jawa yang mengandung nilai budi pekerti serta etika kesopanan sebagai pembangun karakter bangsa. Salah satu cerminan budi pekerti dan etika adalah melalui kemampuan bertutur kata sesuai tingkat tutur Bahasa Jawa (*unggah-ungguh basa*). *Unggah-ungguh basa* merupakan salah satu materi pembelajaran Bahasa Jawa pada jenjang pendidikan formal. Meskipun demikian, masih banyak siswa yang merasa kesulitan menggunakan *unggah-ungguh basa*. Siswa masih merasakan rumitnya penggunaan *unggah-*

ungguh dikarenakan kurangnya pengetahuan *unggah-ungguh* yang harus digunakan ketika berbicara.

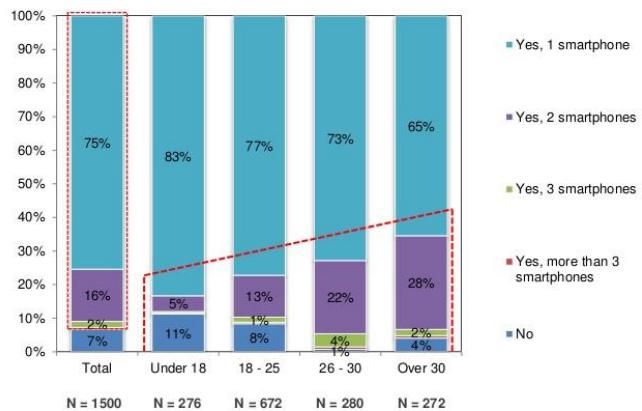
Berdasarkan wawancara kepada guru Bahasa Jawa di SMK Muhammadiyah 2 Yogyakarta, siswa juga mengalami permasalahan penggunaan *unggah-ungguh basa*. Dari hasil wawancara didapatkan informasi bahwa siswa masih kebingungan dalam pemilihan *unggah-ungguh basa*. Saat pembelajaran tentang *unggah-ungguh* yang melibatkan siswa, siswa sering menggunakan Bahasa Indonesia dan Bahasa Jawa *ngoko* ketika siswa tersebut tidak mengetahui bentuk ragam *krama*. Hal ini dikarenakan sebagian siswa ada yang dibiasakan menggunakan Bahasa Indonesia dan sebagian siswa yang lain dibiasakan menggunakan ragam *ngoko* di dalam keluarganya.

Saat ini perkembangan teknologi telah menghasilkan beberapa perangkat *mobile* salah satunya *smartphone*. *Smartphone* merupakan perangkat yang dapat menjalankan beberapa fitur layaknya PDA atau komputer (Shiraishi et al dalam Rustrini, 2016). Kemampuan komputasi layaknya komputer serta daya dukung mobilitas yang tinggi menjadikan *smartphone* memiliki banyak pengguna. Peng, Jinming, & Tianzhou (2012) dalam jurnalnya menyebutkan *smartphone* menjadi perangkat *mobile* yang penggunanya paling tinggi diantara perangkat *mobile* lain serta mencapai 195 juta pengguna pada tahun 2010. Kategori penggunanya tidak terbatas pada golongan maupun usia tertentu. Di Indonesia, hampir semua golongan memiliki *smartphone* seperti disajikan pada Gambar 1.

93% USE SMARTPHONE

The older group owns more smartphones

Are you currently using smartphone?



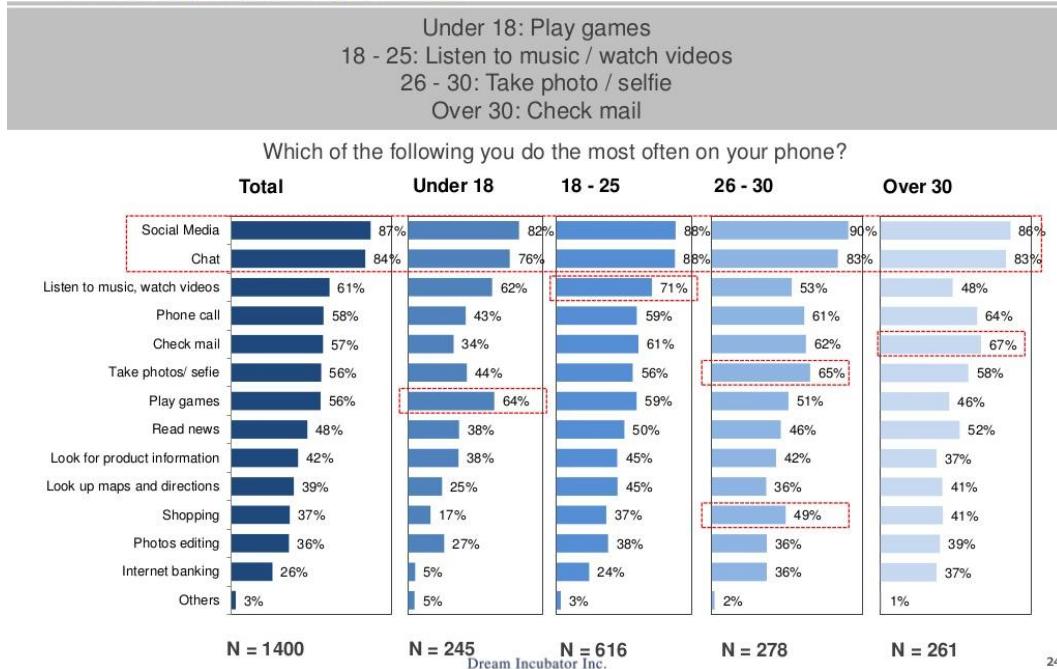
Dream Incubator Inc.

9

Gambar 1. Statistik Kepemilikan *Smartphone* di Indonesia (DI Marketing, 2016)

Meskipun jumlah penggunanya banyak, kegunaan *smartphone* kurang dimanfaatkan sebagai kebutuhan penunjang pembelajaran. Kaitannya dengan subjek penelitian ini siswa SMK yang dikategorikan sebagai anak usia remaja cenderung menggunakan *smartphone* sebagai alat untuk memenuhi kebutuhan hiburan (DI Marketing, 2016) seperti yang disajikan pada Gambar 2.

SOCIAL MEDIA AND CHATTING ARE MOST FREQUENT ACTIVITIES WHEN USING SMARTPHONE



Gambar 2. Statistik Penggunaan Smartphone di Indonesia (DI Marketing, 2016)

Di lain sisi, *smartphone* dengan sistem operasi Android menjadi *smartphone* yang paling banyak digunakan terutama di Indonesia. Berdasarkan data statistik StatCounter (2017) menunjukkan pada tahun 2016 hingga 2017 *smartphone* Android mencapai 65.87% pengguna di Indonesia. Melihat jumlah penggunanya yang cukup besar, maka dapat dibuat aplikasi yang bertujuan untuk mempermudah siswa menggunakan *unggah-ungguh basa*.

Berdasarkan permasalahan yang ada maka perlu dikembangkan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa*. Penelitian ini mengembangkan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) yang diharapkan dapat membantu siswa

mempelajari serta menggunakan *unggah-ungguh basa*. Aplikasi ini dikembangkan dari awal dan belum diketahui kualitasnya sehingga pada tahapan pengembangannya perlu dilakukan pengujian kualitas terhadap aplikasi Ukara Jawa. Pengujian kualitas aplikasi dilakukan untuk menghindari berbagai kesalahan aplikasi sebelum disebarluaskan. Untuk mengetahui kualitas aplikasi Ukara Jawa digunakan standar ISO 25010. Pengujian yang akan dilakukan meliputi aspek *functional suitability, performance efficiency, compatibility, usability, dan reliability*. Selain itu, pengujian materi juga dilakukan untuk mengetahui kelayakan aplikasi dari sisi materi. Penelitian dilakukan dengan mengembangkan serta menguji aplikasi Ukara Jawa pada *platform* Android di SMK Muhammadiyah 2 Yogyakarta. Diharapkan dengan adanya aplikasi Ukara Jawa dapat menarik minat serta mempermudah siswa mempelajari dan melestarikan Bahasa Jawa khususnya *unggah-ungguh basa*.

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, penulis mengidentifikasi berbagai masalah, di antaranya:

1. Siswa merasa rumit menggunakan *unggah-ungguh* karena kurangnya pengetahuan *unggah-ungguh* yang harus digunakan ketika berbicara.
2. Siswa SMK Muhammadiyah 2 Yogyakarta masih kebingungan dalam menggunakan *unggah-ungguh* Bahasa Jawa.
3. Siswa sering menggunakan Bahasa Indonesia dan Bahasa Jawa *ngoko* ketika tidak mengetahui bentuk ragam *krama*.

4. Siswa tidak dibiasakan menggunakan ragam *krama* di dalam keluarganya.
5. *Smartphone* kurang dimanfaatkan sebagai penunjang pembelajaran.
6. Perlunya pengembangan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa*.
7. Belum diketahui kualitas aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) yang meliputi aspek *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi.

C. Batasan Masalah

Berdasarkan identifikasi masalah yang telah dijabarkan, maka penelitian ini dibatasi pada masalah:

1. Perlunya pengembangan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa*.
2. Belum diketahui kualitas aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) yang meliputi aspek *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi.

D. Rumusan Masalah

Berdasarkan batasan masalah di atas maka penelitian ini akan dirumuskan sebagai berikut:

1. Bagaimana cara mengembangkan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa* pada *platform* Android ?

2. Bagaimana kualitas aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) meliputi aspek *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi ?

E. Tujuan Penelitian

Penelitian ini diadakan dengan tujuan:

1. Mengembangkan aplikasi penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa* pada *platform* Android.
2. Mengetahui kualitas aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) yang meliputi aspek *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi.

F. Manfaat Penelitian

Manfaat yang didapat dari penelitian ini adalah sebagai berikut:

1. Manfaat Praktis

Manfaat praktis dari penelitian ini antara lain:

- a. Aplikasi ini menjadi salah satu upaya untuk melestarikan Bahasa Jawa.
- b. Bagi dunia pendidikan, aplikasi ini menjadi salah satu media yang dapat digunakan sebagai penunjang belajar Bahasa Jawa.
- c. Bagi pengguna, aplikasi ini diharapkan dapat mempermudah pengguna dalam berkomunikasi dengan Bahasa Jawa sesuai *unggah-ungguh* serta meningkatkan minat pengguna mempelajari Bahasa Jawa.

- d. Bagi penulis, aplikasi ini dapat menjadi referensi untuk mengembangkan aplikasi pada perangkat *mobile* berbasis *Android*.

2. Manfaat Teoritis

Hasil dari pengembangan dan penelitian aplikasi ini diharapkan dapat menjadi referensi penelitian yang berkaitan dengan pengembangan aplikasi berbasis *Android*.

BAB II

KAJIAN PUSTAKA

A. Kajian Teori

1. Aplikasi

Aplikasi menurut Kamus Besar Bahasa Indonesia (2008: 81) didefinisikan sebagai program komputer atau perangkat lunak yang didesain untuk mengerjakan tugas tertentu. Sejalan dengan pengertian Kamus Besar Bahasa Indonesia, Ventola (2014) menjelaskan aplikasi adalah perangkat lunak yang dikembangkan pada komputer atau perangkat *mobile* untuk menjalankan tugas tertentu. Berdasarkan kategori pengembangannya, aplikasi *mobile* dibagi menjadi aplikasi *native*, aplikasi *web*, dan aplikasi *hybrid* (Göth, 2015).

Aplikasi *native* dikembangkan dengan menggunakan bahasa pemrograman seperti Java atau C++ untuk Android, Objective-C atau Swift untuk iOS, dan C#, Visual Basic atau C++ untuk Windows Phone. Aplikasi *native* dikembangkan secara terbatas untuk sistem operasi tertentu. Aplikasi *web* dikembangkan menggunakan teknologi *web* (HTML, CSS, Javascript) dan *server-side code* seperti Node.js, PHP, ASP.NET, dan lain sebagainya. Aplikasi *web* dikembangkan dengan mengambil sebagian atau seluruh informasi pada *web*. Aplikasi *hybrid* dikembangkan dengan kombinasi aplikasi *native* dan aplikasi *web*. Tujuan pengembangan aplikasi *hybrid* supaya dapat dijalankan pada lintas *platform*.

Pengembangan aplikasi *native* memiliki beberapa kelebihan antara lain memiliki *performance* terbaik serta kemampuan untuk dapat mengakses semua fasilitas dan sumber daya yang terdapat pada perangkat telepon seperti GPS, *bluetooth*, *accelerometer*, kompas, dan lain sebagainya. Selain itu (Hartman, Rokitta, & Peake, 2013) menambahkan aplikasi *native* dapat dikembangkan secara lokal dengan menggunakan bantuan penyimpanan lokal seperti SQLite sehingga aplikasi dapat melanjutkan aktivitasnya tanpa adanya koneksi Internet.

Aplikasi dapat didefinisikan sebagai sebuah perangkat lunak yang dikembangkan pada perangkat tertentu untuk menjalankan tugas tertentu. Aplikasi pada penelitian ini dikembangkan secara *native* dengan bantuan penyimpanan lokal menggunakan SQLite.

2. Penerjemah

a. Pengertian Penerjemahan

Suryawinata & Hariyanto (2003) menjabarkan penjelasan penerjemahan menurut beberapa ahli sebagai berikut:

- 1) Nida dan Taber (1969) menjelaskan penerjemahan sebagai usaha mencipta kembali pesan dalam bahasa sumber ke dalam bahasa sasaran dengan padanan alami yang sedekat mungkin, pertama-tama dalam hal makna dan kemudian gaya bahasanya.
- 2) Brislin (1976: 1) menjelaskan penerjemahan adalah proses pengalihan buah pikiran dan gagasan dari satu bahasa (sumber) ke dalam bahasa lain (sasaran), dalam bentuk tulisan maupun lisan, baik kedua bahasa

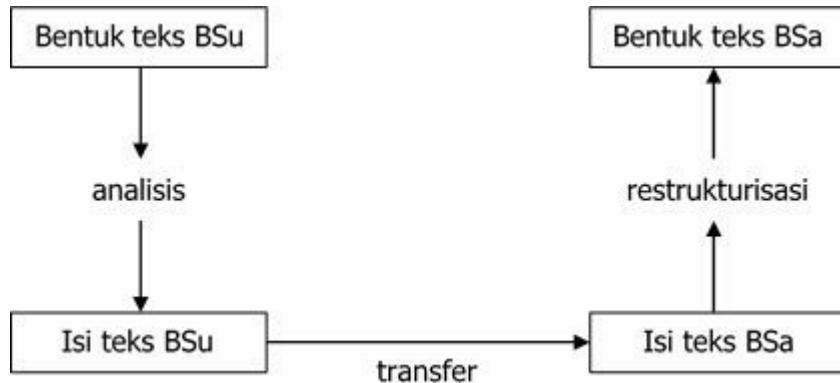
tersebut telah mempunyai sistem penulisan yang telah baku ataupun belum.

- 3) McGuire (1980: 2) menjelaskan penerjemahan merupakan usaha menjadikan bahasa sumber ke bahasa sasaran sehingga (1) makna keduanya menjadi hampir mirip dan (2) struktur bahasa sumber dapat dipertahankan setepat mungkin.
- 4) Newmark (1981: 7) menjabarkan penerjemahan sebagai suatu kiat yang merupakan usaha untuk mengganti suatu pesan atau pernyataan tertulis dalam satu bahasa dengan pesan atau pernyataan yang sama dalam bahasa lain.

Berdasarkan pendapat beberapa ahli di atas, penerjemahan dapat dikatakan sebagai proses penulisan atau pengungkapan ulang sebuah pesan dari bahasa asal ke dalam bentuk bahasa tujuan tanpa mengurangi makna yang terdapat pada bahasa asal.

b. Proses Penerjemahan

Proses penerjemahan merupakan model yang dimaksudkan untuk menerangkan proses internal yang dilakukan saat melakukan penerjemahan (Suryawinata & Hariyanto, 2003). Nida dan Taber (1969: 33) menggambarkan proses penerjemahan sebagai berikut:



Gambar 3. Proses penerjemahan menurut Nida dan Taber

Pada proses penerjemahan di atas terdapat tiga tahap yaitu tahap analisis, transfer, dan restrukturisasi.

- 1) Pada tahap analisis, struktur kalimat yang ada dianalisis menurut hubungan gramatikal, makna kata atau kombinasi kata, makna textual, dan makna kontekstual.
- 2) Pada tahapan transfer, materi yang sudah dianalisis dan dipahami kemudian diolah penerjemah dan dipindah dari BSu (bahasa sumber) ke dalam BSa (bahasa sasaran).
- 3) Tahap restrukturisasi dilakukan penerjemah dengan mencari padanan kata, ungkapan, dan struktur kalimat yang tepat dalam bahasa sasaran sehingga isi, makna, dan pesan yang ada dalam teks bahasa sumber dapat disampaikan sepenuhnya dalam bahasa sasaran.

Pada tahapan analisis, penerjemah memerlukan metode penerjemahan untuk menghasilkan penerjemahan yang baik. Salah satu metode penerjemahan yaitu menggunakan metode kata demi kata. Menurut Nababan

(1999) penerjemahan kata demi kata (*word-for-word translation*) adalah suatu jenis penerjemahan yang pada dasarnya masih sangat terikat pada tataran kata. Dalam melakukan tugasnya, penerjemah hanya mencari padanan kata bahasa sumber dalam bahasa sasaran, tanpa mengubah susunan kata dalam terjemahannya. Susunan kata dalam kalimat terjemahan sama persis dengan susunan kata dalam kalimat aslinya. Penerjemahan tipe ini bisa diterapkan hanya kalau bahasa sumber dan bahasa sasaran mempunyai struktur yang sama.

Penerjemahan pada aplikasi yang akan dikembangkan menggunakan metode kata demi kata. Pemilihan metode ini dikarenakan Bahasa Indonesia dan Bahasa Jawa memiliki kesamaan struktur kalimat yaitu terdiri atas subjek (S), predikat (P), dan objek (O) (Wibawa, Kadarisman, & Mahmudy, 2013).

3. Bahasa Jawa

Bahasa Jawa merupakan bahasa penduduk Jawa yang tinggal di Provinsi Jawa Timur, Jawa Tengah, Daerah Istimewa Yogyakarta, Banten, Lampung, sekitar Medan, sekitar Riau, daerah transmigrasi dan beberapa wilayah di luar negeri seperti Suriname, Belanda, new Caledonia, Pantai Barat Johor (Wedhawati, 2006).

Bahasa Jawa mengenal adanya sistem tingkat tutur bahasa atau *unggah-ungguh basa*. Poedjosoedarmo et al (2013) berpendapat tingkat tutur adalah variasi-variasi bahasa yang perbedaan antara satu dan lainnya ditentukan oleh perbedaan sikap santun yang ada pada diri pembicara terhadap lawan bicara dan yang dibicarakan. Tingkat tutur dalam bahasa Jawa secara garis besar

terbagi menjadi tiga yaitu: (1) *ngoko*, (2) *madya*, dan *krama* (Wedhawati, 2006). Namun menurut Sudaryanto (1994) bentuk *basa madya* atau yang lazim disebut *krama madya* termasuk dalam jenis *krama*.

Tingkat tutur *ngoko* mencerminkan rasa tak berjarak antara 01 (penutur) terhadap 02 (mitra bicara) dan artinya tingkat tutur ini digunakan untuk menyatakan keakraban antara 01 terhadap 02. Tingkat tutur *krama* merupakan tingkat tutur yang memancarkan arti penuh sopan santun. Tingkat tutur *krama* menandakan adanya perasaan segan 01 terhadap 02.

a. Ragam *Ngoko*

Ragam *ngoko* adalah bentuk *unggah-ungguh* Bahasa Jawa yang berintikan leksikon *ngoko*, atau yang menjadi unsur inti di dalam ragam *ngoko* adalah leksikon *ngoko* bukan leksikon yang lain. Afiks yang muncul dalam ragam ini pun semuanya berbentuk *ngoko* (misalnya afiks di-, -e, dan -ake) (Andayani dalam Rohmadi, 2011: 87). Ragam *ngoko* digunakan untuk pembicaraan orang yang lebih tua terhadap orang yang lebih muda atau orang yang memiliki status sosial lebih tinggi terhadap lawan bicara. Selain itu, ragam *ngoko* juga digunakan untuk pembicaraan antara penutur terhadap lawan bicara yang sudah akrab (Wibawa et al, 2004: 49). Ragam *ngoko* mempunyai dua bentuk varian, yaitu *ngoko lugu* dan *ngoko alus*.

1) *Ngoko Lugu*

Ngoko lugu adalah bentuk-bentuk *unggah-ungguh* Bahasa Jawa yang semua kosakatanya berbentuk *ngoko*. Pilihan kosakata *ngoko* bersifat netral (leksikon *ngoko* dan netral). Di dalamnya tidak terselip leksikon *krama*, *krama*

inggil, atau *krama andhap* (Andayani dalam Rohmadi, 2011: 87). *Ngoko lugu* digunakan untuk berbicara antara orang tua kepada anak, cucu, atau pada anak muda lainnya; percakapan orang-orang sederajat, tidak memperhatikan kedudukan dan usia; atasan kepada bawahannya; serta dipakai saat *ngunandika*, sebab yang diajak berbicara adalah diri sendiri (Setiyanto, 2007: 29-30).

Contoh:

Yanti njupukake piring adhine. (Harjawiyana & Supriya, 2001: 128).

'Yanti mengambilkan piring adiknya.'

2) *Ngoko Alus*

Ngoko alus adalah bentuk *unggah-ungguh* yang di dalamnya bukan hanya terdiri atas leksikon *ngoko* dan netral saja melainkan juga terdiri atas leksikon *krama inggil*, *krama andhap*, dan *krama* (Andayani dalam Rohmadi, 2011: 87).

Ngoko alus digunakan untuk berbicara terhadap orang yang memiliki status sama tetapi memiliki rasa hormat; berbicara terhadap orang yang memiliki status lebih tinggi tetapi sudah sangat akrab; berbicara terhadap orang yang memiliki status menengah; membicarakan orang lain yang memiliki status lebih tinggi (Harjawiyana & Supriya, 2001: 49).

Contoh :

Ibu Tini mundhutake ambengan rayine. (Harjawiyana & Supriya, 2001: 128).

'Ibu Tini mengambilkan piring adiknya'

b. *Ragam Krama*

Ragam *krama* adalah bentuk *unggah-ungguh* Bahasa Jawa yang berintikan leksikon *krama* atau yang menjadi unsur inti di dalam ragam *krama*

adalah leksikon *krama* bukan leksikon yang lain. Afiks yang muncul dalam ragam ini pun semuanya berbentuk *krama* (misalnya afiks *dipun-*, *-ipun*, dan *-aken*). Ragam *krama* digunakan oleh mereka yang belum akrab dan oleh mereka yang merasa dirinya lebih rendah status sosialnya daripada lawan bicara. Ragam *krama* memiliki dua bentuk varian, yaitu *krama alus* dan *krama lugu* (Andayani dalam Rohmadi, 2011: 88).

1) *Krama Lugu*

Ragam *krama lugu* diartikan sebagai ragam yang semua kosakatanya terdiri atas leksikon *krama*, *madya*, netral, atau *ngoko* dan dapat ditambah dengan leksikon *krama inggil* atau *krama andhap*. Meskipun begitu, yang menjadi leksikon inti dalam ragam *krama lugu* adalah leksikon *krama*, *madya* dan netral, sedangkan leksikon *krama inggil* atau *krama andhap* yang muncul dalam ragam ini hanya digunakan untuk menghormati lawan bicara (Andayani dalam Rohmadi, 2011: 88).

Contoh:

Mbok Radem mendhetaken lancaran adhinipun. (Harjawiyana & Supriya, 2001: 128).

'Bu Radem mengambilkan piring adiknya.'

2) *Krama Alus*

Ragam *krama alus* yaitu *unggah-ungguh* Bahasa Jawa yang semua kosakatanya menggunakan ragam *krama* atau *krama inggil*. Afiks yang terdapat dalam ragam ini menggunakan ragam *krama* (Wibawa *et al*, 2004: 51). *Krama alus* digunakan untuk berbicara terhadap orang yang memiliki status sama namun sangat menghormati dikarenakan kurang akrab; berbicara

terhadap orang yang memiliki status lebih tinggi; berbicara pada suasana yang memerlukan *unggah-ungguh* lebih lengkap (Harjawiyan & Supriya, 2001: 102).

Contoh:

Ibu Tini mundhutaken ambengan rayinipun. (Harjawiyan & Supriya, 2001: 128).

'Ibu Tini mengambilkan piring adiknya.'

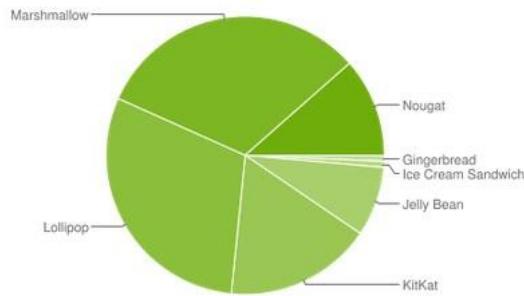
4. **Android**

Android merupakan sistem operasi berbasis Linux yang digunakan untuk perangkat *mobile* seperti *smartphone* (Agustina, 2017). Sejalan dengan hal tersebut, Android adalah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware*, dan aplikasi (Safaat, 2015: 1). Awalnya Android merupakan sebuah perusahaan yang kemudian diakuisisi Google pada tahun 2000. Setelah diakuisisi, sistem operasi Android hanya dikembangkan untuk kebutuhan internal Google dan belum berlisensi *open source*.

Android pertama kali dirilis dengan standar *open source* pada 5 November 2007 bersama *Open Handset Alliance* (OHA). Alasan Google merilis Android secara *open source* untuk mempermudah *user* dalam mengakses layanan Google dan Android digunakan sebagai media baru penayangan iklan yang dapat menambah pendapatan serta sebagian pendapatan tersebut untuk pengembangan OS dan aplikasi Android (Istiyanto, 2013).

Berdasarkan versi pengembangannya, distribusi perangkat dengan sistem operasi Android sampai tanggal 6 Juli 2017 dapat dilihat pada gambar 4.

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.7%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.7%
4.1.x	Jelly Bean	16	2.8%
4.2.x		17	4.1%
4.3		18	1.2%
4.4	KitKat	19	17.1%
5.0	Lollipop	21	7.8%
5.1		22	22.3%
6.0	Marshmallow	23	31.8%
7.0	Nougat	24	10.6%
7.1		25	0.9%



*Data collected during a 7-day period ending on July 6, 2017.
Any versions with less than 0.1% distribution are not shown.*

Gambar 4. Distribusi perangkat berdasarkan versi Android (Developers, 2017)

Alasan pemilihan pengembangan pada *platform* Android karena di masa depan sistem operasi ini berpotensi sangat besar. Pertimbangan aplikasi dikembangkan pada sistem operasi Android (Juhara, 2016) adalah sebagai berikut:

- Keseragaman lingkungan pengembangan aplikasi.

Android menawarkan *platform* pengembangan yang konsisten untuk beragam vendor *mobile phone*. Aplikasi yang dibuat untuk *platform* Android dapat dijalankan pada perangkat *mobile* berbasis Android meski berbeda vendor.

b. Konsistensi akses *hardware*

Pada Android, aplikasi dapat menggunakan *hardware* yang tersedia sepanjang memiliki izin cukup. Developer dapat membuat aplikasi yang kaya fitur, yang terintegrasi dengan semua *hardware* yang disediakan.

c. Ekosistem *mobile phone* bertenaga, kaya fitur, aman, dan terjangkau

Aplikasi Android dibangun menggunakan Java. Meskipun memiliki *Java Virtual Machine* bernama Dalvik VM yang berjalan di atas kernel Linux, tetapi Android bukan implementasi Java ME. Dalvik sendiri sudah mengalami optimasi dan diubah sedemikian rupa sehingga, kode *intermediate* yang dihasilkan kompiler Java tidak akan dapat dijalankan. Optimasi ini bertujuan supaya Dalvik berjalan optimal pada perangkat *embedded* yang minim daya komputasi dan listrik.

Android dirancang untuk menyediakan beragam fitur supaya piranti komunikasi menjadi lebih dari sekedar telefon. Selain layanan telefon dan pesan, Android juga menyediakan fitur kelola lokasi, grafis, multimedia, jaringan, dan database. Karena dibangun di atas kernel Linux, Android mewarisi fitur keamanan dan stabilitas. Android menerapkan aturan ketat terkait penggunaan layanan dan *hardware* yang tersedia. Aplikasi tidak dapat menggunakan fitur *hardware* tanpa sepengetahuan pengguna. Tiap aplikasi wajib menyertakan izin yang cukup. Ketika aplikasi dipasang, pengguna akan diberitahu izin yang diminta aplikasi. Bila izin yang diminta tidak disetujui, instalasi aplikasi akan ditolak. Fitur keamanan ditambah kewaspadaan pengguna dapat menekan serangan *malware*.

Android dirilis menggunakan *Apache License*. Penggunaan lisensi ini menjadikan Android bersifat gratis dan bebas dikembangkan sehingga mampu menekan biaya produksi perangkat *mobile phone*. Vendor *mobile phone* sendiri tidak dikenakan biaya untuk menggunakan Android.

Selain itu, pertimbangan lain mengacu pada beberapa faktor untuk realisasi pembuatan aplikasi (Istiyanto, 2013: 16) sebagai berikut:

- a. Faktor kecepatan, tingkat efisiensi aplikasi dalam menyajikan data, proses dan memberikan *output* data secara cepat sesuai keinginan konsumen.
- b. Aspek produktivitas, kebermanfaatan aplikasi dalam peningkatan produktivitas *user* mencakup alasan penggunaan aplikasi untuk mengatas masalah yang sering dihadapi *user*.
- c. Kreativitas desain, nilai tambah yang dapat menarik minat *user* menggunakan aplikasi tersebut.
- d. Fleksibilitas dan kehandalan solusi alternatif dari keterbatasan aplikasi agar dapat berfungsi normal pada segala kondisi.

5. ***Stemming***

Menurut Agusta (2009) *stemming* merupakan suatu proses yang terdapat dalam sistem *Information Retrieval* yang mentransformasikan kata-kata yang terdapat dalam suatu dokumen ke kata-kata akarnya (*root word*) dengan menggunakan aturan tertentu. Proses *stemming* antara satu bahasa berbeda dengan bahasa lain. Algoritma *stemming* pada penelitian ini digunakan untuk mencari kata dasar Bahasa Indonesia dan Bahasa Jawa.

a. **Stemming Nazief dan Adriani**

Stemming Nazief dan Adriani merupakan algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani untuk mencari kata dasar Bahasa Indonesia. Bentuk kata berafiks Bahasa Indonesia secara umum didefinisikan sebagai berikut.

[DP+[DP+[DP+]]] *root-word* [[+DS][+PP][+P]]

Keterangan:

DP: *Derivation Prefixes*

DS: *Derivation Suffixes*

Inflection Suffixes terdiri atas:

PP: *Possessive Pronouns*

P: *Particles*

Tahapan-tahapan algoritma ini adalah sebagai berikut.

- 1) Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tersebut adalah *root word*. Maka algoritma berhenti.
- 2) *Inflection Suffixes* ("lah", "kah", "ku", "mu", atau "nya") dibuang. Jika berupa *particles* ("lah", "kah", "tah", atau "pun") maka langkah ini diulangi lagi untuk menghapus *Possessive Pronoun* ("ku", "mu", atau "nya"), jika ada. Contoh penghapusan *inflection suffixes*:

kepunyaannya = kepunyaanya + lah

kepunyaannya = kepunyaan + nya

- 3) Hapus *Derivation Suffixes* ("-i" atau "-an"). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a.
 - a) Jika "-an" telah dihapus dan huruf terakhir dari kata tersebut adalah "-k", maka "-k" juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b. Contoh penghapusan *derivation suffixes*:
kepunyaan = kepunya + an
 - b) Akhiran yang dihapus ("-i", "-an", "-kan") dikembalikan, lanjut ke langkah 4.
- 4) Hapus *Derivation Prefix*. Jika pada langkah 3 ada sufiks yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
 - a) Periksa tabel kombinasi awalan-akhiran yang tidak diizinkan (Tabel 1). Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
 - b) For $i = 1$ to 3 , tentukan tipe awalan kemudian hapus awalan. Jika *root word* belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama maka algoritma berhenti.
- 5) Melakukan *recording*.
- 6) Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

Tipe awalan ditentukan melalui langkah-langkah berikut:

1) Jika awalannya adalah: "di-", "ke-", atau "se-" maka tipe awalannya secara berturut-turut adalah "di-", "ke-", atau "se-". Contoh penghapusan tipe awalan *derivation prefix* ("di-", "ke-", atau "se-"):

kepunya = ke + punya

2) Jika awalannya adalah "te-", "me-", "be-", atau "pe-" maka dibutuhkan sebuah proses tambahan untuk menentukan tipe awalannya. Contoh penghapusan tipe awalan *derivation prefix* ("te-", "me-", "be-", atau "pe-"):

tertangkap = ter + tangkap

3) Jika dua karakter pertama buka "di-", "ke-", "se-", "te-", "me-", "be-" atau "pe-" maka algoritma berhenti.

4) Jika tipe awalan adalah "none" maka berhenti. Jika tipe awalannya bukan "none" maka awalan dapat dilihat pada Tabel 2. Hapus awalan jika ditemukan.

Tabel 1. Kombinasi Awalan dan Akhiran yang tidak diizinkan

Awalan yang tidak diizinkan	Akhiran yang tidak diizinkan
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan

Tabel 2. Aturan Pemenggalan Prefiks (Sumber: Santosa, 2016)

No	Format Kata	Penghapusan Prefiks	Contoh	
			Kata Berimbahan	Kata Dasar
1	terV...	ter-V te-rV...	terambil	ambil

No	Format Kata	Penghapusan Prefiks	Contoh	
			Kata Berimbuhan	Kata Dasar
2	terCerV	ter-CerV... dengan C!= 'r'	tergerak	gerak
3	terCP...	ter-CP... dengan C!= 'r' dan P!= 'er'	terhitung	hitung
4	terC ₁ erC ₂	te-C ₁ erC ₂ ... dengan C ₁ != 'r'	terjernih	jernih
5	me{l r w y}V...	me-{l r w y}V...	melihat	lihat
6	mem{b f v}...	mem-{b f v}...	memberi	beri
7	mempe...	mem-pe...	mempercaya	percaya
8	mem{rV V}...	me-m{rV V}... me-p{rV V}	memakai	pakai
9	men{c d j z}...	men-{c d j z}...	mendengar	dengar
10	menV...	me-nV... me-tV	menukar	tukar
11	meng{g h q k}...	meng-{g h q k}...	menghukum	hukum
12	mengV...	meng-V... meng-kV...	mengantuk	kantuk
13	menyV...	meny-sV...	menyapu	sapu
14	berV...	ber-V... be-rV...	berambut	rambut
15	berCAP...	ber-CAP... dengan C!= 'r' dan P!= 'er'	bertenda	tenda
16	berCAerV...	ber-CAerV... dengan C!= 'r'	-	-
17	belajar	bel-ajar	-	-
18	beC ₁ erC ₂	be-C ₁ erC ₂ ... dengan C ₁ != {'r' 'l'}	bekerja	kerja
19	pe{w y}V...	pe-{w y}V...	pewarna	warna
20	perV...	per-V... pe-rV...	perampok	rampok

No	Format Kata	Penghapusan Prefiks	Contoh	
			Kata Berimbuhan	Kata Dasar
21	perCAP	per-CAP... dengan C!=‘r’ dan P!=‘er’	pertajam	tajam
22	perCAerV...	per-CAerV... dengan C!=‘r’	-	-
23	pem{b f V}...	pem-{b f V}...	pembunuhan	bunuhan
24	pem{rV V}	pe-m{rV V}... pe-p{rV V}	pemrogram	program
25	pen{c d j z}	pen-{c d j z}	penjual	Jual
26	penV...	pe-nV... pe-tV...	penabrak	tabrak
27	peng{g h q}...	peng-{g h q}...	penghafal	hafal
28	pengV...	peng-V... peng-kV...	pengukur	ukur
29	penyV...	peny-sV...	penyebar	sebar
30	pelV...	pe-lV... kecuali “pelajar” return “ajar”	pelamar	lamar
31	peCerV...	per-erV... dengan C!={r w y m n}	-	-
32	peCP	pe-CP... dengan C!={r w y m n} dan P!=‘er’	pedagang	dagang
33	peC ₁ erC ₂	pe-C ₁ erC ₂ ... dengan C ₁ !={r w y m n}	pekerja	kerja
Keterangan simbol huruf: C: huruf konsonan V: huruf vokal A: huruf vokal atau konsonan P: partikel atau fragmen dari suatu kata				

b. **Stemming Bahasa Jawa**

Algoritma *stemming* Bahasa Jawa pada penelitian ini menggunakan metode *rule based* (Madia, 2016). Algoritma ini mengubah kata berimbuhan menjadi kata dasar dengan model aturan imbuhan seperti berikut.

[[P+]]Kata Dasar[+K][+S]]

P: Prefiks

K: Konfiks

S: Sufiks

Proses tahapan algoritma ini yaitu:

- 1) *User* melakukan *input* kata berimbuhan bahasa Jawa, kemudian *input* tersebut dicek pada *database*. Jika sudah ada pada *database* maka kata tersebut dianggap sebagai kata dasar dan algoritma *stemming* berhenti. Jika tidak ada pada *database* maka berlanjut pada proses *stemming*.
- 2) Proses ke-2 (proses *stemming*) yaitu proses penghilangan imbuhan. Analisis imbuhan yang terdapat pada proses ini meliputi proses analisis prefiks, konfiks, dan sufiks.
- 3) Hapus prefiks. Jika terdapat adanya imbuhan prefiks, maka proses selanjutnya adalah memisahkan kata dasar dari imbuhan prefiks. Jika imbuhan prefiks tidak ada, maka proses selanjutnya adalah pengecekan imbuhan konfiks. Contoh penghapusan prefiks:

dijupuk 'diambil' = *di* + *jupuk*

- 4) Hapus konfiks. Jika terdapat adanya imbuhan konfiks, maka proses selanjutnya adalah memisahkan kata dasar dari imbuhan konfiks. Tahapan penghapusan konfiks diawali dengan penghapusan prefiks

kemudian dilakukan penghapusan sufiks. Jika imbuhan konfiks tidak ada, maka proses selanjutnya adalah pengecekan imbuhan sufiks. Contoh proses penghapusan konfiks:

- a) Proses penghapusan prefiks: *njupukake* = *n* + *jupukake*
- b) Proses penghapusan sufiks: *jupukake* = *jupuk* + *ake*
- 5) Hapus sufiks. Jika terdapat adanya imbuhan sufiks, maka proses selanjutnya adalah memisahkan kata dasar dari imbuhan sufiks tersebut. Jika imbuhan sufiks tidak ada, maka maka kata terakhir diasumsikan sebagai *root word*. Contoh penghapusan sufiks:

wenehake 'berikan' = *weneh* + *ake*

6. Afiksasi

Menurut Tarigan (1989) afiksasi adalah proses pembubuhan afiks pada suatu satuan, baik satuan itu berupa bentuk tunggal maupun bentuk kompleks, untuk membentuk kata. Proses afiksasi pada penelitian ini mengacu pada aturan *rule based* morfologi berdasarkan hasil penelitian Krishandini (2011) dan penelitian (Asmoko, 2014) yang menyebutkan persamaan afiks Bahasa Jawa dan Bahasa Indonesia.

Tabel 3. Persamaan Afiks Bahasa Jawa dan Bahasa Indonesia (Sumber: Krishandini, 2011)

Afiks Bahasa Jawa	Afiks Bahasa Indonesia
N- (<i>nge-</i> , <i>m-</i> , <i>n-</i> , <i>ng-</i> , <i>ny-</i>)	Me-
di-	di-
Ke-	Ke-
-i	-i
-an	-an
Ka—an	Ke—an
N—i	Me—i

Menurut Asmoko (2014), afiks-afiks pembentuk verba turunan Bahasa Jawa dan Bahasa Indonesia terdiri atas prefiks, simulfiks, dan konfiks. Beberapa padanan afiks Bahasa Jawa dan Bahasa Indonesia adalah sebagai berikut.

- a. Prefiks *N-* Bahasa Jawa berpadanan dengan *meN-* bahasa Indonesia.

Contoh padanan prefiks *N-* dengan prefiks *meN-*:

njupuk 'mengambil'

- b. Prefiks *ke-* Bahasa Jawa berpadanan dengan prefiks *ter-* Bahasa Indonesia. Contoh padanan prefiks *ke-* dengan prefiks *ter-*:

kejupuk 'terambil'

- c. Simulfiks *N—ake* Bahasa Jawa berpadanan dengan *meN—kan* Bahasa Indonesia. Contoh padanan simulfiks *N—ake* dengan simulfiks *meN—kan*:

njupukake 'mengambilkan'

- d. Konfiks *N—i* Bahasa Jawa berpadanan dengan *meN—i* Bahasa Indonesia.

Contoh padanan konfiks *N—i* dengan konfiks *meN—i*:

njupuki 'mengambil'i'

- e. Konfiks *N—ake* Bahasa Jawa berpadanan dengan konfiks *meN—kan* Bahasa Indonesia. Contoh padanan konfiks *N—ake* dengan konfiks *meN—kan*:

ngadolake 'menjualkan'

- f. Konfiks *ke—an* Bahasa Jawa berpadanan dengan konfiks *ke—an* Bahasa Indonesia. Contoh padanan konfiks *ke—an* Bahasa Jawa dengan konfiks *ke—an* Bahasa Indonesia:

kadohan 'kejauhan'

a. Afiksasi Bahasa Indonesia

Proses afiksasi Bahasa Indonesia dibagi menjadi prefiks, infiks, sufiks, dan simulfiks. Pada penelitian ini, proses afiksasi Bahasa Indonesia meliputi:

1) Prefiks me-

Prefiks me- memiliki empat alomorf yaitu mem-, men-, meny-, dan meng- (Ramlan, 1983: 74 dalam Tarigan, 1989).

a) Alomorf mem- dipakai apabila bentuk dasar yang mengikutinya berawal dengan /p, b, f/. Contoh:

me + pakai — memakai

me + baca — membaca

me + fitnah — memfitnah

b) Alomorf men- dipakai apabila bentuk dasar yang mengikutinya berawal dengan /t, d, c, j/. Contoh:

me + tangkap — menangkap

me + dapat — mendapat

me + cangkul — mencangkul

me + jawab — menjawab

c) Alomorf meny- dipakai apabila bentuk dasar yang mengikutinya berawal dengan fonem /s/. Contoh:

me + sapu — menyapu

d) Alomorf meng- dipakai apabila bentuk dasar yang mengikutinya berawal dengan fonem /k, g, x, h, dan vokal/. Contoh:

me + karang — mengarang

me + gambar — menggambar

me + khayal	—	mengkhayal
me + hantam	—	menghantam
me + adu	—	mengadu

2) Prefiks di-

Prefiks di- tidak mengalami perubahan bentuk morfologi apabila kata dasar yang mengikutinya berawal dengan huruf konsonan maupun vokal.

Contoh:

di + pakai	—	dipakai
di + baca	—	dibaca

3) Prefiks ter-

Prefiks ter- tidak mengalami perubahan bentuk morfologi apabila kata dasar yang mengikutinya berawal dengan huruf konsonan maupun vokal.

Contoh:

ter + bawa	—	terbawa
ter + ambil	—	terambil

4) Prefiks ke-

Prefiks ke- tidak mengalami perubahan bentuk morfologi apabila kata dasar yang mengikutinya berawal dengan huruf konsonan maupun vokal.

Contoh:

ke + pakai	—	kepakai
ke + injak	—	keinjak

5) Sufiks –kan

Sufiks –kan tidak mengalami perubahan morfologi apabila kata dasar yang diikutinya berakhir dengan huruf konsonan maupun vokal. Contoh:

membeli + kan — membelikan

mengambil + kan — mengambilkan

6) Sufiks –i

Sufiks –i tidak mengalami perubahan morfologi apabila kata dasar yang diikutinya berakhiran dengan huruf konsonan maupun vokal. Contoh:

mendatang + i — mendatangi

memukul + i — memukuli

7) Sufiks –an

Sufiks –an tidak mengalami perubahan morfologi apabila kata dasar yang diikuti berakhiran dengan huruf konsonan maupun vokal. Contoh:

pukul + an — pukulan

bulan + an — bulanan

b. Afiksasi Bahasa Jawa

Proses afiksasi Bahasa Jawa dilakukan dengan mengimbuhkan awalan, sisipan, akhiran, atau gabungan dari imbuhan-imbuhan itu pada kata dasarnya (Poedjosoedarmo et al, 1979). Proses afiksasi Bahasa Jawa pada penelitian ini meliputi:

1) Awalan N-

Awalan N- mempunyai lima alomorf yaitu: /nge-/; /m-/; /n-/; /ng-/; /ny-/.

a) Alomorf /nge-/ apabila diikuti oleh kata dasar yang bersuku kata satu, seperti:

cet 'cat' — *ngecet* 'mengecat'

bom 'bom' — *ngebom* 'mengebom'

dol 'jual' — *ngedol* 'menjual'

tik '(mesin) tik' — *ngetik* 'mengetik'

bis 'bis' — *ngebis* 'naik bis'

b) Alomorf /m-/ apabila diikuti oleh kata dasar yang dimulai dengan fonem /b, p, w, m/, seperti:

bali 'ulang' — *mbaleni* 'mengulangi'

pacul 'cangkul' — *maculi* 'mencangkul'

weneh 'beri' — *menehi* 'memberi'

maling 'pencuri' — *malingi* 'mencuri'

c) Alomorf /n-/ apabila diikuti oleh kata dasar yang dimulai dengan fonem /t, d, th, dh, n/, seperti:

tabrak 'tabrak' — *nabrak* 'menabrak'

dudut 'tarik' — *ndudut* 'menarik'

thuthuk 'pukul' — *nuthuk* 'memukul'

dhupak 'sepak' — *ndhupak* 'menyepak'

d) Alomorf /ng-/ apabila diikuti oleh kata dasar yang dimulai dengan fonem /k, g, r, l, w/, seperti:

garap 'garap' — *nggarap* 'menggarap'

kukur 'garuk' — *ngukur* 'menggaruk'

yekti 'sungguh' — *ngyektekake* 'membuktikan'

rabi 'kawin' — *ngrabekake* 'mengawinkan'

lali 'lupa' — *nglalekake* 'melupakan'

wadul 'adu' — *ngwadulake* 'mengadu'

obong 'bakar' — *ngobong* 'membakar'

iris 'iris' — *ngiris* 'mengiris'

antem 'hantam' — *ngantem* 'menghantam'

e) Alomorf /ny-/ apabila diikuti oleh kata dasar yang dimulai dengan fonem /s, c, j, ny/, seperti:

sapu 'sapu' — *nyapu* 'menyapu'

cukur 'cukur' — *nyukur* 'mencukur'

jaluk 'minta' — *njaluk* 'meminta'

nyata 'nyata' — *nyatakake* 'membuktikan'

2) Awalan di-

Awalan di- digunakan dalam tingkat tutur *ngoko* dan *madya*. Dalam ragam bahasa krama, awalan di- sering berbentuk dipun- seperti:

ngoko: *dijupuk*

madya: *dipundhut*

krama: *dipunpundhut*

arti: 'diambil'

3) Awalan ke-

Awalan ke- apabila diikuti oleh kata dasar yang dimulai dengan huruf vokal, maka terjadi peluluanan seperti:

ke + ilang 'hilang' + *an* — *kelangan* 'kehilangan'

ke + obong 'bakar' — *kobong* 'kebakar'

ke + udan 'hujan' + *an* — *kudanan* 'kehujanan'

4) Akhiran –i

Akhiran –i mempunyai dua bentuk, yaitu /-i/ dan /-ni/.

a) Bentuk /-i/ dipakai apabila kata dasar yang diikutinya berakhir dengan huruf konsonan, seperti:

njupuk + *i* 'mengambil' — *njupuki* 'mengambil'

nuthuk + *i* 'memukul' — *nuthuki* 'memukuli'

b) Bentuk */-ni/* dipakai apabila kata dasar yang mengikutinya berakhir dengan huruf vokal, seperti:

mara + *ni* 'datang' — *marani* 'mendatangi'

bali + *ni* 'ulang' — *mbaleni* 'mengulangi'

5) Akhiran *-an*

Akhiran *-an* hanya mempunyai satu alomorf saja yaitu */-an/*. Akhiran *-an* dipakai apabila kata dasar yang mengikutinya berakhir dengan huruf konsonan maupun huruf vokal, seperti:

dolan 'main' + *an* — *dolanan* 'mainan'

gawe 'buat' + *an* — *gawean* 'buatan'

6) Akhiran *-ake*

Akhiran *-ake* dipakai dalam tingkat tutur *ngoko* dan *madya*. Dalam tingkat *krama* akhiran itu berbentuk *-aken*. Akhiran *-ake* dan *-aken* masing-masing mempunyai dua alomorf: *-kake* dan *-ake*, serta *-kaken* dan *-kaken*. Akhiran *-kake* dan *-kaken* terdapat pada kata dasar yang berakhir pada huruf vokal, sedangkan *-ake* dan *-aken* mengikuti kata dasar yang berakhir pada konsonan, seperti:

gawe 'buat' + *kake* — *nggawekake* 'membuatkan'

gawa 'bawa' + *kake* — *nggawakake* 'membawakan'

takon 'tanya' + *ake* — *nakokake* 'menanyakan'

jupuk 'ambil' + *ake* — *njupukake* 'mengambilkan'

7. Model Pengembangan

a. Pengembangan Perangkat Lunak

Rosa A.S dan M. Shalahuddin (2014: 26) menjelaskan pengembangan perangkat lunak atau *Software Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak yang sudah teruji dengan baik. SDLC memiliki tahapan-tahapan yang secara umum sebagai berikut:

1) Inisiasi

Pada tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2) Pengembangan konsep sistem

Tahap ini pengembang mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

3) Perencanaan

Tahapan ini pengembang melakukan rencana manajemen proyek dan dokumen perencanaan lainnya.

4) Analisis Kebutuhan

Tahapan ini pengembang melakukan analisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*.

5) Desain

Tahapan ini pengembang mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap. Tahapan ini dilakukan untuk mencari solusi bagaimana sistem dapat memenuhi fungsi-fungsi yang dibutuhkan.

6) Pengembangan

Tahapan pengembangan dilakukan dengan melakukan konversi desain ke dalam sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan.

7) Integrasi dan Pengujian

Tahapan ini dilakukan dengan mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional.

8) Implementasi

Tahapan ini dilakukan dengan melakukan penerapan perangkat lunak pada lingkungan produksi (lingkungan *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9) Operasi dan pemeliharaan

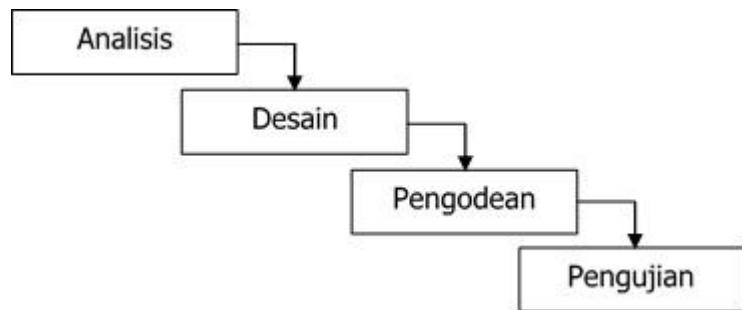
Tahapan ini dilakukan dengan mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan *user*).

10) Disposisi

Tahapan ini dilakukan dengan mendeskripsikan aktivitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktivitas *user*.

b. Model *Waterfall*

Model *waterfall* merupakan salah satu model pengembangan perangkat lunak. Berikut adalah gambar prosedur pengembangan model *waterfall*.



Gambar 5. Prosedur Pengembangan Model *Waterfall*

Menurut Rosa A.S. & Shalahuddin (2014) model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekvensial atau terurut dimulai dari analisis, desain, pengodean, dan tahap pengujian.

1) Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mendapatkan spesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan pengguna. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

2) Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi

program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini perlu didokumentasikan.

3) Pembuatan kode program

Desain yang telah ada ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4) Pengujian

Pengujian fokus pada perangkat lunak secara dari sisi logis dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisasi kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

Pada penelitian ini, aplikasi dikembangkan menggunakan model *waterfall*. Pemilihan model ini dikarenakan kebutuhan pelanggan sudah sangat dipahami serta kemungkinan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil. Selain itu, model *waterfall* menyediakan struktur tahap pengembangan sistem yang jelas serta sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (Rosa A.S. & Shalahuddin, 2014).

8. Kualitas Perangkat Lunak

Di dalam pengembangan perangkat lunak perlu dilakukan pengujian untuk mengetahui kelayakan perangkat lunak yang dikembangkan. Pengujian diperlukan untuk menemukan segala kesalahan dan segala kemungkinan yang akan menimbulkan kesalahan sesuai dengan spesifikasi perangkat lunak yang telah ditentukan sebelum disebarluaskan kepada pengguna (Simarmata, 2010).

Simarmata (2010: 261) menjelaskan kualitas perangkat lunak sebagai keberadaan karakteristik dari suatu produk yang dibabarkan dalam kebutuhannya. Terdapat beberapa model kualitas diantaranya model kualitas McCall, Boehm, FURPS, Dromey, ISO 9126 dan ISO 25010 (Miguel, Mauricio, & Rodriguez, 2014). Beberapa model kualitas tersebut dibabarkan pada Tabel 4.

Tabel 4. Perbandingan Model Kualitas Perangkat Lunak

Karakteristik	McCall	Boehm	FURPS	Dromey	ISO-9126	ISO-25010
<i>Accuracy</i>					X	X
<i>Adaptability</i>			X			X
<i>Analyzability</i>					X	X
<i>Attractiveness</i>					X	X
<i>Changeability</i>					X	X
<i>Correctness</i>	X					X
<i>Efficiency</i>	X	X		X	X	X
<i>Flexibility</i>	X					
<i>Functionality</i>			X	X	X	X
<i>Human Engineering</i>		X				
<i>Installability</i>					X	X
<i>Integrity</i>	X					X
<i>Interoperability</i>	X					X
<i>Maintainability</i>	X			X	X	X
<i>Maturity</i>					X	X
<i>Modifiability</i>						X
<i>Operability</i>					X	X
<i>Performance</i>			X		X	X
<i>Portability</i>	X	X		X	X	X
<i>Reliability</i>	X	X	X	X	X	X
<i>Resource utilization</i>					X	X
<i>Reusability</i>	X			X		X
<i>Stability</i>					X	X
<i>Suitability</i>					X	X
<i>Supportability</i>			X		X	X
<i>Testability</i>	X	X			X	X
<i>Transferability</i>						X
<i>Understandability</i>		X			X	X
<i>Usability</i>	X		X	X	X	X

Berdasarkan Tabel 4, model kualitas ISO 25010 merupakan model yang memiliki kriteria kualitas paling lengkap. Pada penelitian ini menggunakan model kriteria ISO 25010 dikarenakan dapat menguji perangkat lunak dari berbagai karakteristik.

Model kualitas ISO 25010 memiliki delapan karakteristik untuk menguji kualitas produk. Delapan karakteristik tersebut dijabarkan pada Tabel 5.

Tabel 5. Model Kualitas Produk ISO 25010

No.	Karakteristik	Sub Karakteristik
1	<i>Functional suitability</i>	<i>Functional completeness</i>
		<i>Functional correctness</i>
		<i>Functional appropriateness</i>
2	<i>Performance efficiency</i>	<i>Time behaviour</i>
		<i>Resource utilization</i>
		<i>Capacity</i>
3	<i>Compatibility</i>	<i>Co-existence</i>
		<i>Interoperability</i>
4	<i>Usability</i>	<i>Appropriateness recognizability</i>
		<i>Learnability</i>
		<i>Operability</i>
		<i>User error protection</i>
		<i>User interface aesthetics</i>
5	<i>Reliability</i>	<i>Maturity</i>
		<i>Availability</i>
		<i>Fault tolerance</i>
		<i>Recoverability</i>
6	<i>Security</i>	<i>Confidentiality</i>
		<i>Integrity</i>
		<i>Non-repudiation</i>
		<i>Accountability</i>
		<i>Authenticity</i>
7	<i>Maintainability</i>	<i>Modularity</i>
		<i>Reusability</i>
		<i>Analyzability</i>
		<i>Modifiability</i>
		<i>Testability</i>
8	<i>Portability</i>	<i>Adaptability</i>
		<i>Installability</i>
		<i>Replaceability</i>

Berdasarkan karakteristik ISO 25010, penelitian ini hanya menggunakan lima aspek pengujian. Aspek yang digunakan yaitu *functional suitability*, *performance efficiency*, *compatibility*, *usability*, dan *reliability*. Aspek *security*, *maintainability*, dan *portability* tidak diujikan pada penelitian ini.

Aspek *security* berkaitan dengan pengujian keamanan sebuah aplikasi sedangkan aplikasi yang akan dikembangkan tidak berkaitan dengan keamanan perangkat lunak sehingga aspek *security* tidak dapat diuji. Motogna (2016) mempresentasikan cara pengujian *maintainability* dengan membandingkan versi yang berbeda pada sebuah perangkat lunak. Perangkat lunak pada penelitian ini dikembangkan dari awal dan tidak memiliki versi pengembangan lain sehingga pengujian *maintainability* tidak dapat diukur. Brown (2003) dalam Franke (2012) menyatakan perangkat lunak dikatakan *portable (portability)* apabila perangkat lunak yang dibuat dapat dijalankan lebih dari satu *platform*. Pada penelitian ini, perangkat lunak yang dikembangkan terbatas pada *platform* berbasis Android sehingga pengujian *portability* tidak cocok.

a. *Functional Suitability*

Karakteristik sebuah produk atau sistem mampu menyediakan fungsi yang memenuhi kebutuhan ketika digunakan dalam kondisi tertentu. Karakteristik ini dibagi menjadi tiga subkarakteristik sebagai berikut.

- 1) *Functional completeness*, yaitu karakteristik sejauh mana dalam menyediakan fungsi dapat mencakup semua tugas dan tujuan pengguna secara spesifik.

- 2) *Functional correctness*, yaitu karakteristik sejauh mana produk atau sistem mampu menyediakan hasil yang tepat sesuai dengan kebutuhan.
- 3) *Functional appropriateness*, yaitu karakteristik sejauh mana fungsi yang disediakan mampu menyelesaikan tugas dan tujuan secara spesifik.

Pengujian aspek *functionality* dilakukan dengan menghitung jumlah fitur yang ada pada aplikasi kemudian dibandingkan dengan jumlah fitur yang berhasil dijalankan (Niknejad, 2011). Hasil pengujian kemudian dianalisis menggunakan metode statistik deskriptif.

b. *Performance Efficiency*

Karakteristik sejauh mana kemampuan kerja sebuah aplikasi terhadap sumber daya digunakan dalam kondisi tertentu. Karakteristik ini dibagi menjadi tiga subkarakteristik sebagai berikut.

- 1) *Time behaviour*, yaitu karakteristik sejauh mana respon dan lama proses sebuah produk atau sistem saat menjalankan fungsinya sesuai dengan kriteria.
- 2) *Resource utilization*, yaitu karakteristik sejauh mana jumlah dan jenis sumber daya yang digunakan produk atau sistem saat menjalankan fungsinya sesuai dengan kriteria.
- 3) *Capacity*, yaitu karakteristik sejauh mana batas maksimal sebuah produk atau sistem mampu memenuhi kriteria.

Pengujian *performance* menurut David (2011) dilakukan dengan menghitung penggunaan memori/CPU, konsumsi baterai, dan beban pada server pada berbagai kondisi. Oleh karena itu, pengujian yang akan dilakukan meliputi *time behaviour*, *resource utilization CPU*, dan *resource utilization*

memory. Menurut Álvarez dalam Testing Experience (2012) pengujian *performance* dilakukan untuk mengetahui kerusakan yang terjadi pada aplikasi seperti *crash* dan *memory leak*. Pengujian *performance efficiency* menggunakan *tools* yang disediakan dari <https://www.monkop.com/>.

c. *Compatibility*

Karakteristik sebuah produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen lain dan/atau dapat menjalankan fungsi yang dibutuhkan secara bersamaan ketika berbagi *environment* perangkat keras atau perangkat lunak yang sama. Karakteristik ini dibagi menjadi dua subkarakteristik sebagai berikut.

- 1) *Co-existence*, yaitu karakteristik sejauh mana produk atau sistem dapat menjalankan fungsi yang dibutuhkan secara efisien saat berbagi sumber daya dengan produk atau sistem lain tanpa memberi dampak terhadap produk atau sistem tersebut.
- 2) *Interoperability*, yaitu karakteristik sejauh mana dua atau lebih sistem, produk atau komponen dapat bertukar informasi dan menggunakan informasi tersebut.

Menurut David (2011) pengujian *compatibility* dilakukan dengan menguji aplikasi pada berbagai macam *browser*, sistem operasi, jenis perangkat, ukuran perangkat, variasi kecepatan koneksi, pergantian teknologi, dan berbagai standar protokol. Pengujian *compatibility* meliputi aspek *co-existence* serta dilakukan dengan menguji coba instalasi aplikasi pada berbagai sistem operasi dan jenis perangkat yang disediakan dari <https://www.monkop.com/>. Hasil pengujian kemudian dianalisis menggunakan metode statistik deskriptif.

d. *Usability*

Karakteristik sejauh mana sebuah produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu secara efektif, efisien dan memuaskan dalam konteks penggunaan. Karakteristik ini dibagi menjadi enam subkarakteristik sebagai berikut.

- 1) *Appropriateness recognizability*, yaitu karakteristik sejauh mana pengguna mengetahui sebuah produk atau sistem sesuai dengan kebutuhan mereka.
- 2) *Learnability*, yaitu karakteristik sejauh mana produk atau sistem dapat digunakan pengguna tertentu untuk mencapai tujuan mempelajari sebuah produk atau sistem secara efektif, efisien dan bebas dari resiko serta memenuhi kepuasan dalam konteks penggunaan.
- 3) *Operability*, yaitu karakteristik sejauh mana produk atau sistem mudah dioperasikan dan dikontrol.
- 4) *User error protection*, yaitu karakteristik sejauh mana produk atau sistem melindungi pengguna dalam melakukan kesalahan.
- 5) *User interface aesthetics*, yaitu karakteristik sejauh mana tampilan antarmuka memenuhi kesenangan dan kepuasan pengguna.
- 6) *Accessibility*, yaitu karakteristik sejauh mana produk atau sistem dapat digunakan oleh pengguna secara luas untuk mencapai tujuan tertentu dalam konteks penggunaan.

Pengujian *usability* menggunakan kuesioner *Computer System Usability Questionnaire* oleh Lewis (1995) yang telah dipublikasikan pada *International Journal of Human Computer Interaction*. Pemilihan kuesioner ini dikarenakan kuesioner ini telah mencakup subkarakteristik *usability* ISO 25010.

e. *Reliability*

Karakteristik sejauh mana sistem, produk, atau komponen dapat menjalankan fungsinya dalam kondisi dan waktu tertentu. Karakteristik ini dibagi menjadi empat subkarakteristik sebagai berikut.

- 1) *Maturity*, yaitu karakteristik sejauh mana sistem, produk, atau komponen memenuhi kriteria reliabilitas dibawah kondisi normal.
- 2) *Availability*, yaitu karakteristik sejauh mana sistem, produk, atau komponen.
- 3) *Fault tolerance*, yaitu karakteristik sejauh mana sistem, produk atau komponen.
- 4) *Recoverability*, yaitu karakteristik ketika produk atau sistem mengalami kejadian atau kegagalan dapat mengembalikan data dan menjalankan kondisi sistem yang diharapkan.

Huang, Ai, & Wang (2014) menyebutkan pendekatan pengujian *reliability* adalah dengan menjalankan fungsionalitas perangkat lunak di bawah kondisi normal ketika dilakukan *stress test*. Oleh karena itu, pengujian aspek *reliability* dilakukan dengan menggunakan *tools* yang disediakan <https://testobject.com/>. Pengujian *stress testing* dijalankan dengan melakukan *click* dan *touch* secara acak fungsi-fungsi pada aplikasi.

B. Hasil Penelitian yang Relevan

Beberapa penelitian relevan yang telah dilakukan antara lain:

1. Penelitian yang dilakukan Intan Nur Farida dengan judul "*Applikasi Kamus Indonesia - Jawa Berbasis Android*". Tujuan penelitian ini adalah

menghasilkan aplikasi kamus Bahasa Jawa berbasis Android. Penelitian ini menghasilkan aplikasi kamus Bahasa Jawa yang mampu menerjemahkan Bahasa Indonesia ke Bahasa Jawa *ngoko*, *krama madya*, dan *krama inggil*. Kekurangan aplikasi dari penelitian ini adalah kamus hanya dapat dilakukan untuk menerjemahkan Bahasa Indonesia ke Bahasa Jawa. Kekurangan lain adalah penerjemahan kamus masih sebatas penerjemahan kata. Sedangkan aplikasi yang akan dikembangkan dapat melakukan penerjemahan baik dari Bahasa Indonesia maupun Bahasa Jawa sesuai saran pengembangan dari penelitian yang dilakukan Intan Nur Farida.

2. Penelitian yang dilakukan Irfan Bawa Suteja dengan judul "*Applikasi Translator Bahasa Indonesia-Bahasa Jawa dengan Unggah Ungguh Basa Berbasis Web*". Penelitian ini dilakukan tahun 2012 yang bertujuan untuk mengembangkan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa. Penelitian ini menghasilkan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa dengan berbasis *web* beserta *unggah-ungguh basa*. Kekurangan dari penelitian ini adalah kemampuan komputasi untuk melakukan penerjemahan tergantung dari kecepatan serta daya dukung akses koneksi maupun *server* yang digunakan. Kekurangan lain ketika diakses melalui *smartphone*, aplikasi tersebut memiliki ketidaksesuaian *user interface* serta tidak dapat melakukan penerjemahan dengan baik saat diakses melalui *browser* seperti *opera mini*. Sedangkan pada penelitian ini, aplikasi dikembangkan tanpa membutuhkan akses koneksi dan berjalan secara *offline*. Aplikasi juga akan dikembangkan dengan *user*

interface serta fungsi-fungsinya dapat berjalan dengan baik pada *platform* Android. Saran dari penelitian ini adalah pengembangan aplikasi dibuat untuk perangkat *mobile* Android.

3. Penelitian tentang "Pembuatan Kamus Elektronik Kalimat Bahasa Indonesia dan Bahasa Jawa untuk Aplikasi Mobile Menggunakan Interpolation Search" yang dilakukan Nur Afifah (2010). Penelitian ini menghasilkan kamus penerjemah Bahasa Indonesia dan Bahasa Jawa berbasis kata dan kalimat menggunakan metode *interpolation search* untuk melakukan pencarian arti kata. Aplikasi ini dikembangkan pada perangkat *mobile* berbasis J2ME.

C. Kerangka Pikir

Bahasa Jawa merupakan salah satu warisan kebudayaan Indonesia. Oleh karena itu diperlukan upaya pelestarian agar tetap menjadi bagian dari kekayaan budaya Indonesia. Salah satu upaya pelestarian Bahasa Jawa melalui pembelajaran Bahasa Jawa pada jenjang formal. Topik bahasan Bahasa Jawa salah satunya adalah materi tentang *unggah-ungguh basa*. Siswa SMK Muhammadiyah 2 Yogyakarta masih merasa kebingungan dalam pemilihan *unggah-ungguh* Bahasa Jawa. Di lain sisi, banyak siswa yang memiliki perangkat *smartphone* namun kurang dimanfaatkan sebagai penunjang pembelajaran dan hanya digunakan untuk hiburan.

Telah ada penelitian relevan tentang *unggah-ungguh* Bahasa Jawa yang mampu menghasilkan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa. Penelitian tersebut memiliki beberapa kekurangan dan diberikan saran

pengembangan lebih lanjut. Aplikasi yang akan dikembangkan diharapkan dapat menutupi kekurangan tersebut.

Proses pengembangan dimulai dengan analisis berupa analisis isi program, analisis *hardware*, dan analisis *software*. Proses selanjutnya dilakukan desain aplikasi yang dikembangkan. Desain aplikasi meliputi desain UML (*Unified Modelling Language*), desain sistem dengan bantuan *flowchart* dan desain *interface*. Langkah selanjutnya dilakukan implementasi desain ke dalam bentuk kode program. Setelah melewati proses implementasi dilakukan pengujian aplikasi. Aspek pengujian meliputi faktor kualitas *functional suitability, performance efficiency, compatibility, usability, reliability*, dan materi. Faktor kualitas yang digunakan mengacu pada standar ISO 25010 serta instrumen pengujian materi yang disesuaikan dengan spesifikasi aplikasi yang dikembangkan.

D. Pertanyaan Penelitian

Berdasarkan tujuan penelitian, pertanyaan penelitian dijabarkan sebagai berikut:

1. Bagaimana cara mengembangkan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) dengan *unggah-ungguh basa* pada *platform* Android ?
2. Bagaimana tingkat kelayakan aplikasi penerjemah Bahasa Indonesia-Bahasa Jawa (Ukara Jawa) berdasarkan aspek *functionality suitability, performance efficiency, compatibility, usability, reliability*, dan materi ?

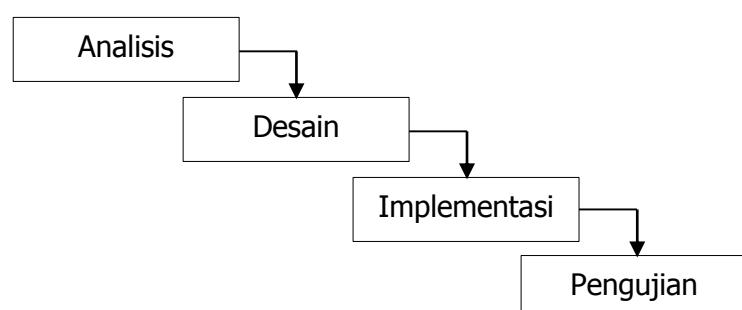
BAB III

METODE PENELITIAN

A. Model Pengembangan

Metode penelitian pada penelitian ini menggunakan metode penelitian dan pengembangan atau *Research and Development* (R&D). Metode R&D merupakan metode penelitian yang digunakan untuk menghasilkan suatu produk dan menguji efektivitas produk tersebut (Sugiyono dalam Krisnayuni, 2017).

Penerapan metode R&D memerlukan model pengembangan yang menggambarkan langkah pengembangan secara sistematis. Model pengembangan pada penelitian ini menggunakan model *waterfall*. Tahapan model SDLC *waterfall* dimulai dari analisis, desain, pengodean, hingga tahap pengujian (Rosa A. S. dan M. Shalahuddin, 2014: 29). Tahapan pengodean pada penelitian ini disebut dengan tahapan implementasi karena dilakukan implementasi desain ke dalam kode program. Tahapan model *waterfall* digambarkan sebagai berikut:



Gambar 6. Model Pengembangan *Waterfall*

B. Prosedur Pengembangan

Prosedur pengembangan pada penelitian ini menggunakan tahapan *waterfall* yang meliputi tahap analisis, desain, implementasi, dan pengujian.

1. Analisis

Analisis dilakukan sebagai langkah awal peneliti untuk mengetahui kebutuhan-kebutuhan yang diperlukan pengguna. Analisis dilakukan peneliti dengan kajian literatur dan observasi. Kajian literatur dilakukan untuk mengetahui konsep dan teori dasar sebagai landasan isi materi aplikasi. Observasi dilakukan dengan mengacu pada aplikasi yang telah ada dan wawancara dengan guru Bahasa Jawa untuk mendapatkan kriteria aplikasi yang akan dikembangkan. Dari hasil observasi dan wawancara didapatkan pendekatan bahwa:

- a) Aplikasi Ukara Jawa dapat digunakan untuk mendukung pembelajaran *unggah-ungguh basa*.
- b) Aplikasi dapat menyajikan arti penerjemahan.
- c) Aplikasi memiliki fitur pencarian seperti tombol pencarian.
- d) Aplikasi memiliki fitur tukar bahasa asal dan bahasa tujuan penerjemahan.
- e) Aplikasi memiliki fitur hapus masukan penerjemahan.
- f) Aplikasi memiliki fitur salin arti terjemahan.
- g) Perlu diberikan bantuan penggunaan pada aplikasi untuk mempermudah pengguna dalam menerapkan *unggah-ungguh basa*.

2. Desain

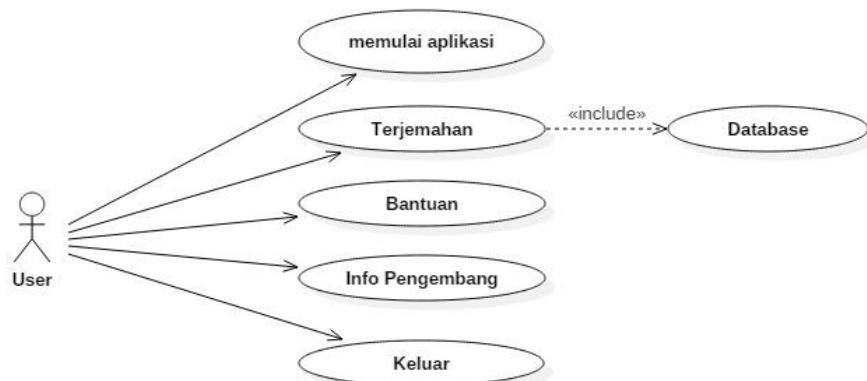
Tahap desain dilakukan untuk menggambarkan aplikasi yang akan dikembangkan. Tahapan desain dilaksanakan berdasarkan hasil analisis aplikasi. Desain yang dilakukan meliputi desain UML, desain sistem, dan desain *interface*. Desain sistem digunakan untuk menggambarkan alur penerjemahan dengan bantuan *flowchart* dan perancangan basis data. Desain *interface* dilakukan untuk menggambarkan tampilan antarmuka aplikasi.

a. Desain *Unified Modelling Language* (UML)

Tahapan desain UML meliputi desain *use case diagram* dan *sequence diagram*.

1) *Use Case Diagram*

Use Case merupakan pemodelan untuk memodelkan kelakuan sistem yang akan dibuat. *Use case diagram* terdiri dari sebuah aktor dan interaksi yang dilakukannya. *Use case* menggambarkan tentang hubungan aktor dengan sistem. Diagram *use case* aplikasi Ukara Jawa digambarkan pada Gambar 7.



Gambar 7. *Use Case* Aplikasi Ukara Jawa

Use Case Diagram di atas dijabarkan menjadi definisi aktor, definisi *use case*, dan skenario *use case*.

a) Definisi Aktor

Tabel 6. Definisi Aktor

No	Aktor	Deskripsi
1	<i>User</i>	<i>User</i> merupakan pengguna yang dapat mengakses halaman terjemahan, halaman bantuan, halaman info pengembang, dan menu keluar.

b) Definisi *Use Case*

Tabel 7. Definisi *Use Case*

No	<i>Use Case</i>	Deskripsi
1	Memulai Aplikasi	Proses awal ketika <i>user</i> menjalankan aplikasi, pada tahapan awal dilakukan inisialisasi <i>database</i> dan menampilkan halaman utama
2	Terjemahan	Merupakan fitur untuk menampilkan halaman penerjemahan.
3	Bantuan	Merupakan fitur untuk menampilkan bantuan penggunaan aplikasi.
4	Info Pengembang	Merupakan fitur untuk menampilkan tentang pengembang aplikasi.
5	Keluar	Merupakan proses ketika user mengakhiri penggunaan aplikasi.

c) Skenario *Use Case*

(1) Nama *Use Case* : Memulai Aplikasi
 Kondisi awal : Membuka Aplikasi Ukara Jawa
 Kondisi akhir : Menampilkan halaman menu utama aplikasi

Tabel 8. Skenario *Use Case* Memulai Aplikasi

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. <i>User</i> menjalankan aplikasi	
	2. Menjalankan <i>splash screen</i> dan melakukan inisialisasi <i>database</i>
	3. Menampilkan halaman menu utama aplikasi

(2) Nama *Use Case* : Terjemahan

Kondisi awal : Membuka Menu Terjemahan

Kondisi akhir : Menampilkan hasil terjemahan

Tabel 9. Skenario *Use Case* Terjemahan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. <i>User</i> menekan menu Terjemahan	
	2. Menampilkan halaman menu Terjemahan
3. Memasukkan <i>input</i> yang akan diterjemahkan	
4. Menekan tombol pencarian	
	5. Mengecek <i>database</i> kosakata berdasarkan <i>input</i>
	6. Jika data yang dimasukkan ditemukan pada <i>database</i> , maka data akan ditampilkan
Alternatif alur 5a. Jika <i>input</i> tidak ditemukan maka akan dilakukan proses <i>stemming</i> .	

Aksi Aktor	Reaksi Sistem
Alternatif alur 6a. Jika data <i>input</i> hasil <i>stemming</i> ditemukan, maka dilakukan restrukturisasi data berdasarkan perbandingan kesamaan afiks.	
Alternatif alur 6b. Jika data <i>input</i> hasil <i>stemming</i> tidak ditemukan, maka sistem akan menampilkan "Terjemahan tidak ditemukan"	

(3) Nama *Use Case* : Bantuan

Kondisi awal : Membuka menu Bantuan

Kondisi akhir : Menampilkan halaman Bantuan

Tabel 10. Skenario *Use Case* Bantuan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. <i>User</i> menekan menu Bantuan	
	2. Menampilkan halaman bantuan penggunaan

(4) Nama *Use Case* : Info Pengembang

Kondisi awal : Membuka menu Info Pengembang

Kondisi akhir : Menampilkan halaman Info Pengembang

Tabel 11. Skenario *Use Case* Info Pengembang

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. <i>User</i> menekan menu Info Pengembang	
	2. Menampilkan halaman identitas pengembang aplikasi

(5) Nama <i>Use Case</i>	: Keluar
Kondisi awal	: Menekan menu Keluar
Kondisi akhir	: Mengakhiri penggunaan aplikasi

Tabel 12. Skenario *Use Case* Keluar

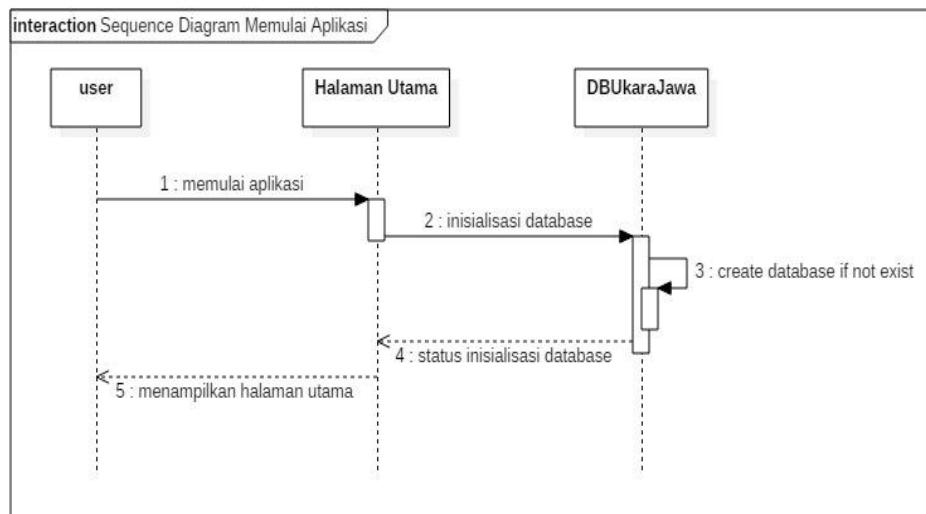
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. <i>User</i> menekan menu Keluar	
	2. Melakukan proses untuk mengakhiri penggunaan aplikasi

2) *Sequence Diagram*

Sequence diagram adalah diagram yang menggambarkan interaksi antar objek yang menunjukkan rangkaian pesan dan juga digunakan untuk menggambarkan skenario. *Sequence diagram* terdiri atas dimensi vertikal yang berupa waktu yang dimulai dari atas dan berjalan ke bawah sampai bagian dasar dari diagram, sedangkan dimensi horizontal yang berupa objek-objek yang terkait diurutkan dari kiri ke kanan (Kendall & Kendall, 2003). *Sequence diagram* aplikasi Ukara Jawa meliputi *sequence diagram* memulai aplikasi, *sequence diagram* terjemahan, *sequence diagram* bantuan, *sequence diagram* info pengembang, dan *sequence diagram* keluar.

a) *Sequence Diagram* Memulai Aplikasi

Sequence diagram memulai aplikasi menggambarkan rangkaian pesan yang dikirim dan diterima antar objek. Berikut adalah *sequence diagram* memulai aplikasi Ukara Jawa.

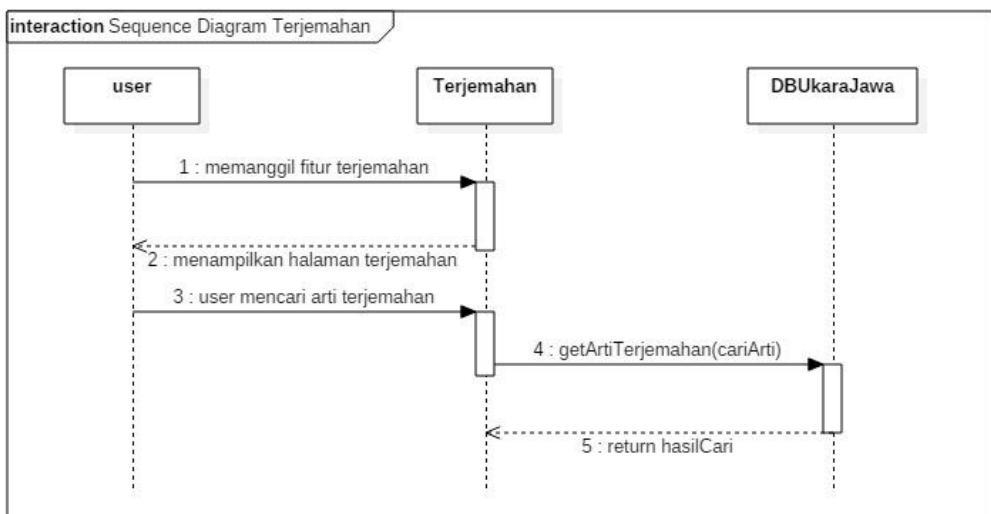


Gambar 8. *Sequence Diagram Memulai Aplikasi*

Sequence diagram memulai aplikasi diawali dengan *user* menjalankan aplikasi. Ketika aplikasi dijalankan, sistem melakukan inisialisasi untuk pengecekan *database*. Apabila *database* tidak ditemukan, sistem melakukan pembuatan *database*. Kemudian sistem menampilkan halaman utama kepada *user*.

b) *Sequence Diagram* Terjemahan

Berikut adalah *sequence diagram* melakukan proses Terjemahan.

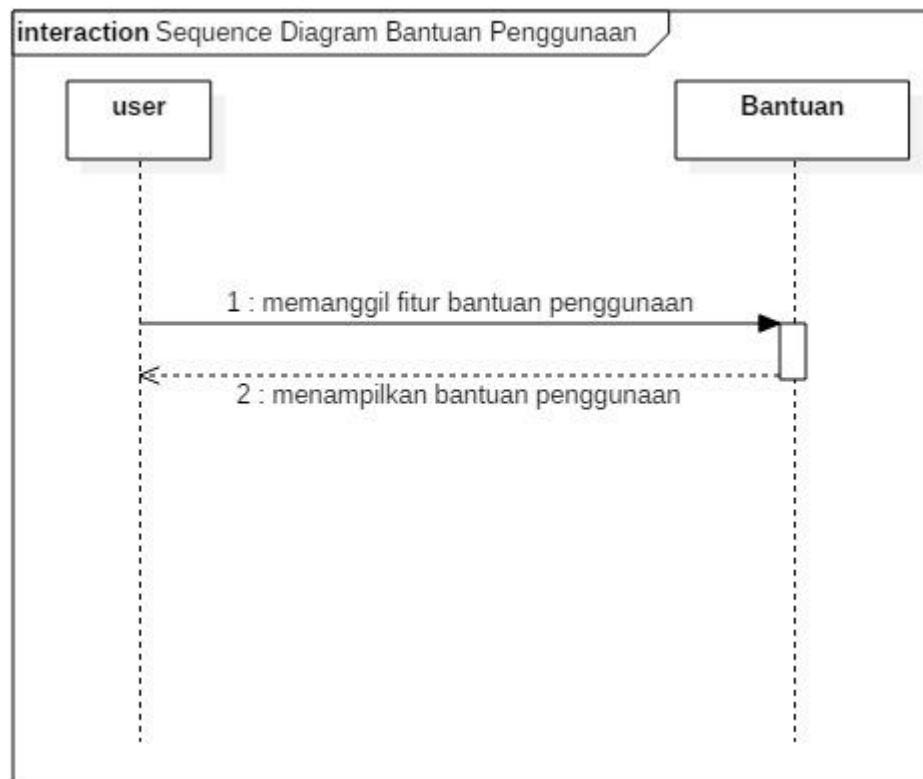


Gambar 9. *Sequence Diagram* Terjemahan

Sequence diagram terjemahan diawali dengan *user* memanggil fitur terjemahan dan sistem menampilkan halaman terjemahan. Ketika *user* melakukan pencarian arti terjemahan, sistem melakukan proses pencarian terjemahan pada *database*. Setelah dilakukan proses pencarian, sistem menampilkan hasil arti terjemahan.

c) *Sequence Diagram* Bantuan

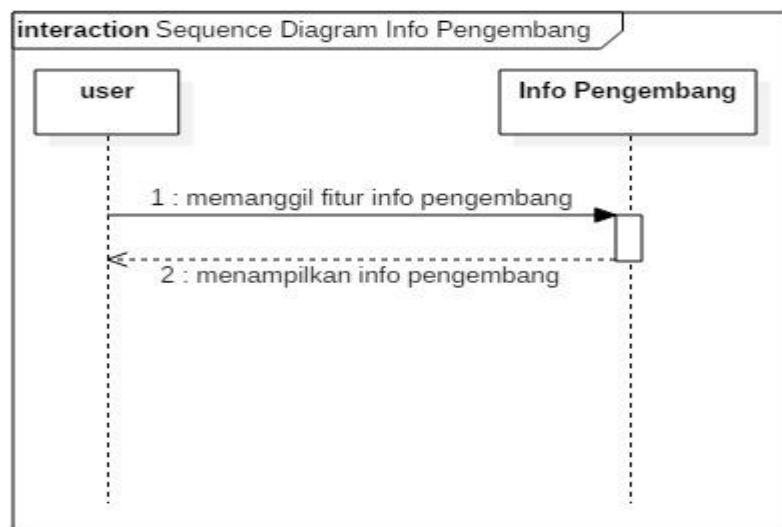
Sequence diagram bantuan menampilkan proses rangkaian pesan yang dikirim objek *user* kepada objek bantuan. Berikut adalah *sequence diagram* bantuan.



Gambar 10. *Sequence Diagram* Bantuan

d) *Sequence Diagram Info Pengembang*

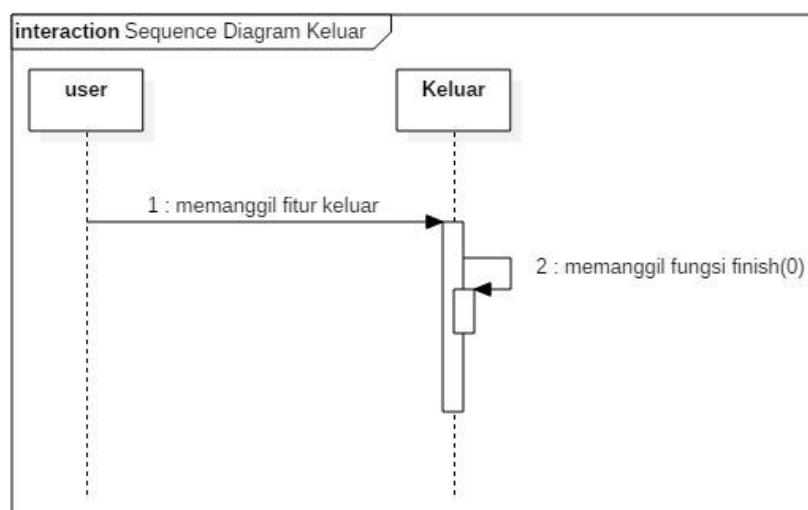
Sequence diagram info pengembang menampilkan proses yang dikirim objek *user* kepada objek info pengembang. Berikut adalah *sequence diagram* info pengembang.



Gambar 11. *Sequence Diagram* Info Pengembang

e) *Sequence Diagram Keluar*

Sequence diagram keluar menampilkan proses antar objek *user* dan objek keluar. Berikut adalah *sequence diagram* keluar.



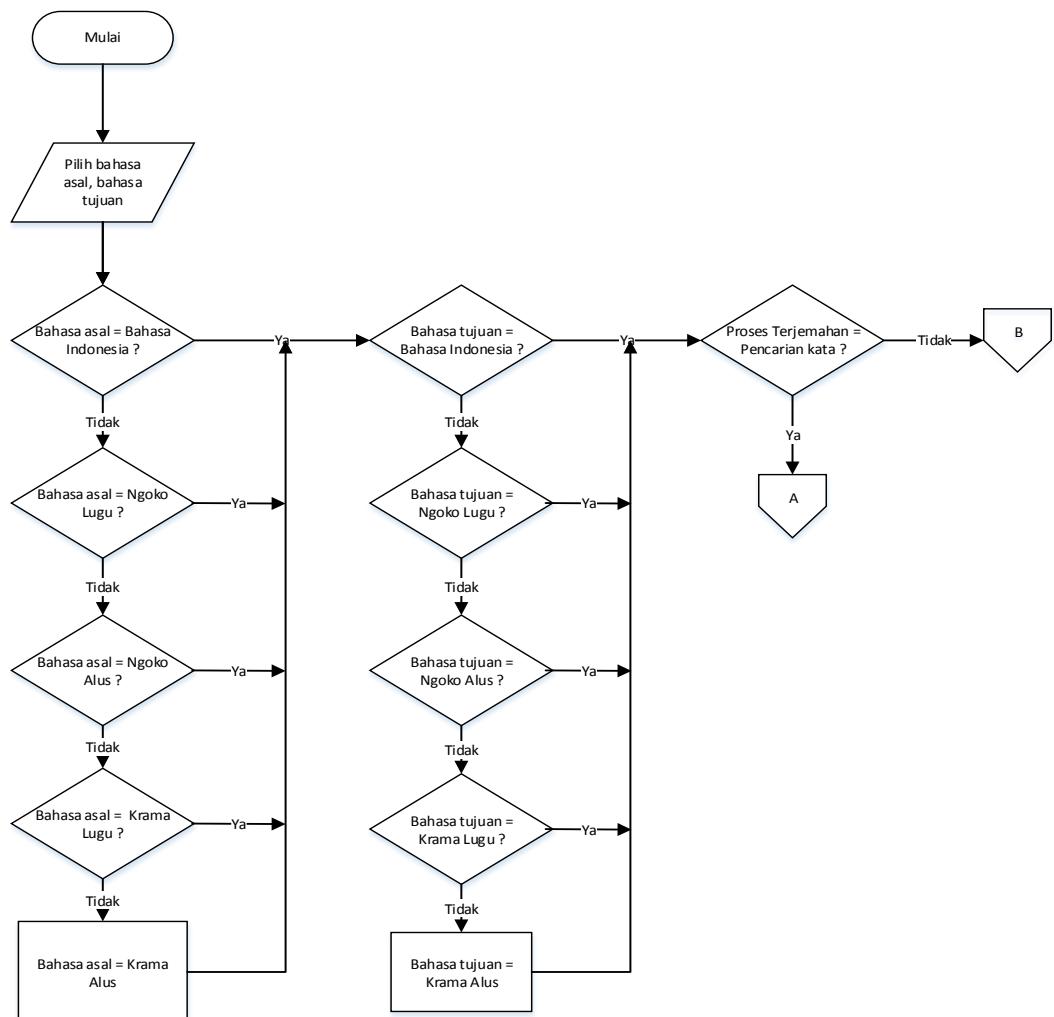
Gambar 12. *Sequence Diagram* Keluar

b. Desain Sistem

Perancangan desain sistem dalam pengembangan aplikasi Ukara Jawa dilaksanakan untuk menggambarkan alur proses penerjemahan dengan *flowchart* serta perancangan basis data pada aplikasi. Perancangan desain sistem juga digunakan sebagai alat dokumentasi.

1) *Flowchart* pemilihan bahasa terjemahan

Flowchart untuk menggambarkan proses pemilihan bahasa terjemahan yang ditunjukkan pada Gambar 13.

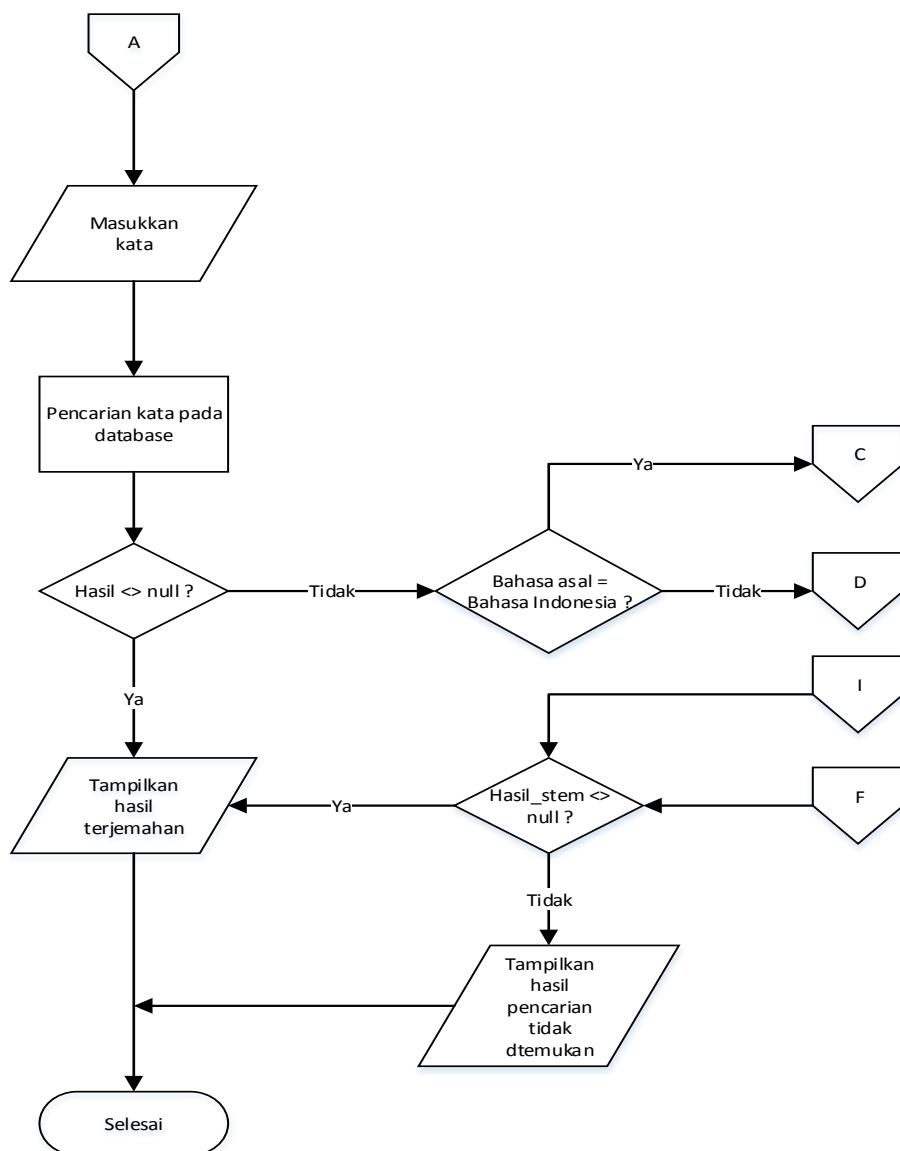


Gambar 13. *Flowchart* pemilihan bahasa terjemahan

2) *Flowchart* pencarian terjemahan

a) *Flowchart* pencarian arti kata

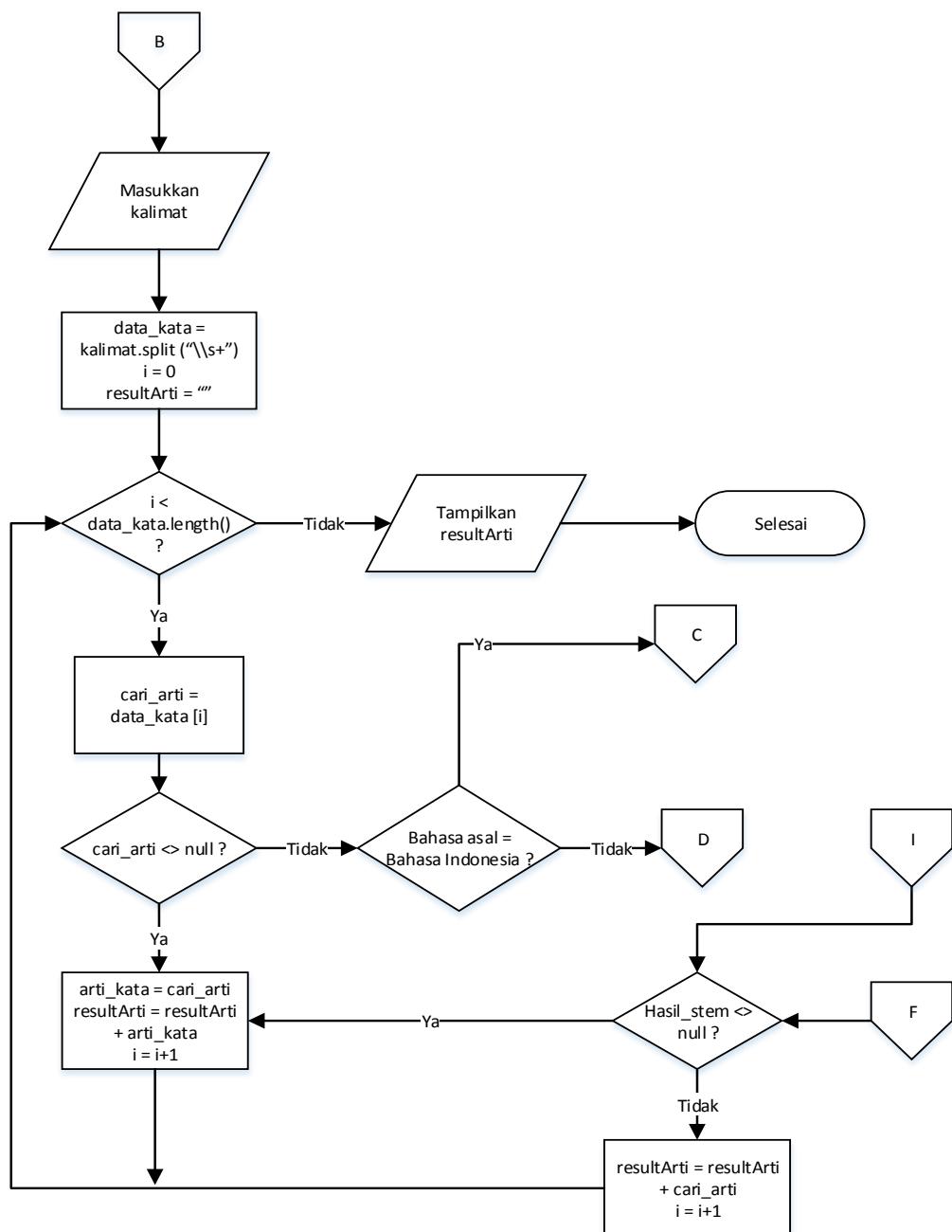
Flowchart untuk menggambarkan proses pencarian terjemahan berdasarkan kata yang ditunjukkan pada Gambar 14.



Gambar 14. *Flowchart* pencarian arti kata

b) *Flowchart* pencarian arti kalimat

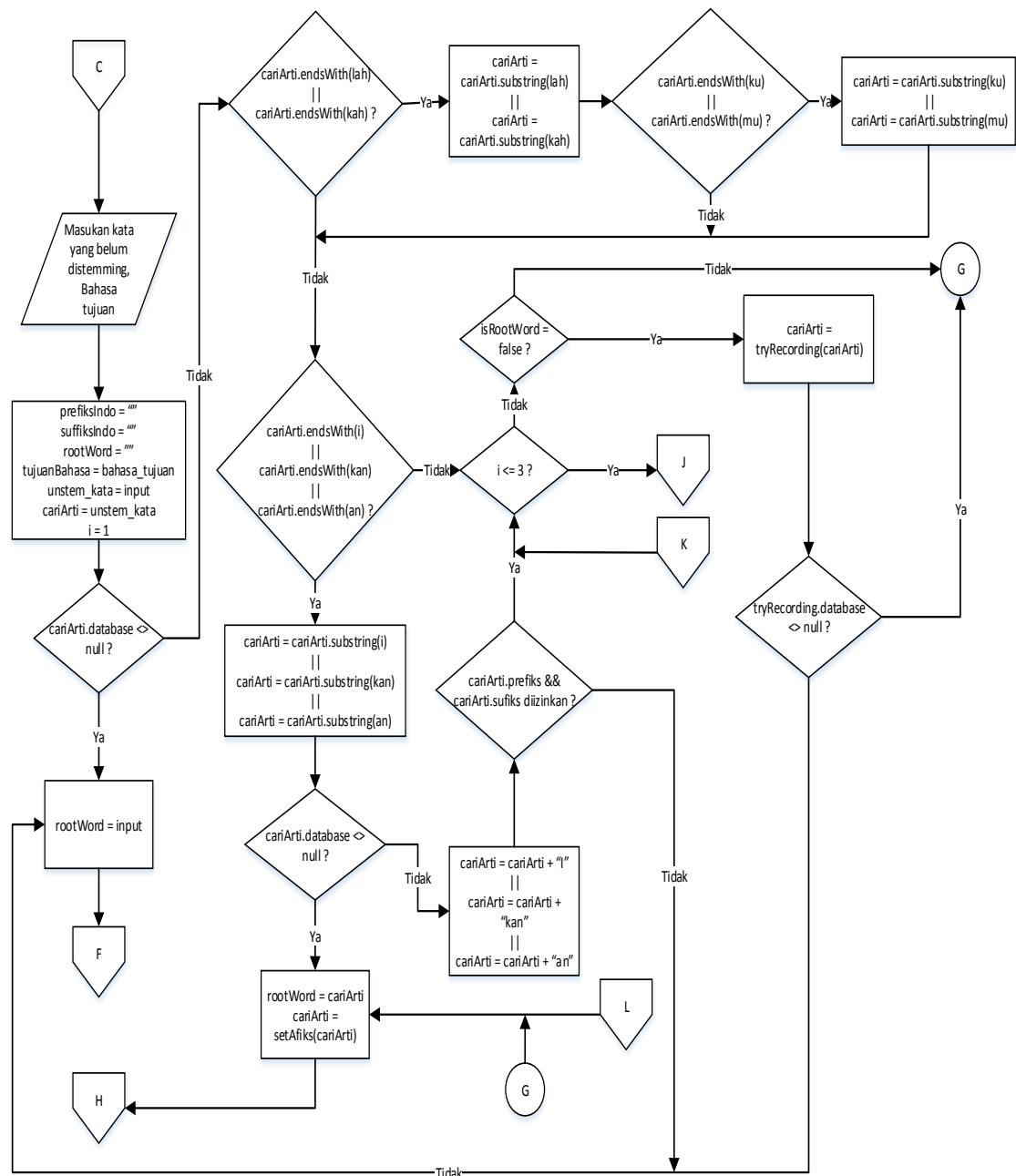
Flowchart untuk menggambarkan proses pencarian terjemahan berdasarkan kalimat yang ditunjukkan pada Gambar 15.



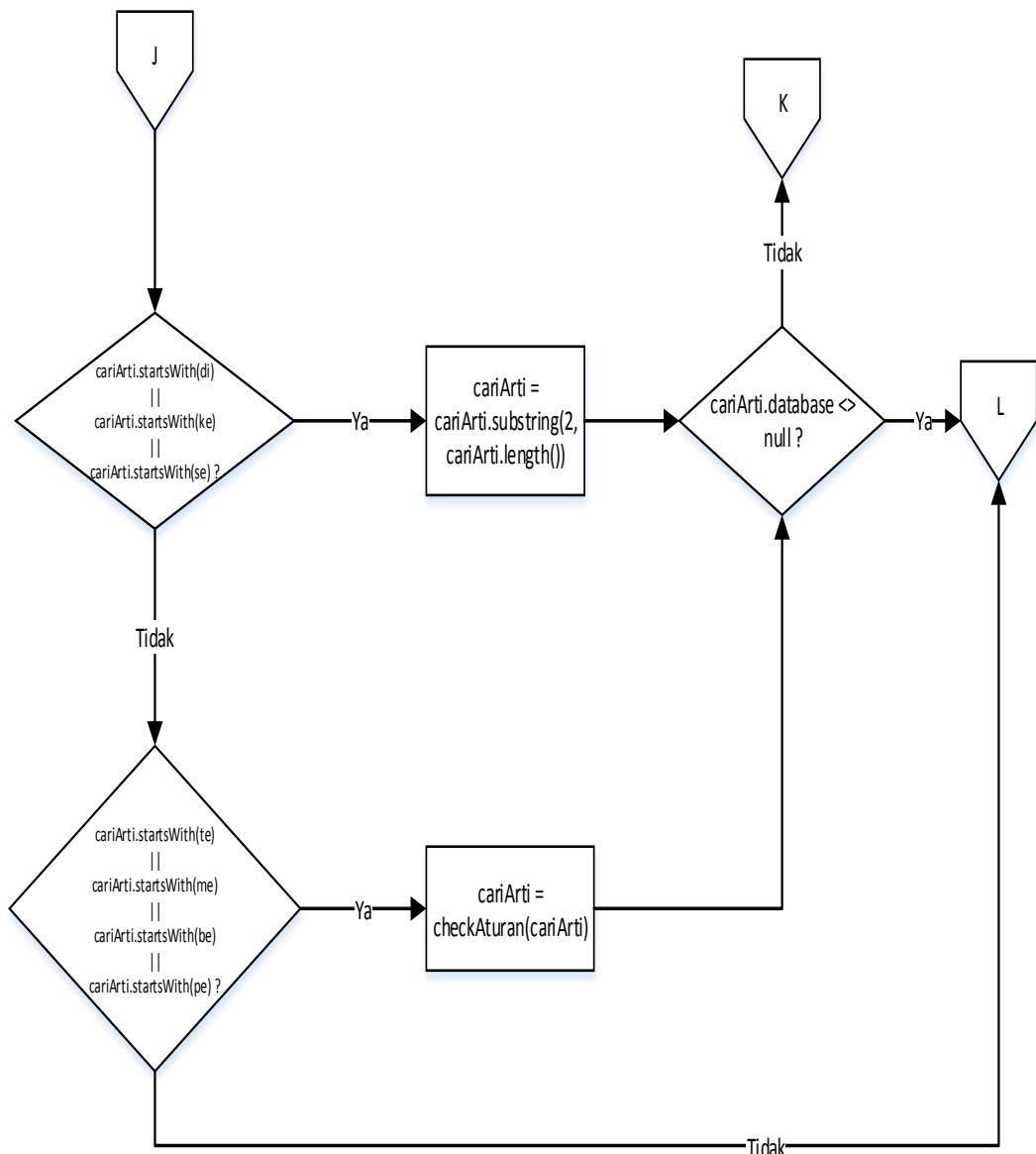
Gambar 15. *Flowchart* pencarian arti kalimat

3) Flowchart algoritma *stemming* Nazief dan Adriani

Flowchart untuk menggambarkan proses algoritma *stemming* Nazief dan Adriani yang ditunjukkan pada Gambar 16 dan Gambar 17.



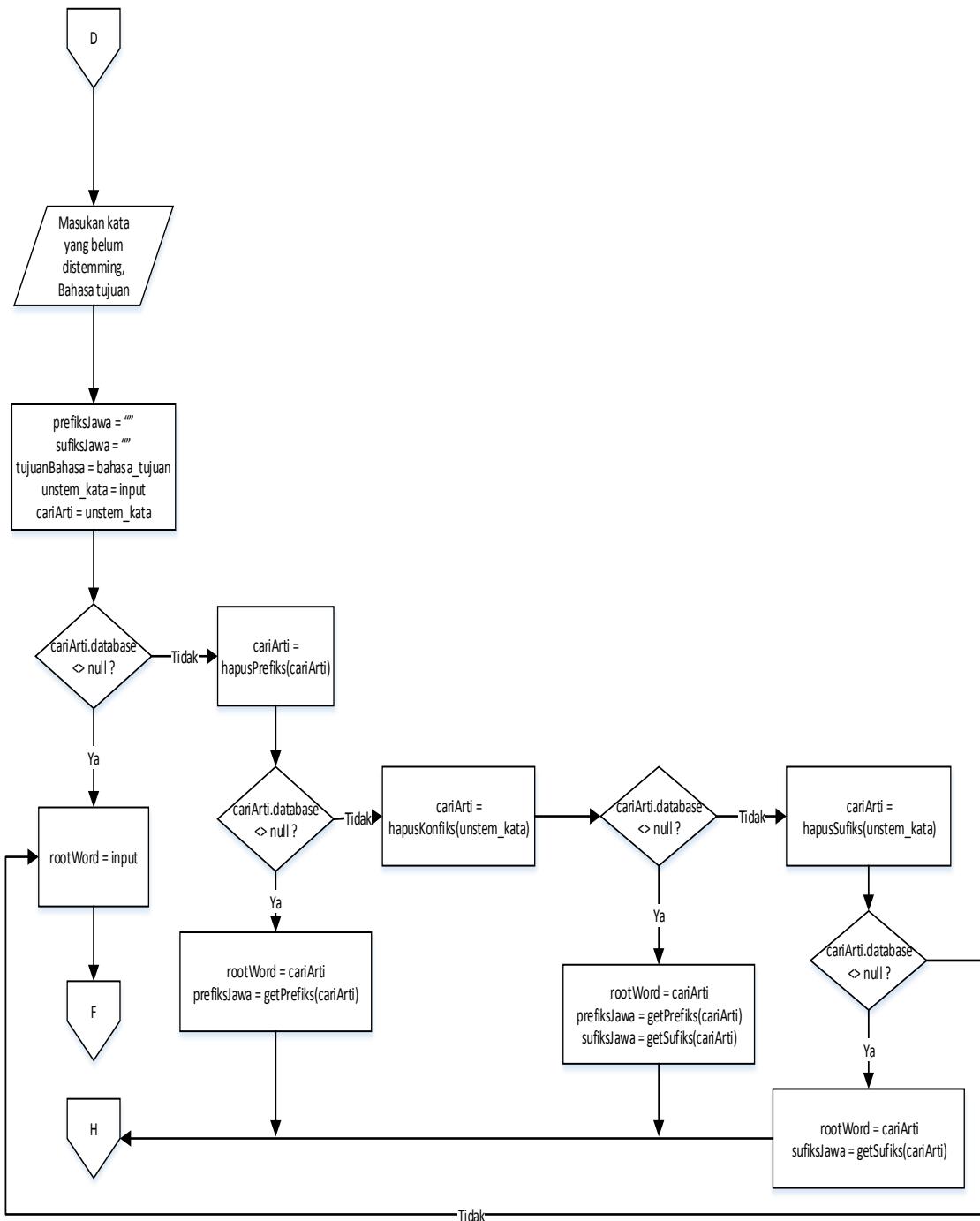
Gambar 16. Flowchart algoritma *stemming* Nazief dan Adriani



Gambar 17. *Flowchart* algoritma *stemming* Nazief dan Adriani (Lanjutan)

4) *Flowchart* algoritma *stemming* Bahasa Jawa

Flowchart untuk menggambarkan proses algoritma *stemming* Bahasa Jawa yang ditunjukkan pada Gambar 18.



Gambar 18. *Flowchart* algoritma *stemming* Bahasa Jawa

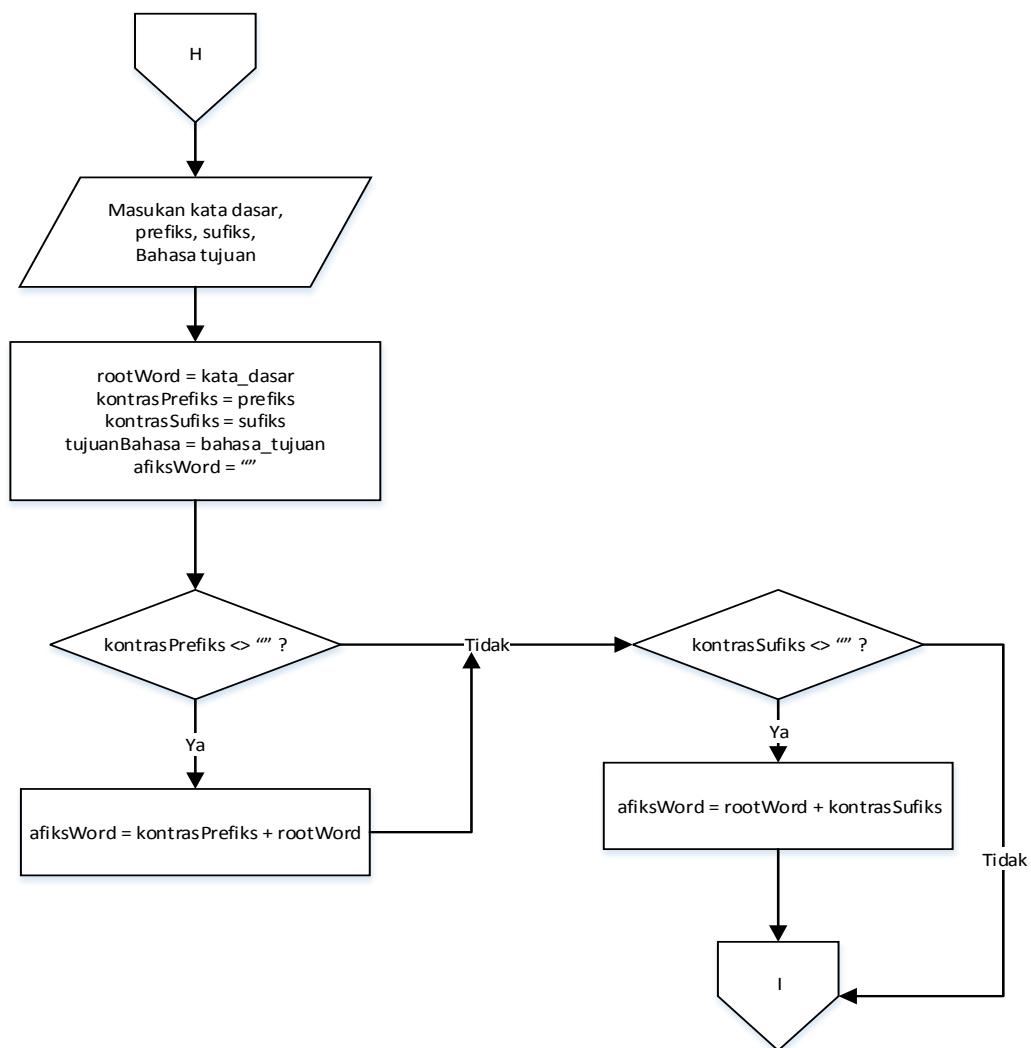
Penjelasan proses algoritma *stemming* Bahasa Jawa sebagai berikut:

- Algoritma dimulai dengan *input* kata berimbuhan dan *input* bahasa tujuan terjemahan.

- b) Selanjutnya dilakukan proses inisialisasi variabel untuk tahapan *stemming*.
- c) *Input* kata berimbahan dicek terlebih dahulu pada *database*, jika ditemukan maka *input* dianggap sebagai *root word* (kata dasar). Jika tidak ditemukan maka tahapan selanjutnya dilakukan pengecekan prefiks.
- d) Jika pada *input* ditemukan prefiks, maka dilakukan proses penghapusan prefiks. Kemudian hasil penghapusan prefiks dicek pada *database*. Jika ditemukan pada *database*, maka hasil penghapusan prefiks dianggap sebagai *root word*. Jika tidak ditemukan, dilakukan pengembalian prefiks pada *input* dan tahapan selanjutnya dilakukan pengecekan konfiks.
- e) Jika pada *input* ditemukan konfiks, maka dilakukan penghapusan konfiks. Penghapusan konfiks dimulai dari penghapusan prefiks kemudian penghapusan sufiks. Jika hasil penghapusan konfiks ditemukan pada *database*, maka hasil tersebut dianggap sebagai *root word*. Jika tidak ditemukan, dilakukan pengembalian konfiks pada *input* dan tahapan selanjutnya dilakukan pengecekan sufiks.
- f) Jika pada *input* ditemukan sufiks, maka dilakukan penghapusan sufiks. Jika hasil penghapusan sufiks ditemukan pada *database*, maka hasil tersebut dianggap sebagai *root word*. Jika tidak ditemukan, maka *input* diasumsikan sebagai *root word*.

5) *Flowchart* proses afiksasi

Flowchart untuk menggambarkan proses afiksasi yang ditunjukkan pada Gambar 19.



Gambar 19. *Flowchart* proses afiksasi

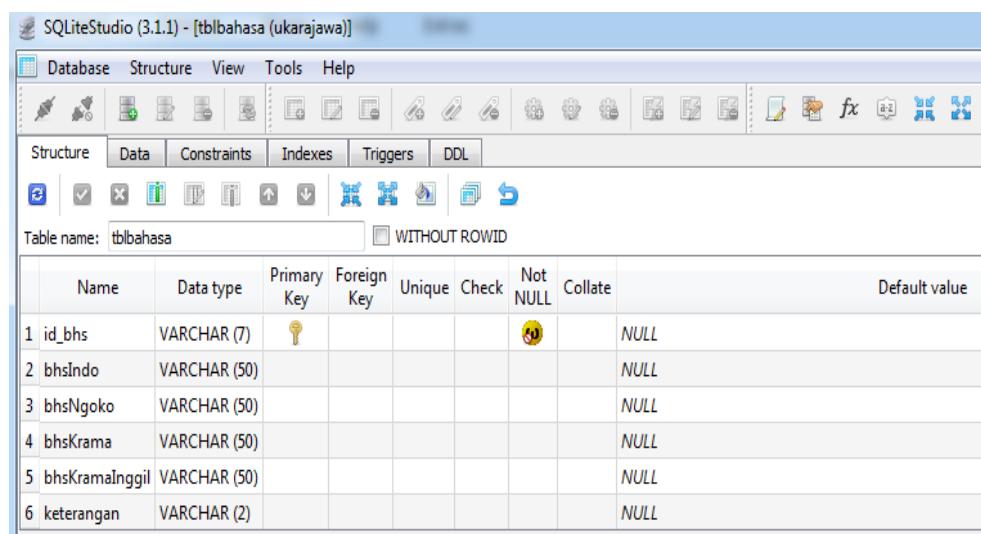
Penjelasan *flowchart* proses afiksasi adalah sebagai berikut:

- Algoritma dimulai dengan *input* kata dasar, prefiks, dan sufiks hasil *stemming*.

- b) Tahapan selanjutnya dilakukan inisialisasi variabel untuk proses afiksasi. Langkah selanjutnya dilakukan pengecekan prefiks.
- c) Jika pada proses *stemming* ditemukan prefiks, maka dilakukan proses penambahan prefiks pada kata dasar dengan mengacu pada aturan morfologi berdasarkan prefiks pada Tabel 3. Langkah selanjutnya dilakukan pengecekan sufiks.
- d) Jika pada proses *stemming* ditemukan sufiks, maka dilakukan proses penambahan sufiks pada kata dasar dengan mengacu pada Tabel 3. Proses algoritma afiksasi berhenti.

6) Perancangan Basis Data

Perancangan basis data menggunakan *database* SQLite dengan *software* SQLiteStudio. Pada basis data memuat tabel yang digunakan untuk proses aplikasi (lihat Gambar 20).



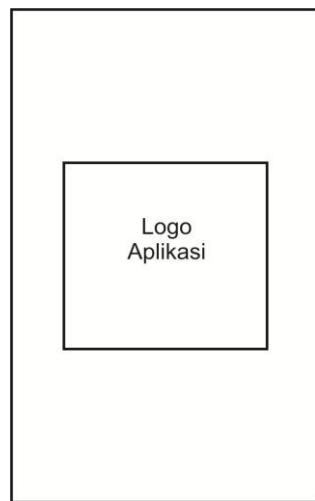
Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1 id_bhs	VARCHAR (7)	key				NULL		
2 bhsIndo	VARCHAR (50)					NULL		
3 bhsNgoko	VARCHAR (50)					NULL		
4 bhsKrama	VARCHAR (50)					NULL		
5 bhsKramaInggil	VARCHAR (50)					NULL		
6 keterangan	VARCHAR (2)					NULL		

Gambar 20. Desain Perancangan Basis Data

c. Desain *Interface*

1) Desain *Interface Splash Screen*

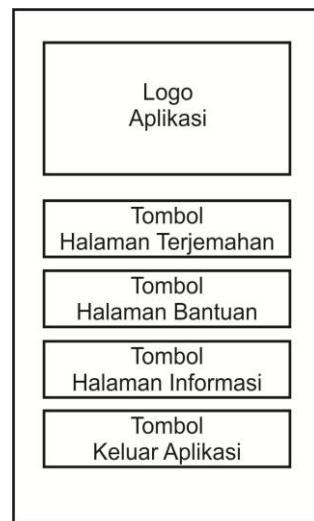
Gambaran tentang desain *interface splash screen* pada aplikasi Ukara Jawa dapat dilihat pada Gambar 21.



Gambar 21. Desain *Interface Splash Screen*

2) Desain *Interface* Halaman Utama

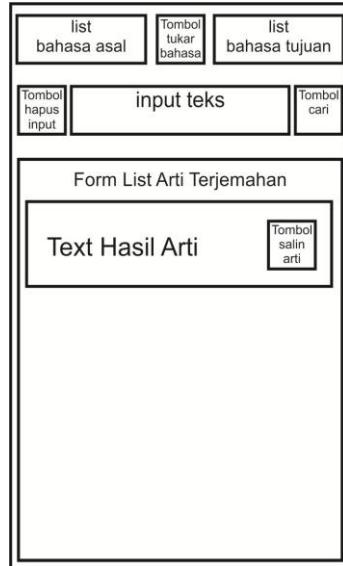
Gambaran tentang desain *interface* Halaman Utama pada aplikasi Ukara Jawa dapat dilihat pada Gambar 22.



Gambar 22. Desain *Interface* Halaman Utama

3) Desain *Interface* Halaman Terjemahan

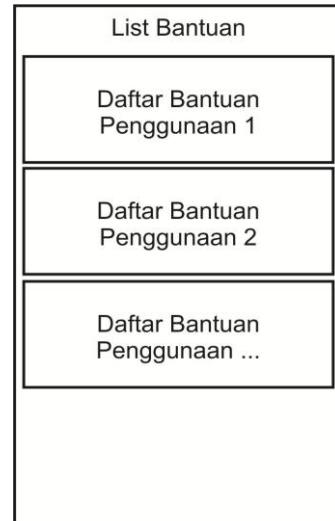
Gambaran tentang desain *interface* Halaman Terjemahan pada aplikasi Ukara Jawa dapat dilihat pada Gambar 23.



Gambar 23. Desain *Interface* Halaman Terjemahan

4) Desain *Interface* Halaman Bantuan

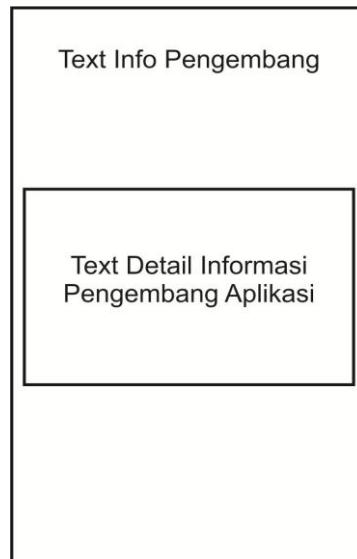
Gambaran tentang desain *interface* Halaman Bantuan pada aplikasi Ukara Jawa dapat dilihat pada Gambar 24.



Gambar 24. Desain *Interface* Halaman Bantuan

5) Desain *Interface* Halaman Info Pengembang

Gambaran tentang desain *interface* Halaman Info Pengembang pada aplikasi Ukara Jawa dapat dilihat pada Gambar 25.



Gambar 25. Desain *Interface* Halaman Info Pengembang

3. Implementasi

Implementasi merupakan tahapan dengan menerapkan desain yang telah ada ke dalam kode program. Implementasi pengodean pada penelitian ini terdiri atas implementasi *layout interface* aplikasi dan implementasi logika pemrograman. Pada tahap implementasi, fungsi-fungsi pada aplikasi dapat diketahui apakah sudah berjalan dengan baik.

4. Pengujian

Tahap pengujian merupakan tahap yang menentukan kelayakan aplikasi yang telah dikembangkan. Pengujian dilakukan mengacu pada kriteria standar ISO 25010 yang meliputi *functional suitability, performance efficiency, compatibility, usability, dan reliability*. Selain itu, kriteria

pengujian juga dilakukan dengan validasi materi terkait isi materi pada aplikasi yang dilakukan oleh ahli materi. Pengujian aspek *functional suitability* dilakukan dengan metode *test case* terhadap fungsi-fungsi aplikasi oleh ahli *software*. Pengujian aspek *performance efficiency* dan *compatibility* dilakukan peneliti dengan menggunakan *tools* yang telah disediakan pada website <https://www.monkop.com/>. Pengujian aspek *usability* diisi oleh siswa SMK Muhammadiyah 2 Yogyakarta dengan menggunakan instrumen kuesioner dari Lewis (1995). Pengujian aspek *reliability* dilakukan dengan menggunakan *tools* yang disediakan pada website <https://testobject.com/>. Pengujian aspek materi dilakukan dengan validasi oleh guru Bahasa Jawa.

C. Tempat dan Waktu Penelitian

Pelaksanaan penelitian untuk pengembangan produk, pengujian produk, dan revisi produk bertempat di Laboratorium Program Studi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta. Tempat penelitian untuk pengujian pengujian materi dan pengujian *usability* dilakukan di SMK Muhammadiyah 2 Yogyakarta. Sedangkan waktu pelaksanaan penelitian dilakukan pada bulan Mei hingga Juli 2017.

D. Sumber Data / Subjek Penelitian

Subjek penelitian melibatkan tiga ahli *software*, dua ahli materi, 30 responden siswa SMK Muhammadiyah 2 Yogyakarta, dan dokumentasi terhadap aplikasi. Alasan jumlah pengambilan sampel responden sebanyak 30 siswa mengacu pada pendapat Roscoe dalam (Sugiyono, 2012: 131) yang

menyebutkan bahwa ukuran sampel yang layak dalam penelitian adalah antara 30 sampai dengan 500. Ahli *software* pada penelitian ini untuk menguji kelayakan aplikasi pada aspek *functional suitability*. Ahli materi pada penelitian ini untuk menguji isi materi pada aplikasi. Responden siswa pada penelitian ini sebagai subjek penelitian untuk menguji aspek *usability*. Sedangkan dokumentasi terhadap aplikasi dilakukan untuk pengujian aspek *performance efficiency, compatibility, dan reliability*.

E. Teknik Pengumpulan Data

1. Wawancara

Wawancara dilakukan dengan pengumpulan data melalui tanya jawab peneliti dengan narasumber. Wawancara pada penelitian ini digunakan untuk mengumpulkan data sebagai acuan analisis kebutuhan. Wawancara dilakukan kepada guru Bahasa Jawa SMK Muhammadiyah 2 Yogyakarta.

2. Observasi

Teknik pengumpulan data observasi dilakukan dengan mengamati langsung terhadap objek yang diteliti. Teknik observasi pada penelitian ini dilakukan untuk mengumpulkan data pengujian pada aspek *performance efficiency, compatibility, dan reliability*.

3. Angket atau Kuesioner

Angket diberikan kepada responden dengan memberikan serangkaian pertanyaan. Data yang didapatkan melalui angket digunakan untuk

menguji kelayakan aplikasi pada aspek *functionality suitability, usability*, dan aspek uji materi.

F. Instrumen Penelitian

Instrumen penelitian adalah suatu alat yang digunakan untuk mengukur fenomena alam maupun sosial yang diamati (Sugiyono, 2012: 148). Instrumen penelitian digunakan untuk mengetahui kelayakan aplikasi yang dikembangkan.

Pada penelitian ini, instrumen penelitian yang digunakan dibagi menjadi instrumen *functional suitability*, instrumen *performance efficiency*, instrumen *compatibility*, instrumen *usability*, instrumen *reliability*, dan instrumen uji materi. Skala jawaban instrumen *functionality suitability*, *instrument compatibility*, *instrument reliability*, dan instrumen uji materi menggunakan Skala Guttman. Skala jawaban instrumen *usability* menggunakan Skala Likert.

Tabel 13. Skala Guttman

No	Kategori	Skor
1	Sukses/Ya	1
2	Gagal/Tidak	0

Tabel 14. Skala Likert

No	Kategori	Skor
1	Sangat Setuju (SS)	4
2	Setuju (S)	3
3	Tidak Setuju (TS)	2
4	Sangat Tidak Setuju (STS)	1

1. Instrumen *Functionality Suitability*

Instrumen *functionality suitability* yang digunakan ditinjau dari aspek *functional correctness*. Berikut kisi-kisi instrumen *functionality suitability*.

Tabel 15. Kisi-kisi Instrumen *Functionality Suitability*

No	Aspek	Indikator	Sub Indikator	No. Butir
1	<i>Functionality Suitability</i>	<i>Functional Correctness</i>	Semua fungsi dapat berjalan dengan benar	1-3

2. Instrumen *Performance Efficiency*

Pengujian *performance efficiency* menggunakan *tools* pada *website* <https://www.monkop.com/>. Metode pengujian dengan melakukan analisis *time behaviour*, *resource utilization CPU*, dan *resource utilization memory* yang digunakan aplikasi.

3. Instrumen *Compatibility*

Instrumen *compatibility* digunakan untuk menganalisis dan menguji aplikasi Ukara Jawa dengan metode observasi. Pengujian *compatibility* meliputi aspek *co-existence*, pengujian berbagai sistem operasi dan jenis perangkat. Pengujian aspek *co-existence* menggunakan daftar instrumen sebagai berikut (lihat Tabel 16).

Tabel 16. Instrumen *Compatibility*

No	Aplikasi yang dijalankan	Ketercapaian Hasil	
		Sukses	Gagal
1	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Whatsapp</i> ?		
2	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Line</i> ?		
3	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Google Translate</i> ?		
4	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Opera Mini</i> ?		
5	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Messaging</i> ?		
6	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Gmail</i> ?		
7	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Play Store</i> ?		
8	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Facebook</i> ?		
9	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Contacts</i> ?		
10	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Office Suite</i> ?		

Untuk pengujian pengujian berbagai sistem operasi dan jenis perangkat menggunakan *tools* pada *website* <https://www.monkop.com/> dengan melakukan uji coba instalasi aplikasi pada beberapa perangkat serta versi Android yang berbeda.

4. Instrumen *Usability*

Instrumen *usability* menggunakan angket IBM *Computer Usability Satisfaction Questionnaires* yang telah tervalidasi berdasarkan penelitian Lewis (1995). Aspek *usability* yang diujikan mencakup beberapa kriteria seperti Tabel 17.

Tabel 17. Kisi-kisi Instrumen *Usability*

Aspek	Kriteria	No. Butir
<i>Usability</i>	<i>Operability</i>	1-6
	<i>Learnability</i>	7-12
	<i>Appropriateness recognizability</i>	13-15
	<i>User interface aesthetics</i>	16-19

Aspek *usability* akan digunakan untuk pengujian yang melibatkan pengguna khususnya siswa SMK Muhammadiyah 2 Yogyakarta. Instrumen *usability* yang akan digunakan dapat dilihat pada Tabel 18.

Tabel 18. Instrumen *Usability*

No	Pernyataan	Penilaian			
		STS	TS	S	SS
1	Secara keseluruhan, saya merasa puas dengan kemudahan penggunaan aplikasi ini.				
2	Cara penggunaan aplikasi ini sangat mudah.				
3	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan lebih mudah ketika menggunakan aplikasi ini.				

No	Pernyataan	Penilaian			
		STS	TS	S	SS
4	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan cepat ketika menggunakan aplikasi ini.				
5	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan efisien menggunakan aplikasi ini.				
6	Saya merasa nyaman menggunakan aplikasi ini.				
7	Aplikasi ini sangat mudah dipelajari.				
8	Saya merasa lebih produktif ketika menggunakan aplikasi ini.				
9	Jika terjadi <i>error</i> , aplikasi ini memberikan pesan kesalahan yang jelas dan memberitahu cara mengatasinya.				
10	Kapanpun saya melakukan kesalahan, saya bisa mengakses kembali aplikasi dengan cepat.				
11	Informasi yang disediakan aplikasi ini sangat jelas.				
12	Sangat mudah menemukan informasi yang saya butuhkan pada aplikasi ini.				
13	Informasi yang diberikan aplikasi ini sangat mudah dipahami.				

No	Pernyataan	Penilaian			
		STS	TS	S	SS
14	Informasi yang diberikan sangat efektif dalam membantu saya menggunakan aplikasi ini.				
15	Tata letak informasi yang ditampilkan aplikasi ini sangat jelas.				
16	Tampilan aplikasi ini menyenangkan.				
17	Saya suka menggunakan tampilan aplikasi seperti ini.				
18	Aplikasi ini memiliki fitur dan kemampuan sesuai keinginan saya.				
19	Secara keseluruhan, saya sangat puas dengan aplikasi ini.				

5. Instrumen *Reliability*

Pengujian aspek *reliability* dengan melakukan *stress test* pada aplikasi menggunakan *tools* yang disediakan <https://testobject.com/>. Metode *stress testing* dengan melakukan *click* dan *touch* pada aplikasi secara acak.

6. Instrumen Uji Materi

Instrumen uji materi berisi kesesuaian materi pada aplikasi ditinjau berdasarkan aspek kemanfaatan materi dan isi materi. Berikut kisi-kisi instrumen uji materi (lihat Tabel 19).

Tabel 19. Kisi-kisi Instrumen Uji Materi

No	Aspek	Indikator	No Butir
1	Kemanfaatan Materi	Kejelasan petunjuk penggunaan	1-2
		Kemudahan materi	33-36
2	Isi Materi	Ketepatan materi	3-32

G. Teknik Analisis Data

1. Aspek *Functionality* *Suitability*, *Compatibility*, *Usability*, *Reliability*, dan Uji Materi

Jenis data yang didapatkan untuk aspek *functionality* *suitability*, *compatibility*, *usability*, *reliability*, dan uji materi adalah data kuantitatif. Data yang didapatkan kemudian dilakukan teknik analisis deskriptif. Data kuantitatif yang berwujud angka-angka hasil hitungan dapat diproses dengan cara dijumlah, dibandingkan dengan jumlah yang diharapkan (Arikunto, 2010).

Perhitungan persentase kelayakan menggunakan rumus sebagai berikut:

$$\text{Persentase Kelayakan (\%)} = \frac{\sum \text{Skor yang diobservasi}}{\sum \text{Skor yang diharapkan}} \times 100\%$$

Setelah mendapatkan perhitungan persentase rata-rata kemudian dilakukan konversi untuk mendapatkan nilai kualitatif yang diklasifikasikan menjadi kategori Sangat Tidak Layak, Tidak Layak, Cukup Layak, Layak, dan Sangat Layak. Kriteria konversi persentase ke dalam pernyataan kualitatif dijabarkan pada Tabel 20.

Tabel 20. Skala Persentase Kelayakan

Presentase Pencapaian	Interpretasi
81 - 100 %	Sangat Layak
61 - 80 %	Layak
41 – 60 %	Cukup Layak
21 – 40 %	Tidak Layak
0 – 20%	Sangat Tidak Layak

2. Aspek *Performance Efficiency*

Analisis pengujian aspek *performance efficiency* dengan melihat hasil pengujian pada aspek *time behaviour*, *resource utilization CPU*, dan *resource utilization memory*.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Hasil Penelitian

1. Tahap Analisis

Pada tahap analisis dilakukan dengan mengumpulkan informasi yang dibutuhkan untuk pengembangan aplikasi. Tahap analisis meliputi analisis fungsional dan nonfungsional.

a. Analisis Fungsional

Analisis fungsional merupakan analisis terhadap fungsi-fungsi yang diperlukan pada aplikasi Ukara Jawa antara lain:

- 1) Aplikasi dapat melakukan proses pencarian terjemahan.
- 2) Aplikasi dapat menghapus masukan terjemahan.
- 3) Aplikasi dapat melakukan pertukaran bahasa asal dan tujuan terjemahan.
- 4) Aplikasi dapat melakukan salin arti penerjemahan.

b. Analisis Nonfungsional

Analisis nonfungsional merupakan analisis kebutuhan yang diperlukan selama pengembangan aplikasi. Analisis nonfungsional meliputi analisis isi program, analisis *hardware*, dan analisis *software*.

1) Analisis Isi Program

Tahap analisis isi program terkait materi yang akan disajikan. Konten materi yang disajikan bersumber dari buku Tingkat Tutur Bahasa Jawa

karangan Poedjosoedarmo, et al dan Kamus Unggah-Ungguh Basa Jawa karangan Haryana Harjawiyan & TH. Supriya.

2) Analisis Hardware

Analisis *hardware* meliputi spesifikasi *hardware* yang mendukung untuk pengembangan aplikasi. Dukungan *hardware* yang dibutuhkan yaitu *hardware* yang dapat menjalankan lingkungan pengembangan Android Studio serta Corel Draw X7. Berikut spesifikasi *hardware* minimal untuk Android Studio.

- a) Microsoft Windows 7/8/10
- b) 3 GB RAM
- c) 2 GB *hard disk space*

Sedangkan untuk spesifikasi minimal Corel Draw X7 adalah sebagai berikut.

- a) Microsoft Windows 7/8/10
- b) Intel Core i3 atau AMD Athlon 64
- c) 2 GB RAM
- d) 1 GB *hard disk space*

Pengembangan aplikasi Ukara Jawa menggunakan spesifikasi yang disarankan yaitu:

- a) Microsoft Windows 7
- b) Intel Core i5
- c) 4 GB RAM

Untuk *hardware* yang dapat menjalankan aplikasi Ukara Jawa yaitu *smartphone* yang memiliki sistem operasi Android dengan versi 4.0.x atau yang terbaru.

3) Analisis Software

Tahap analisis *software* dilakukan terhadap *software* yang digunakan selama pengembangan antara lain:

- a) Android Studio digunakan untuk tahap implementasi pengodean aplikasi Ukara Jawa.
- b) Corel Draw X7 digunakan untuk pembuatan desain, serta *resource* yang digunakan pada aplikasi Ukara Jawa.
- c) Microsoft Office Visio 2013 untuk mendesain *flowchart* aplikasi Ukara Jawa.
- d) StarUML untuk menggambarkan desain UML aplikasi Ukara Jawa.
- e) SQLiteStudio digunakan untuk perancangan desain basis data.

2. Tahap Desain

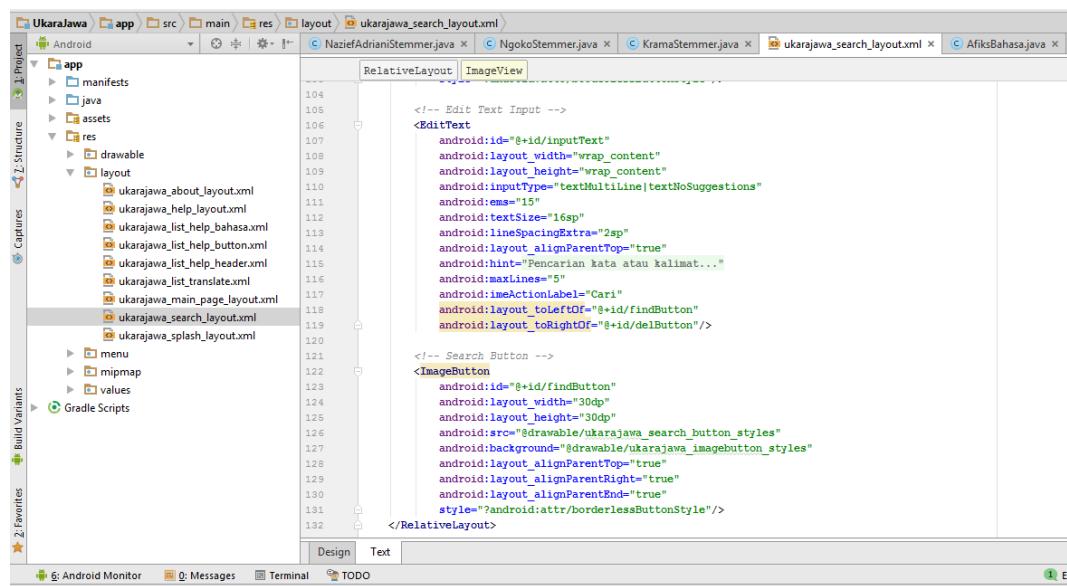
Berdasarkan hasil analisis maka dapat diketahui kebutuhan pengembangan aplikasi. Dari penerapan analisis kemudian dirancang sehingga dapat digambarkan aplikasi yang akan dikembangkan.

3. Tahap Implementasi

Pada tahap implementasi aplikasi Ukara Jawa menggunakan Android Studio dengan target pengembangan aplikasi mulai dari versi 4.0 sampai dengan versi 7.1. Tahapan implementasinya adalah sebagai berikut:

a. Implementasi *Layout Interface* Aplikasi

Implementasi *layout interface* aplikasi menggunakan *script XML* yang merupakan standar pengaturan *layout* Android. *Layout* yang digunakan pada aplikasi Ukara Jawa antara lain *ukarajawa_about_layout*, *ukarajawa_help_layout*, *ukarajawa_list_help_bahasa*, *ukarajawa_list_help_button*, *ukarajawa_list_help_header*, *ukarajawa_list_translate*, *ukarajawa_main_page_layout*, *ukarajawa_search_layout*, dan *ukarajawa_splash_layout*.



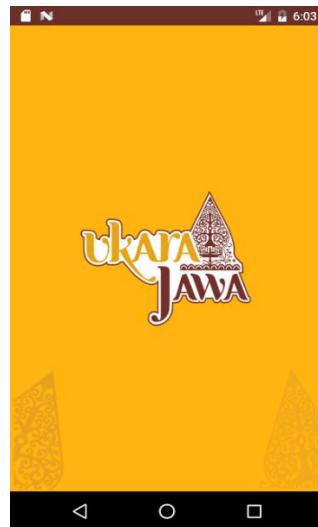
Gambar 26. Potongan *script* implementasi pada *ukarajawa_search_layout*

Berikut ini adalah hasil implementasi *layout interface* aplikasi Ukara Jawa.

1) Implementasi *layout interface splash screen*

Hasil implementasi *layout interface splash screen* dapat dilihat pada

Gambar 27.



Gambar 27. Implementasi *layout interface* *splash screen*

Layout interface *splash screen* menunjukkan gambaran ketika memulai aplikasi dengan menampilkan logo aplikasi.

2) Implementasi *layout interface* Halaman Utama

Hasil implementasi *layout interface* Halaman Utama dapat dilihat pada

Gambar 28.



Gambar 28. Implementasi *layout interface* Halaman Utama

Layout interface Halaman Utama menampilkan menu-menu yang terdapat pada aplikasi.

3) Implementasi *layout interface* Halaman Terjemahan

Hasil implementasi *layout interface* Halaman Terjemahan dapat dilihat pada Gambar 29.

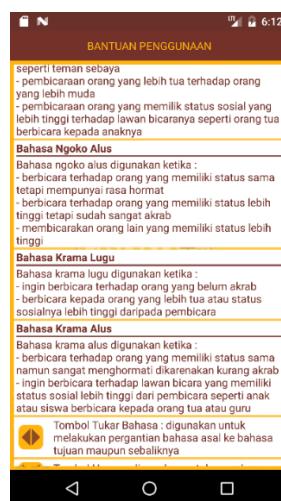


Gambar 29. Implementasi *layout interface* Halaman Terjemahan

Layout interface Halaman Terjemahan menunjukkan tampilan halaman yang digunakan untuk melakukan proses penerjemahan.

4) Implementasi *layout interface* Halaman Bantuan

Hasil implementasi *layout interface* Halaman Bantuan dapat dilihat pada Gambar 30.



Gambar 30. Implementasi *layout interface* Halaman Bantuan

Layout interface Halaman Bantuan menampilkan halaman bantuan sebagai petunjuk penggunaan aplikasi.

5) Implementasi *layout interface* Halaman Info Pengembang

Hasil implementasi *layout interface* Halaman Info Pengembang dapat dilihat pada Gambar 31.



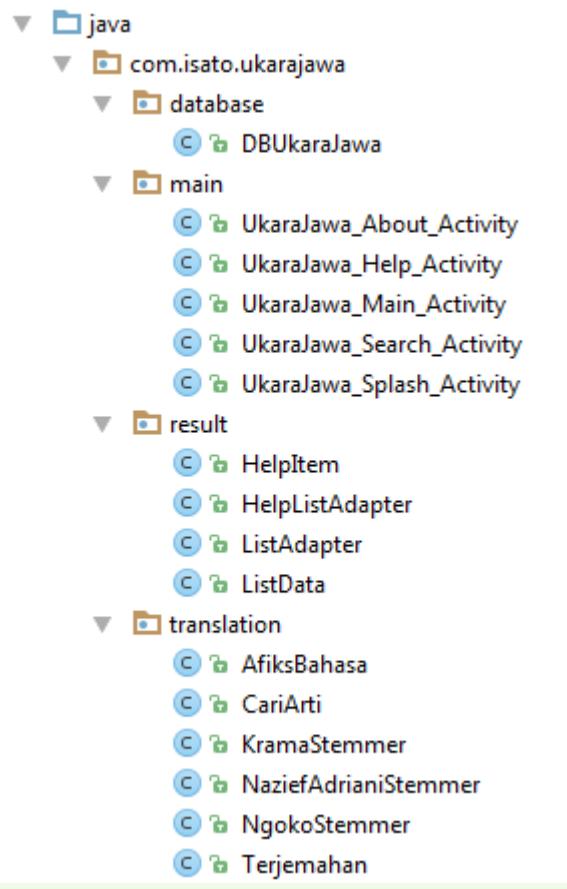
Gambar 31. Implementasi *layout interface* Halaman Info Pengembang

Layout interface halaman info pengembang menampilkan halaman informasi tentang identitas pengembang aplikasi.

b. Implementasi Logika Pemrograman

Aplikasi tidak dapat dijalankan apabila tidak ada perintah yang digunakan. Perintah yang dimaksudkan untuk menjalankan fungsi-fungsi pada aplikasi. Untuk itu diperlukan bahasa pemrograman supaya fungsi dapat dijalankan. Android secara *default* menggunakan bahasa pemrograman Java. Pada aplikasi Ukara Jawa *file* Java yang digunakan dibagi ke dalam empat *package* yang meliputi *package* *com.isato.ukarajawa.database*, *package*

com.isato.ukarajawa.main, package *com.isato.ukarajawa.result*, dan package *com.isato.ukarajawa.translation* (lihat Gambar 32).



Gambar 32. *Package file Java* pada Aplikasi Ukara Jawa

- 1) Implementasi Logika Pemrograman *Stemming* Bahasa Indonesia (Nazief Adriani)

Potongan hasil implementasi logika pemrograman *stemming* Bahasa Indonesia (Nazief Adriani) disajikan pada Gambar 33.

```

219      /**
220      * 4b. For i = 1 to 3, tentukan tipe awalan kemudian hapus awalan.
221      * Jika root word belum juga ditemukan lakukan langkah 5,
222      * jika sudah maka algoritma berhenti.
223      * Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
224      */
225  private void tipeIndoAwalan() {
226      /**
227      * Jika awalannya adalah: "di-", "ke-", atau "se-"
228      * maka tipe awalannya secara berturut-turut adalah "di-", "ke-", atau "se-".
229      */
230      //indoDasar = false;
231      int i = 0;
232      while (i < 3) {
233          if ((akarKataIndo.startsWith("di") || akarKataIndo.startsWith("ke")
234              || akarKataIndo.startsWith("se")) && akarKataIndo.length() > 2) {
235              if (akarKataIndo.startsWith("di")) {
236                  prefiksIndo = "di";
237              } else if (akarKataIndo.startsWith("ke")) {
238                  prefiksIndo = "ke";
239              } else prefiksIndo = "se";
240
241              if (isKataDasarIndo(akarKataIndo.substring(2, akarKataIndo.length())))
242                  akarKataIndo = akarKataIndo.substring(2, akarKataIndo.length());
243                  indoDasar = true;
244              }
245              if (!indoDasar) {
246                  akarKataIndo = cariIndo;
247                  prefiksIndo = "";
248                  suffiksIndo = "";

```

Gambar 33. Potongan Implementasi Algoritma *Stemming* Nazief Adriani

2) Implementasi Logika Pemrograman *Stemming* Bahasa Jawa (*Ngoko* dan *Krama*)

Potongan hasil implementasi logika pemrograman *stemming* Bahasa Jawa *Ngoko* disajikan pada Gambar 34.

```

140     private void hapusNgokoPrefiks() {
141         ngokoDasar = false;
142         //N-
143         if (cariNgoko.length() > 3) {
144             if (cariNgoko.startsWith("ng") || cariNgoko.startsWith("m") || cariNgoko.startsWith("n"))
145                 || cariNgoko.startsWith("ny")) {
146                 //nge-
147                 if (cariNgoko.startsWith("nge")) {
148                     tempNgoko = cariNgoko.substring(3, cariNgoko.length());
149                     if (isKataDasarNgoko(tempNgoko)) {
150                         akarKataNgoko = tempNgoko;
151                         prefiksNgoko = "nge";
152                         ngokoDasar = true;
153                     }
154                 else if (isKataDasarNgoko("k" + cariNgoko.substring(2, cariNgoko.length()))) {
155                     tempNgoko = "k" + cariNgoko.substring(2, cariNgoko.length());
156                     akarKataNgoko = tempNgoko;
157                     prefiksNgoko = "nge";
158                     ngokoDasar = true;
159                 }
160             }
161             //ng-
162             else if (cariNgoko.startsWith("ng")) {
163                 //ng- /k,g,r,l,v/
164                 tempNgoko = cariNgoko.substring(2, cariNgoko.length());
165                 if (isKataDasarNgoko("k" + tempNgoko)) {
166                     tempNgoko = "k" + cariNgoko.substring(2, cariNgoko.length());
167                     akarKataNgoko = tempNgoko;
168                     ngokoDasar = true;
169                 } else if (isKataDasarNgoko(tempNgoko)) {
170                     akarKataNgoko = tempNgoko;

```

Gambar 34. Potongan Hasil Implementasi Algoritma *Stemming Ngoko*

Potongan hasil implementasi logika pemrograman *stemming* Bahasa Jawa

Krama disajikan pada Gambar 35.

```

322     }
323     if (kramaDasar) {
324         prefiksKrama = "ny";
325     }
326 }
327 //m-
328 else if (cariKrama.startsWith("m")) {
329     tempKrama = cariKrama.substring(1, cariKrama.length());
330     akarKonfiksKrama = hapusSuffiksKonfiksKrama(tempKrama);
331     if (!kramaDasar) {
332         tempKrama = "p" + cariKrama.substring(1, cariKrama.length());
333         akarKonfiksKrama = hapusSuffiksKonfiksKrama(tempKrama);
334         if (!kramaDasar) {
335             tempKrama = "w" + cariKrama.substring(1, cariKrama.length());
336             akarKonfiksKrama = hapusSuffiksKonfiksKrama(tempKrama);
337         }
338     }
339     if (kramaDasar) {
340         prefiksKrama = "m";
341     }
342 }
343 }
344 //dipun-
345 else if (cariKrama.startsWith("dipun")) {
346     tempKrama = cariKrama.substring(5, cariKrama.length());
347     akarKonfiksKrama = hapusSuffiksKonfiksKrama(tempKrama);
348     if (kramaDasar) {
349         prefiksKrama = "dipun";
350     }
351 }

```

Gambar 35. Potongan Hasil Implementasi Algoritma *Stemming Krama*

3) Implementasi Logika Pemrograman Afiksasi Bahasa

Potongan hasil implementasi logika pemrograman afiksasi bahasa disajikan pada Gambar 36.

```
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
```

```
else if (prefiks.equals("nge") || prefiks.equals("ng") || prefiks.equals("ny"))
    || prefiks.equals("m") || prefiks.equals("n")) {
    //me- + l,m,n,w,y,ny,ng
    if (kataDasar.charAt(0) == 'l' || kataDasar.charAt(0) == 'm'
        || (kataDasar.charAt(0) == 'n' && kataDasar.charAt(1) == 'y')
        || (kataDasar.charAt(0) == 'n' && kataDasar.charAt(1) == 'g')
        || kataDasar.charAt(0) == 'n' || kataDasar.charAt(0) == 'w'
        || kataDasar.charAt(0) == 'r' || kataDasar.charAt(0) == 'y') {
        hasilAfiks = "me" + kataDasar;
    }
    //me- + vowel atau me- + k,g,h,kh --> meng-
    else if (isVowel(kataDasar.charAt(0)) || kataDasar.charAt(0) == 'g'
        || kataDasar.charAt(0) == 'h' || kataDasar.charAt(0) == 'k') {
        if (kataDasar.charAt(0) == 'k' && isVowel(kataDasar.charAt(1))) {
            hasilAfiks = "meng" + kataDasar.substring(1, kataDasar.length());
        }
        else {
            hasilAfiks = "meng" + kataDasar;
        }
    }
    //me- + d,t,c,j,sy,z --> men-
    else if (kataDasar.charAt(0) == 'd' || kataDasar.charAt(0) == 't'
        || kataDasar.charAt(0) == 'c' || kataDasar.charAt(0) == 'j'
        || (kataDasar.charAt(0) == 's' && kataDasar.charAt(1) == 'y')
        || kataDasar.charAt(0) == 'z') {
        if (kataDasar.charAt(0) == 't') {
            hasilAfiks = "men" + kataDasar.substring(1, kataDasar.length());
        }
    }
}
```

Gambar 36. Potongan Hasil Implementasi Logika Pemrograman Afiksasi Bahasa

4. Tahap Pengujian

Tahap terakhir dari penelitian ini adalah dengan melakukan pengujian. Pengujian yang dilakukan menggunakan instrumen penelitian sesuai dengan standar ISO 25010 yaitu *functionality suitability, performance efficiency, compatibility, usability, reliability*, serta uji materi. Hasil pengujian adalah sebagai berikut.

a. Pengujian *Functionality Suitability*

Pengujian *functionality suitability* dilakukan menggunakan kuesioner *checklist* yang diisi oleh ahli *software* sebanyak tiga orang. Hasil pengujian *functionality suitability* adalah sebagai berikut.

Tabel 21. Hasil Pengujian *Functionality Suitability*

No	Aktivitas/ Menu	Skenario Uji	Pertanyaan	Ketercapaian	
				Ya	Tidak
1	Membuka Aplikasi	Memilih ikon <i>launcher</i> aplikasi Ukara Jawa	Apakah halaman utama dapat ditampilkan dan muncul semua menu (Terjemahan, Bantuan, Info Pengembang, dan Keluar) dengan benar ?	3	
2	Halaman Utama	Memilih menu Terjemahan	Apakah pengguna dapat membuka menu Terjemahan ?	3	
		Menu Terjemahan (menekan tombol <i>back</i>)	Apakah pengguna dapat kembali ke halaman utama dengan menekan tombol <i>back</i> ?	3	
		Memilih menu Bantuan	Apakah halaman Bantuan Penggunaan dapat ditampilkan dengan benar ?	3	
		Memilih menu Info Pengembang	Apakah halaman Info Pengembang dapat ditampilkan dengan benar ?	3	
		Memilih menu Keluar	Ketika memilih menu Keluar, apakah muncul <i>pop-up</i> konfirmasi untuk keluar ?	3	
			Ketika menekan tombol "Tidak", apakah pengguna dapat kembali ke Halaman Utama ?	3	
			Apakah pengguna dapat mengakhiri penggunaan aplikasi ketika menekan tombol "Ya" ?	3	

No	Aktivitas/ Menu	Skenario Uji	Pertanyaan	Ketercapaian	
				Ya	Tidak
3	Menu Terjemahan	Memilih bahasa asal terjemahan	Apakah bahasa asal terjemahan (Indonesia, <i>ngoko lugu</i> , <i>ngoko alus</i> , <i>krama lugu</i> , dan <i>krama alus</i>) ditampilkan dengan benar ?	3	
		Memilih bahasa tujuan terjemahan	Daftar bahasa tujuan terjemahan (Indonesia, <i>ngoko lugu</i> , <i>ngoko alus</i> , <i>krama lugu</i> , dan <i>krama alus</i>) apakah ditampilkan dengan benar ?	3	
		Menekan tombol tukar bahasa: 	Apakah pertukaran bahasa asal dengan bahasa tujuan terjemahan dapat dijalankan dengan benar ?	3	
		Menekan tombol cari terjemahan maupun <i>icon</i> di <i>keyboard</i> : 	Ketika memasukkan <i>input</i> terjemahan, apakah pengguna dapat melakukan pencarian arti terjemahan ?	3	
		Menekan tombol hapus <i>input</i> terjemahan: 	Ketika terdapat <i>input</i> terjemahan, apakah pengguna dapat menghapus keseluruhan <i>input</i> terjemahan ?	3	
		Menekan tombol salin arti/ <i>copy</i> : 	Apakah pengguna dapat melakukan salin/ <i>copy</i> arti pada bagian hasil terjemahan ?	3	

Berdasarkan pada Tabel 21, dapat diketahui persentase masing-masing butir pengujian.

$$Ya = \frac{42}{42} \times 100\% = 100\%$$

$$Tidak = \frac{0}{42} \times 100\% = 0\%$$

Tabel 22. Analisis Data Pengujian *Functionality Suitability*

No	Skor Pengujian	Skor Maksimal	Persentase (%)
1	3	3	100
2	3	3	100
3	3	3	100
4	3	3	100
5	3	3	100
6	3	3	100
7	3	3	100
8	3	3	100
9	3	3	100
10	3	3	100
11	3	3	100
12	3	3	100
13	3	3	100
14	3	3	100
Total	42	42	100

Berdasarkan hasil analisis perhitungan pada Tabel 22, maka diperoleh persentase **100%** untuk pengujian *functionality suitability* oleh tiga orang ahli *software*. Hasil persentase tersebut kemudian dikonversikan menjadi nilai kualitatif berdasarkan skala persentase kelayakan. Berdasarkan hasil konversi nilai tersebut, pengujian *functionality suitability* termasuk dalam kategori **“Sangat Layak”**. Hasil pengujian menunjukkan bahwa fungsi-fungsi yang terdapat pada aplikasi Ukara Jawa sudah berjalan dengan baik.

b. Pengujian *Performance Efficiency*

Pengujian *performance efficiency* menggunakan *tools* yang disediakan <https://www.monkop.com/>. Pengujian *performance efficiency* dilakukan dengan metode *automated testing* menggunakan 11 perangkat yang disediakan. Berikut ini ringkasan hasil pengujian *performance efficiency*.

1) *time behaviour*

Pengujian aspek *time behaviour* meliputi penggunaan waktu yang dibutuhkan untuk *launching time* dan *rendering time* aplikasi. Masing-masing rincian waktu yang dibutuhkan adalah sebagai berikut.

a) *Launching Time*

Launching time merupakan waktu yang dibutuhkan aplikasi saat dijalankan. Hasil pengujian *launching time* disajikan pada Gambar 37.

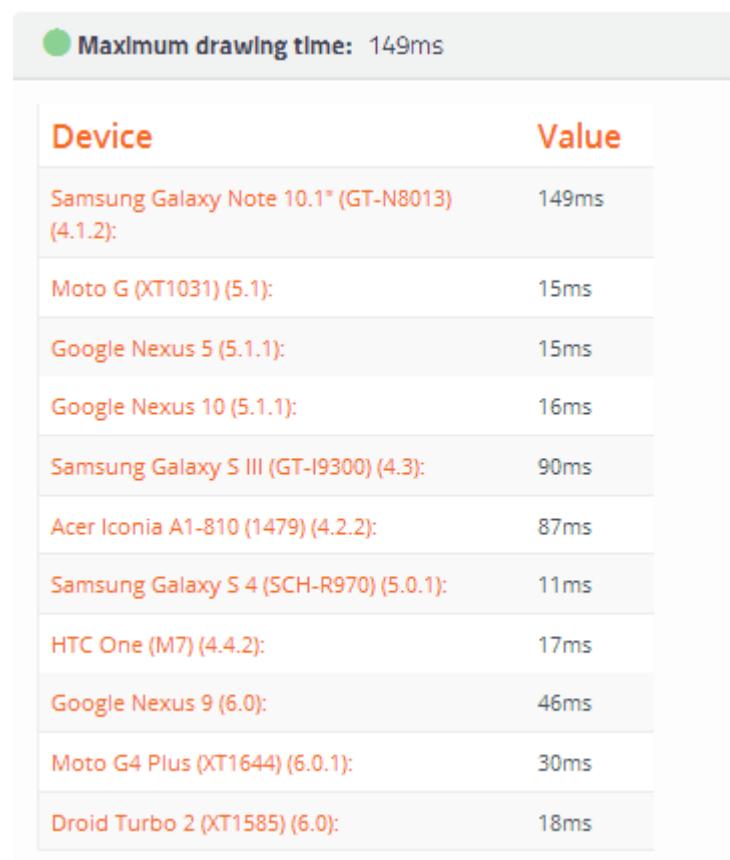
App launching time: 1261ms	
Device	Value
Samsung Galaxy Note 10.1" (GT-N8013) (4.1.2):	470ms
Moto G (XT1031) (5.1):	947ms
Google Nexus 5 (5.1.1):	271ms
Google Nexus 10 (5.1.1):	424ms
Samsung Galaxy S III (GT-I9300) (4.3):	765ms
Acer Iconia A1-810 (1479) (4.2.2):	617ms
Samsung Galaxy S 4 (SCH-R970) (5.0.1):	901ms
HTC One (M7) (4.4.2):	360ms
Google Nexus 9 (6.0):	1261ms
Moto G4 Plus (XT1644) (6.0.1):	303ms
Droid Turbo 2 (XT1585) (6.0):	708ms

Gambar 37. Hasil Pengujian *Launching Time*

Berdasarkan Gambar 37, *launching time* yang dibutuhkan paling tinggi yaitu 1261 ms. Kriteria yang ditetapkan <https://www.monkop.com/> bahwa *launching time* aplikasi paling tidak hanya membutuhkan waktu selama 3 detik.

b) *Rendering Time*

Rendering time merupakan waktu yang dibutuhkan aplikasi untuk menyajikan grafik atau animasi setiap *frame*. Hasil pengujian *rendering time* disajikan pada Gambar 38.



Device	Value
Samsung Galaxy Note 10.1" (GT-N8013) (4.1.2):	149ms
Moto G (XT1031) (5.1):	15ms
Google Nexus 5 (5.1.1):	15ms
Google Nexus 10 (5.1.1):	16ms
Samsung Galaxy S III (GT-I9300) (4.3):	90ms
Acer Iconia A1-810 (1479) (4.2.2):	87ms
Samsung Galaxy S 4 (SCH-R970) (5.0.1):	11ms
HTC One (M7) (4.4.2):	17ms
Google Nexus 9 (6.0):	46ms
Moto G4 Plus (XT1644) (6.0.1):	30ms
Droid Turbo 2 (XT1585) (6.0):	18ms

Gambar 38. Hasil Pengujian *Rendering Time*

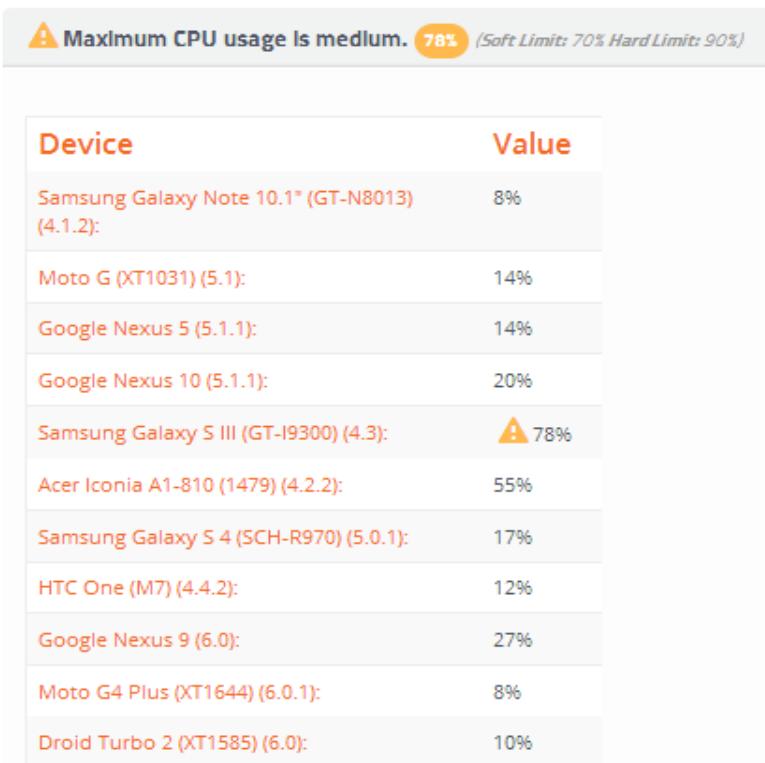
Berdasarkan Gambar 38, *rendering time* yang dibutuhkan paling tinggi adalah sebesar 149 ms. Standar yang ditetapkan

<https://www.monkop.com/> bahwa *rendering time* setiap *frame* adalah sebesar 16 ms.

Dari penjabaran aspek *time behaviour*, dapat disimpulkan bahwa *launching time* aplikasi berjalan kurang dari 3 detik. Sedangkan *rendering time* yang dibutuhkan aplikasi adalah 149 ms. Hal ini disebabkan aplikasi memuat banyak gambar yang dibutuhkan setiap fungsinya. Namun dengan tidak adanya *error* pada setiap prosesnya menunjukkan bahwa aplikasi berjalan dengan baik.

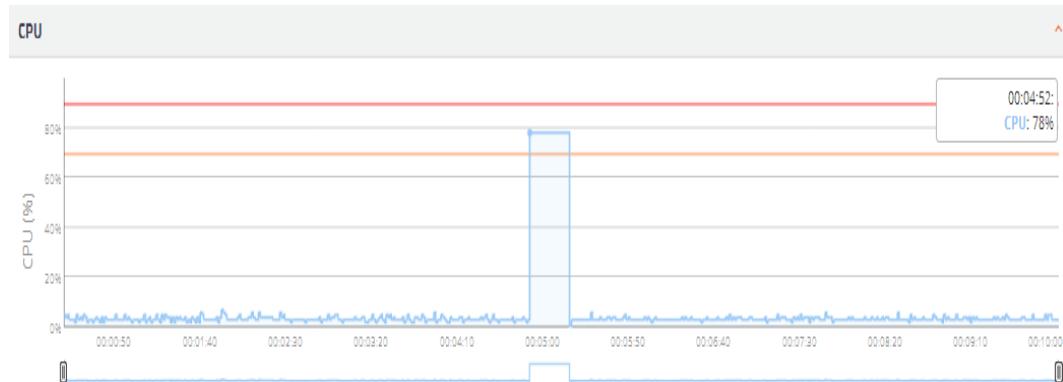
2) *resource utilization CPU*

Resource utilization CPU merupakan jumlah sumber daya CPU (CPU *usage*) yang digunakan ketika menjalankan aplikasi. Hasil pengujian *resource utilization CPU* dapat dilihat pada Gambar 39.



Gambar 39. Hasil Pengujian *CPU Usage*

Berdasarkan Gambar 39, terdapat penggunaan CPU yang dibutuhkan paling tinggi sebesar 78% pada perangkat Samsung Galaxy S III (GT-I900). Rincian lebih lanjut disajikan pada Gambar 40.



Gambar 40. CPU *Usage* pada perangkat Samsung Galaxy S III

Meskipun tergolong cukup tinggi, secara keseluruhan rata-rata penggunaan CPU yang dibutuhkan paling tinggi sebesar 10% (lihat Gambar 41)

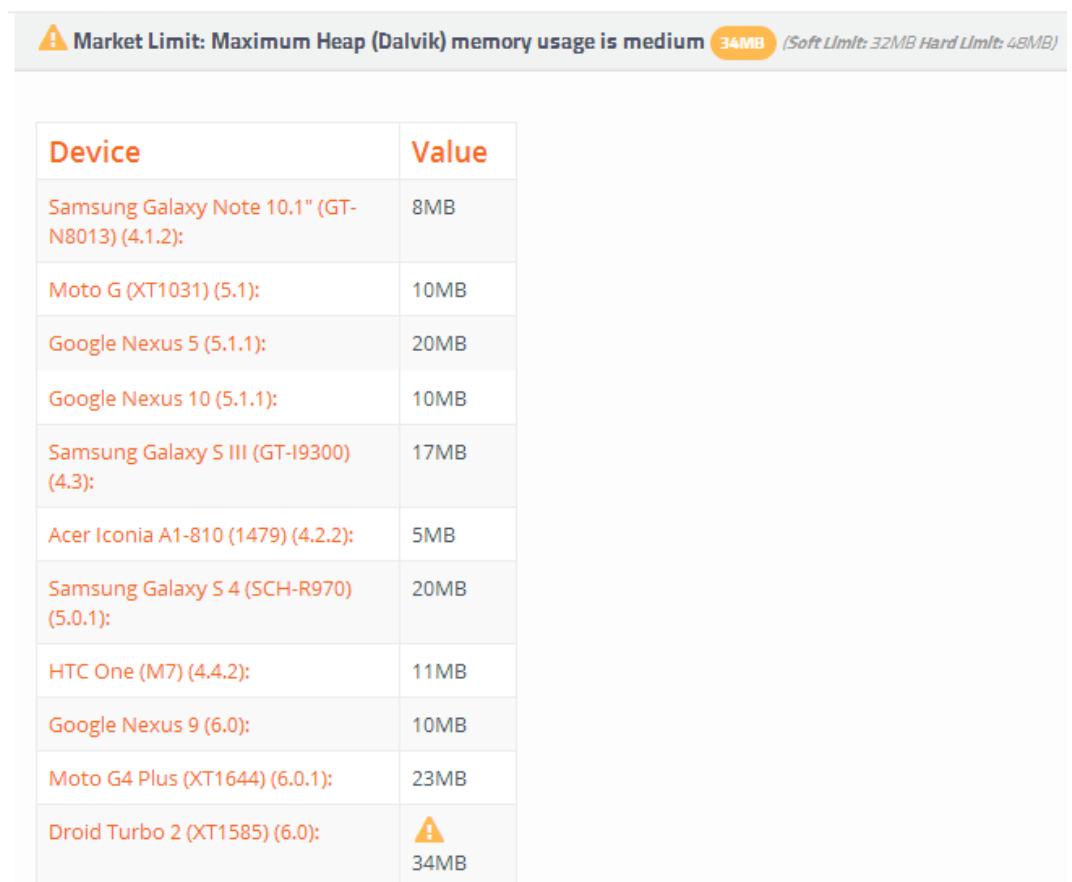
Average CPU: 10%	
Device	Value
Samsung Galaxy Note 10.1" (GT-N8013) (4.1.2):	3%
Moto G (XT1031) (5.1):	9%
Google Nexus 5 (5.1.1):	6%
Google Nexus 10 (5.1.1):	8%
Samsung Galaxy S III (GT-I9300) (4.3):	3%
Acer Iconia A1-810 (1479) (4.2.2):	5%
Samsung Galaxy S 4 (SCH-R970) (5.0.1):	5%
HTC One (M7) (4.4.2):	4%
Google Nexus 9 (6.0):	10%
Moto G4 Plus (XT1644) (6.0.1):	3%
Droid Turbo 2 (XT1585) (6.0):	3%

Gambar 41. Rata-rata Hasil Pengujian CPU *Usage*

Hasil pengujian tersebut masih berada pada standar yang ditetapkan Little Eye Labs (2013) yaitu 5% - 15%.

3) *resource utilization memory*

Resource utilization memory merupakan jumlah sumber daya *memory* perangkat yang dibutuhkan saat menjalankan aplikasi. Hasil pengujian *resource utilization memory* dapat dilihat pada Gambar 42.



The screenshot shows a table of memory usage for various devices. The table has two columns: 'Device' and 'Value'. The 'Value' column shows memory usage in MB. A warning message at the top of the table reads: 'Market Limit: Maximum Heap (Dalvik) memory usage is medium 34MB (Soft Limit: 32MB Hard Limit: 48MB)'. The table data is as follows:

Device	Value
Samsung Galaxy Note 10.1" (GT-N8013) (4.1.2):	8MB
Moto G (XT1031) (5.1):	10MB
Google Nexus 5 (5.1.1):	20MB
Google Nexus 10 (5.1.1):	10MB
Samsung Galaxy S III (GT-I9300) (4.3):	17MB
Acer Iconia A1-810 (1479) (4.2.2):	5MB
Samsung Galaxy S 4 (SCH-R970) (5.0.1):	20MB
HTC One (M7) (4.4.2):	11MB
Google Nexus 9 (6.0):	10MB
Moto G4 Plus (XT1644) (6.0.1):	23MB
Droid Turbo 2 (XT1585) (6.0):	34MB

Gambar 42. Hasil Pengujian *Resource Utilization Memory*

Berdasarkan Gambar 42, hasil pengujian *resource utilization memory* paling tinggi adalah sebesar 34 MB. Hasil tersebut masih berada pada standar yang ditetapkan <https://www.monkop.com/> yaitu kurang dari 48 MB.

Dari keseluruhan pengujian *performance efficiency*, bahwa aplikasi Ukara Jawa dapat berjalan dengan baik dan tidak terjadi *crash* dan *memory leak* sehingga dapat disimpulkan aplikasi dikategorikan “**Layak**”.

c. Pengujian *Compatibility*

Pengujian *compatibility* dilakukan dengan pengujian aspek *co-existence*, pengujian berbagai sistem operasi dan jenis perangkat. Berikut ini ringkasan hasil pengujian *compatibility*.

1) Pengujian aspek *co-existence*

Pengujian aspek *co-existence* dilakukan dengan menjalankan aplikasi Ukara Jawa bersamaan dengan aplikasi lain pada satu perangkat. Berikut ini hasil pengujian aspek *co-existence* (Tabel 23).

Tabel 23. Hasil Pengujian Aspek *Co-existence*

No	Aplikasi yang dijalankan	Ketercapaian Hasil	
		Sukses	Gagal
1	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Whatsapp</i> ?	Ya	
2	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Line</i> ?	Ya	
3	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Google Translate</i> ?	Ya	
4	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Opera Mini</i> ?	Ya	
5	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Messaging</i> ?	Ya	
6	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Gmail</i> ?	Ya	

No	Aplikasi yang dijalankan	Ketercapaian Hasil	
		Sukses	Gagal
7	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Play Store</i> ?	Ya	
8	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Facebook</i> ?	Ya	
9	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Contacts</i> ?	Ya	
10	Apakah aplikasi Ukara Jawa dapat berjalan bersamaan dengan <i>Office Suite</i> ?	Ya	

Berdasarkan pada Tabel 23, dapat diketahui persentase masing-masing butir pengujian.

$$\text{Sukses} = \frac{10}{10} \times 100\% = 100\%$$

$$\text{Gagal} = \frac{0}{10} \times 100\% = 0\%$$

Hasil persentase tersebut kemudian dikonversikan menjadi nilai kualitatif berdasarkan skala persentase kelayakan. Berdasarkan hasil konversi nilai tersebut, pengujian aspek *co-existence* termasuk pada kategori "**Sangat Layak**".

2) Pengujian berbagai sistem operasi dan jenis perangkat

Pengujian berbagai sistem operasi dan jenis perangkat menggunakan 11 perangkat yang disediakan <https://www.monkop.com/>. Berikut ini daftar perangkat yang digunakan untuk pengujian (Gambar 43).

Ride	Device	OS	Size/Density
●	Google Nexus 5	5.1 (API:22)	normal/xxhdpi
●	Samsung Galaxy Note 10.1" (GT-N8013)	4.1 (API:16)	xlarge/mdpi
●	Samsung Galaxy S III (GT-I9300)	4.3 (API:18)	normal/xhdpi
●	Samsung Galaxy S 4 (SCH-R970)	5.0.x (API:21)	normal/xxhdpi
●	HTC One (M7)	4.4 (API:19)	normal/xxhdpi
●	Google Nexus 9	6.0 (API:23)	xlarge/xhdpi
●	Google Nexus 10	5.1 (API:22)	xlarge/xhdpi
●	Moto G (XT1031)	5.1 (API:22)	normal/xhdpi
●	Acer Iconia A1-810 (1479)	4.2 (API:17)	xlarge/mdpi
●	Droid Turbo 2 (XT1585)	6.0 (API:23)	normal/xxxhdpi
●	Moto G4 Plus (XT1644)	6.0 (API:23)	normal/xxhdpi

Gambar 43. Daftar Perangkat Pengujian *Compatibility*

Tahapan pengujian dilakukan dengan uji coba instalasi, proses menjalankan aplikasi, dan proses *uninstall*. Berikut ini sampel hasil pengujian pada perangkat Samsung Galaxy S4 (Gambar 44).



Gambar 44. Sampel Pengujian Perangkat Samsung Galaxy S4

Hasil pengujian berbagai sistem operasi dan jenis perangkat disajikan pada Tabel 24.

Tabel 24. Hasil Pengujian Berbagai Sistem Operasi dan Jenis Perangkat

No	Pertanyaan	Ketercapaian Pengujian		
		Install	Process	Uninstall
1	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Google Nexus 5 ?	Ya	Ya	Ya
2	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Samsung Galaxy Note 10.1" ?	Ya	Ya	Ya
3	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Samsung Galaxy S III ?	Ya	Ya	Ya
4	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Samsung Galaxy S 4 ?	Ya	Ya	Ya
5	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat HTC One ?	Ya	Ya	Ya
6	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Google Nexus 9 ?	Ya	Ya	Ya
7	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Google Nexus 10 ?	Ya	Ya	Ya

No	Pertanyaan	Ketercapaian Pengujian		
		Install	Process	Uninstall
8	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Moto G ?	Ya	Ya	Ya
9	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Acer Iconia A1-810 ?	Ya	Ya	Ya
10	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Droid Turbo 2 ?	Ya	Ya	Ya
11	Apakah pengujian <i>install</i> , <i>process</i> , dan <i>uninstall</i> berhasil dilakukan pada perangkat Moto G4 Plus ?	Ya	Ya	Ya

Berdasarkan Tabel 24, maka dapat diperoleh persentase masing-masing butir pengujian.

$$\begin{aligned}
 \text{Ya} &= \frac{\sum(\text{install} + \text{process} + \text{uninstall})}{\sum \text{Skor maksimal}} \times 100\% \\
 &= \frac{33}{33} \times 100\% = 100\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Tidak} &= \frac{\sum(\text{install} + \text{process} + \text{uninstall})}{\sum \text{Skor maksimal}} \times 100\% \\
 &= \frac{0}{33} \times 100\% = 0\%
 \end{aligned}$$

Selanjutnya dilakukan analisis data pengujian berbagai sistem operasi dan jenis perangkat.

Tabel 25. Analisis Data Pengujian Berbagai Sistem Operasi dan Jenis Perangkat

No	Skor Pengujian	Skor Maksimal	Persentase (%)
1	3	3	100
2	3	3	100
3	3	3	100
4	3	3	100
5	3	3	100
6	3	3	100
7	3	3	100
8	3	3	100
9	3	3	100
10	3	3	100
11	3	3	100
Total	33	33	100

Dari hasil perhitungan di atas, diperoleh persentase **100%** untuk pengujian berbagai sistem operasi dan jenis perangkat. Hasil persentase tersebut kemudian dikonversikan menjadi nilai kualitatif berdasarkan skala persentase kelayakan. Berdasarkan hasil konversi nilai tersebut, pengujian berbagai sistem operasi dan perangkat mempunyai skala **“Sangat Layak”**.

Dari hasil pengujian aspek *co-existence* dan pengujian berbagai sistem operasi dan jenis perangkat, dapat disimpulkan bahwa aplikasi Ukara Jawa pada pengujian *compatibility* dapat dikategorikan **“Sangat Layak”**.

d. Pengujian *Usability*

Pengujian *usability* dilakukan terhadap siswa SMK Muhammadiyah 2 Yogyakarta sebanyak 30 responden. Responden mencoba aplikasi Ukara Jawa pada perangkat yang dimiliki responden kemudian responden mengisi

kuesioner yang telah distandardkan oleh Lewis (1995). Kuesioner ini terdiri atas 19 butir pertanyaan dengan skala jawaban mulai dari Sangat Tidak Setuju (STS) sampai Sangat Setuju (SS). Skala jawaban tersebut dikonversikan ke dalam skala 1-4. Hasil pengujian *usability* dapat dilihat pada Tabel 26.

Tabel 26. Hasil Pengujian *Usability*

NO RESPONDEN	PERNYATAAN																			TOTAL SKOR	SKOR MAX	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19			
1	2	3	2	3	3	2	3	3	2	3	3	3	3	3	4	3	4	3	3	55	76	
2	3	4	4	4	3	3	3	3	3	3	3	3	3	4	3	3	3	3	3	61	76	
3	3	3	3	3	3	4	3	3	3	3	3	3	3	4	3	3	4	2	3	60	76	
4	3	3	3	3	3	2	3	3	3	3	3	4	3	4	3	3	3	4	4	60	76	
5	3	3	4	4	3	4	3	3	3	3	4	3	3	3	3	3	3	2	3	60	76	
6	3	3	3	3	3	3	3	2	3	3	3	4	3	3	3	3	4	3	3	58	76	
7	4	4	4	4	4	3	4	4	4	4	3	4	4	4	2	4	4	3	4	71	76	
8	2	3	3	3	3	4	3	3	3	3	3	3	3	3	2	2	3	2	2	54	76	
9	3	3	4	4	3	3	3	3	3	3	3	3	3	4	3	3	3	4	4	62	76	
10	3	4	3	3	3	4	3	3	4	3	4	3	3	3	3	3	3	2	3	60	76	
11	3	3	4	4	3	4	3	3	3	3	3	3	3	3	3	4	3	2	3	60	76	
12	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	4	58	76	
13	3	4	4	4	3	3	4	3	3	4	3	3	4	4	3	4	3	3	4	66	76	
14	3	3	3	3	4	3	3	4	3	3	3	4	3	3	3	3	3	2	3	59	76	
15	3	3	3	3	4	4	3	3	3	3	2	3	3	3	3	4	3	3	3	59	76	
16	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	3	57	76	
17	4	4	4	3	3	4	4	3	4	3	4	3	4	3	3	3	3	4	2	66	76	
18	3	3	3	3	3	4	3	3	3	3	3	3	3	3	3	3	3	3	2	57	76	
19	3	4	3	3	3	3	3	2	3	3	3	3	3	3	4	3	3	3	2	57	76	
20	3	3	4	4	3	3	4	3	3	3	4	3	3	4	3	3	3	2	3	61	76	
21	4	3	3	3	3	3	4	3	3	3	3	3	3	3	3	4	3	3	3	60	76	
22	4	4	3	3	4	3	3	3	3	4	4	3	3	3	3	4	4	3	3	64	76	
23	3	3	4	4	4	4	3	2	2	2	3	3	3	3	4	4	4	4	3	62	76	
24	3	3	3	4	4	3	4	4	2	3	4	3	4	4	4	4	4	4	4	67	76	
25	4	3	3	4	3	3	4	3	3	4	3	3	4	4	2	4	3	4	3	64	76	
26	3	3	4	3	3	3	3	3	3	3	3	3	3	3	3	4	3	4	3	61	76	
27	4	3	3	3	3	3	4	3	4	3	3	3	3	3	3	3	3	3	3	60	76	
28	3	4	3	4	4	3	4	3	3	2	4	3	4	3	4	3	3	3	4	64	76	
29	3	3	4	3	3	3	4	4	2	3	3	4	3	4	3	3	3	3	3	61	76	
30	3	4	4	3	3	3	4	3	2	2	3	3	3	4	3	3	3	4	3	61	76	
JUMLAH																					1825	2280

Berdasarkan ringkasan hasil pengujian dapat diperoleh persentase kelayakan sebagai berikut.

$$\text{Persentase Kelayakan (\%)} = \frac{\sum \text{Skor yang diobservasi}}{\sum \text{Skor yang diharapkan}} \times 100\%$$

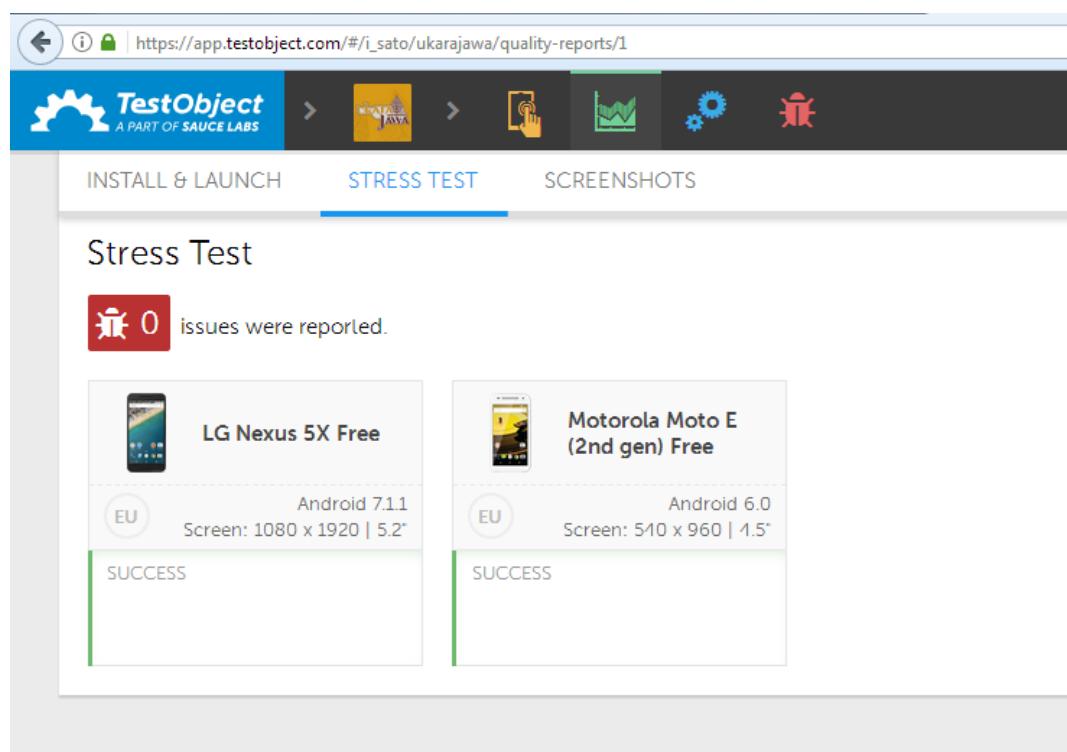
$$= \frac{1825}{2280} \times 100\%$$

$$= 80,04\%$$

Hasil perhitungan kelayakan adalah 80,04%. Hasil persentase tersebut kemudian dikonversikan menjadi nilai kualitatif. Berdasarkan hasil konversi nilai tersebut, pengujian *usability* termasuk dalam kategori “**Layak**”.

e. Pengujian *Reliability*

Pengujian *reliability* menggunakan *tools* yang disediakan www.testobject.com. Metode pengujian dilakukan dengan *stress testing* pada aplikasi. *Stress testing* dilakukan menggunakan dua perangkat yang disediakan secara gratis. Berikut ini hasil pengujian *reliability* (lihat Gambar 45).



Gambar 45. Hasil Pengujian *Reliability*

Berdasarkan hasil yang disajikan Gambar 45 selanjutnya dibuat ringkasan hasil pengujian *reliability* yang disajikan Tabel 27.

Tabel 27. Ringkasan Hasil Pengujian *Reliability*

No	Pertanyaan	Ketercapaian Hasil	
		Sukses	Gagal
1	Apakah <i>stress testing</i> aplikasi pada perangkat LG Nexus 5X berhasil dan tidak terjadi <i>error</i> ?	Ya	
2	Apakah <i>stress testing</i> aplikasi pada perangkat Motorola Moto E (2 nd gen) berhasil dan tidak terjadi <i>error</i> ?	Ya	

Berdasarkan Tabel 26 maka dapat diketahui persentase masing-masing butir pengujian.

$$\text{Sukses} = \frac{2}{2} \times 100\% = 100\%$$

$$\text{Gagal} = \frac{0}{2} \times 100\% = 0\%$$

Selanjutnya hasil persentase tersebut kemudian dikonversikan menjadi data kualitatif berdasarkan skala persentase kelayakan. Dari hasil konversi, menunjukkan pengujian *reliability* termasuk pada kategori "**Sangat Layak**". Hasil pengujian menunjukkan *stress testing* yang dilakukan terhadap aplikasi berhasil dan tidak terjadi *error*.

f. Pengujian Aspek Materi

Pengujian aspek materi dilakukan menggunakan kuesioner berupa *checklist* yang berisi daftar materi serta sampel penerjemahan pada aplikasi. Kuesioner ini diisi oleh dua ahli bidang Bahasa Jawa. Ringkasan analisis data hasil pengujian aspek materi disajikan pada Tabel 28.

Tabel 28. Analisis Data Hasil Pengujian Aspek Materi

NO PERTANYAAN	RESPONDEN		TOTAL SKOR	SKOR MAX	NO PERTANYAAN	RESPONDEN		TOTAL SKOR	SKOR MAX
	1	2				1	2		
1	1	1	2	2	19	1	1	2	2
2	1	1	2	2	20	1	1	2	2
3	1	1	2	2	21	1	1	2	2
4	1	1	2	2	22	1	1	2	2
5	1	1	2	2	23	1	1	2	2
6	1	1	2	2	24	1	1	2	2
7	1	1	2	2	25	1	1	2	2
8	1	1	2	2	26	1	1	2	2
9	1	1	2	2	27	1	1	2	2
10	1	1	2	2	28	1	1	2	2
11	1	1	2	2	29	1	1	2	2
12	1	1	2	2	30	1	1	2	2
13	1	1	2	2	31	1	1	2	2
14	1	1	2	2	32	1	1	2	2
15	1	1	2	2	33	1	1	2	2
16	1	1	2	2	34	1	1	2	2
17	1	1	2	2	35	1	1	2	2
18	1	1	2	2	36	1	1	2	2
JUMLAH			36	36	JUMLAH			36	36

Berdasarkan Tabel 28, selanjutnya dapat diperoleh persentase masing-masing butir pertanyaan.

$$\begin{aligned}
 \text{Persentase Kelayakan} &= \frac{\text{Total Skor 1} + \text{Total Skor 2}}{\text{Total Skor Max}} \times 100\% \\
 &= \frac{36+36}{36+36} \times 100\% \\
 &= \frac{72}{72} \times 100\% \\
 &= 100\%
 \end{aligned}$$

Berdasarkan perhitungan di atas diperoleh persentase **100%** dari pengujian aspek materi. Hasil persentase tersebut kemudian dikonversikan

menjadi nilai kualitatif berdasarkan skala persentase kelayakan. Berdasarkan hasil konversi nilai tersebut, menunjukkan bahwa pengujian aspek materi termasuk pada kategori “**Sangat Layak**”. Dari hasil tersebut, aplikasi Ukara Jawa sudah layak untuk digunakan pengambilan data.

B. Pembahasan

Aplikasi Ukara Jawa dikembangkan melalui tahap analisis, desain, implementasi, dan pengujian. Pada tahap analisis meliputi analisis fungsional dan analisis nonfungsional. Analisis fungsional dilakukan untuk mengumpulkan informasi terkait fungsi-fungsi yang dibutuhkan pada aplikasi Ukara Jawa. Analisis nonfungsional digunakan untuk mengumpulkan informasi yang dibutuhkan dalam tahap pengembangan aplikasi Ukara Jawa. Pada tahap desain terdiri atas desain *unified modelling language* (UML), desain sistem, dan desain *interface*. Desain UML digunakan untuk menggambarkan rancangan alur proses aplikasi terhadap sistem. Desain sistem digunakan untuk menggambarkan rancangan basis data serta alur kerja sistem yang digambarkan dengan *flowchart*. Pembuatan *flowchart* alur kerja sistem menggunakan *software* Microsoft Visio 2013. Desain *interface* digunakan untuk menggambarkan tampilan *interface* aplikasi. Pembuatan desain *interface* dengan menggunakan *software* Corel Draw X7. Pada tahap implementasi terdiri atas implementasi *layout interface* aplikasi dan implementasi logika pemrograman. Pembuatan *layout interface* aplikasi dan logika pemrograman menggunakan *software* Android Studio. Tahap pengujian meliputi pengujian kualitas berdasarkan standar ISO 25010 dan pengujian aspek materi. Pengujian berdasarkan standar ISO 25010 meliputi pengujian

functional suitability, pengujian *performance efficiency*, pengujian *compatibility*, pengujian *usability*, dan pengujian *reliability*. Hasil masing-masing pengujian kemudian dievaluasi sehingga dapat diketahui kelayakan aplikasi yang dikembangkan. Ringkasan hasil pengujian aplikasi dapat dilihat pada Tabel 29.

Tabel 29. Ringkasan Hasil Pengujian Aplikasi

Aspek	Hasil	Skala Kelayakan
<i>Functionality suitability</i>	Fungsi-fungsi pada aplikasi 100% dapat dijalankan dengan benar	Sangat Layak
<i>Performance efficiency</i>	Pengujian <i>time behaviour launching time</i> aplikasi diperoleh nilai terbesar 1261 ms dan pengujian <i>rendering time</i> aplikasi diperoleh nilai terbesar 149 ms. Pengujian <i>resource utilization CPU</i> diperoleh nilai terbesar 10%. Pengujian <i>resource utilization memory</i> diperoleh nilai terbesar 34 MB. Aplikasi berjalan dengan baik dan tidak terjadi <i>crash</i> dan <i>memory leak</i> .	Layak
<i>Compatibility</i>	Pengujian aspek <i>co-existence</i> diperoleh persentase sebesar 100% dan pengujian uji coba instalasi aplikasi pada berbagai sistem operasi dan	Sangat Layak

Aspek	Hasil	Skala Kelayakan
	jenis perangkat diperoleh persentase sebesar 100%	
<i>Usability</i>	Pengujian <i>usability</i> diperoleh persentase sebesar 80,04%	Layak
<i>Reliability</i>	Aplikasi berhasil melalui pengujian <i>stress testing</i> dan tidak terjadi <i>error</i> dengan persentase 100%	Sangat Layak
Materi	Materi yang disajikan aplikasi dinyatakan 100% layak	Sangat Layak

BAB V

SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil penelitian dan pembahasan, dapat disimpulkan sebagai berikut:

1. Penelitian ini menghasilkan aplikasi Ukara Jawa sebagai alat penerjemah Bahasa Indonesia dan Bahasa Jawa dengan *unggah-ungguh basa* pada *platform* Android. Aplikasi dikembangkan menggunakan *tool* Android Studio.
2. Hasil pengujian kualitas aplikasi Ukara Jawa yang meliputi aspek *functionality suitability, performance efficiency, compatibility, usability, reliability*, dan aspek materi adalah sebagai berikut.
 - a. Hasil pengujian kualitas aspek *functionality suitability* dilakukan oleh tiga ahli dan menunjukkan semua fungsi aplikasi dapat dijalankan dengan benar sehingga kualitas aplikasi mempunyai skala "**Sangat Layak**" dengan persentase **100%**.
 - b. Hasil pengujian kualitas aspek *performance efficiency* dengan *tools* dari <https://www.monkop.com/> menunjukkan skala kualitas "**Layak**" dengan tidak terjadinya *crash* dan *memory leak*. Penggunaan waktu yang dibutuhkan *launching* aplikasi paling tinggi sebesar 1261 ms (1,261 detik) dan waktu *rendering* yang dibutuhkan aplikasi paling tinggi sebesar 149 ms (0,149 detik). Penggunaan *CPU* yang digunakan aplikasi paling tinggi sebesar 10% dan masih berada pada

standar yang ditetapkan yaitu antara 5% sampai 15%. Penggunaan *memory* yang digunakan aplikasi paling tinggi sebesar 34 MB.

- c. Hasil pengujian kualitas aspek *compatibility* menunjukkan skala kualitas "**Sangat Layak**" karena aplikasi dapat berjalan bersamaan dengan aplikasi lain tanpa memberikan dampak yang merugikan serta dapat dijalankan pada berbagai sistem operasi dan jenis perangkat dengan tahapan *install*, *process*, dan *uninstall*.
- d. Hasil pengujian kualitas aspek *usability* dilakukan oleh 30 siswa SMK Muhammadiyah 2 Yogyakarta dengan persentase **80,04%** dan skala kualitas "**Layak**".
- e. Hasil pengujian kualitas aspek *reliability* dengan *tools* dari www.testobject.com dengan metode *stress testing* diperoleh persentase sebesar **100%** dan skala kualitas "**Sangat Layak**".
- f. Hasil pengujian kualitas aspek materi dilakukan oleh dua ahli Bahasa Jawa dan menunjukkan bahwa materi yang disajikan aplikasi memperoleh persentase sebesar **100%** dengan skala kualitas "**Sangat Layak**".

B. Saran

Pengembangan aplikasi Ukara Jawa masih terdapat banyak kekurangan sehingga perlu dikembangkan lagi. Peneliti memiliki saran untuk pengembangan berikutnya antara lain:

1. Sumber data kosakata yang digunakan aplikasi masih bersifat lokal sehingga untuk memperbarui atau menambah kosakata perlu ditambahkan fitur *update* kosakata.
2. Penerjemahan yang dilakukan masih terbatas pada penerjemahan berbasis kata demi kata sehingga untuk menerjemahkan frase kata menampilkan hasil yang tidak sesuai. Saran pengembangan selanjutnya ditambahkan algoritma yang dapat menerjemahkan berbasis frase kata.
3. Pengembangan aplikasi Ukara Jawa dapat diimplementasikan pada sistem operasi lain seperti iOS dan Windows Phone.

DAFTAR PUSTAKA

Afifah, N., Santoso, T. B., & Yuliana, M. (2010). Pembuatan Kamus Elektronik Kalimat Bahasa Indonesia dan Bahasa Jawa untuk Aplikasi Mobile Menggunakan Interpolation Search. *Seminar Proyek Akhir Jurusan Teknik Telekomunikasi*. Surabaya: Politeknik Elektronika Negeri Surabaya.

Agusta, L. (2009). Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani untuk Stemming Dokumen Teks Bahasa Indonesia. *Konferensi Nasional Sistem dan Informatika*. Bali.

Agustina, G. D. (2017). Developing An Android Based Application As Balinese Shadow Puppet Educational Media. Diambil kembali dari <http://journal.uny.ac.id/index.php/jptk/article/download/13387/9401>

Arikunto, S. (2010). *Prosedur Penelitian Suatu Pendekatan Praktik*. Jakarta: PT. Rineka Cipta.

Asmoko, Y. W. (2014). Pembentukan Verba Turunan Bahasa Jawa dengan Bahasa Indonesia Berdasarkan Kamus (Analisis Kontrastif). *Skripsi S1*. Yogyakarta: Universitas Negeri Yogyakarta.

David, A. B. (2011). *Mobile Application Testing (Best Practices to Ensure Quality)*. Amdocs.

Departemen Pendidikan Nasional. (2008). *Kamus Besar Bahasa Indonesia Pusat Bahasa Edisi Keempat*. Jakarta: PT. Gramedia Pustaka Utama.

Developers, A. (2017). *Dashboards: Platform Version*. Dipetik Juli 17, 2017, dari <https://developer.android.com/about/dashboards/index.html>

DI Marketing. (2016). Smartphone Usage In Indonesia. Dipetik Mei 7, 2017, dari <http://www.di-onlinesurvey.com/en/2016/07/07/smartphone-usage-in-indonesia/>

Farida, I. N. (t.thn.). Aplikasi Kamus Indonesia-Jawa Berbasis Android. *Nusantara of Engineering, Vol. 2* (No. 1).

Franke, D., Kowalewski, S., & Weise, C. (2012). A Mobile Software Quality Model. Diambil kembali dari <http://ieeexplore.ieee.org/document/6319241/>

Göth, B. R. (2015). *Testing techniques for mobile device applications*. Diploma Thesis, Masaryk University, Faculty of Informatics.

Harian Jogja. (2016). KONGRES BAHASA JAWA Ini Jalur Pelestarian Bahasa Jawa Hasil Keputusan KBJ 2016. Dipetik Mei 7, 2017, dari <http://www.harianjogja.com/baca/2016/11/14/kongres-bahasa-jawa-ini-jalur-pelestarian-bahasa-jawa-hasil-keputusan-kbj-2016-768608>

Harjawiyyana, H., & Supriya, T. (2001). *Marsudi Unggah-Ungguh Basa Jawa*. Kanisius: Yogyakarta.

Hartman, R., Rokitta, C., & Peake, D. (2013). *Oracle Application Express for Mobile Web Applications*. Apress.

Huang, L., Ai, J., & Wang, J. (2014). Combined Reliability Test for Software-hardware. *International Conference on Cyber Technology in Automation, Control and Intelligent Systems*. Diambil kembali dari <http://ieeexplore.ieee.org/document/6917543/>

Istiyanto, J. E. (2013). *Pemrograman Smart Phone Menggunakan SDK Android dan Hacking Android*. Yogyakarta: Graha Ilmu.

Juhara, Z. P. (2016). *Panduan Lengkap Pemrograman Android*. Yogyakarta: CV. ANDI.

Kendall, K. E., & Kendall, J. (2003). *Analisis dan Perancangan Sistem*. Jakarta: Penerbit Indeks.

Krishandini. (2011). *Analisis kontrastif afiksasi verba Bahasa Jawa dengan Bahasa Indonesia*. Hasil Penelitian. Diambil kembali dari <http://repository.ipb.ac.id/handle/123456789/53640>

Krisnayuni, K. (2017). Developing Project Based Learning E-Module For The Course of Video Editing. Diambil kembali dari <http://journal.uny.ac.id/index.php/jptk/article/download/13389/9405>

Krjogja. (2016). Kongres Bahasa Jawa VI. Dipetik Mei 7, 2017, dari http://krjogja.com/web/news/read/15182/Kongres_Bahasa_Jawa_VI

Lewis, J. (1995). IBM Computer Usability Satisfaction Questionnaires. *International Journal of Human-Computer Interaction*, 57-58.

Little Eye Labs. (2013). *Little Eye Labs Data Sheet v 2.4*. Diambil kembali dari <http://cdn.littleeye.co/downloads/LittleEye-DataSheet.pdf?v=2.4.2014.01.09>

Madia, M. (2016). Stemming Bahasa Jawa Untuk Mencari Akar Kata Dalam Bahasa Jawa Dengan Aturan Analisis Kontrastif Afiksasi Verba. *Skripsi S1*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.

Miguel, J. P., Mauricio, D., & Rodriguez, G. (2014). A Review of Software Quality Models for the Evaluation of Software Products. *International Journal of Software Engineering & Applications*, 31-53.

Motogna, S., et al. (2017). An Approach to Assess Maintainability Change. Diambil kembali dari <http://ieeexplore.ieee.org/document/7501279/>

Nababan, R. (1999). *Teori Menerjemah Bahasa Inggris*. Yogyakarta: Pustaka Pelajar.

Niknejad, A. (2011). *A Quality Evaluation of an Android Smartphone Application*. University of Gothenburg, Department of Applied Information Technology.

Peng, B., Jinming, Y., & Tianzhou, C. (2012). The Android Application Development College. *International Conference on High Performance Computing and Communications*, (pp. 1677-1681).

Poedjosoedarmo, S., et al. (1979). *Morfologi Bahasa Jawa*. Jakarta: Pusat Pembinaan dan Pengembangan Bahasa Departemen Pendidikan dan Kebudayaan.

Poedjosoedarmo, S., et al. (2013). *Tingkat Tutur Bahasa Jawa*. (R. Sukesti, Penyunt.) Yogyakarta: Kementerian Pendidikan dan Kebudayaan Badan Pengembangan dan Pembinaan Bahasa Balai Bahasa Provinsi Daerah Istimewa Yogyakarta.

Rohmadi, M., & Hartono, L. (2011). *Kajian Bahasa, Sastra, dan Budaya Jawa : Teori dan Pembelajarannya*. Surakarta: Pelangi Press.

Rosa A.S., & Shalahuddin, M. (2014). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.

Rustrini, N. P. (2016). A Descriptive Survey On Smartphones Features For Supporting The Academic Activities At Universitas Pendidikan Ganesha. Diambil kembali dari <http://journal.uny.ac.id/index.php/jptk/article/view/13188>

Safaat, N. (2015). *Android - Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.

Santosa, B. P. (2016). Aplikasi Penterjemah Kalimat Bahasa Indonesia ke Bahasa Jawa Disertai Transliterasi Aksara Jawa Berbasis Web dengan Metode Analisis Kontrastif Averbia dan Decision Tree. *Skripsi S1*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.

Setiyanto, A. B. (2007). *Parama Sastra Bahasa Jawa*. Yogyakarta: Panji Pustaka.

Simarmata, J. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta: Penerbit ANDI.

StatCounter, G. (2017). *Operating System Market Share in Indonesia*. Dipetik Februari 8, 2017, dari <http://gs.statcounter.com/os-market-share/all/indonesia#monthly-201601-201701>

Sudaryanto. (1994). *Pemanfaatan Potensi Bahasa*. Yogyakarta: Gadjah Mada.

Sugiyono. (2012). *Metode Penelitian Pendidikan (Pendekatan Kuantitatif, Kualitatif, dan R&D)*. Bandung: CV ALFABETA.

Suryawinata, Z., & Hariyanto, S. (2003). *Translation: Bahasan Teori dan Penuntun Praktis Menerjemahkan*. Yogyakarta: Kanisius.

Suteja, I. B. (2012). Aplikasi Translator Bahasa Indonesia-Bahasa Jawa dengan Unggah-Ungguh Bassa Berbasis Web. *Skripsi S1*. Yogyakarta: Universitas Negeri Yogyakarta.

Tarigan, H. G. (1989). *Pengajaran Kosakata*. Bandung: Penerbit Angkasa.

Testing Experience. (2012). Mobile App Testing. Diambil kembali dari http://www.hettel.dk/lib/exe/fetch.php?media=pdf:testingexperience:testingexperience19_09_12.pdf

Ventola, C. (2014). Mobile devices and apps for health care professionals: uses and benefits. *Pharmacy and Therapeutics*, 356-364. Diambil kembali dari <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4029126/>

Wedhawati, et al. (2006). *Tata Bahasa Jawa Mutakhir*. Yogyakarta: Kanisius.

Wibawa, A. P., Kadarisman, A. E., & Mahmudy, W. F. (2013). Indonesian-to-Javanese Machine Translation. *International Journal of Innovation, Management and Technology*, Vol. 4 (No. 4). Diambil kembali dari <http://www.ijimt.org/papers/440-M0004.pdf>

Wibawa, S., et al. (2004). *Buku Pegangan Kuliah Mata Pelajaran Bahasa Jawa*. Yogyakarta: Pendidikan Guru Sekolah Dasar Fakultas Ilmu Pendidikan Universitas Negeri Yogyakarta.

LAMPIRAN

Lampiran 1. Surat Keputusan Pengangkatan Pembimbing Skripsi

**KEPUTUSAN DEKAN
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 169 /ELK/Q-I/IX/2016
TENTANG
PENGANGKATAN PEMBIMBING TUGAS AKHIR SKRIPSI
BAGI MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

Menimbang : 1. Bawa sehubungan dengan telah dipenuhi syarat untuk penulisan Tugas Akhir Skripsi bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta, perlu diangkat pembimbing.
2. Bawa 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 93 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/UN34/KP/2011.

M E M U T U S K A N

Menetapkan

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

Nama Pembimbing : Totok Sukardiyono, MT
Bagi mahasiswa :
Nama/No.Mahasiswa : **Imam Santoso /10520244057**
Jurusan/Prodi : Pendidikan Teknik Elektronika / Pendidikan Teknik Informatika
Judul Skripsi : *Pengembangan Aplikasi Penerjemah Bahasa Indonesia - Bahasa Jawa pada Platform Berbasis Android*

Kedua : Dosen pembimbing diserahi 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.



Tembusan Yth :

1. Wakil Dekan II. FT UNY
2. Ketua Jurusan Pendidikan Teknik Elektronika
3. Kasub. Bag. Pendidikan FT UNY
4. Yang bersangkutan

Lampiran 2. Surat Izin Penelitian



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS NEGERI YOGYAKARTA
FAKULTAS TEKNIK

Alamat: Karangmalang, Yogyakarta 55281
Telp. (0274) 568168 psw: 276, 289, 292. (0274) 586734. Fax. (0274) 586734:
Website : <http://ft.uny.ac.id>, email : ft@uny.ac.id, teknik@uny.ac.id



Certificate No. QSC 00592

No : 756/H34/PL/2017
Lamp : -
Hal : Ijin Penelitian

9 Mei 2017

Yth.

1. Kepala Majelis Pendidikan Dasar dan Menengah Pimpinan Daerah Muhammadiyah Kota Yogyakarta
2. Kepala Sekolah SMK Muhammadiyah 2 Yogyakarta

Dalam rangka pelaksanaan Tugas Akhir Skripsi kami mohon dengan hormat bantuan Saudara memberikan ijin untuk melaksanakan penelitian dengan judul Pengembangan Aplikasi Penerjemah Bahasa Indonesia - Bahasa Jawa Pada Platform Berbasis Android, bagi Mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta tersebut di bawah ini:

No	Nama	No. Mhs.	Program Studi	Lokasi
1.	Iman Santoso	10520244057	Pend. Teknik Informatika	SMK Muhammadiyah 2 Yogyakarta

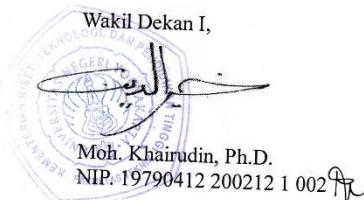
Dosen Pembimbing/Dosen Pengampu

Nama : Totok Sukardiyono, MT
NIP : 19670930 199303 1 005

Adapun pelaksanaan penelitian dilakukan mulai 10 Mei - 20 Juni 2017

Demikian permohonan ini, atas bantuan dan kerjasama yang baik selama ini, kami mengucapkan terima kasih.

Wakil Dekan I,



Moh. Khairudin, Ph.D.

NIP. 19790412 200212 1 002

Tembusan :
Ketua Jurusan

Lampiran 3. Surat Rekomendasi Izin Penelitian PDM


**MAJELIS PENDIDIKAN DASAR DAN MENENGAH
PIMPINAN DAERAH MUHAMMADIYAH KOTA YOGYAKARTA**
Jalan Sultan Agung 14, Telepon (0274)375917, Faks. (0274) 411947, Yogyakarta 55151
e-mail: dikdasmenpdm_yk@yahoo.com

IZIN PENELITIAN/SKRIPSI/OBSERVASI/TESIS
No. : 446/REK/III.4/F/2017

Setelah membaca surat dari : **Fakultas Teknik Universitas Negeri Yogyakarta.**
No. : 756/H34/PL/2017 Tgl. : 9 Mei 2017
Perihal : **Surat Izin Penelitian**

dan berdasar Putusan Sidang Majelis Dikdasmen PDM Kota Yogyakarta, hari **Senin** tanggal **18 Sya'ban 1438 H**, bertepatan tanggal **15 Mei 2017 M** yang salah satu agenda sidangnya membahas pemberian izin penelitian/praktek kerja/observasi, maka dengan ini kami memberikan izin kepada:

Nama Terang : **IMAN SANTOSO** NIM. **10520244057**
Pekerjaan : Mahasiswa pada prodi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta
alamat **Karangmalang Yogyakarta**
Pembimbing : **Totok Sukardiyo, M.T.**

untuk melakukan observasi/penelitian/pengumpulan data dalam rangka menyusun Skripsi :
Tentang : **PENGEMBANGAN APLIKASI PENERJEMAH BAHASA INDONESIA - BAHASA JAWA PADA PLATFORM BERBASIS ANDROID**
Lokasi : **SMK Muhammadiyah 2 Yogyakarta**

dengan ketentuan sebagai berikut:

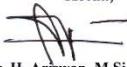
1. Menyerahkan tembusan surat ini kepada pejabat yang dituju.
2. Wajib menjaga rasa tertib dan menaati ketentuan-ketentuan yang berlaku di sekolah/setempat.
3. Wajib memberi laporan hasil penelitian/praktek kerja/observasi dalam bentuk CD kepada Majelis Pendidikan Dasar dan Menengah Pimpinan Daerah Muhammadiyah Kota Yogyakarta.
4. Izin ini tidak dibolehkan dilakukan untuk tujuan yang dapat mengganggu kestabilan Persyarikatan dan hanya diperlukan untuk keperluan ilmiah.
5. Surat izin ini dapat diujukan kembali untuk mendapat perpanjangan bila di-perlukan.
6. Surat izin ini dapat dibatalkan sejak si-waktu bila tidak dipenuhi ketentuan-ketentuan tersebut di atas.

MASA BERLAKU3 (TIGA) BULAN :
16-05-2017 sampai dengan 16-08-2017

Tanda tangan Pemegang Izin,

Iman Santos

Yogyakarta, 16 Mei 2017

Ketua,

Dr. H. Ariswan, M.Si., DEA
NBM. 820.325

Sekretaris,

Bilang. S.Fd., M.Eng
NBM. 728.558



Tembusan:

1. PDM Kota Yogyakarta.
2. Wk.Dekan I FT UNY
3. Kepala SMK Mu. 2 Yk

Lampiran 4. Surat Permohonan Validasi Instrumen TAS

Hal : Permohonan Validasi Instrumen TAS
Lampiran : 1 Bendel

Kepada Yth,
Bapak Muhammad Izzuddin Mahali, M.Cs.
Dosen Jurusan Pendidikan Teknik Elektronika dan Informatika
Di Fakultas Teknik UNY

Sehubungan dengan rencana pelaksanaan Tugas Akhir Skripsi (TAS), dengan ini
saya :

Nama : Iman Santoso
NIM : 10520244057
Program Studi : Pendidikan Teknik Informatika
Judul TAS : Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa
Jawa Pada Platform Berbasis Android

dengan hormat mohon Bapak berkenan memberikan validasi terhadap instrumen
penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini saya
lampirkan: (1) proposal TAS, (2) kisi-kisi instrumen penelitian TAS, dan (3) draf instrumen
penelitian TAS.

Demikian permohonan saya, atas bantuan dan perhatian Bapak diucapkan terima
kasih.

Yogyakarta, 24 Mei 2017

Pemohon,



Iman Santoso

NIM. 10520244057

Mengetahui,

Kaprodi Pend. Teknik Informatika



Handaru Jati, Ph.D.

NIP. 19740511 199903 1 002

Pembimbing TAS,



Drs. Totok Sukardiyono, M.T.

NIP. 19670930 199303 1 005

Hal : Permohonan Validasi Instrumen TAS
Lampiran : 1 Bendel

Kepada Yth,
Bapak Nurkhamid, Ph.D.
Dosen Jurusan Pendidikan Teknik Elektronika dan Informatika
Di Fakultas Teknik UNY

Sehubungan dengan rencana pelaksanaan Tugas Akhir Skripsi (TAS), dengan ini saya :

Nama : Iman Santoso
NIM : 10520244057
Program Studi : Pendidikan Teknik Informatika
Judul TAS : Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa
Pada Platform Berbasis Android

dengan hormat mohon Bapak berkenan memberikan validasi terhadap instrumen penelitian TAS yang telah saya susun. Sebagai bahan pertimbangan, bersama ini saya lampirkan: (1) proposal TAS, (2) kisi-kisi instrumen penelitian TAS, dan (3) draf instrumen penelitian TAS.

Demikian permohonan saya, atas bantuan dan perhatian Bapak diucapkan terima kasih.

Yogyakarta, 30 Mei 2017

Pemohon,



Iman Santoso
NIM. 10520244057

Mengetahui,

Kaprodi Pend. Teknik Informatika



Handaru Jati, Ph.D.

NIP. 19740511 199903 1 002

Pembimbing TAS,



Drs. Totok Sukardiyono, M.T.
NIP. 19670930 199303 1 005

Lampiran 5. Surat Keterangan Validasi Instrumen TAS

SURAT PERNYATAAN VALIDASI INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI

Saya yang bertanda tangan di bawah ini :

Nama : Muhammad Izzuddin Mahali, M.Cs.
NIP : 19841209 201504 1 001
Jurusan : Pendidikan Teknik Elektronika dan Informatika

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa :

Nama : Iman Santoso
NIM : 10520244057
Program Studi : Pendidikan Teknik Informatika
Jurusan : Pendidikan Teknik Elektronika dan Informatika
Judul TAS : Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa
pada Platform berbasis Android

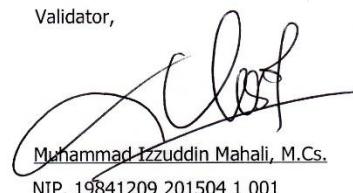
Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan :

Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan dengan saran/perbaikan
sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 15. Juni 2017.....

Validator,



Muhammad Izzuddin Mahali, M.Cs.

NIP. 19841209 201504 1 001

Catatan :

Beri tanda √

**SURAT PERNYATAAN VALIDASI
INSTRUMEN PENELITIAN TUGAS AKHIR SKRIPSI**

Saya yang bertanda tangan di bawah ini :

Nama : Nurkhamid, Ph.D.
NIP : 19680707 199702 1 001
Jurusan : Pendidikan Teknik Elektronika dan Informatika

menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa :

Nama : Iman Santoso
NIM : 10520244057
Program Studi : Pendidikan Teknik Informatika
Jurusan : Pendidikan Teknik Elektronika dan Informatika
Judul TAS : Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa
pada Platform berbasis Android

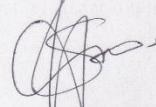
Setelah dilakukan kajian atas instrumen penelitian TAS tersebut dapat dinyatakan :

Layak digunakan untuk penelitian
 Layak digunakan dengan perbaikan
 Tidak layak digunakan untuk penelitian yang bersangkutan dengan saran/perbaikan
sebagaimana terlampir.

Demikian agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 3 Juli 2017

Validator,



Nurkhamid, Ph.D.

NIP. 19680707 199702 1 001

Catatan :

Beri tanda √

Lampiran 6. Hasil Validasi Instrumen TAS

Hasil Validasi Instrumen Penelitian TAS			
Nama Mahasiswa	: Iman Santoso	NIM	: 10520244057
Judul TAS	: Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android		
No	Variabel	Saran/Tanggapan	
1	Perbaikan beberapa kalimat yang belum sesuai	di revisi dan disesuaikan	
2.	Aspek Functionalit, termasuk keterangannya	di tambahkan keterangan	
	Uji yang belum ada	Uji yang belum mewujud	
Komentar Umum/Lain-lain :			
Yogyakarta, Validator, 15 Januari 2017			
 Muhammad Izzuddin Mahali, M.Cs. NIP. 19841209 201504 1 001			

Hasil Validasi Instrumen Penelitian TAS

Nama Mahasiswa	: Iman Santoso	NIM	: 10520244057
Judul TAS	: Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android		

Yogyakarta,
Validator,


Nurkhamid, Ph.D.
NIP. 19680707 199702 1 001

Lampiran 7. Sampel Kuesioner Instrumen *Functionality Suitability*

INSTRUMEN *Functionality Suitability* Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android

Nama : Ar Riyani, S.Pd
Pekerjaan : Guru Produktif TKJ
Petunjuk :

1. Instrumen ini diciptakan oleh orang yang memiliki keahlian di bidang *software engineering*.
2. Instrumen ini digunakan untuk menguji fungsi-fungsi pada aplikasi.
3. Berilah tanda *check* (✓) pada kolom Ketercapaian.
4. Berikan komentar dan saran pada bagian yang telah disediakan.

No	Aktivitas/Menu	Skenario Uji	Pertanyaan	Ketercapaian	
				Ya	Tidak
1	Membuka Aplikasi	Memilih ikon <i>launcher</i> aplikasi Ukara Jawa	Apakah halaman utama dapat ditampilkan dan muncul semua menu (Terjemahan, Bantuan, Info Pengembang, dan Keluar) dengan benar ?	✓	
2	Halaman Utama	Memilih menu Terjemahan	Apakah pengguna dapat membuka menu Terjemahan ?	✓	

No	Aktivitas/Menu	Skenario Uji	Pertanyaan	Ketercapaian	
				Ya	Tidak
	Halaman Utama (Lanjutan)	Menu Terjemahan (menekan tombol <i>back</i>)	Apakah pengguna dapat kembali ke halaman utama dengan menekan tombol <i>back</i> ?	✓	
		Memilih menu Bantuan	Apakah halaman Bantuan Penggunaan dapat ditampilkan dengan benar ?	✓	
		Memilih menu Pengembang	Apakah halaman Info Pengembang dapat ditampilkan dengan benar ?	✓	
		Memilih menu Keluar	Ketika memilih menu Keluar, apakah muncul <i>popup</i> konfirmasi untuk keluar ?	✓	
			Ketika menekan tombol "Tidak", apakah pengguna dapat kembali ke Halaman Utama ?	✓	
			Apakah pengguna dapat mengakhiri penggunaan aplikasi ketika menekan tombol "Ya" ?	✓	
3	Menu Terjemahan	Memilih bahasa asal terjemahan	Apakah bahasa asal terjemahan (Indonesia, <i>ngoko lugu</i> , <i>ngoko alus</i> , <i>krama lugu</i> , dan <i>krama alus</i>) ditampilkan dengan benar ?	✓	

Komentar dan Saran

Sebuah bagus, mudah dioperasikan, user friendly. Database dibuat / diperbaruiak form katanya

Yogyakarta, 12 Juli 2019

Validator,



Apriyanti, S.Pd

NIM. 121 0550

Lampiran 8. Sampel Kuesioner Instrumen Uji Materi

INSTRUMEN UJI MATERI	
Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android	
Nama	: <u>Yunita, S. Pd</u>
Pekerjaan	: <u>Guru Bahasa Jawa</u>
Petunjuk	
<p>1. Validasi ini dilakukan oleh orang yang memiliki keahlian di bidang Bahasa Jawa.</p> <p>2. Validasi ini digunakan untuk menguji isi materi pada aplikasi yang mencakup aspek kemanfaatan dan isi materi.</p> <p>3. Berilah tanda <i>check</i> (✓) pada kolom Penilaian.</p> <p>4. Berikan komentar dan saran pada bagian yang telah disediakan.</p>	

No	Pertanyaan	Penilaian	
		Ya	Tidak
1	Apakah petunjuk Bahasa <i>ngoko lugu, ngoko alus, krama lugu dan krama alus</i> sudah sesuai ?	✓	
2	Apakah petunjuk penggunaan dapat dipahami dengan mudah ?	✓	
	Berikut ini pertanyaan untuk nomor (3-7). Apakah terjemahan dari Bahasa Indonesia sudah sesuai dengan Bahasa <i>ngoko alus</i> ?		
3	Bahasa Indonesia : Bapak sedang bepergian. Terjemahan Ngoko Alus : <i>Bapak lagi tindakan.</i>	✓	
4	Bahasa Indonesia : Pak Lurah yang baru itu namanya siapa ? Terjemahan Ngoko Alus : <i>Pak Lurah sing anyar kuwi astmane sapa ?</i>	✓	
5	Bahasa Indonesia : Bapak sudah pulang dari kantor. Terjemahan Ngoko Alus : <i>Bapak wis kondur saka kantor.</i>	✓	
6	Bahasa Indonesia : Surat dari Pak camat sudah diambil Mas Bayu. Terjemahan Ngoko Alus : <i>Layang saka Pak Camat wis dipundhut Mas Bayu.</i>	✓	
7	Bahasa Indonesia : Bapak disuruh Pak RT datang ke rumahnya. Terjemahan Ngoko Alus : <i>Bapak diaturi Pak RT rawuh menyang daleme.</i>	✓	
	Berikut ini pertanyaan untuk nomor (8-12). Apakah terjemahan dari Bahasa <i>ngoko lugu</i> sudah sesuai dengan Bahasa <i>ngoko alus</i> ?		
8	Bahasa Ngoko Lugu : <i>Bapak arep lunga nang Surabaya.</i> Terjemahan Ngoko Alus : <i>Bapak arep tindak wonten Surabaya.</i>	✓	

No	Pertanyaan	Penilaian	
		Ya	Tidak
9	Bahasa Ngoko Lugu : Mau esuk Pak Guru wis teka nang sekolah. Terjemahan Ngoko Alus : Mau esuk Pak Guru wis rawuh wonten sekolah.	✓	
10	Bahasa Ngoko Lugu : Simbah saiki lagi lara. Terjemahan Ngoko Alus : Simbah saiki lagi gerah.	✓	
11	Bahasa Ngoko Lugu : Ibu mau ngomong yen Bapak lagi lunga karo Mas Raka. Terjemahan Ngoko Alus : Ibu mau ngandika yen Bapak lagi tindak karo Mas Raka.	✓	
12	Bahasa Ngoko Lugu : Pakdhe lagi ngungokake radio. Terjemahan Ngoko Alus : Pakdhe lagi midhangetake radio.	✓	
Berikut ini pertanyaan untuk nomor (13-17). Apakah terjemahan dari Bahasa Indonesia sudah sesuai dengan Bahasa krama lugu ?			
13	Bahasa Indonesia : Saya ingin pergi ke sekolah. Terjemahan Krama Lugu : Kulé badhé kesah dhatereng sekolah.	✓	
14	Bahasa Indonesia : Sekarang aku sedang ada di rumah. Terjemahan Krama Lugu : Samenika kulé saweg wonten ing griya.	✓	
15	Bahasa Indonesia : Kamu tadi mengambil apa di meja ? Terjemahan Krama Lugu : Sampeyan wau mendhet menapa ing meja ?	✓	
16	Bahasa Indonesia : Nas Cahyo besok akan membelikan Bapak motor baru. Terjemahan Krama Lugu : Mas Cahyo benjing badhé numbasaken Bapak motor enggal.	✓	
17	Bahasa Indonesia : Mbak Rama, kamu tadi dicari Bapak. Terjemahan Krama Lugu : Mbak Rama, sampeyan wau dipunpadosi Bapak.	✓	

No	Pertanyaan	Penilaian		
		Ya	Tidak	
Berikut ini pertanyaan untuk nomor (18-22). Apakah terjemahan dari Bahasa <i>Ngoko lugu</i> sudah sesuai dengan Bahasa <i>krama lugu</i> ?				
18	Bahasa Ngoko Lugu : Kowe sida teka nang omahku ?	✓		
19	Terjemahan Krama Lugu : Sampeyan siyos dhateng ing griya kula ?	✓		
20	Bahasa Ngoko Lugu : Kowe mau njupuk apa nang meja ?	✓		
21	Terjemahan Krama Lugu : Sampeyan wau mendhet menapa ing meja ?	✓		
22	Bahasa Ngoko Lugu : Aku ketemu kowe ana ing desa.	✓		
23	Terjemahan Krama Lugu : Kula kepanggih sampeyan wonten ing desa.	✓		
24	Bahasa Ngoko Lugu : Aku diwenehi undangan saka Pak RW.	✓		
	Terjemahan Krama Lugu : Kula dipunparingi undangan saking Pak RW.	✓		
	Bahasa Ngoko Lugu : Aku oleh melu kowe apa ora ?	✓		
	Terjemahan Krama Lugu : Kula angosat tumut sampeyan menapa boten ?	✓		
Berikut ini pertanyaan untuk nomor (23-27). Apakah terjemahan dari Bahasa Indonesia sudah sesuai dengan Bahasa <i>krama alus</i> ?				
23	Bahasa Indonesia : Bu Eri pergi ke pasar akan membeli sayur.	✓		
24	Terjemahan Krama Alus : Bu Eri tindak dhateng péken badhe mundhut sayur.	✓		
	Bahasa Indonesia : Bapak sedang tidur, katanya akan bangun nanti jam lima.			
	Terjemahan Krama Alus : Bapak saweg sare, ngandikanipun badhe wungu mangke jam gansal.	✓		

No	Pertanyaan	Penilaian	
		Ya	Tidak
25	Bahasa Indonesia : <i>Permisi Pak, apa Bapak yang mempunyai mobil itu ?</i> Terjemahan Krama Alus : <i>Nuvun sewu Pak, menpa Bapak ingkang kagungan motor menika ?</i>	✓	
26	Bahasa Indonesia : Ibu dan Bapak sedang makan nasi goreng. Terjemahan Krama Alus : <i>Ibu kaliyan Bapak saweg dhahar sekul goreng.</i>	✓	
27	Bahasa Indonesia : Saya menyuruh Pak RT supaya bisa datang di pengajian. Terjemahan Krama Alus : <i>Kula ngaturi Pak RT kersanipun saged rawuh wonten pengaosan.</i>	✓	
28	Berikut ini pertanyaan untuk nomor (28-32). Apakah terjemahan dari Bahasa ngoko lugu sudah sesuai dengan Bahasa krama alus ? Bahasa Ngoko Lugu : <i>Bapak lagi nukokake iwak nang pasar.</i>	✓	
29	Terjemahan Krama Alus : <i>Bapak saweg mundhutaken ulam wonten peken.</i>	✓	
30	Bahasa Ngoko Lugu : <i>Ibu arep lunga nang endi ?</i>	✓	
31	Terjemahan Krama Alus : <i>Ibu badine tindak wonten pundi ?</i>	✓	
32	Bahasa Ngoko Lugu : <i>Aku diakon Bapak tuku beras nang pasar.</i>	✓	
33	Terjemahan Krama Alus : <i>Kula dipundhawuin Bapak mundhut was wonten peken.</i>	✓	
34	Bahasa Ngoko Lugu : <i>Buku iki regane pira ?</i>	✓	
35	Terjemahan Krama Alus : <i>Buku menika reginipun pinten ?</i>	✓	
36	Bahasa Ngoko Lugu : <i>Adhiku lagi turu jam telu bengi.</i>	✓	
37	Terjemahan Krama Alus : <i>Adhi kula saweg tiem jam tiga dulu.</i>	✓	

No	Pertanyaan	Penilaian	
		Ya	Tidak
33	Apakah hasil yang disajikan aplikasi dapat membantu mempelajari penggunaan bahasa <i>ngoko lugu</i> ?	✓	
34	Apakah hasil penerjemahan dapat membantu pengguna mempelajari bahasa <i>ngoko alus</i> ?	✓	
35	Hasil penerjemahan ke bahasa <i>krama lugu</i> apakah dapat membantu pengguna mempelajari bahasa <i>krama lugu</i> ?	✓	
36	Apakah pengguna dapat terbantu dengan hasil penerjemahan ke bahasa <i>krama alus</i> ?	✓	

Komentar dan Saran

Yogyakarta, 11 Juli 2017

Validator,


-f₄₅

(Yunianti)

Lampiran 9. Sampel Kuesioner Instrumen *Usability*

INSTRUMEN *USABILITY*

Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android

A. IDENTITAS RESPONDEN

Nama : Oktiwi
Kelas : X Aka

B. PETUNJUK PENGISIAN

1. Mohon dengan hormat bantuan dari ketersediaan Anda untuk menjawab seluruh pernyataan yang ada.
2. Sebelum menjawab, mohon dibaca dengan teliti setiap pernyataan pada angket.
3. Jika terdapat pernyataan yang kurang jelas, silakan bertanya kepada peneliti.
4. Berilah tanda *check* (✓) pada kolom penilaian dengan pendapat Anda sesuai dengan keadaan yang sebenarnya.
5. Jawaban diberikan pada kolom penilaian yang sudah disediakan dengan skala penilaian:
SS = Sangat Setuju
S = Setuju
TS = Tidak Setuju
STS = Sangat Tidak Setuju

No	Pernyataan	Penilaian			
		STS	TS	S	SS
1	Secara keseluruhan, saya merasa puas dengan kemudahan penggunaan aplikasi ini.	✗			✓
2	Cara penggunaan aplikasi ini sangat mudah.			✓	
3	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan lebih mudah ketika menggunakan aplikasi ini.			✓	

No	Pernyataan	Penilaian			
		STS	TS	S	SS
4	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan cepat ketika menggunakan aplikasi ini.			✓	
5	Saya dapat memenuhi kebutuhan belajar Bahasa Jawa (terutama tentang <i>unggah-ungguh basa</i>) dengan efisien menggunakan aplikasi ini.			✓	
6	Saya merasa nyaman menggunakan aplikasi ini.			✓	
7	Aplikasi ini sangat mudah dipelajari.			✓	
8	Saya merasa lebih produktif ketika menggunakan aplikasi ini.			✓	
9	Jika terjadi <i>error</i> , aplikasi ini memberikan pesan kesalahan yang jelas dan memberitahu cara mengatasinya.				✓
10	Kapanpun saya melakukan kesalahan, saya bisa mengakses kembali aplikasi dengan cepat.			✓	
11	Informasi yang disediakan aplikasi ini sangat jelas.			✓	
12	Sangat mudah menemukan informasi yang saya butuhkan pada aplikasi ini.			✓	
13	Informasi yang diberikan aplikasi ini sangat mudah dipahami.			✓	
14	Informasi yang diberikan sangat efektif dalam membantu saya menggunakan aplikasi ini.			✓	
15	Tata letak informasi yang ditampilkan aplikasi ini sangat jelas.			✓	
16	Tampilan aplikasi ini menyenangkan.			✓	
17	Saya suka menggunakan tampilan aplikasi seperti ini.			✓	
18	Aplikasi ini memiliki fitur dan kemampuan sesuai keinginan saya.			✓	
19	Secara keseluruhan, saya sangat puas dengan aplikasi ini.			✓	

C. SARAN

Mungkin bisa ditambah Pengaturan warna background
yg dapat diubah

Atas perhatian dan kerjasama Anda pada penelitian ini diucapkan terima kasih.

Yogyakarta, 17 Juli 2017



Lampiran 10. Kartu Bimbingan

<p>JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA Alamat : Kampus Karangmalang Yogyakarta 55281 Telp. : (0274) 554686 ; (0274) 586168 ext. 293</p>			
<p>KARTU BIMBINGAN SKRIPSI (Untuk Mahasiswa) FRM/EKA/05-00 25 Januari 2008</p>			
<p>Nama Mahasiswa : Iman Santoso No. Mahasiswa : 10520299057 E-mail : isato.iman@gmail.com Program Studi : Pendidikan Teknik Informatika Jenjang : S1 Kelas : G Dosen Pembimbing : Totok Subardiyono, M.T. HP : 085771356090 Judul : Pengembangan Aplikasi Penerjemah Bahasa Indonesia-Bahasa Jawa pada Platform berbasis Android</p>			
No	Tanggal	Uraian Bimbingan	Tandatangan Pembimbing
1.	3/02/17	Bimbingan Bab I	
2.	15/02/17	Revisi & Bimbingan Bab I	
3.	9/03/17	Bimbingan Bab I & II	
4.	12/03/17	Bimbingan Bab III	
5.	4/05/17	Revisi bimbingan Bab III	
6.	17/05/17	Konsultasi Instrumen Penelitian	
7.	7/06/17	Bimbingan Bab IV & V	
8.	28/06/17	Revisi bimbingan Bab IV & V	
9.	21/07/17	Revisi Bab VI	
10.	28/07/17	Melengkapi naskah T4S	

Rekomendasi Pembimbing :
1. Mahasiswa yang bersangkutan siap untuk diuji.

Tanggal Persetujuan : 28-7-2017 Tandatangan Dosen Pembimbing : 

2. Kartu Bimbingan ini wajib dilampirkan pada saat pendaftaran ujian Skripsi.