

**PENGEMBANGAN APLIKASI ANDROID *MOBILE DICTIONARY AND
EMULATOR* PERINTAH DASAR *COMMAND LINE INTERFACE LINUX
DAN WINDOWS***

TUGAS AKHIR SKRIPSI

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



Oleh :

Dian Kartika Sari
NIM 14520244012

PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2018

LEMBAR PERSETUJUAN

Tugas Akhir Skripsi dengan Judul

PENGEMBANGAN APLIKASI ANDROID *MOBILE DICTIONARY AND EMULATOR* PERINTAH DASAR *COMMAND LINE INTERFACE LINUX DAN WINDOWS*

Disusun oleh:

DIAN KARTIKA SARI

NIM 14520244012

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

Yogyakarta, 24 Juli 2018

Mengetahui,
Ketua Program Studi
Pendidikan Teknik Informatika



Handaru Jati, Ph.D.
NIP. 19740511 199903 1 002

Disetujui,
Dosen Pembimbing



Nurkhamid S.Si., M.Kom., Ph.D.
NIP. 19680707 199702 1 001

HALAMAN PENGESAHAN

Tugas Akhir Skripsi

PENGEMBANGAN APLIKASI ANDROID *MOBILE DICTIONARY AND EMULATOR* PERINTAH DASAR *COMMAND LINE INTERFACE LINUX* DAN WINDOWS

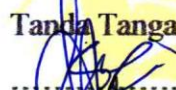


Disusun oleh :

Dian Kartika Sari
NIM. 14520244012

Telah dipertahankan di depan Tim Penguji Tugas Akhir Skripsi Program Studi Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Yogyakarta

Pada tanggal 7 Agustus..... 2018

TIM PENGUJI

Nama/Jabatan	Tanda Tangan	Tanggal
Nurkhamid S.Si., M.Kom., Ph.D. Ketua penguji/Pembimbing		16/08 - 2018
Dr. Fatchul Arifin, M.T. Sekretaris		16/08 - 2018
Prof. Drs. Herman Dwi Sarjono, M.Sc., MT., Ph.D. Penguji		16/8 2018

Yogyakarta, 20 Agustus..... 2018

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,


Dr. Widarto, M.Pd.


NIP 19631230 198812 1 001

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Dian Kartika Sari
NIM : 14520244012
Program Studi : Pendidikan Teknik Informatika
Judul TAS : Pengembangan Aplikasi Android *Mobile Dictionary and Emulator* Perintah Dasar *Command Line Interface* Linux dan Windows

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 kutipan dengan mengikuti tata penulisan karya ilmiah yang telah lazim.

Yogyakarta,

Yang menyatakan,



Dian Kartika Sari

NIM . 14520244012

HALAMAN MOTTO

“ bersabar dan tetaplah bersabar menjalani kehidupan, yakinlah keberuntungan dan keberhasilan senantiasa mengiringi ”

(my self)

“Innallaha Ma'ash Shobirin”

(Al-Quran)

“Segala sesuatu bergantung pada niatnya ”

(Hadits Bukhari-Muslim)

“Sesungguhnya Allah Tidak Akan Mengubah Nasib Suatu Kaum Kecuali Kaum Itu Sendiri Yang Mengubah Apa-apa Yang Pada Diri Mereka”

(Al-Quran)

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur kepada Allah SWT, karya ini saya persembahkan kepada :

1. Kedua orang tua saya yang telah memberikan dorongan dan kasih sayangnya serta selalu mendoakan sepanjang waktu untuk keberhasilan saya.
2. Untuk keluarga besar yang selalu memberi dukungannya.
3. Teman-teman seperjuangan Tim INFINITE Universitas Negeri Yogyakarta yang telah memberikan pelajaran dan pengalaman yang sangat berharga serta kekeluargaan yang terjalin.
4. Untuk rekan-rekan jurusan Pendidikan Teknik Elektronika dan Informatika angkatan 2014 khususnya prodi Pendidikan Teknik Informatika kelas F yang senantiasa berbagi ilmu.
5. Perpustakaan Universitas Negeri Yogyakarta untuk kemajuan ilmu pengetahuan dan teknologi.

Terima kasih atas segala doa, bimbingan, kasih sayang, dan pengorbanan yang tidak pernah berhenti.

PENGEMBANGAN APLIKASI ANDROID *MOBILE DICTIONARY AND EMULATOR* PERINTAH DASAR *COMMAND LINE INTERFACE* LINUX DAN WINDOWS

Oleh :

Dian Kartika Sari
NIM. 14520244012

ABSTRAK

Berdasarkan observasi yang dilakukan oleh peneliti ketika melakukan PLT di SMK N 1 Bantul, beberapa siswa kompetensi keahlian TKJ mengalami kendala dalam mengoperasikan sistem operasi berbasis teks. Siswa tidak mengetahui fungsi dari perintah *command line interface* (CLI) yang digunakan. Penelitian ini bertujuan (1) mengembangkan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows, (2) mengetahui kualitas aplikasi yang dikembangkan berdasarkan standar kualitas perangkat lunak ISO/IEC 25010 pada aspek *functional suitability, compatibility, performance efficiency, dan usability*.

Metode yang digunakan adalah *Research and Development* dengan prosedur proses pengembangan aplikasi menggunakan *Rational Unified Process* (RUP) melalui 4 tahapan yaitu *inception, elaboration, construction, dan transition*. Pengumpulan kebutuhan *user* menggunakan teknik observasi dan wawancara. Proses *analysis & design* menggunakan bahasa pemodelan UML (*use case, expanded use case, activity diagram, dan sequence diagram*). Proses konstruksi yang dilakukan menggunakan Android Studio. Pengujian dilakukan menggunakan standar ISO 25010 pada karakteristik *functional suitability, usability, compatibility, dan performance efficiency*.

Hasil penelitian ini adalah 1) aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows yang dikembangkan menggunakan IDE Android Studio, dan 2) hasil pengujian menunjukkan aplikasi telah memenuhi standar ISO 25010 pada aspek (a) *functional suitability* seluruh aplikasi berjalan 100%, (b) *compatibility* aplikasi kompatibel 100% dari sisi *coexistence*, berbagai sistem operasi dan tipe perangkat, (c) *usability* aplikasi memperoleh skor 88,1% dan masuk pada kategori sangat layak (d) *performance efficiency* aplikasi berjalan dengan baik tanpa terjadi *memory leak* atau *force close*. Rata-rata penggunaan CPU adalah 49,6%, penggunaan memori 22,8 MB, dan *time behaviour* 0,039 *seconds/thread*.

Kata kunci : kamus, emulator, *command line*, Android, aplikasi *mobile*, ISO 25010

**DEVELOPMENT OF ANDROID MOBILE APPLICATION
DICTIONARY AND EMULATOR COMMAND LINE INTERFACE LINUX
AND WINDOWS**

By :

Dian Kartika Sari
NIM. 14520244012

ABSTRACT

Based on observations made by researchers when conducting PLT in SMK N 1 Bantul, some students of TKJ's competency expertise experienced obstacles in operating a text-based operating system. Students do not know the function of the CLI command used. This study aims to (1) develop Android mobile applications dictionary basic commands CLI Linux and, (2) know the quality of applications developed based on ISO / IEC 25010 software quality standards on the functional suitability, compatibility, performance efficiency, and usability.

The method used is Research and Development with the procedure of application development process using Rational Unified Process (RUP) through 4 stages, namely inception, elaboration, construction, and transition. Collecting user needs using observation and interview techniques. Analysis & design process using UML modeling languages (use cases, expanded use cases, activity diagrams, and sequence diagrams). The construction process is carried out using Android Studio. Testing is done using ISO 25010 standards on functional suitability, usability, compatibility, and performance efficiency characteristics.

The results of this study are 1) Android mobile application dictionary basic commands CLI Linux and Windows for students TKJ SMK N 1 Bantul developed using Android Studio IDE, and 2) test results show the application meets the ISO 25010 standards on (a) functional suitability all applications running 100%, (b) compatibility of 100% compatible applications in terms of coexistence, various operating systems and device types, (c) usability of application scored 88.1% and entered in very decent category, used for research, (d) the performance efficiency of the application goes well without going leak memory or force close. The average CPU usage is 49.6%, memory usage is 22.8 MB, and time behavior is 0.039 seconds / thread.

Keywords: dictionary, emulator, command line, Android, mobile application, ISO 25010

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas rahmat dan karunia-NYA sehingga penulis dapat menyelesaikan Tugas Akhir Skripsi dalam rangka memenuhi sebagian persyaratan untuk mendapatkan gelar Sarjana Pendidikan Teknik Informatika Universitas Negeri Yogyakarta dengan judul “Pengembangan Aplikasi Android *Mobile Dictionary and Emulator* Perintah Dasar *Command Line Interface* Linux dan Windows”. Tugas Akhir Skripsi ini dapat diselesaikan tidak lepas dari bantuan dan arahan dari berbagai pihak. Berkenaan dengan hal tersebut, penulis menyampaikan ucapan terima kasih kepada yang terhormat :

1. Bapak Nurkhamid, S.Si., M.Kom., Ph.D selaku Dosen Pembimbing Tugas Akhir Skripsi yang telah banyak memberikan semangat, dorongan, dan bimbingan selama penyusunan Tugas Akhir Skripsi ini.
2. Bapak Nurkhamid, S.Si., M.Kom., Ph.D dan Ibu Dariyati, S.Kom selaku Validator instrumen penelitian TAS yang memberikan saran/masukan perbaikan sehingga penelitian TAS dapat terlaksana sesuai dengan tujuan.
3. Tim penguji, selaku Ketua Penguji, Sekretaris, dan Penguji yang memberikan koreksi perbaikan terhadap Tugas Akhir Skripsi ini.
4. Bapak Fatchul Arifin, M.T selaku ketua jurusan Pendidikan Teknik Elektronika dan Bapak Handaru Jati, Ph.D beserta dosen dan staf yang telah memberikan bantuan dan fasilitas selama penyusunan pra proposal sampai dengan selesainya Tugas Akhir Skripsi ini.

5. Bapak Dr. Widarto, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta yang memberikan persetujuan pelaksanaan Tugas Akhir Skripsi.
6. Ibu Ir. Retno Yuniar Dwi Aryani selaku kepala sekolah SMK N 1 Bantul, yang telah memberi izin dalam pelaksanaan penelitian Tugas Akhir Skripsi ini.
7. Para guru dan staf SMK N 1 Bantul yang telah memberi bantuan memperlancar pengambilan data selama proses penelitian Tugas Akhir Skripsi ini.
8. Seluruh pihak, secara langsung maupun tidak langsung, yang tidak dapat disebutkan di sini atas bantuan dan perhatiannya selama penyusunan Tugas Akhir Skripsi ini.

Demikian Tugas Akhir Skripsi ini penulis susun. Semoga segala bantuan yang telah diberikan semua pihak di atas menjadi amalan yang bermanfaat dan mendapatkan balasan dari Allah SWT dan Tugas Akhir Skripsi ini menjadi informasi bermanfaat bagi pembaca atau pihak lain yang membutuhkan.

Yogyakarta, 28 Juli 2018

Penulis,



Dian Kartika Sari
NIM. 14520244012

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERNYATAAN	iv
HALAMAN MOTTO	v
HALAMAN PERSEMBAHAN	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN.....	xvii
BAB I PENDAHULUAN.....	1
A. Latar Belakang	1
B. Identifikasi Masalah.....	6
C. Batasan Masalah.....	7
D. Rumusan Masalah.....	7
E. Tujuan Penelitian	8
F. Spesifikasi Produk yang Dikembangkan	8
G. Manfaat Peneliti	9
BAB II KAJIAN PUSTAKA	11
A. Kajian Teori	11
1. Pengembangan Aplikasi Android <i>Mobile Dictionary and Emulator</i> Perintah Dasar CLI Linux dan Windows	11
a. <i>Dictionary and Emulator</i> Perintah Dasar <i>Command Line Interface</i> Linux dan Windows.....	11
1) Perintah Dasar <i>Command Line Interface</i> Linux dan Windows	11
2) Kamus	15
3) Emulator.....	19
b. Pengembangan Aplikasi Android <i>Mobile</i>	20

c. Model Pengembangan.....	25
d. Perangkat Pengembangan.....	30
1) UML (<i>Unified Modelling Language</i>).....	30
2) Android SDK	35
3) Android Studio.....	36
2. Pengujian Perangkat Lunak.....	37
a. <i>Functional Suitability</i>	38
b. <i>Compatibility</i>	39
c. <i>Usability</i>	39
d. <i>Performance Efficiency</i>	40
B. Penelitian yang Relevan.....	41
C. Kerangka Pikir	42
D. Pertanyaan Penelitian	43
BAB III METODE PENELITIAN.....	44
A. Model Penelitian	44
B. Prosedur Pengembangan	45
1. <i>Inception</i>	45
2. <i>Elaboration</i>	46
3. <i>Construction</i>	47
4. <i>Transition</i>	47
C. Subjek, Tempat, dan waktu Penelitian.....	48
D. Variabel Penelitian	48
1. Variabel Penelitian	48
2. Definisi Operasional Variabel	49
E. Metode dan Alat Pengumpul Data	49
1. Wawancara	50
2. Observasi	50
3. Kuesioner.....	50
F. Instrumen Penelitian.....	51
1. Instrumen Uji Materi	51
2. Instrumen <i>Functional Suitability</i>	51
3. Instrumen <i>Usability</i>	53
4. Instrumen <i>Compatibility</i>	54
5. Instrumen <i>Performance Efficiency</i>	55
G. Teknik Analisis Data.....	56
1. Analisis Pengujian Aspek <i>Functional Suitability</i> dan <i>Compatibility</i>	56
2. Analisis Pengujian Aspek <i>Usability</i>	56
3. Analisis Pengujian Aspek <i>Performance Efficiency</i>	60
BAB IV HASIL DAN PEMBAHASAN	61
A. Hasil Penelitian	61
1. Pengembangan Aplikasi	61
1. Tahap <i>Inception</i>	61
2. Tahap <i>Elaboration</i>	69

3. Tahap <i>Construction</i>	80
4. Tahap <i>Transition</i>	88
2. Pengujian Aplikasi.....	93
a. Hasil Uji Materi	93
b. Hasil Uji <i>Functional Suitability</i>	94
c. Hasil Uji <i>Usability</i>	96
d. Hasil Uji <i>Compatibility</i>	97
e. Hasil Uji <i>Performance Efficiency</i>	101
B. Pembahasan.....	104
BAB V KESIMPULAN DAN SARAN.....	108
A. Kesimpulan	108
B. Keterbatasan Produk	109
C. Pengembangan Lebih Lanjut	110
D. Saran.....	110
DAFTAR PUSTAKA	111
LAMPIRAN.....	113

DAFTAR GAMBAR

Gambar 1. Contoh Tampilan <i>Command Line Interface</i>	12
Gambar 2. Lapisan-Lapisan Pengembangan Perangkat Lunak.....	21
Gambar 3. Data Jumlah Distribusi Platform Perangkat Android.....	23
Gambar 4. Grafik Perbandingan Sistem Operasi Mobile di Indonesia.....	23
Gambar 5. Proses Iteratif RUP.....	26
Gambar 6. Alur Hidup RUP.....	27
Gambar 7. Diagram UML (Rosa, 2011 : 121).....	31
Gambar 8. Contoh <i>Use Case Diagram</i>	32
Gambar 9. Contoh <i>Class Diagram</i>	33
Gambar 10. Contoh <i>Sequence Diagram</i>	34
Gambar 11. Contoh <i>Activity Diagram</i>	35
Gambar 12. <i>Android SDK Manager for Windows</i>	36
Gambar 13. Tampilan Android Studio.....	37
Gambar 14. Diagram <i>Use Case</i>	70
Gambar 15. <i>Activity Diagram</i> Melihat Penjelasan CLI.....	75
Gambar 16. <i>Sequence Diagram</i> Melihat Penjelasan CLI.....	76
Gambar 17. <i>Class Diagram</i> Aplikasi.....	76
Gambar 18. Potongan HTA.....	77
Gambar 19. Potongan <i>Script XML</i> Pada <i>activity_main.xml</i>	83
Gambar 20. Potongan File Java Pada <i>Class MainActivity.java</i>	85
Gambar 21. <i>Icon Run 'app'</i>	86
Gambar 22. Pilihan <i>Device</i> Android yang Terhubung.....	86
Gambar 23. <i>Install Driver adb</i>	87
Gambar 24. Layar Logcat pada Android Monitor.....	87

Gambar 25. <i>Build Variant release</i>	90
Gambar 26. <i>Build APK</i>	90
Gambar 27. Proses Pengisian <i>Generate Signed APK</i>	91
Gambar 28. Pengaturan Penyimpanan	91
Gambar 29. Hasil <i>Build Release</i>	91
Gambar 30. Hasil <i>Rating</i> Aplikasi	92
Gambar 31. Detail Produk Aplikasi	92
Gambar 32. Hasil Pengujian <i>Compatibility</i> dengan Firebase Test Lab	100
Gambar 33. Hasil Pengujian <i>Compatibility</i> dengan Google Play Store.....	100
Gambar 34. Hasil Pengujian <i>Performance Efficiency</i>	101

DAFTAR TABEL

Tabel 1. Opsi/Parameter Perintah cp.....	13
Tabel 2. Opsi/Parameter Perintah copy.....	14
Tabel 3. <i>Workflows</i> Model Pengembangan RUP.....	28
Tabel 4. Tabel Instrumen Pengujian Aspek <i>Functional Suitability</i>	51
Tabel 5. <i>USE Questionnaire</i>	53
Tabel 6. Tabel Instrument Sub Karakteristik <i>Coexistence</i>	54
Tabel 7. Tabel Kriteria Interpretasi Skor	56
Tabel 8. Kelas Interval	59
Tabel 9. Tabel Interpretasi Kelas Interval.....	59
Tabel 10. Estimasi Penyelesaian Tiap Tahapan (<i>Stage</i>)	68
Tabel 11. Definisi Aktor	71
Tabel 12. Definisi <i>Use Case</i>	71
Tabel 13. <i>Expanded Use Case</i>	73
Tabel 14. <i>Expanded Use Case</i> Melihat Penjelasan CLI	74
Tabel 15. Daftar <i>Activity Diagram</i>	74
Tabel 16. Daftar <i>Sequence Diagram</i>	75
Tabel 17. <i>Storyboard</i>	78
Tabel 18. Konfirmasi Desain <i>Mockup</i>	78
Tabel 19. <i>Layout</i> Aplikasi.....	81
Tabel 20. File Java	83
Tabel 21. Hasil Perbaikan	89
Tabel 22. Daftar Ahli Pengujian <i>Functional Suitability</i>	94
Tabel 23. Hasil Uji <i>Functional Suitability</i>	94

Tabel 24. Hasil uji <i>coexistence</i>	97
Tabel 25. Hasil dan Perangkat Pengujian <i>Compatibility</i>	99
Tabel 26. Ringkasan Uji <i>Performance Efficiency</i>	102
Tabel 27. Hasil Pengujian Aplikasi Berdasarkan ISO/IEC 25010.....	105

DAFTAR LAMPIRAN

Lampiran 1. Surat Keputusan Dosen Pembimbing	114
Lampiran 2. Surat Ijin Penelitian dari Fakultas Teknik UNY	115
Lampiran 3. Surat Ijin Penelitian dari BAKESPOL DIY	116
Lampiran 4. Surat Ijin Penelitian dari DISPORA Yogyakarta	117
Lampiran 5. Surat Selesai Penelitian di SMK N 1 Bantul	118
Lampiran 6. Surat Permohonan Validasi Instrumen TAS	119
Lampiran 7. Surat Pernyataan Instrumen Penelitian TAS	120
Lampiran 8. Hasil Uji <i>Functional Suitability</i>	122
Lampiran 9. Hasil <i>Usability</i>	129
Lampiran 10. Karakteristik dan Sub Karakteristik Standar ISO 25010.....	132
Lampiran 11. Instrumen Pengujian Aspek <i>Functional Suitability</i>	134
Lampiran 12. Instrumen Aspek <i>Usability</i>	142
Lampiran 13. <i>Expanded Use Case</i>	146
Lampiran 14. <i>Activity Diagram</i>	149
Lampiran 15. <i>Sequence Diagram</i>	153
Lampiran 16. <i>Hierarchy Analysis Task</i>	156
Lampiran 17. Hasil <i>Storyboard</i>	159
Lampiran 18. <i>Mockup Design User Interface</i>	168
Lampiran 19. Hasil Perbaikan Aplikasi	169
Lampiran 20. Hasil Uji Materi	172
Lampiran 21. Hasil Uji <i>Usability</i>	173
Lampiran 22. <i>Measuring Usability with the USE Questionnaire</i>	174
Lampiran 23. Dokumentasi Penelitian.....	176

BAB I

PENDAHULUAN

A. Latar Belakang

Sekolah Menengah Kejuruan (SMK) merupakan salah satu jenjang pendidikan menengah atas yang dipersiapkan pemerintah sebagai lanjutan dari jenjang SMP/MTs. SMK lebih dipersiapkan untuk menghasilkan lulusan yang mampu memenuhi tuntutan dunia kerja. Hal ini sesuai dengan UU No.20 tahun 2003 tentang aplikasi pendidikan nasional, yang menyatakan bahwa salah satu tujuan pendidikan menengah kejuruan adalah untuk menyiapkan peserta didik agar menjadi manusia produktif, mampu bekerja mandiri, mengisi lowongan pekerjaan yang ada sebagai tenaga kerja tingkat menengah sesuai dengan kompetensi dalam program keahlian yang dipilihnya.

Lulusan siswa SMK dalam menghadapi tuntutan dunia kerja ditentukan oleh kemampuan siswa dalam menguasai kompetensi kejuruan sesuai dengan kompetensi keahlian masing-masing. Semakin tinggi penguasaan siswa terhadap kompetensi kejuruan, semakin tinggi pula keterampilan dan kemampuan yang dimiliki siswa dalam menghadapi dunia kerja.

Salah satu kompetensi keahlian yang diselenggarakan di SMK N 1 Bantul adalah Teknik Komputer dan Jaringan (TKJ). Kompetensi keahlian Teknik Komputer dan Jaringan SMK N 1 Bantul mengharuskan siswanya untuk menguasai beberapa kompetensi kejuruan. Kompetensi kejuruan tersebut antara lain melakukan instalasi sistem operasi (OS) berbasis teks, melakukan

administrasi server, membangun aplikasi keamanan jaringan, dan merancang bangun sebuah jaringan baik jaringan kabel maupun jaringan *wireless*. Beberapa kompetensi kejuruan tersebut memiliki kesamaan yaitu menggunakan OS komputer linux berbasis teks. Tetapi tidak jarang juga menggunakan *command line interface* (CLI) pada OS Windows.

Pengoperasian OS berbasis teks pada Linux dan Windows berbeda dengan pengoperasian OS berbasis *Graphical User Interface* (GUI). Pengoperasian OS berbasis teks membutuhkan keahlian tersendiri bagi penggunanya. Dibutuhkan pengetahuan tentang CLI berikut fungsinya-fungsinya dan cara penulisannya untuk melakukan konfigurasi.

Berdasarkan observasi yang dilakukan oleh peneliti ketika melakukan kegiatan PLT tahun 2017 di SMK N 1 Bantul, beberapa siswa SMK N 1 Bantul kompetensi keahlian Teknik Komputer dan Jaringan mengalami kendala dalam mengoperasikan OS berbasis teks. Siswa sering menanyakan fungsi dan sintaks perintah CLI yang dituliskan kepada peneliti. Hasil wawancara yang dilakukan peneliti kepada guru pengajar tentang kendala pembelajaran juga menyatakan hal yang sama. Beberapa siswa masih mengalami kendala dalam mengoperasikan OS berbasis teks. Sebagian siswa juga masih bingung membedakan mana perintah CLI yang digunakan pada OS Linux dan Windows. Siswa sekedar menyalin perintah yang ada di *lab sheet* yang diberikan guru pengampu sebagai pedoman praktik tanpa mengetahui apa yang ia lakukan dan apa fungsi dari perintah yang ia ketikkan. Selama ini belum ada pemberian materi khusus dari guru pengampu

mengenai perintah-perintah CLI baik untuk OS Linux maupun Windows. Keadaan ini menyebabkan penyampaian materi kompetensi kejuruan oleh guru tidak dapat diterima siswa secara maksimal.

Sebenarnya referensi untuk mengetahui arti perintah CLI bisa didapat dari buku, *online documentation*, dan sistem operasi itu sendiri. Berdasarkan wawancara dan pengamatan peneliti, sebagian besar siswa tidak memiliki sumber referensi buku karena harganya yang mahal dan dari pihak sekolah tidak menyediakan fasilitas buku yang mendukung materi CLI yang bisa dipinjam oleh siswa. Sehingga siswa harus mencari sumber referensi lain dari internet berupa *online documentation*. Untuk menggunakan sumber referensi tersebut, siswa harus membuka halaman tertentu di browser seperti ss64.com/bash/ untuk kumpulan perintah CLI Linux dan docs.microsoft.com untuk kumpulan perintah CLI Windows. Pada *online documentation* terdapat penjelasan tentang fungsi perintah, sintaks, parameter dan contoh penggunaan. Namun untuk mengimplementasikan perintah yang dipelajari, siswa harus membuka laptop atau komputer mereka yang terinstal OS yang ingin dicoba perintahnya. Padahal tidak semua siswa memiliki laptop atau komputer dengan *double OS* (Linux dan Windows). Sehingga dibutuhkan sumber referensi yang bisa digunakan untuk mengimplementasikan perintah yang dipelajari secara langsung dalam sistem operasi. Selain dari buku maupun *online documentation*, sistem operasi telah menyediakan menu untuk mengetahui penjelasan dari kumpulan perintah CLI yaitu melalui menu *apropos* pada Linux dan menu *help* pada Windows. Sumber referensi tersebut juga tidak

efisien karena pengguna harus menghidupkan komputer terlebih dahulu untuk memanfaatkannya. Mobilitas sumber referensi menjadi aspek yang penting. Mobilitas yang tinggi membuat sumber referensi dapat dimanfaatkan pengguna dimanapun dan kapanpun.

Di lain sisi, perkembangan teknologi sangat pesat terutama di bidang teknologi digital selain teknologi internet itu sendiri sudah berkembang teknologi perangkat *smartphone*. Berdasarkan hasil survei yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) menyatakan bahwa pada tahun 2017, jumlah pengguna internet di Indonesia mencapai 143,26 juta jiwa. Angka tersebut meningkat dibandingkan pada tahun sebelumnya, yakni tahun 2016 yang tercatat mencapai 132,7 juta jiwa. Jumlah pengguna internet pada tahun 2017 tersebut mencakup 54,68 persen dari total populasi Indonesia yang mencapai 262 juta orang. APJII juga memberlakukan kategori wilayah dan usia untuk memetakan tingkat penetrasi pengguna internet di Indonesia dalam survei tersebut. Berdasarkan kategori wilayah pulau, tingkat penetrasi pengguna internet di pulau Kalimantan menduduki peringkat teratas dengan angka 72,19 persen. Kemudian, disusul oleh Jawa dan Bali-Nusa, dengan angka tingkat penetrasi pengguna internet masing-masing 57,70 persen dan 54,23 persen. Sementara menurut kategori usia, pengguna internet ternyata didominasi oleh usia 13-18 tahun. Tingkat penetrasinya mencapai 75,50 persen. APJII juga menyatakan bahwa sebanyak 66,78 persen pengguna internet, mengakses internet melalui *smartphone*. Data tersebut menunjukkan sebuah asumsi bahwa sebagian besar

peserta didik di tingkat Sekolah Menengah Atas (SMA/SMK) merupakan pengguna *smartphone*. Asumsi yang dikembangkan dapat menjadi sebuah peluang untuk memanfaatkan teknologi *mobile* dalam dunia pendidikan dengan harapan media pendidikan yang digunakan dapat menjadi media bantu belajar siswa. Berdasarkan observasi yang telah dilakukan di SMK N 1 Bantul, didapatkan informasi bahwasannya siswa baik kelas satu hingga kelas tiga hampir semuanya telah menggunakan teknologi *smartphone*. Tentu hal ini sangat disayangkan apabila perkembangan teknologi *smartphone* tidak dimanfaatkan untuk menunjang pendidikan. Berdasarkan hal inilah yang menjadi dasar pembuatan aplikasi Android *mobile dictionary and emulator* perintah dasar *command line* Linux dan Windows.

Aplikasi *mobile* merupakan suatu program yang berjalan pada perangkat bergerak seperti *handphone*. Terutama *smartphone* yang mampu menampung banyak sebuah program yang tentunya berfungsi untuk memudahkan penggunaannya dalam menjalani aktivitasnya. Berdasarkan riset yang dilakukan oleh Wardhana mengenai analisis sentimen pada *review* pengguna aplikasi *mobile* untuk evaluasi faktor *usability* menyatakan bahwa rata-rata nilai akurasi klasifikasi faktor *usability* pada perangkat *mobile* adalah sebesar 82% dengan tingkat akurasi sentimen sebesar 79%. Hal ini digunakan oleh pengembang aplikasi untuk mengembangkan aplikasi dengan fitur unggul berdasarkan sentimen dari *user*. Oleh karena itu untuk memperoleh sentimen positif dari *user*, pengembang harus meningkatkan kualitas aplikasi melalui serangkaian uji coba.

Pengembangan perangkat lunak yang belum dilakukan pengujian belum bisa diketahui apakah perangkat lunak tersebut layak atau tidak digunakan oleh pengguna. Untuk melakukan uji coba kelayakan pada perangkat lunak, bisa menggunakan standar ISO yang telah ditentukan misalnya, ISO 25010 yang menjadi standar pengujian internasional dalam penentuan kualitas perangkat lunak. Maka dari itu untuk menguji dan mengetahui kualitas aplikasi Android *mobile dictionary and emulator* perintah dasar *command line interface* Linux dan Windows menggunakan standar ISO 25010.

B. Identifikasi Masalah

Berdasarkan latar belakang masalah di atas, terdapat beberapa permasalahan yang dapat diidentifikasi yaitu:

1. Siswa SMK harus mampu memenuhi tuntutan dunia kerja.
2. Siswa SMK harus menguasai seluruh kompetensi kejuruan sesuai kompetensi keahlian masing-masing.
3. Siswa SMK N 1 Bantul kompetensi keahlian Teknik Komputer dan Jaringan mengalami kendala dalam mengoperasikan sistem operasi (OS) Linux dan Windows berbasis teks.
4. Sebagian siswa masih bingung membedakan mana perintah CLI yang digunakan pada OS Linux dan Windows.
5. Siswa sekedar menyalin perintah yang ada di *lab sheet* yang diberikan guru pengampu sebagai pedoman praktik tanpa mengetahui apa yang ia lakukan dan apa fungsi dari perintah yang ia ketikkan.

6. Guru tidak memberikan materi tentang perintah-perintah dasar CLI secara khusus.
7. Tidak semua siswa memiliki buku tentang perintah CLI.
8. Belum ditemukan sumber referensi yang layak digunakan pengguna untuk mengetahui arti, sintaks, parameter, dan contoh penerapan perintah CLI Linux maupun Windows secara langsung dalam sistem operasi yang dapat digunakan dimanapun dan kapanpun.

C. Batasan Masalah

Berdasarkan identifikasi masalah yang ada, batasan masalah yang dibuat peneliti dalam penelitian ini adalah belum ditemukan sumber referensi yang layak digunakan pengguna untuk mengetahui arti, sintaks, parameter, dan contoh penerapan perintah CLI Linux maupun Windows secara langsung dalam sistem operasi yang dapat digunakan dimanapun dan kapanpun.

D. Rumusan Masalah

Berdasarkan batasan masalah di atas, dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana mengembangkan sumber referensi untuk mengetahui arti, sintaks, parameter, dan contoh penerapan perintah secara langsung dalam sistem operasi yang dapat digunakan dimanapun dan kapanpun baik untuk perintah CLI Linux dan Windows?
2. Bagaimana mengetahui kelayakan sumber referensi yang digunakan pengguna untuk mengetahui arti, sintaks, parameter, dan contoh penerapan perintah CLI

Linux maupun Windows secara langsung dalam sistem operasi yang dapat digunakan dimanapun dan kapanpun?

E. Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan dari penelitian ini adalah :

1. Mengembangkan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sebagai sumber belajar tambahan untuk siswa TKJ.
2. Mengetahui tingkat kualitas dan kelayakan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows agar tidak terjadi kesalahan dari segi fitur dan fungsionalitasnya ketika digunakan oleh pengguna dengan melakukan pengujian menggunakan standar ISO 25010.

F. Spesifikasi Produk yang Dikembangkan

Spesifikasi dari produk yang dikembangkan adalah aplikasi Android *mobile* kamus perintah dasar CLI Linux dan Windows yang mampu memberikan referensi daftar perintah-perintah dasar setiap sistem operasi tersebut dan memiliki fitur *filtering*, dimana fitur ini bekerja ketika pengguna memasukkan satu huruf atau lebih ketika melakukan pencarian. Aplikasi ini juga dilengkapi dengan fitur implementasi perintah yang berfungsi untuk menerapkan perintah ke sistem operasi secara langsung. Selain itu, aplikasi ini dilengkapi dengan fitur *share* yang berfungsi untuk berbagi ke setiap *user* tentang pengertian dan sintaks perintah. Aplikasi akan dikembangkan pada *platform mobile* Android.

G. Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1) Manfaat Teoritis

Manfaat teoritis yang didapat dalam penelitian ini antara lain:

- a. Hasil penelitian ini dapat memberikan kontribusi bagi dunia pendidikan dan teknologi informasi.
- b. Hasil penelitian ini dapat digunakan sebagai referensi bagi orang lain yang hendak melakukan penelitian yang relevan.

2) Manfaat Praktis

Manfaat praktis yang didapat dalam penelitian ini antara lain:

a. Bagi Peneliti

- 1) Menambah pengetahuan, wawasan, dan keterampilan dalam pengembangan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows.
- 2) Menambah pengalaman peneliti dalam melakukan penelitian pengembangan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows.

b. Bagi Siswa

- 1) Mempermudah dalam pencarian informasi tentang perintah dasar CLI Linux dan Windows.
- 2) Mempermudah siswa untuk memahami arti dan fungsi perintah dasar CLI Linux dan Windows.

3) Mempermudah siswa mengimplementasikan perintah CLI Linux dan Windows di emulator terminal dan *command prompt*.

c. Bagi Masyarakat Umum

1) Memberikan pengetahuan tentang *command line interface* dan perintah-perintahnya yang ada pada OS Linux dan Windows.

2) Mempermudah masyarakat umum untuk mengetahui fungsi, sintaks, parameter, dan contoh penggunaan perintah CLI Linux dan Windows.

3) Mempermudah masyarakat umum untuk mengimplementasikan perintah CLI Linux dan Windows secara langsung di emulator terminal maupun *command prompt*.

BAB II

KAJIAN PUSTAKA

A. Kajian Teori

1. Pengembangan Aplikasi Android *Mobile Dictionary and emulator* Perintah Dasar CLI Linux dan Windows

a. *Dictionary and Emulator* Perintah Dasar *Command Line Interface* Linux dan Windows

1) Perintah Dasar *Command Line Interface* Linux dan Windows

Hutomo (2014:11) menerangkan *command line interface* (CLI) adalah tipe antarmuka dimana pengguna berinteraksi dengan sistem operasi (OS) melalui *text-terminal*. Mekanisme interaksi pengguna dengan OS atau perangkat lunak komputer menggunakan CLI dengan mengetikkan perintah untuk menjalankan tugas tertentu. CLI merupakan kontras dari penggunaan antarmuka penunjuk dengan mengklik pilihan perintah untuk menjalankan tugas tertentu pada *Graphic User Interface* (GUI), atau penggunaan menu untuk memilih pilihan perintah pada *Text User Interface* (TUI).

Konsep CLI dimulai sewaktu *teletypewriter machine* (TTY) dihubungkan ke komputer pada tahun 1950-an dan terus berkembang bersama dengan aplikasi GUI seperti Microsoft Windows, Mac OS, dan X Window System (Hutomo, 2014:11). Meskipun konsepnya sama, tiap-tiap OS memiliki nama atau istilah yang berbeda untuk *command line*-nya.

Pengguna Linux mengenal *command line* pada Linux sebagai terminal. Unix memberi nama *command line*-nya sebagai bash, ash, ksh, dan lain sebagainya. Microsoft Disk Operating System (MS-DOS) memberi nama *command.com* atau *command prompt* pada Windows Vista, Microsoft menamakannya *PowerShell*, sedangkan Apple menamainya *command shell*.

```

root ~ # ping google.com
PING google.com (74.125.95.103) 56(84) bytes of data:
64 bytes from iw-in-f103.1e100.net (74.125.95.103): icmp_seq=1 ttl=47 time=15.3
ms
^C
--- google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 15.453/15.453/15.453/0.000 ms
root ~ # ls
Desktop README
root ~ # cd /
root / # ls
bin dev home lost+found mnt proc sbin srv tmp var
boot etc lib media opt root sources sys usr
root / # pacman -Ss pidgin
extra/libpurple 2.6.6-1
  7M library extracted from Pidgin
extra/pidgin 2.6.6-1
  Multi-protocol instant messaging client
extra/pidgin-encryption 3.0-3
  A Pidgin plugin providing transparent RSA encryption using NSS
extra/purple-plugin-pack 2.6.3-1
  Plugin pack for Pidgin
extra/telepathy-haze 0.3.4-1 (telepathy)
  A telepathy-backend to use libpurple (Pidgin) protocols.
community/quifications 2.16-1
  A set of GUI popup notifications for pidgin
community/pidgin-fonobutton 0.1.6-1
  Adds a video-chat button to the conversation window
community/pidgin-libnotify 0.14-3
  pidgin plugin that enables popups when someone logs in or messages you.
community/pidgin-musictracker 0.4.21-2
  A plugin for Pidgin which displays the music track currently playing.
community/pidgin-otr 3.2.0-1
  Off-the-Record Messaging Plugin for Pidgin
root / #

```

Gambar 1. Contoh Tampilan *Command Line Interface*

Perintah CLI pada Linux maupun Windows memiliki daftar perintah masing-masing. Setiap perintah memiliki fungsi, sintaks, opsi/parameter dan cara penggunaan yang berbeda-beda untuk setiap OS. Contoh perintah CLI Linux dari *online documentation* dari ss64.com/bash/ dan CLI Windows dari docs.microsoft.com:

a) Perintah CLI cp pada OS Linux

➤ Fungsi

Perintah cp berfungsi untuk meng-*copy* file dari satu lokasi ke lokasi lainnya. Jika pada lokasi tujuan file tersebut sudah ada, maka file akan ditulis

kembali, jika pada lokasi tujuan folder yang di-copy sudah ada, maka file di-copy ke dalam folder tersebut.

➤ Sintaks

cp [opsi]...SOURCE DEST

cp [opsi]...SOURCE...DIRECTORY

cp [opsi]...-target-directory=DIRECTORY SOURCE

➤ Opsi/Parameter

Tabel 1. Opsi/Parameter Perintah cp

Opsi	Deskripsi
-a	--archive, sama seperti -dpR.
-b	--backup, membuat file <i>backup</i> (cadangan) pada setiap file tujuan.
-f	--force, jika file di lokasi tujuan yang sudah ada tidak dapat dibuka, maka hapus data tersebut dan ulangi lagi proses mengcopy.
-p	--preserve, melindungi atribut file jika memungkinkan.
-preserve[=ATTR_LIST]	Melindungi atribut khusus file (default: mode, ownership, timestamps) .
--no preserve = ATTR_LIST	Tidak melindungi atribut khusus pada file.

➤ Contoh

Mengcopy dua file

cp file1 file2

Perintah cp di atas mengcopy isi pada file1.php ke file2.php.

b) Perintah CLI copy pada OS Windows

➤ Fungsi

Perintah copy digunakan untuk menyalin satu atau lebih file dari satu lokasi ke lokasi yang lain.

➤ Sintaks

`copy[/d][/v][/y/y][/z][/a/b]<source>[/a/b]<+source>[/a/b][+...]] [<Destinasi>] [/a | /b]`

➤ Opsi/Parameter

Tabel 2. Opsi/Parameter Perintah copy

Opsi	Deskripsi
/d	Memungkinkan file yang dienkripsi disalin untuk disimpan sebagai file deskripsi di tempat tujuan.
/v	Memverifikasi bahwa file baru ditulis dengan benar.
/y	Menyembunyikan perintah untuk mengkonfirmasi bahwa Anda ingin menimpa file tujuan.
/-y	Meminta Anda untuk mengkonfirmasi bahwa Anda ingin menimpa file tujuan yang ada.

Opsi	Deskripsi
/z	Menyalin file berjaringan dalam mode yang dapat dihidupkan ulang.
/a	Menunjukkan file teks ASCII.
/b	Menunjukkan file biner.
<source>	Wajib. Menentukan lokasi dari mana anda ingin menyalin file atau serangkaian file. Sumber dapat dari huruf drive dan titik dua, nama direktori, nama file, atau kombinasi keduanya.
<destination>	Wajib. Menentukan lokasi tempat anda ingin menyalin file atau serangkaian file. Tujuan dapat dari huruf drive dan titik dua, nama direktori, nama file, atau kombinasi keduanya.

➤ Contoh

Untuk menyalin file bernama Robin.typ dari drive dan direktori saat ini ke direktori yang sudah ada bernama Birds yang terletak di drive C.

copy robin.typ c:birds

2) Kamus

Menurut Kamus Besar Bahasa Indonesia (KBBI) kamus memiliki beberapa pengertian. Pengertian yang pertama, kamus adalah buku acuan yang memuat kata dan ungkapan, biasanya disusun menurut abjad berikut keterangan tentang makna, pemakaian, atau terjemahannya. Pengertian yang kedua, kamus adalah buku yang

memuat kumpulan istilah atau nama yang disusun menurut abjad beserta penjelasan tentang makna dan pemakaiannya.

Chaer (2007:179), menyebutkan pengertian kamus yang dikemukakan oleh beberapa para ahli yaitu:

- a) Kridalaksana menyebutkan bahwa kamus adalah buku referensi yang memuat daftar kata atau gabungan kata dengan keterangan mengenai berbagai segi maknanya dan penggunaannya dalam bahasa, biasanya disusun menurut abjad.
- b) Labrousse menyebutkan bahwa kamus adalah buku berisi kumpulan kata-kata sebuah bahasa yang disusun secara alfabetis diikuti dengan definisi atau terjemahannya dalam bahasa lain.
- c) Keraf mendefinisikan kamus sebagai sebuah buku referensi, memuat daftar kata-kata yang terdapat dalam sebuah bahasa, disusun secara alfabetis, disertai keterangan cara menggunakan kata itu.

Berdasarkan definisi kamus dari Kamus Besar Bahasa Indonesia dan pendapat para ahli diatas dapat diambil kesimpulan bahwa kamus merupakan sebuah buku yang memuat sekumpulan kata atau istilah-istilah asing yang disusun menurut abjad berikut keterangan tentang makna dan pemakaiannya. Kamus sudah banyak diterapkan dalam berbagai ilmu pengetahuan, hal ini terpicu karena kebutuhan akan perkembangan ilmu pengetahuan yang semakin meningkat sehingga tuntutan akan pengetahuan juga semakin besar. Salah satu keuntungan penggunaan kamus adalah semua informasi terpusat dan mendetail, sehingga memudahkan dalam proses pencarian kata atau istilah yang dikehendaki.

Penyusunan sebuah kamus ada banyak hal yang harus diperhatikan dengan cermat. Hal-hal yang perlu diperhatikan antara lain, adalah tujuan menyusun kamus, untuk siapa kamus disusun, bagaimana tingkat pendidikan calon pengguna kamus yang ditetapkan sebagai calon konsumen, berapa jumlah kata entri/lema yang akan dimuat, bagaimana menyusun kata entri/lema, dari mana sumber datanya, bagaimana teknik penyajian definisinya, berapa lama waktu yang disediakan, berapa jumlah dana dan tenaga yang diperlukan, serta sistem kerja yang bagaimana yang akan digunakan (secara manual atau dengan peralatan canggih). Langkah-langkah yang perlu dilakukan secara berurutan dalam menyusun kamus adalah sebagai berikut (Sugiono, 2002:144):

- a) Menyiapkan segala sesuatu yang diperlukan
- b) Menetapkan tujuan dan sasaran konsumen yang dipilih
- c) Menentukan sumber data
- d) Menetapkan tata cara pengumpulan data
- e) Melakukan kegiatan pengumpulan data sesuai dengan tata cara yang telah ditetapkan
- f) Menata hasil pengumpulan data sesuai dengan sistem kerja yang dipilih
- g) Melakukan penyeleksian data sesuai dengan teknik leksikografi. Menurut Sunarso (1990:14), proses penyeleksian data ada beberapa langkah yang harus diperhatikan, yaitu:
 - Data dilihat secara sepintas secara keseluruhan dan sekaligus memisahkan data yang diperlukan dari data yang harus disisihkan

- Data dikelompokkan berdasarkan bentuk kata atau entri masukan
 - Data dipilah atas dasar kategori makna
 - Data yang disisihkan jangan dibuang, tetapi disimpan sebagai data cadangan
 - Data disusun menurut urutan abjad per kelompok bentuk entri masukan
- h) Memberi definisi makna secara baik dan tepat sesuai dengan teknik leksikografi.
- i) Melakukan penyuntingan terhadap hasil pemberian definisi
- j) Melakukan pengolahan data sesuai dengan sistem kerja yang dipilih
- k) Melakukan penataan data hasil pengolahan menjadi naskah kamus
- l) Melakukan koreksi dan penyuntingan naskah kamus
- m) Menyiapkan naskah kamus siap cetak
- n) Memeriksa hasil cetak
- o) Menyiapkan naskah penerbitan

Menurut Olivia (2008:32), kamus memiliki beberapa jenis seperti kamus yang memberikan informasi tentang sejarah kata-kata *Oxford English Dictionary*, kamus sinonim seperti *Webster Dictionary of Synonym*, kamus singkatan, kamus pelajaran tertentu, kamus ejaan, dan sebagainya. Kamus yang di kembangkan dalam penelitian ini adalah kamus jenis pelajaran tertentu yakni pelajaran teknik komputer dan jaringan materi *command line interface*.

Berdasarkan uraian tentang kamus dan perintah CLI Linux dan Windows dapat disimpulkan bahwa dalam penelitian ini kamus perintah dasar *command line interface* Linux dan Windows adalah kamus yang berisi kumpulan perintah-

perintah dasar CLI Linux dan Windows yang disusun sesuai abjad dan memuat penjelasan tentang perintah tersebut berupa fungsi, sintaks, parameter, dan contoh penggunaan perintah. Kamus ini dikembangkan dengan basis teknologi *mobile* Android.

3) Emulator

Emulator atau lebih tepatnya peranti lunak emulator memungkinkan suatu program atau peranti lunak yang dibuat pada awalnya oleh suatu sistem komputer dan untuk dijalankan dalam sistem itu (atau dijalankan dalam suatu sistem yang didedikasikan), dapat dijalankan dalam sistem komputer yang sama sekali berbeda (DeveloperWorks, 2004). Sebagai contoh suatu program Windows dapat dijalankan di sistem operasi Linux dengan menggunakan peranti lunak emulator Wine. Ada pula program yang mengemulasikan suatu komputer dalam komputer, misalnya VMware. Contoh lain adalah program-program emulator untuk menjalankan permainan komputer yang awalnya hanya bisa dijalankan pada konsolnya masing-masing, misalnya Nintendo, Atari, PlayStation, Xbox dan lain-lain.

Memprogram emulator adalah hal yang amat sulit dan dimulai dari mengumpulkan informasi tentang sistem yang hendak di-*emulate* dan mengakali komputer agar mampu meniru *hardware* dan instruksi-instruksi yang dijalankan oleh sistem tersebut (DeveloperWorks, 2004). Salah satu contohnya adalah menulis program untuk mengemulasi cara kerja *processor* kuno seperti Apple II atau Intel 8086.

Emulator terdiri dari dua jenis: *single-system/dedicated* emulator dan *multi-system emulator*. *Single-system/dedicated* emulator hanya dapat meng-*emulate* satu jenis sistem, sedangkan *multi-system* emulator dapat meng-*emulate* beberapa jenis sistem sekaligus. Pada dasarnya kualitas *single-system/dedicated* emulator lebih baik daripada *multi-system* emulator karena hanya didesain khusus untuk satu jenis sistem sehingga kemungkinan untuk terjadi *compatibility* problem dapat diminimalisasi. Selain itu kebutuhan *resources* untuk *single-system/dedicated* emulator biasanya lebih kecil dibandingkan *multi-system* emulator. Ada tiga faktor utama yang menentukan kemungkinan suatu sistem di-*emulate* atau tidak:

- Kepopuleran: semakin populer suatu sistem, maka semakin banyak orang yang berusaha untuk meng-*emulate* sistem tersebut.
- Tersedianya informasi: semakin banyak informasi yang tersedia tentang suatu sistem (baik *hardware* maupun *software*) akan amat membantu proses penulisan emulator.
- Kesulitan teknis: banyak sistem yang diproteksi dengan menggunakan enkripsi. Hal ini akan membuat sulit mengemulasi sistem tersebut.

b. Pengembangan Aplikasi Android *Mobile*

Pengembangan perangkat lunak (*software engineering*) merupakan sebuah proses berlapis yang memungkinkan *developer* mengembangkan perangkat lunak komputer yang berkualitas tinggi (Pressman, 2012:15). Menurut Rosa A.S & Shalahuddin (2011:4), pengembangan perangkat lunak merupakan pembangunan dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang

dipercaya dan bekerja secara efisien menggunakan mesin. Berdasarkan dua pengertian tersebut, dapat disimpulkan bahwa pengembangan perangkat lunak adalah proses yang aplikasiatis untuk membangun perangkat lunak yang berkualitas. Terdapat empat lapisan dalam pengembangan perangkat lunak. Lapisan-lapisan tersebut tersaji dalam Gambar 2.



Gambar 2. Lapisan-Lapisan Pengembangan Perangkat Lunak (Pressman, 2012)

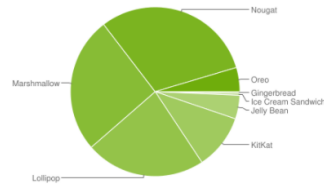
Pondasi dalam pengembangan perangkat lunak yaitu *process layer* atau lapisan proses. Proses rekayasa perangkat lunak berfungsi untuk menghubungkan lapisan-lapisan teknologi dan memungkinkan pengembangan perangkat lunak yang rasional dan tepat waktu. Proses mendefinisikan suatu perangkat kerja yang harus dibangun sedemikian rupa sehingga dapat terjadi penghantaran teknologi rekayasa perangkat lunak yang efektif. Proses-proses perangkat lunak membentuk dasar bagi kendali manajemen proyek perangkat lunak dan membangun konteks dimana metode-metode teknis dapat diterapkan, dimana produk-produk kerja (model, dokumen, data, laporan, formulir, dsb) dibuat, dimana tonggak waktu dibuat, dimana kualitas perangkat lunak dipastikan, dan dimana perubahan dapat diatur dengan tepat.

Android merupakan sistem operasi populer yang berbasis linux yang dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan tablet (Enterprise, 2005:1). Android diciptakan pertama kali oleh Andy Rubin yang bekerja membuat sistem operasi pada perusahaan Android Inc, dan pada tahun 2005 perusahaan Android Inc diakuisisi oleh Google. Android telah dirilis dalam berbagai versi seperti tertera dalam Tabel versi *platform* Android.

Android memiliki berbagai keunggulan sebagai *software* yang memakai basis kode komputer yang bisa didistribusikan secara terbuka (*open source*) sehingga pengguna bisa membuat aplikasi baru di dalamnya. Android memiliki aplikasi Native Google yang terintegrasi seperti Gmail, Google Maps, dan Google Calendar. Para penggemar *open source* kemudian membangun komunitas yang membangun dan berbagi Android berbasis *firmware* dengan sejumlah penyesuaian dan fitur-fitur tambahan, seperti kemampuan untuk menyimpan *download* aplikasi pada *microSD card*.

Berikut adalah data dari jumlah distribusi *platform* perangkat lunak menggunakan Android. Data ini dikumpulkan secara periodik satu minggu sekali dan pembaruan data terakhir pada tanggal 16 April 2018 (*Developers*, 2018).

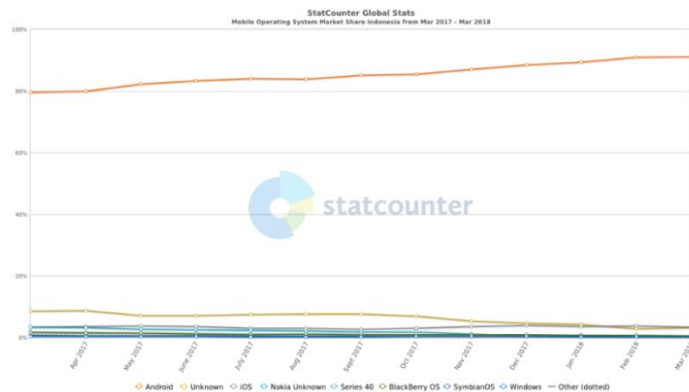
Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.4%
4.1.x	Jelly Bean	16	1.7%
4.2.x		17	2.2%
4.3		18	0.6%
4.4	KitKat	19	10.5%
5.0	Lollipop	21	4.9%
5.1		22	18.0%
6.0	Marshmallow	23	26.0%
7.0	Nougat	24	23.0%
7.1		25	7.8%
8.0	Oreo	26	4.1%
8.1		27	0.5%



Data collected during a 7-day period ending on April 16, 2018.
Any versions with less than 0.1% distribution are not shown.

Gambar 3. Data Jumlah Distribusi *Platform* Perangkat Android

Berikut ini adalah grafik perbandingan sistem operasi *mobile* yang ada di Indonesia dari Maret 2017 sampai dengan Maret 2018. Data diambil secara periodik satu bulan sekali dan pembaruan data terakhir pada Maret 2018 (StatCounter, 2018).



Gambar 4. Grafik Perbandingan Sistem Operasi *Mobile* di Indonesia

Berdasarkan grafik di atas, dapat disimpulkan bahwa *platform* Android yang sekarang banyak digunakan adalah versi KitKat, Lollipop, Marshmallow dan Nougat. Aplikasi ini dikembangkan pada sistem operasi Android karena sistem

operasi ini terus berkembang dan memiliki potensi yang besar di masa depan. Hal ini sependapat dengan Irsyad (2016:7) yang menyebutkan kelebihan pengembangan aplikasi pada *platform* Android:

1) *User Friendly*

Android sangat mudah dioperasikan, orang yang buta Android pun akan mampu mengoperasikannya dalam waktu yang singkat.

2) Bersifat *Open Source*

Karena Android dibangun di atas kernel Linux, maka siapapun dapat mengembangkan dan memodifikasi Android tanpa harus membayar.

3) Merakyat

Sistem operasi Android sangat cocok untuk berbagai kalangan. Dari kelas bawah sampai kelas atas sangat banyak yang menggunakan Android. Jadi tidak heran sistem operasi ini sangat populer di masyarakat.

4) Dukungan Berbagai Aplikasi

Android didukung oleh ribuan bahkan jutaan aplikasi yang tersedia untuk menunjang kinerja Android.

Menurut Widianti (2000:6) sebuah *software* atau perangkat lunak yang bertugas sebagai *front end* pada sebuah aplikasi yang digunakan untuk mengolah berbagai macam data sehingga menjadi sebuah informasi yang bermanfaat untuk penggunaannya dan juga aplikasi yang berkaitan. Jadi aplikasi merupakan sebuah program yang dibuat dalam sebuah perangkat lunak dengan komputer untuk

memudahkan pekerjaan atau tugas-tugas seperti penerapan, penggunaan dan penambahan data yang dibutuhkan.

Menurut Buyens (2001:8) *mobile* dalam konteks *software* dapat diartikan sebagai perpindahan dari suatu tempat ke tempat yang lain. Sedangkan dari kamus *cambridge mobile* kata *mobile* sering digunakan untuk mendeskripsikan sebuah layanan di dalam *handphone* atau komputer kecil. Dari dua sumber di atas, dapat disimpulkan, *mobile* merupakan layanan dalam *handphone* yang dapat dijalankan walaupun pengguna berpindah dari suatu tempat ke tempat yang lain.

Jadi aplikasi *mobile* merupakan potongan dari *software* yang dirancang berjalan di perangkat *mobile*, seperti *smartphone* atau tablet. Aplikasi *mobile* ketika selesai dibuat maka akan didistribusikan melalui pihak ketiga, yang biasa disebut sebagai *marketplace* atau *apps store*.

Berdasarkan uraian penjelasan diatas mengenai pengembangan perangkat lunak, Android, dan aplikasi *mobile*, maka dapat disimpulkan bahwa pengembangan aplikasi Android *mobile* adalah proses yang aplikasiatis yang berkualitas pada sistem operasi Android yang dirancang dapat berjalan di perangkat *mobile*, seperti *smartphone* atau tablet.

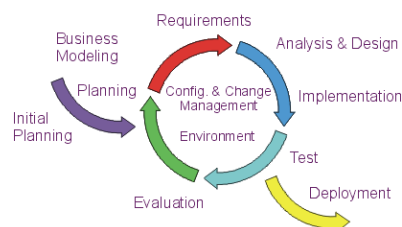
c. Model Pengembangan

Menurut Budi Sutedja (2002) dalam Ramadhina (2015:328) memaparkan “Untuk membangun suatu sistem yang kompleks secara sistematis dan terintegrasi, dibutuhkan metode-metode pembangunan sistem agar dapat menuntun pembuat untuk menghasilkan suatu sistem yang standar”. Model

pengembangan merupakan tahapan atau langkah-langkah yang dilakukan dalam pengembangan (Wulandari, 2015:27). Dalam penelitian ini model pengembangan yang digunakan yaitu *Rational Unified Process* (RUP).

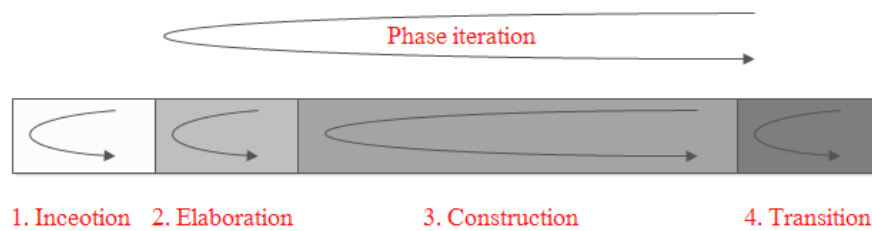
Menurut Rosa A. S. & Shalahuddin (2011:05), *Unified Process* merupakan sebuah proses pengembangan perangkat lunak yang dilakukan secara iteratif dan inkremental. Iterasi bisa dilakukan di dalam setiap tahap untuk menghasilkan perbaikan fungsi inkremental dimana setiap iterasi akan memperbaiki iterasi berikutnya. Salah satu *Unified Process* yang terkenal yaitu *Rational Unified Process*.

Rational Unified Process (RUP) adalah pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (*iterative*), fokus pada arsitektur (*architecture-centric*), lebih diarahkan berdasarkan penggunaan kasus (*use case driven*). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik (*well defined*) dan penstrukturan yang baik (*well structured*). RUP menyediakan pendefinisian struktur yang baik untuk alur hidup proyek perangkat lunak. RUP adalah sebuah produk proses perangkat lunak yang dikembangkan oleh *Rational Software* yang diakuisisi oleh IBM di bulan Februari 2003. Proses pengulangan/*iteratif* pada RUP dapat dilihat pada Gambar berikut :



Gambar 5. Proses Iteratif RUP

Rational Unified Process (RUP) adalah contoh dari model proses modern yang telah diturunkan dari dengan UML dan pengembangan perangkat lunak terpadu yang terkait (Sommerville, 2011:50). RUP memiliki empat buah tahap atau fase yang dapat dilakukan secara iteratif yaitu *inception*, *elaboration*, *construction*, dan *transition*. Berikut ini adalah Gambar alur hidup RUP.



Gambar 6. Alur Hidup RUP

Berikut ini penjelasan untuk setiap fase pada RUP :

a) *Inception* (Permulaan)

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan aplikasi yang akan dibuat (*requirements*). Tahap yang dibutuhkan yaitu memahami ruang lingkup proyek seperti biaya, waktu, kebutuhan dan risiko.

b) *Elaboration* (Perluasan/Perencanaan)

Tahap ini lebih difokuskan pada perencanaan arsitektur aplikasi. Tahap ini juga dapat mendeteksi apakah arsitektur aplikasi yang diinginkan dapat dibuat atau tidak. Tahap ini lebih pada analisis dan desain aplikasi serta implementasi aplikasi yang fokus pada aplikasi *prototype*.

c) *Construction* (Konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur aplikasi. Tahap ini lebih pada implementasi dan pengujian aplikasi yang fokus pada implementasi perangkat lunak kode program. Tahap ini menghasilkan produk perangkat lunak di mana menjadi syarat dari *Initial Operational capability Milestone* atau batas/tonggak kemampuan operasional awal.

d) *Transition* (Transisi)

Tahap ini lebih pada *deployment* atau instalasi aplikasi agar dapat dimengerti oleh *user*. Aktivitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian aplikasi apakah sudah memenuhi harapan *user*.

Pada setiap tahap RUP memiliki alur fokus aktivitas yang harus dilakukan selama proses pengembangan atau yang disebut dengan *workflows*. Menurut Sommerville (2003:51), *workflows* RUP terdiri dari dua bagian yaitu, enam *core process workflows* dan tiga *core supporting workflows*. Berikut penjelasan *workflow* pada RUP di Tabel 3.

Tabel 3. *Workflows* Model Pengembangan RUP.

<i>Workflow</i>	Deskripsi
<i>Business modelling</i>	<i>Business modelling</i> dimodelkan dengan menggunakan <i>use case</i> .
<i>Requirements</i>	Aktor yang berinteraksi dengan sistem didefinisikan dan <i>use case</i> dikembangkan untuk model sistem <i>requirements</i> .

Workflow	Deskripsi
<i>Analysis & design</i>	Model desain dibuat dan didokumentasikan menggunakan <i>architectural models</i> , <i>component models</i> , <i>object models</i> , dan <i>sequence models</i> .
<i>Implementation</i>	Komponen dalam sistem diimplementasikan dan disusun kedalam implementasi subsistem. Pembuatan kode otomatis dari model desain membantu mempercepat proses ini.
<i>Testing</i>	Pengujian adalah proses berulang yang dilakukan bersamaan dengan implementasi.
<i>Deployment</i>	Produk <i>release</i> dibuat, disebar dan diinstal ke <i>user</i> .
<i>Configuration & change management</i>	<i>Workflow</i> pendukung yang menangani perubahan pada sistem.
<i>Project management</i>	<i>Workflow</i> pendukung yang menangani perubahan pada sistem.
<i>Environment</i>	<i>Workflow</i> ini berkaitan dengan pembuatan perangkat lunak yang sesuai yang tersedia untuk pengembangan perangkat lunak.

Menurut Rosa A. S. & Shalahuddin (2011 : 106 - 109), RUP memiliki kelebihan dibanding *waterfall* yaitu :

- 1) RUP mengakomodasi perubahan kebutuhan perangkat lunak.
- 2) Integrasi bukanlah sebuah proses besar dan cepat di akhir proyek.

- 3) Risiko biasanya ditemukan atau dialamatkan selama proses integrasi awal.
- 4) Manajemen berarti membuat perubahan taktik pada produk.
- 5) Mendukung fasilitas penggunaan kembali.
- 6) Kesalahan dapat ditemukan dan diperbaiki pada beberapa iterasi sehingga menghasilkan arsitektur yang baik dan aplikasi berkualitas tinggi.
- 7) Lebih baik menggunakan “anggota proyek” dibandingkan susunan secara seri pada tim proyek.
- 8) Anggota tim belajar selama proyek berjalan.

Pengembangan perangkat lunak dapat diperbaiki seiring proses pengembangan perangkat lunak.

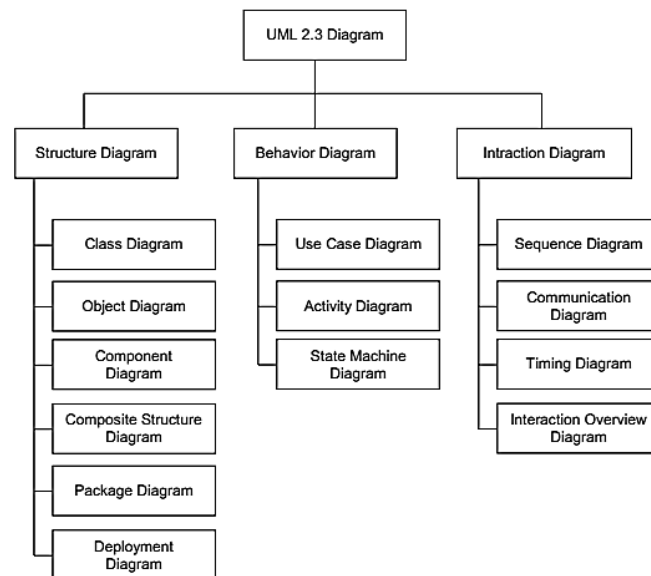
d. Perangkat Pengembangan

1) UML (*Unified Modelling Language*)

Menurut Rosa A. S & M Shalahudin (2011:117 - 118) desain UML adalah standar bahasa yang digunakan di dunia industri untuk mendefinisikan kebutuhan awal aplikasi, desain perancangan, dan arsitektur dalam pemrograman berorientasi objek. UML muncul karena kebutuhan pemodelan visual, mengGambarkan, membangun, dan juga dokumentasi terhadap sistem perangkat lunak. UML digunakan sebagai standarisasi bahasa sehingga mempermudah pengGambaran dari pemahaman terhadap suatu produk oleh banyak orang. UML terdiri dari 13 macam diagram yang dikelompokkan menjadi tiga kategori. Berikut pembagian kategori dapat dilihat pada Gambar 7.

Berikut ini penjelasan singkat dari pembagian kategori tersebut :

- a) *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk mengGambarkan suatu struktur statis dari aplikasi yang dimodelkan.
- b) *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk mengGambarkan kelakuan aplikasi atau rangkaian perubahan yang terjadi pada sebuah aplikasi.
- c) *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambar interaksi aplikasi dengan aplikasi lain maupun interaksi antar sub aplikasi pada suatu aplikasi.



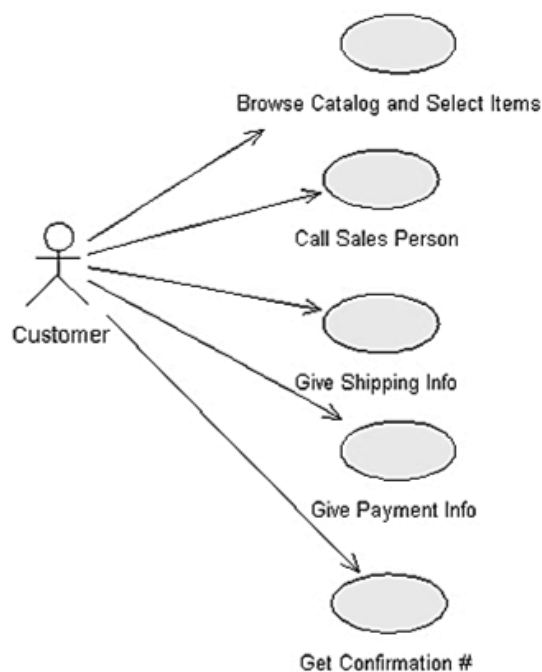
Gambar 7. Diagram UML (Rosa, 2011:121)

Pada penelitian ini diagram UML yang digunakan adalah *use case diagram*, *class diagram*, *sequence diagram*, dan *activity diagram*. Berikut penjelasan masing-masing diagram :

a. *Use Case Diagram*

Menurut Rosa A.S & Shalahuddin (2011:130), *use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) aplikasi informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan aplikasi informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah aplikasi informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Use case diagram dapat membantu pengembang dalam menentukan fungsi dan fitur dari sebuah perangkat lunak dari pandangan pengguna. Suatu *use case diagram* menggambarkan bagaimana pengguna berinteraksi dengan sistem melalui langkah-langkah yang disediakan untuk memenuhi tujuan tertentu (Pressman, 2012:162).

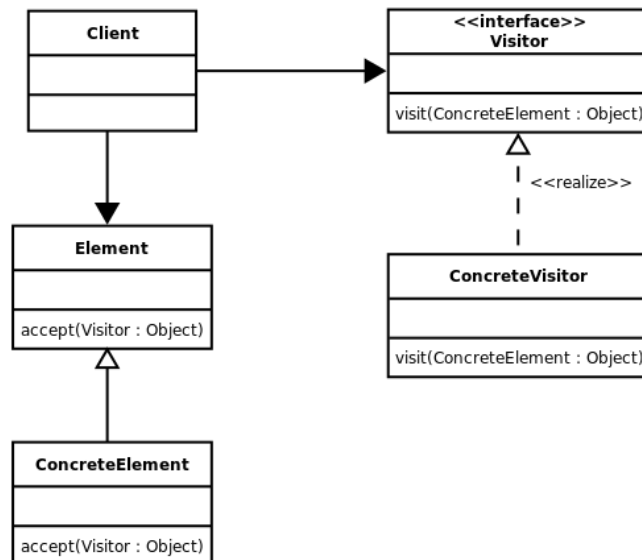


Gambar 8. Contoh *Use Case Diagram*

b. Class Diagram

Class diagram mengGambarkan struktur aplikasi dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun aplikasi (Rosa A.S & Shalahuddin, 2011:122). Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

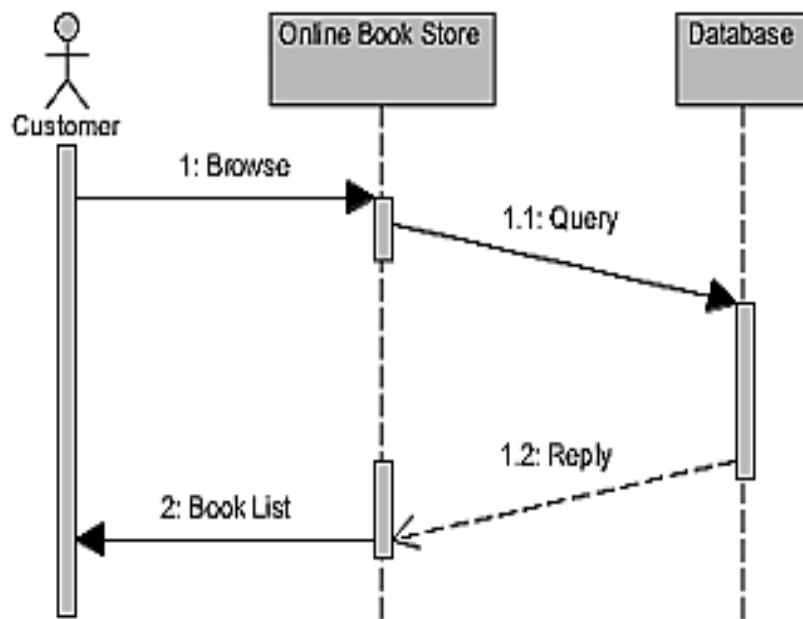


Gambar 9. Contoh *Class Diagram*

c. Sequence Diagram

Sequence diagram mengGambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek (Rosa A.S & Shalahuddin, 2011:137). Oleh karena itu untuk mengGambarkan *sequence diagram* maka harus diketahui objek-objek yang

terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Sequence diagram juga digunakan untuk memperlihatkan bagaimana event-event yang terjadi, bisa mengaitkan transisi dari suatu objek ke objek lainnya (Pressman, 2012:237).

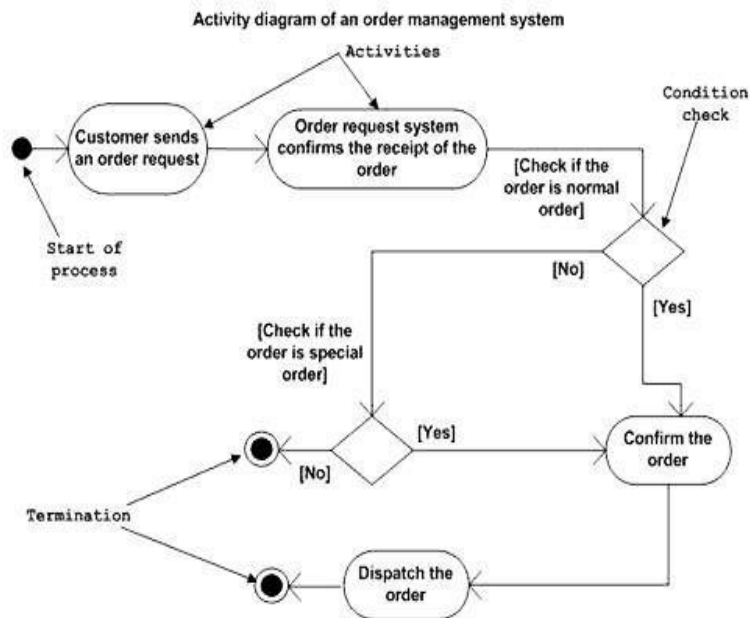


Gambar 10. Contoh *Sequence Diagram*

d. *Activity Diagram*

Pressman (2012:195) menjelaskan bahwa *activity diagram* adalah diagram UML yang melengkapi *use case* dengan memberikan representasi grafik dan aliran-aliran interaksi di dalam suatu skenario. Diagram ini mirip dengan *flowchart* namun yang membedakan adalah *activity diagram* dapat menunjukkan proses yang terjadi bersamaan. Menurut Rosa A.S & Shalahuddin (2011:134), *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah aplikasi atau proses bisnis. Yang perlu diperhatikan adalah bahwa *activity*

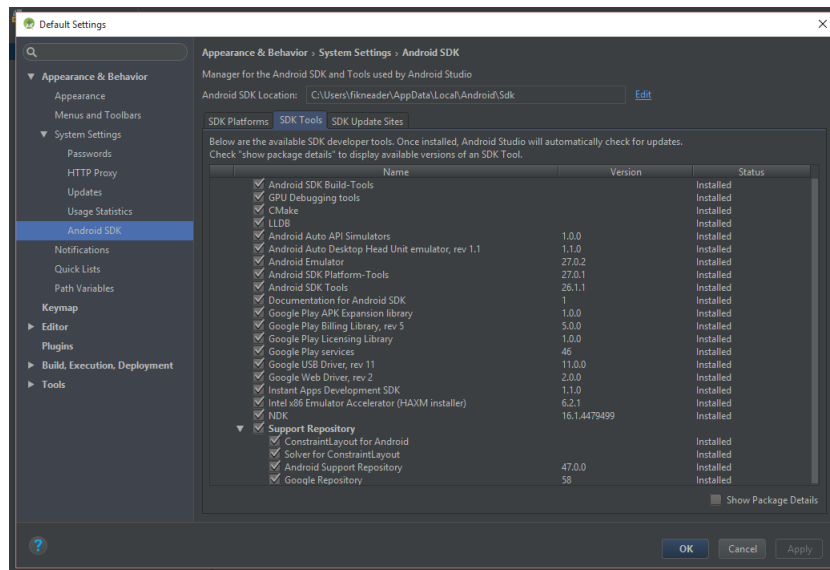
diagram menggambarkan aktivitas aplikasi bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh aplikasi.



Gambar 11. Contoh *Activity Diagram*

2) Android SDK

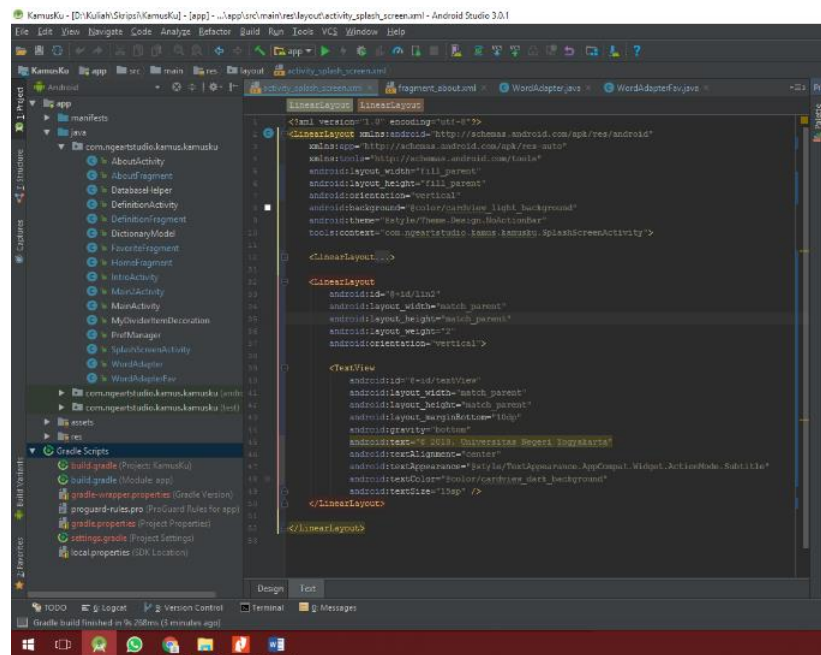
Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java (Nazruddin, 2014). Android SDK menyertakan contoh *project* dengan sumber kode, perangkat pengembangan, sebuah emulator, dan *library* yang dibutuhkan untuk membangun aplikasi Android. Aplikasi ditulis dalam bahasa pemrograman Java dan dijalankan di atas Dalvik, sebuah mesin virtual didesain untuk dapat berjalan di lapisan atas kernel Linux.



Gambar 12. Android SDK Manager for Windows

3) Android Studio

Android Studio adalah sebuah IDE (*Integrated Development Environment*) yang digunakan untuk pengembangan aplikasi Android (Developers, 2018). Android Studio merupakan IDE resmi untuk Android. Android Studio memiliki fitur editor kode cerdas (*Intelligent Code Editor*) yang memiliki kemampuan penyelesaian kode, optimalisasi, dan analisis kode yang canggih. Selain itu fitur *new project wizards* membuat proses memulai proyek baru menjadi jauh lebih mudah bahkan dapat mengimpor contoh kode Google dari gitHub. Berbagai modul baru digunakan dalam Android Studio ini, salah satunya adalah pengembangan aplikasi multi layar yang memudahkan pengembangan untuk membangun sebuah aplikasi untuk ponsel dan tablet Android, Android Wear, Android TV, Android Auto, dan Android Google Glass.



Gambar 13. Tampilan Android Studio

2. Pengujian Perangkat Lunak

Sebuah perangkat lunak perlu dijaga kualitasnya di mana kualitas bergantung pada kepuasan pelanggan. Pengujian diperlukan tidak hanya untuk meminimalisasi kesalahan secara teknik tapi juga kesalahan non teknis (misalnya pengujian pesan kesalahan yang muncul).

Pengujian perangkat lunak adalah sebuah elemen suatu topik yang memiliki cakupan luas dan sering dikaitkan dengan verifikasi dan validasi (Rosa A.S & Shalahuddin, 2011:211). Sedangkan menurut Pressman (2012:485), kualitas perangkat lunak dapat didefinisikan sebagai suatu proses perangkat lunak yang efektif diterapkan dalam arti kata proses perangkat lunak yang menyediakan nilai yang dapat diukur untuk mereka yang menghasilkan dan untuk mereka yang menghasilkannya. Perangkat lunak yang bermanfaat memiliki tiga poin penting

yaitu efektivitas proses perangkat lunak, produk yang mampu mengirimkan konten serta fungsi, dan mampu memberi nilai lebih bagi pengguna perangkat lunak.

Salah satu tolok ukur kualitas perangkat lunak menggunakan standar ISO 25010, yang dibuat oleh *International Organization for Standardization (ISO)* dan *International Electrotechnical Commission (IEC)* (Wagner, 2013 : 2). ISO 25010 menggantikan standar ISO/IEC 9126 (ISO, 2011). Model kualitas produk mengategorikan sifat kualitas produk menjadi delapan karakteristik yaitu *functional suitability, reliability, performance efficiency, usability, security, compatibility, maintainability, dan portability*. Karakteristik dan sub karakteristik standar pengujian ISO 25010 dapat dilihat di Lampiran 10.

Pengujian ISO 25010 untuk *mobile* meliputi empat aspek yaitu: *functional suitability, compatibility, usability, dan performance* (David, 2011:2). Oleh karena itu dalam pengujian perangkat yang dikembangkan akan menggunakan empat dari delapan karakteristik dalam ISO 25010 yaitu *functional suitability, compatibility, usability, dan performance*. Berikut penjelasan empat aspek tersebut :

a. *Functional Suitability*

Karakteristik yang menguji sejauh mana produk atau aplikasi dapat menyediakan fungsi untuk memenuhi kebutuhan yang dapat digunakan pada kondisi tertentu. *Functional suitability* terbagi menjadi tiga sub karakteristik:

- 1) *Functional completeness*, mengukur sejauh mana fungsi mencakup semua tugas dan tujuan pengguna.
- 2) *Functional correctness*, mengukur sejauh mana produk atau aplikasi memberikan hasil yang tepat sesuai kebutuhan.
- 3) *Functional appropriateness*, mengukur sejauh mana fungsi dapat memfasilitasi untuk menyelesaikan tugas dan tujuan tertentu.

b. *Compatibility*

Sejauh mana produk, aplikasi atau komponen dapat saling bertukar informasi dengan produk lain, aplikasi atau komponen, dan/atau menjalankan fungsi yang diperlukan, berbagi perangkat keras atau perangkat lunak dalam lingkungan yang sama secara sementara. *Compatibility* terbagi atas sub karakteristik:

- 1) *Coexistence*, sejauh mana suatu produk dapat menjalankan fungsi yang dibutuhkan secara efisien sementara berbagi sumber daya dengan produk lainnya, tanpa berdampak merugikan pada produk lainnya.
- 2) *Interoperability*, sejauh mana dua atau lebih aplikasi, produk atau komponen dapat saling bertukar informasi dan menggunakan informasi yang telah ditukar.

c. *Usability*

Sejauh mana produk atau aplikasi dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektif, efisien dan kepuasan dalam konteks penggunaan tertentu. Karakteristik ini terbagi menjadi enam sub karakteristik:

- 1) *Appropriateness recognizability*, sejauh mana pengguna dapat mengenali apakah suatu produk atau aplikasi yang sesuai dengan kebutuhan mereka.
- 2) *Learnability*, sejauh mana produk atau aplikasi dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dari pembelajaran untuk menggunakan produk atau aplikasi dengan efektif, efisien, bebas dari risiko dan kepuasan dalam konteks tertentu dalam penggunaan.
- 3) *Operability*, sejauh mana produk atau aplikasi memiliki atribut yang membuatnya mudah dioperasikan dan dikontrol.
- 4) *User error protection*, sejauh mana aplikasi melindungi dari pengguna jika membuat kesalahan.
- 5) *User interface aesthetics*, sejauh mana *user interface* memungkinkan interaksi yang menyenangkan dan memuaskan bagi pengguna.
- 6) *Accessibility*, sejauh mana produk atau aplikasi dapat digunakan oleh orang dalam jangkauan luas dari karakteristik dan kemampuan untuk mencapai suatu tujuan tertentu dalam konteks tertentu dalam penggunaan.

d. *Performance Efficiency*

Kinerja relatif terhadap jumlah sumber daya yang digunakan pada kondisi tertentu. Karakteristik ini terbagi menjadi tiga sub karakteristik:

- 1) *Time behaviour*, mengukur waktu respon dan proses dan kesalahan memenuhi persyaratan ketika menjalankan fungsi.

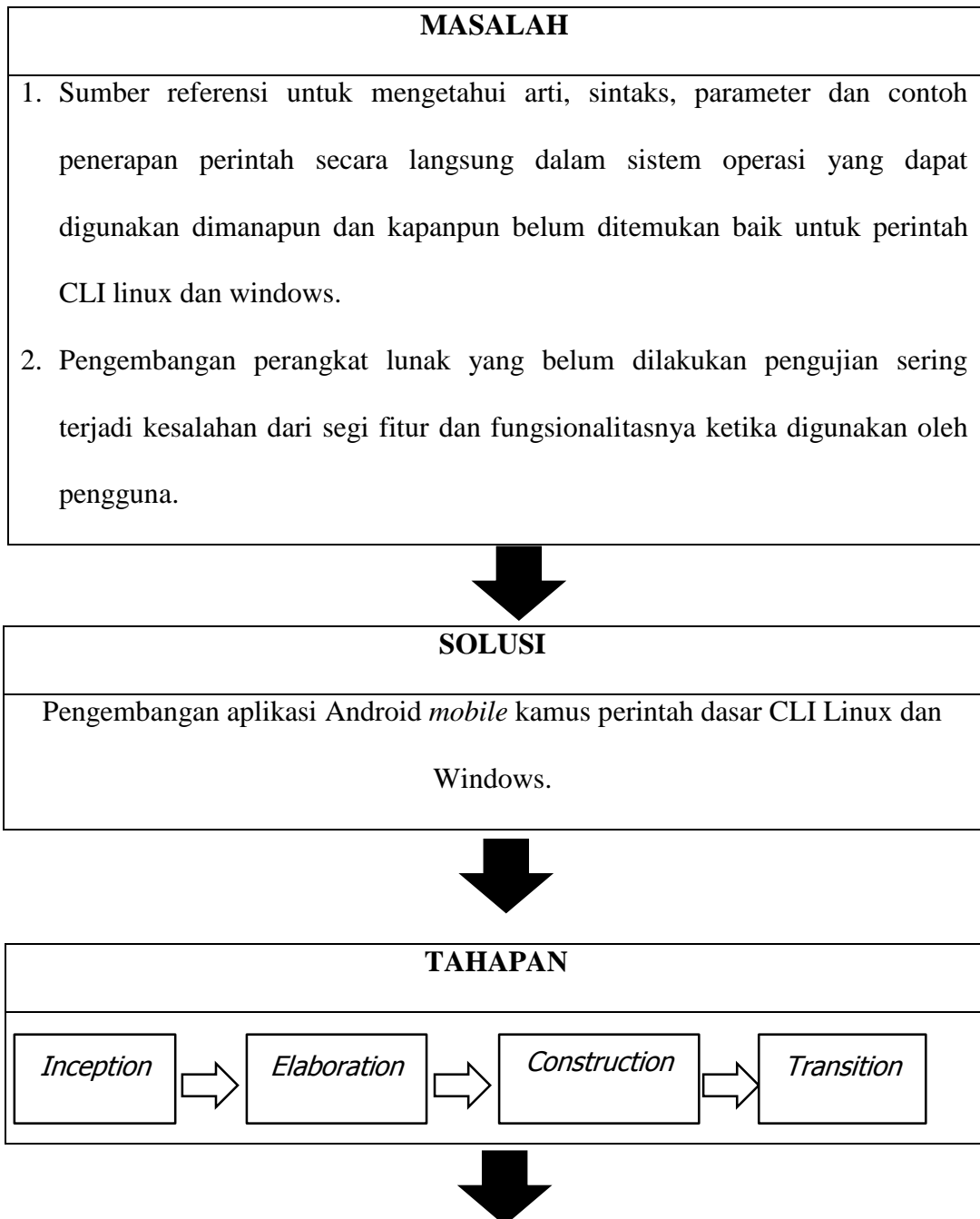
- 2) *Resource utilization*, mengukur jumlah dan jenis sumber daya yang digunakan oleh produk atau aplikasi sehingga dapat memenuhi persyaratan ketika menjalankan fungsi.
- 3) *Capacity*, mengukur batas maksimal parameter produk atau aplikasi dapat memenuhi persyaratan.

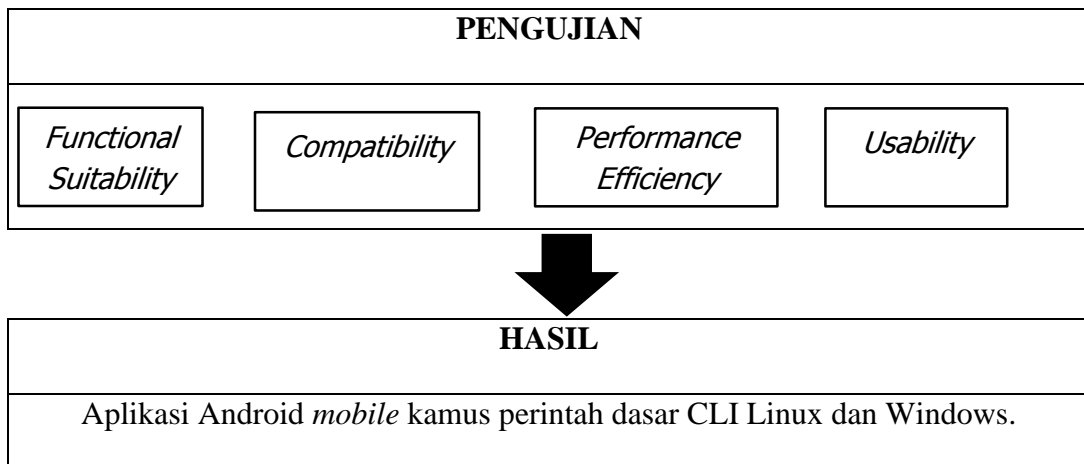
B. Penelitian yang Relevan

1. Hasil penelitian dengan judul “Pengembangan Aplikasi Mobile Kamus Istilah Jaringan Komputer Pada Platform Android” oleh Rohmad Dwi Jayanto pada tahun 2017. Tujuan dari penelitian ini merancang dan membuat aplikasi kamus istilah jaringan komputer pada platform Android. Penelitian ini menghasilkan aplikasi *mobile* kamus istilah jaringan komputer berbasis Android.
2. Hasil penelitian dengan judul “Pengembangan Aplikasi Android Kamus Binomial Nomenklatur (Binnom) Sebagai Media Bantu Belajar Siswa Kelas X MA Negeri Yogyakarta 1 Mata Pelajaran Biologi” oleh Didi Yulianto pada tahun 2016. Tujuan dari penelitian ini mengembangkan aplikasi kamus Binnom. Penelitian ini menghasilkan aplikasi kamus Binnom berbasis Android sebagai media bantu belajar siswa.
3. Hasil penelitian dengan judul “Pengembangan kamus Bahasa Jawa Bahasa Indonesia menggunakan java 2 micro edition (J2ME)” oleh Kartika Yudha Pratama pada tahun 2013. Tujuan dari penelitian ini adalah merancang dan membuat aplikasi kamus Bahasa Jawa Bahasa Indonesia versi *mobile* pada

ponsel menggunakan J2ME. Penelitian ini menghasilkan kamus Bahasa Jawa Bahasa Indonesia versi *mobile* dengan J2ME.

C. Kerangka Pikir





D. Pertanyaan Penelitian

Pertanyaan penelitian dari pengembangan Aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sebagai berikut:

1. Bagaimanakah mengembangkan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sesuai dengan model pengembangan *Rational Unified Process* (RUP) ?
2. Apakah aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memenuhi aspek *functional suitability* ?
3. Apakah aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memenuhi aspek *usability* ?
4. Apakah aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memenuhi aspek *compatibility* ?
5. Apakah aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memenuhi aspek *performance efficiency* ?

BAB III

METODE PENELITIAN

A. Model Penelitian

Model pengembangan dalam penelitian “Pengembangan Aplikasi Android *Mobile* Kamus Perintah Dasar CLI Linux Dan Windows” ini menggunakan metode *Research and Development* (R&D). Menurut Borg and Gall (1989:782) dalam Munir (2013:309) menerangkan yang dimaksud dengan *research* dan *development* (R&D) adalah proses mengembangkan dan memvalidasi produk pendidikan (*a process used develop and validate educational product*). Metode penelitian dan pengembangan adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut (Sugiyono, 2014). Metode penelitian dan pengembangan bukan bertujuan untuk menghasilkan teori baru maupun menguji teori yang sudah ada, melainkan untuk menghasilkan sebuah produk baru atau mengembangkan produk yang sudah ada agar dapat bermanfaat bagi sarannya. Metode penelitian dan pengembangan ini merupakan yang paling relevan untuk digunakan dalam penelitian.

Produk yang dihasilkan dalam penelitian ini adalah aplikasi Android *mobile* kamus perintah dasar CLI Linux dan Windows. Pengguna dari aplikasi ini adalah siswa SMK khususnya jurusan Teknik Komputer dan Jaringan. Oleh karena itu untuk mendapatkan produk yang sesuai, maka dalam pengembangan perangkat lunak ini peneliti menggunakan model pengembangan *Rational Unified Process* (RUP). Alasan pemilihan model pengembangan RUP karena RUP menggunakan

proses iteratif dan *incremental* sehingga mampu mengakomodasi perubahan kebutuhan perangkat lunak (Rossa A.S. & Shalahuddin, 2011).

B. Prosedur Pengembangan

Prosedur pengembangan menggunakan model pengembangan perangkat lunak *Rational Unified Process* (RUP) memiliki tahapan mulai dari *inception*, *elaboration*, *construction*, dan *transition*. Berikut penjelasan dari setiap tahapan:

1. *Inception*

Tahapan *inception* lebih pada memodelkan proses bisnis yang dibutuhkan (*business modelling*) dan mendefinisikan kebutuhan *user* akan aplikasi yang akan dibangun (*requirements*). Berikut tahapan yang dibutuhkan pada tahap *inception* :

- a. Memahami ruang lingkup proyek seperti biaya, waktu, kebutuhan, dan risiko.
- b. Memahami model bisnis yang ada dalam proyek.
- c. Membangun model bisnis dari proyek.
- d. Memahami kebutuhan *user* dengan tepat.

Hasil yang diharapkan dari tahap ini adalah memenuhi *Life Cycle Objective Milestone* dengan kriteria sebagai berikut :

- a. Umpan balik dari pendefinisian ruang lingkup aplikasi, perkiraan biaya, perkiraan jadwal.
- b. Kebutuhan dimengerti dengan pasti dan sejalan dengan kasus primer yang dibutuhkan.
- c. Kredibilitas perkiraan biaya, jadwal, penentuan skala prioritas, risiko, dan proses pengembangan.

Pada tahap ini untuk mengumpulkan data menggunakan teknik observasi dan wawancara. Peneliti melakukan wawancara dengan guru TKJ terutama pengampu mata pelajaran sistem operasi, administrasi server, rancang bangun jaringan, dan jaringan nirkabel. Hasil dari observasi dan wawancara ini yaitu analisis kebutuhan berupa spesifikasi produk yang dibutuhkan dalam pengembangan aplikasi. Kemudian dilakukan pendefinisian ruang lingkup, pembuatan model bisnis, dan perkiraan jadwal. Jika pada tahap ini target yang diharapkan tidak tercapai maka dapat diulangi kembali dengan direncanakan ulang agar kriteria yang diharapkan dapat tercapai.

2. *Elaboration*

Tahap ini lebih difokuskan pada perencanaan arsitektur aplikasi. Tahap ini juga dapat mendeteksi apakah arsitektur aplikasi yang diinginkan dapat dibuat atau tidak. Mendeteksi risiko-risiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih pada analisis dan desain aplikasi serta implementasi aplikasi yang fokus pada *prototype* aplikasi. Kegiatan yang dilakukan pada tahap ini yaitu:

- a. Pembuatan *use case* dimana aktor dan kasus yang berkaitan telah diidentifikasi.
- b. Pembuatan *expanded use case* (deskripsi dari *use case*) dari arsitektur perangkat lunak yang akan dikembangkan.
- c. Perancangan *Unified Modelling Language* (UML) yang meliputi *activity diagram*, *sequence diagram*, dan *class diagram*.
- d. Perancangan *design user interface* yang meliputi pembuatan *storyboard*, *Hierarchy Task Analysis* (HTA), dan *mockup*.

- e. *Prototype* dengan digital *mockup* untuk mengurangi risiko teknis.
- f. Memperbaiki model bisnis yang telah dibuat.

3. *Construction*

Tahap ini fokus pada pengembangan komponen dan fitur-fitur aplikasi. Tahapan ini lebih pada implementasi dan pengujian aplikasi yang fokus pada implementasi perangkat lunak pada kode program. Kegiatan yang dilakukan pada tahap ini yaitu:

- a. Perbaiki perancangan UML dan *user interface*.
- b. Implementasi *user interface* dalam *layout* Android Studio.
- c. Implementasi kode aplikasi (*coding*) di Android Studio dengan menggunakan bahasa Java.
- d. Pengujian fungsionalitas.

4. *Transition*

Tahap ini fokus pada tahap *deployment* atau instalasi aplikasi agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana lebih dari *Initial Operational Capability Milestone*. Kegiatan pada tahap ini lebih pada pelatihan *user*, pemeliharaan, dan pengujian aplikasi apakah sudah memenuhi kebutuhan *user* atau belum.

C. **Subjek, Tempat, dan Waktu Penelitian**

Pada penelitian pengembangan digunakan subjek penelitian untuk menguji aspek *usability* dan *functional suitability*. Pengujian *functional suitability* menggunakan empat responden ahli pengembang perangkat lunak. Sedangkan

untuk menguji aspek *usability* menggunakan 30 responden yang diambil dari siswa kelas XI TKJ 2 SMK N 1 Bantul. Subjek untuk menguji *performance efficiency* dan *compatibility* menggunakan dokumentasi aplikasi.

Waktu penelitian dilaksanakan 28 Mei 2018 sampai dengan 30 Juni 2018. Tempat penelitian untuk pengembangan aplikasi, pengujian aplikasi, dan revisi aplikasi dilaksanakan di laboratorium aplikasi informasi program studi Pendidikan Teknik Informatika Universitas Negeri Yogyakarta dan Laboratorium Komputer SMK N 1 Bantul. Tempat untuk melakukan uji coba terhadap pengguna dilaksanakan di SMK N 1 Bantul, Yogyakarta.

D. Variabel Penelitian

1. Variabel Penelitian

Variabel penelitian pada penelitian pengembangan aplikasi Android *mobile dictionary and emulator* perintah dasar *command line interface* Linux dan Windows adalah pengujian kualitas perangkat lunak yang mengacu pada standar ISO 25010 yang diantaranya meliputi :

- a. *Functional Suitability*
- b. *Usability*
- c. *Performance efficiency*
- d. *Compatibility*

2. Definisi Operasional Variabel

Berikut merupakan operasional dari tiap variabel dalam penelitian ini :

- a. *Functional Suitability*

Perangkat lunak memiliki kemampuan untuk menjalankan fungsi dengan baik dan lancar sesuai dengan kebutuhan *user*.

b. *Usability*

Perangkat lunak memiliki kemampuan untuk memenuhi kebutuhan *user*, kemudahan, dan kenyamanan pada saat menggunakan perangkat lunak.

c. *Compatibility*

Perangkat lunak memiliki kemampuan untuk bekerja dalam aplikasi operasi atau berbagai spesifikasi yang berbeda dalam menjalankan setiap fungsi yang ada.

d. *Performance efficiency*

Perangkat lunak memiliki kemampuan untuk bekerja dengan berbagai sumber daya perangkat yang telah ada agar dapat menyesuaikan dengan baik.

E. Metode dan Alat Pengumpul Data

Metode dan alat pengumpulan data yang digunakan dalam penelitian pengembangan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sebagai sumber tambahan bahan belajar siswa jurusan TKJ di SMK N 1 Bantul adalah sebagai berikut :

1. Wawancara

Wawancara adalah cara pengumpulan data yang digunakan memperoleh informasi secara langsung dari sumbernya (Sudaryono, 2011:88). Wawancara dilakukan agar memperoleh informasi secara mendalam. Wawancara dalam penelitian ini dilaksanakan untuk mengumpulkan data yang digunakan untuk

analisis kebutuhan. Wawancara dilakukan dengan Ibu Dariyati selaku Ketua Jurusan TK, Bapak Rohmad Dwiyanto selaku guru pengampu mata pelajaran administrasi server, serta siswa kelas XI TKJ 2 SMK N 1 Bantul.

2. Observasi

Observasi adalah teknik untuk mengumpulkan data dengan mengamati kegiatan yang sedang berlangsung (Sudaryono, 2011:90). Teknik ini digunakan dalam mengumpulkan data di sekolah untuk membantu proses analisis kebutuhan dan proses pengujian perangkat lunak pada aspek *performance efficiency* dan *compatibility*.

3. Kuesioner

Kuesioner/angket adalah teknik pengumpulan data secara tidak langsung artinya peneliti tidak langsung bertanya jawab kepada responden (Sudaryono, 2011:84). Instrumen atau alat pengumpulan datanya yang disebut angket berisi sejumlah pertanyaan yang harus dijawab oleh responden. Pengumpulan data menggunakan teknik ini dilakukan untuk menguji perangkat lunak dari aspek *functional suitability* dan *usability*. Kuesioner akan dibagikan kepada 30 responden yaitu siswa kelas XI TKJ 2.

F. Instrumen Penelitian

Instrumen pada pengembangan aplikasi Android *mobile* kamus perintah dasar CLI Linux dan Windows terdiri dari instrumen untuk pengujian perangkat lunak berdasarkan standar ISO 25010 pada empat aspek yang diujikan yaitu, *functional suitability*, *usability*, *compatibility*, *performance efficiency*.

1. Instrumen Uji Materi

Pengujian materi bertujuan untuk memastikan konten yang ditampilkan pada aplikasi *Android mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sudah sesuai dengan sumber materi dan layak untuk digunakan dalam penelitian. Uji materi dilakukan oleh satu orang ahli dalam bidang jaringan dan administrasi server, yaitu Ibu Dariyati, S.Kom selaku ketua jurusan TKJ SMK N 1 Bantul dan guru pengampu administrasi server.

2. Instrumen *Functional Suitability*

Aspek yang digunakan untuk mengukur aspek *functional suitability* berupa daftar menggunakan *checklist* pada *test-case* yang berisi daftar fungsi aplikasi sesuai analisis pada kebutuhan fungsional. Aspek *functionality* diuji oleh ahli dalam bidang pengembangan perangkat lunak. Instrumen untuk mengukur aspek *functional suitability* dapat dilihat pada Tabel 4.

Tabel 4. Tabel Instrumen Pengujian Aspek *Functional Suitability*

No	Fitur	Deskripsi	Langkah	Kegiatan	Hasil yang diharapkan	Hasil Keluaran	
						Berhasil	Gagal
1.	Beranda	Fitur untuk menampilkan halaman utama (dashboard)	1	Sentuh icon beranda yang ada di menu bagian bawah.	Halaman utama muncul		

Dan seterusnya... (Angket lengkap terlampir pada Lampiran 11).

Keterangan instrumen *test case* di atas sebagai berikut:

- a. Kolom fitur: berisi fitur yang terdapat di dalam aplikasi
- b. Kolom deskripsi: berisi penjelasan dari fitur
- c. Kolom langkah: berisi tahapan untuk menjalankan fungsi fitur
- d. Kolom kegiatan: berisi keterangan langkah yang dibutuhkan
- e. Kolom hasil: berisi deskripsi hasil yang diharapkan
- f. Kolom hasil keluaran: berisi indikator gagal atau berhasil suatu fitur dijalankan

Dari *test case* ini didapatkan hasil pengujian *functional suitability*. Sedangkan pernyataan kesimpulan dari para ahli setelah melakukan pengujian *functional suitability* merupakan *functional completeness* dan *functional appropriateness*.

3. Instrumen *Usability*

Pengujian aspek *usability* pengembangan aplikasi Android *mobile* kamus perintah dasar CLI Linux dan Windows menggunakan kuesioner yang dibagikan langsung kepada pengguna. Pengujian ini menggunakan angket *USE Questionnaire* oleh Lund (2001). Penggunaan *Angket USE Questionnaire* berdasarkan kesesuaian terhadap kriteria aspek *usability* yang sudah mencakup sub karakteristik *usability* ISO 25010. Untuk memudahkan pengguna dalam mengisi kuesioner, terlebih dahulu kuesioner diubah ke dalam bahasa Indonesia agar mudah dipahami oleh responden. Kuesioner berjumlah 30 pernyataan yang dibagi menjadi empat kriteria yaitu *usefulness*, *ease of use*, *ease of learning*, dan *satisfaction*. Pilihan jawabannya yaitu Sangat Tidak Setuju (STS), Tidak Setuju

(TS), Ragu-Ragu (RG), Setuju (S), dan Sangat Setuju (SS). Instrumen *USE Questionnaire* ada pada Tabel 5.

Tabel 5. *USE Questionnaire*

No	Pernyataan	Skala Penilaian				
		SS	S	RG	TS	STS
<i>Usefulness</i>						
1	Aplikasi ini membantu saya menjadi lebih efektif					
2	Aplikasi ini membantu saya menjadi lebih produktif					
Dan seterusnya... (Angket lengkap terlampir pada Lampiran 12)						

4. Instrumen *Compatibility*

Aspek yang diuji pada pengujian ini adalah sebagai berikut:

1) *Coexistence*

Pengujian ini akan menganalisis apakah aplikasi ini dapat berjalan berdampingan dengan aplikasi lain tanpa merugikan salah satu aplikasi. Pengujian menggunakan metode observasi dengan cara menjalankan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows pada *device* Android bersamaan dengan aplikasi lain. Dengan demikian dapat diketahui apakah aplikasi dapat berjalan dengan baik atau tidak. Observasi dilakukan menggunakan daftar *checklist* pada Tabel 6.

Tabel 6. Tabel Instrument Sub Karakteristik *Coexistence*

No.	Aplikasi yang dijalankan		Berhasil	Gagal
1.	Aplikasi “clict”	<i>Play store</i>		
2.	Aplikasi “clict”	<i>Google chrome</i>		
3.	Aplikasi “clict”	<i>Whatsapp</i>		
4.	Aplikasi “clict”	<i>Instagram</i>		
5.	Aplikasi “clict”	<i>Facebook Lite</i>		
6.	Aplikasi “clict”	<i>WPS Office</i>		

2) Pengujian pada Berbagai Sistem Operasi dan Tipe Perangkat

Pengujian ini dilakukan dengan *tools* dari Google yaitu Google Firebase Test Lab. *Tools* ini memungkinkan aplikasi dipasang di berbagai sistem operasi Android yang ada di sistem Google Firebase secara *online/virtual*. Pengujian ini dilakukan dengan cara observasi yang hasilnya didapat dengan menggunakan *tools* Google Firebase Test Lab. Selain itu pengujian dilakukan dengan Google Play Store. Pengujian ini dilakukan dengan cara observasi yang hasilnya didapat setelah file apk diunggah. Dari hasil pengujian tersebut dapat diambil kesimpulan apakah aplikasi sudah layak dinyatakan kompatibel dari segi sistem operasi dan tipe perangkat atau belum.

5. Instrumen *Performance Efficiency*

Aspek yang akan diuji pada pengujian ini adalah sebagai berikut:

a. *Time Behaviour*

Pengujian pada sub karakteristik *time behaviour* dilakukan dengan cara observasi waktu yang dibutuhkan sebuah perangkat untuk menjalankan seluruh

fungsi yang terdapat pada aplikasi kamus istilah komputer dan jaringan. Pengujian dilakukan dengan *tools* pengujian dari Appachhi. Dengan menggunakan *tools* ini, aplikasi akan dipasang pada berbagai jenis perangkat Android, setelah selesai pemasangan akan keluar hasil berupa total *thread*/fungsi yang dapat dijalankan perangkat dalam satu detik. Setelah hasil didapatkan dari seluruh perangkat, kemudian akan dilakukan penghitungan rata-rata waktu yang diperlukan oleh perangkat dalam menjalankan fungsi aplikasi.

b. *Resource utilization* pada CPU

Pengujian pada sub karakteristik ini menghitung penggunaan CPU oleh aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dengan cara observasi. Pengujian yang dilakukan dengan menggunakan *tools* pengujian dari Appachhi, akan didapatkan hasil penggunaan CPU per detik yang kemudian akan dihitung rata-rata penggunaan CPU pada berbagai perangkat yang menjadi sampel. Sehingga akan didapatkan hasil rata-rata penggunaan CPU oleh aplikasi.

c. *Resource Utilization* pada *memory*

Pengujian pada sub karakteristik ini menghiung penggunaan *memory* oleh aplikasi. Pengujian dilakukan dengan *tools* pengujian dari Appachhi dengan *tools* ini akan menghasilkan rata-rata penggunaan *memory* yang digunakan oleh aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows.

G. Teknik Analisis Data

Teknik analisis data untuk seluruh aspek pengujian sesuai dengan standar ISO 25010 adalah sebagai berikut :

1. Analisis Pengujian Aspek *Functional Suitability* dan *Compatibility*.

Pengujian aspek *functional suitability* dan *compatibility* menggunakan skala Guttman sebagai skala pengukuran dalam instrumen pengujian. Skala Guttman hanya memiliki dua jawaban berhasil dan gagal. Jawaban berhasil diberi nilai satu dan jawaban gagal diberi nilai nol. Teknik analisis data yang digunakan adalah analisis deskriptif dengan rumus perhitungan sebagai berikut.

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

Setelah mendapatkan hasil skor persentase dengan menggunakan rumus tersebut, kemudian dikonversikan ke dalam pernyataan sesuai dengan Tabel 7.

Tabel 7. Interpretasi Persentase Kelayakan (Sudaryono, 2011)

No	Persentase Pencapaian	Interpretasi
1	0% - 20%	Sangat Tidak Layak
2	21% - 40%	Kurang Layak
3	41% - 60%	Cukup Layak
4	61% - 80%	Layak
5	81% - 100%	Sangat Layak

2. Analisis Pengujian Aspek *Usability*

Analisis yang digunakan dalam pengujian aspek *usability* menggunakan *Likert*. Skala *Likert* yang terdapat dalam instrumen *USE Questionnaire* dapat

menggunakan skala lima maupun skala tujuh dalam penilaiannya. Penelitian ini menggunakan skala lima karena untuk mempermudah dalam mengisi instrumen.

Pada analisis kuantitatif setiap item instrumen yang menggunakan skala *Likert* mempunyai gradasi dari positif sampai negatif (Sugiyono, 2014) sebagai berikut :

- a. Sangat setuju (SS) diberi skor 5
- b. Setuju (S) diberi skor 4
- c. Ragu-ragu (RG) diberi skor 3
- d. Tidak setuju (TS) diberi skor 2
- e. Sangat tidak setuju (STS) diberi skor 1

Dari hasil pengujian *usability*, dianalisis dengan menghitung jawaban berdasarkan skor setiap jawaban dari responden. Berikut rumus menghitung skor pengujian *usability*:

$$\text{Skor total} = (J_{ss} \times 5) + (J_s \times 4) + (J_{rg} \times 3) + (J_{ts} \times 2) + (J_{sts} \times 1)$$

Keterangan :

J_{ss} = Jumlah responden menjawab Sangat Setuju

J_s = Jumlah responden menjawab Setuju

J_{rr} = Jumlah responden menjawab Ragu-ragu

J_{ts} = Jumlah responden menjawab Tidak Setuju

J_{sts} = Jumlah responden menjawab Sangat Tidak Setuju

Kemudian mencari persentase skor untuk mendapatkan kriteria interpretasi skor hasil pengujian *usability* dengan rumus :

$$\text{Persentase skor}(\%) = \frac{\text{Skor total}}{i \times r \times 5} \times 100\%$$

Keterangan :

Skor total = skor total hasil jawaban responden

i = jumlah pertanyaan

r = jumlah responden

Setelah mendapatkan hasil berupa nilai kuantitatif dari perhitungan sebelumnya, kemudian nilai dikonversi menjadi nilai kualitatif berskala lima dengan skala *likert*. Kemudian dicocokkan dengan Tabel interpretasi kelayakan berdasarkan data angket. Tabel interpretasi didapatkan dengan cara mencari kelas interval seperti Tabel 8 lalu dikonversikan dalam persen pada Tabel 9.

Perhitungan bisa didapat dari:

Jumlah pertanyaan = 30

Datum terkecil = Skor *likert* paling kecil x jumlah pertanyaan

= 1 x 30

= 30

Datum terbesar = Skor *likert* paling besar x jumlah pertanyaan

= 5 x 30

= 150

Jangkauan data = Datum terbesar – Datum terkecil

= 150 – 30

= 120

Banyak kelas interval (k)

$$\begin{aligned}k &= 1 + 3,3 \log (n) \\ &= 1 + 3,3 \log (30) \\ &= 1 + 3,3 (1,48) \\ &= 5,884 \\ &= 6 \text{ (dibulatkan)}\end{aligned}$$

Panjang interval = Jangkauan data / banyak kelas interval

$$= 120 / 6$$

$$= 20$$

Tabel 8. Kelas Interval

Kelas Interval	Persentase (%)
30 – 49	20% - 33%
50 – 69	33% - 46%
70 – 89	47% - 59%
90 – 109	60% - 73%
110 – 129	73% - 86%
130 – 150	87% - 100%

Maka interpretasi kelayakannya menjadi:

Tabel 9. Tabel Interpretasi Kelas Interval

Persentase Pencapaian	Interpretasi
20% - 33%	Sangat Tidak Layak

Persentase Pencapaian	Interpretasi
33% - 46%	Tidak Layak
47% - 59%	Kurang Layak
60% - 73%	Cukup Layak
73% - 86%	Layak
87% - 100%	Sangat Layak

3. Analisis Pengujian Aspek *Performance Efficiency*.

Analisis *performance efficiency* yang diuji antara lain pada aspek *time behaviour*, *resource utilization* pada CPU, dan *resource utilization* pada *memory*. Analisis ini juga menggunakan *tools* dari Appachi yang menjalankan seluruh fungsi yang ada di aplikasi Android *mobile* kamus perintah dasar CLI Linux dan Windows. Menurut Yulianto (2016), aplikasi memenuhi kriteria pengujian *performance efficiency* jika saat eksekusi pengujian tidak terjadi *error*, pada *memory* tidak terjadi hang atau *memory leak* pada saat aplikasi dieksekusi, dan penggunaan CPU tidak melebihi batas yang menyebabkan aplikasi *force close* atau *launch fail*.

BAB IV

HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Penelitian ini bertujuan untuk mengembangkan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dan menjamin kualitas aplikasi yang dikembangkan dengan diuji berdasarkan standar ISO 25010 yang meliputi aspek *functional suitability*, *compatibility*, *usability*, dan *performance efficiency*. Berikut hasil pengembangan aplikasi dan hasil pengujian yang telah dilakukan:

1. Pengembangan Aplikasi

Aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dikembangkan dengan menggunakan metode pengembangan *Rational Unified Process* (RUP) dengan empat tahap yaitu *inception*, *elaboration*, *construction*, dan *transition*. Berikut langkah-langkah yang dilakukan pada proses pengembangan aplikasi:

a. Tahap *Inception*

Kegiatan pertama dalam penelitian ini adalah tahap *inception* yaitu tahapan analisis aplikasi yang dikembangkan. Tahapan ini meliputi beberapa kegiatan utama diantaranya mendefinisikan kebutuhan aplikasi yang akan dibuat (*requirements*), mendeskripsikan ruang lingkup proyek (*project management*), dan *environment*.

1) *Business Modelling*

a) Observasi dan Wawancara

Pada tahap ini untuk menganalisis kebutuhan *user* dalam pengembangan aplikasi kamus *command line* Linux dan Windows menggunakan teknik wawancara dan teknik observasi. Teknik wawancara dilakukan dengan Bapak Rohmad Dwiyanto, S.Pd selaku guru pengampu mata pelajaran administrasi server, Ibu Dariyati, S.Kom selaku ketua jurusan Teknik Komputer dan Jaringan dan guru pengampu mata pelajaran administrasi server, serta siswa kelas XI TKJ. Berikut hasil wawancara yang telah dilakukan peneliti :

(1) Hasil Wawancara dengan Ibu Dariyati, S.Kom (Ketua Jurusan TKJ)

- Siswa jurusan TKJ minimal harus menguasai kompetensi instalasi sistem operasi, sistem operasi jaringan, jaringan dasar, administrasi jaringan, dan rancang bangun jaringan.
- Semua kompetensi minimal yang harus dikuasai siswa tersebut, sama-sama berhubungan dengan perintah dasar Linux yaitu dengan menggunakan Debian.
- Peralatan praktik untuk beberapa kompetensi belum memadai, siswa kurang memahami apa yang disampaikan guru merupakan beberapa kendala yang menyebabkan proses pembelajaran tidak berjalan maksimal.
- Tidak semua guru memberikan sumber belajar siswa dalam bentuk buku atau *ebook*. Sehingga guru mengharuskan siswa mencatat materi yang

diberikan guru melalui slide presentasi dan mencari referensi sumber belajar lain dari internet.

(2) Hasil Wawancara dengan Bapak Rohmad Dwiyanto, S.Pd (Guru Pengampu Administrasi Server)

- Guru telah memberikan *ebook* dan materi yang terkait dengan mata pelajaran administrasi server.
- Siswa kurang memahami dan terkadang lupa dengan perintah-perintah Linux untuk pengoperasian server berbasis *text*.
- Perintah-perintah Linux yang diberikan hanya sebatas yang ada di *e-book* untuk perintah-perintah Linux yang lain siswa melakukan pengembangan sendiri dengan mencari referensi lain melalui internet.
- Guru tidak memberikan materi khusus tentang perintah-perintah CLI.

(3) Hasil wawancara dengan siswa kelas XI TKJ 2

- Siswa mengalami kesulitan belajar pada kompetensi administrasi jaringan, rancang bangun jaringan, dan komunikasi data.
- Siswa sulit mengingat perintah-perintah Linux beserta fungsinya untuk membangun jaringan dan mengelola servernya karena menurut mereka hafalannya terlalu banyak.
- Sudah diberikan bahan belajar untuk siswa berupa *e-book* tentang administrasi server namun siswa menilai *ebook* tersebut kurang lengkap.
- Siswa mencari referensi sumber belajar lain dari internet, akibatnya terkadang hasil yang diperoleh siswa berbeda-beda pemahamannya

karena sebagian besar sumber yang diambil dari blog-blog pribadi yang belum diketahui kebenarannya.

Teknik observasi dilakukan ketika peneliti melaksanakan PLT (Praktek Lapangan Terbimbing) di SMK N 1 Bantul. Hasil observasi yang dilakukan oleh peneliti adalah sebagai berikut :

- SMK N 1 Bantul telah menggunakan kurikulum 2013 dalam proses pembelajarannya siswa dituntut untuk lebih aktif dalam mencari sumber bacaan dan materi yang akan dipelajari, namun banyaknya sumber bacaan yang ada di internet menyebabkan perbedaan persepsi dari siswa sehingga dibutuhkan sumber bacaan atau materi yang sama untuk semua siswa.
- Dibutuhkan sebuah media yang dapat memudahkan siswa dalam mengetahui arti perintah *command line* dan dapat digunakan setiap waktu secara mandiri.
- *Ebook* Jaringan komputer 2013 yang diterbitkan oleh Kementerian Pendidikan dan Kebudayaan yakni Administrasi Server menjadi buku pegangan siswa dalam proses pembelajaran jaringan komputer.
- Tidak semua siswa mengetahui perintah *command line* beserta fungsinya.
- Dibutuhkan media pembelajaran yang dapat mendukung dan memaksimalkan kegiatan belajar mengajar didalam maupun diluar kelas.
- Sebagian besar siswa memiliki dan menggunakan *smartphone* serta tidak ada larangan untuk membawanya ke sekolah.

- Penggunaan *smartphone* di kelas diizinkan oleh guru ketika memang harus mencari sumber-sumber referensi belajar yang berkaitan dengan materi yang sedang dibahas.
- Koneksi internet yang disediakan oleh sekolah terbatas hanya di lab-lab tertentu. Sehingga bagi siswa yang sedang belajar pada lab yang tidak terdapat koneksi internet melalui wifi harus menggunakan koneksi masing-masing.

b) Deskripsi Aplikasi

Aplikasi ini berfungsi untuk membantu siswa dalam proses pencarian arti atau fungsi perintah CLI Linux dan Windows. Siswa dapat mencari arti atau fungsi, sintaks, parameter dan contoh penggunaan perintah CLI yang diinginkan. Selain itu siswa juga dapat mengetahui apa yang dimaksud dengan perintah CLI. Aplikasi ini juga dilengkapi dengan emulator OS, sehingga siswa dapat mengimplementasikan perintah yang dicari secara langsung tanpa harus membuka laptop atau komputer. Dengan aplikasi ini, siswa juga dapat mengukur sejauh mana pemahamannya tentang perintah CLI tersebut melalui fungsi kuis yang disediakan.

2) Requirements

Aktivitas yang dilakukan pada *requirements* yaitu melakukan analisis kebutuhan fungsional. Analisis kebutuhan fungsional didasarkan pada hasil observasi dan wawancara. Kebutuhan fungsi pada pengembangan aplikasi

Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows adalah sebagai berikut :

- a) Siswa dapat mengetahui penjelasan tentang *command line*.
- b) Siswa dapat melihat daftar perintah *command line* Linux dan Windows.
- c) Siswa dapat melihat hasil pencarian.
- d) Siswa dapat mengetahui penjelasan perintah *command line* yang dicari.
- e) Siswa dapat mengimplementasikan perintah yang dicari pada emulator OS Linux maupun Windows.
- f) Siswa dapat saling berbagi perintah yang dicari.
- g) Siswa dapat melihat halaman bantuan untuk membantu siswa dalam menggunakan aplikasi tersebut.
- h) Siswa dapat melihat informasi dari aplikasi.
- i) Siswa dapat mengerjakan kuis yang diberikan dalam bentuk pilihan ganda.
- j) Siswa dapat mengetahui perolehan skor dari kuis yang telah dikerjakan.

3) *Analysis & Design*

Belum dilakukan kegiatan *analysis & design* karena pada tahap *inception* fokus pada kegiatan mendefinisikan kebutuhan sistem yang akan dibuat (*requirements*), mendeskripsikan ruang lingkup proyek (*project management*), dan *environment*.

4) ***Implementation***

Belum dilakukan kegiatan *implementation* karena pada tahap *inception* fokus pada kegiatan mendefinisikan kebutuhan sistem yang akan dibuat (*requirements*), mendeskripsikan ruang lingkup proyek (*project management*), dan *environment*.

5) ***Test***

Belum dilakukan kegiatan *test* karena pada tahap *inception* fokus pada kegiatan mendefinisikan kebutuhan sistem yang akan dibuat (*requirements*), mendeskripsikan ruang lingkup proyek (*project management*), dan *environment*.

6) ***Deployment***

Belum dilakukan kegiatan *deployment* karena pada tahap *inception* fokus pada kegiatan mendefinisikan kebutuhan sistem yang akan dibuat (*requirements*), mendeskripsikan ruang lingkup proyek (*project management*), dan *environment*.

7) ***Configuration & Change Management***

Kegiatan ini tidak dilakukan karena hanya dikerjakan oleh satu orang sehingga tidak memerlukan proses integrasi.

8) ***Project Management***

Membuat penjadwalan (*scheduling*) dalam pembuatan produk. Tujuan dari penjadwalan ini sebagai pedoman bagi pengembang agar penelitian dapat berjalan dengan efektif dan sesuai dengan estimasi waktu yang telah ditetapkan pada Tabel 10.

Tabel 10. Estimasi Penyelesaian Tiap Tahapan (*Stage*)

	Februari	Maret	April	Mei
<i>Inception</i>				
<i>Elaboration</i>				
<i>Implementation</i>				
<i>Transition</i>				

9) Environment

Dalam proses pengembangan aplikasi dibutuhkan spesifikasi perangkat-perangkat pengembangan, berikut adalah daftar spesifikasinya :

a) Kebutuhan Perangkat Keras

(1) 1 unit laptop

- Processor : Intel Core B940 @2.00GHz
- RAM : 2GB
- Hard Disk Drive : 500GB
- *Operating System* : Windows 7 64 bit
- Merk : Acer 4750Z

(2) 1 unit perangkat Android

- Processor : Quad Core 1.25 Ghz
- RAM : 2 BG
- Memori Internal : 16 GB
- *Operating System* : Android Nougat 7.0
- Merk : Advan i5C Plus

b) Kebutuhan Perangkat Lunak

- (1) Android Studio digunakan untuk pengembangan aplikasi Android.
- (2) Android SDK digunakan sebagai kerangka kerja atau *framework* untuk membangun aplikasi Android.
- (3) Corel Draw digunakan untuk mendesain *user interface* dan membuat *mockup*.
- (4) Adobe Photoshop digunakan untuk mendesain *user interface* dan membuat *mockup*.
- (5) Microsoft Visio 2013 digunakan untuk membuat UML.

b. Tahapan *Elaboration*

Kegiatan yang dilakukan pada tahap *elaboration* lebih difokuskan pada perencanaan arsitektur sistem, seperti pembuatan *use case*, perancangan UML, dan desain *user interface*.

1) *Business Modelling*

Iterasi kedua pada aktivitas *business modelling* yaitu melengkapi deskripsi produk dengan menambahkan identifikasi aktor, kata kerja, kata benda, dan masalah. Berikut deskripsi produk setelah dilakukan perbaikan :

Aplikasi ini berfungsi untuk membantu siswa dalam proses pencarian arti atau fungsi perintah CLI Linux dan windows. Siswa dapat mencari arti atau fungsi, sintaks, dan contoh penggunaan perintah CLI yang diinginkan. Selain itu siswa juga dapat mengetahui apa yang dimaksud dengan perintah CLI. Aplikasi ini juga dilengkapi dengan emulator OS, sehingga siswa dapat mengimplementasikan perintah yang dicari secara langsung tanpa harus membuka

laptop atau komputer. Dengan aplikasi ini, siswa juga dapat mengukur sejauh mana pemahamannya tentang perintah CLI tersebut melalui fungsi kuis yang disediakan.

Biru = Aktor

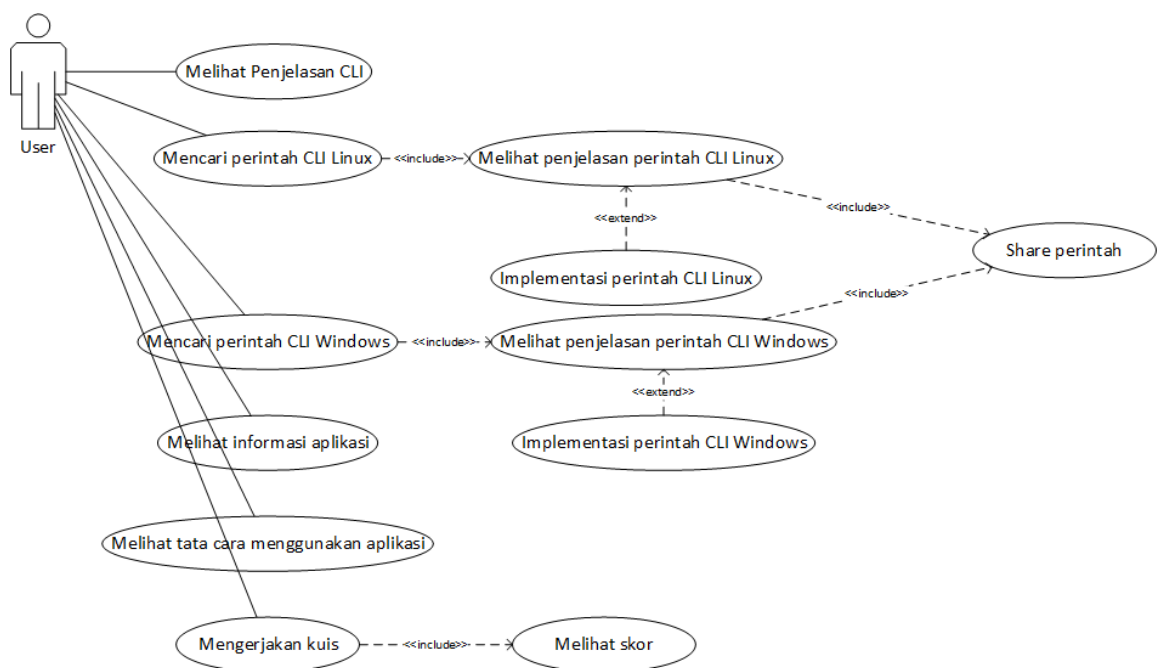
Hijau = Kata Kerja

Orange = Kata Benda

Merah = Permasalahan

2) Requirement

Berdasarkan analisis requirement pada tahap inception maka digambarkan pemodelan UML pada Gambar 14 dengan diagram use case sebagai berikut :



Gambar 14. Diagram Use Case

a) Definisi Aktor

Tabel 11. Definisi Aktor

No.	Aktor	Deskripsi
1.	User	Orang yang menggunakan aplikasi.

b) Definisi *Use Case*

Tabel 12. Definisi *Use Case*

No.	<i>Use Case</i>	Definisi
1.	Melihat penjelasan CLI	<i>Use case</i> ini berfungsi untuk melihat penjelasan tentang apa yang dimaksud dengan perintah <i>command line interface</i> (CLI).
2.	Mencari perintah CLI Linux	<i>Use case</i> ini berfungsi untuk mencari perintah CLI Linux.
3.	Mencari perintah CLI Windows	<i>Use case</i> ini berfungsi untuk mencari perintah CLI Windows.
4.	Melihat informasi aplikasi	<i>Use case</i> ini berfungsi untuk mengetahui informasi tentang versi dan <i>developer</i> dari aplikasi.
5.	Melihat tata cara menggunakan aplikasi	<i>Use case</i> ini berfungsi untuk mengetahui bagaimana cara menggunakan aplikasi tersebut.
6.	Mengerjakan kuis	<i>Use case</i> ini berfungsi untuk

		menyediakan kumpulan soal pilihan ganda bagi <i>user</i> agar dapat dijadikan tolok ukur sejauh mana <i>user</i> memahami perintah CLI.
7.	Melihat penjelasan perintah CLI Linux	<i>Use case</i> ini berfungsi untuk mengetahui penjelasan perintah CLI Linux yang dicari.
8.	Melihat penjelasan perintah CLI Windows	<i>Use case</i> ini berfungsi untuk mengetahui penjelasan perintah CLI Windows yang dicari.
9.	Melihat skor	<i>Use case</i> ini berfungsi untuk menampilkan jumlah skor yang diperoleh setelah <i>user</i> mengerjakan soal dalam kuis.
10.	Implementasi perintah CLI Linux	<i>Use case</i> ini berfungsi untuk mengimplementasikan perintah CLI Linux dalam emulator OS Linux.
11.	Implementasi perintah CLI Windows	<i>Use case</i> ini berfungsi untuk mengimplementasikan perintah CLI Windows dalam emulator OS Linux.

No.	Use Case	Definisi
12.	Share perintah	Use case ini berfungsi untuk saling berbagi informasi tentang perintah CLI Linux dan Windows yang dicari dengan sesama <i>user</i> .

3) *Analysis & Design*

a) *Expanded Use Case*

Expanded use case adalah penjabaran dari proses interaksi antara aktor dan sistem. Dalam prosesnya dibagi empat yaitu penjabaran kondisi sebelum adanya interaksi (*Precondition*), proses interaksi (*main flow*), proses setelah interaksi (*Post Conditions*), dan hal yang terjadi ketika ada *error*/kesalahan (*Exception*). Berikut hasil dari *expanded use case* yang dihasilkan pada daftar Tabel 13.

Tabel 13. *Expanded use case*

No.	Use Case
1.	Melihat penjelasan CLI
2.	Mencari perintah CLI Linux
3.	Mencari perintah CLI Windows
4.	Melihat informasi aplikasi
5.	Melihat tata cara menggunakan aplikasi
6.	Mengerjakan kuis

Berikut *expanded use case* nomor 1 yaitu melihat penjelasan CLI. *Expanded use case* nomor 2 sampai 6 terdapat pada Lampiran 13.

Tabel 14. *Expanded Use Case* Melihat Penjelasan CLI

<i>Use case 1 : Melihat penjelasan CLI</i>	
No.	Deskripsi
1.	<i>User memilih penjelasan command line interface</i>
No.	Deskripsi
2.	Sistem menampilkan penjelasan <i>command line interface</i>

b) *Activity Diagram*

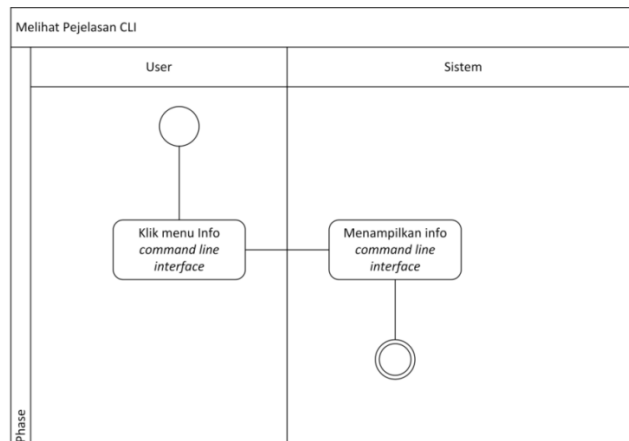
Berdasarkan hasil *expanded use case*, maka dihasilkan *activity diagram*.

Berikut adalah daftar *activity diagram* pada Tabel 15.

Tabel 15. Daftar *Activity Diagram*

No.	<i>Activity Diagram</i>
1.	Melihat penjelasan CLI
2.	Mencari perintah CLI Linux
3.	Mencari perintah CLI Windows
4.	Melihat informasi aplikasi
5.	Melihat tata cara menggunakan aplikasi
6.	Mengerjakan kuis

Berikut *activity diagram* melihat penjelasan CLI. *Activity diagram* nomor 2 sampai 6 terdapat pada Lampiran 14.



Gambar 15. Activity Diagram Melihat Penjelasan CLI

c) *Sequence Diagram*

Berdasarkan hasil *activity diagram*, maka dihasilkan *sequence diagram*.

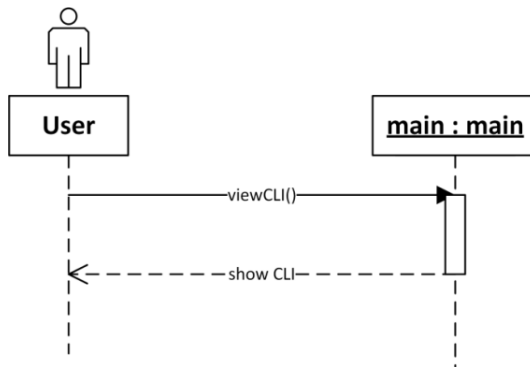
Berikut adalah daftar *sequence diagram* pada Tabel 16.

Tabel 16. Daftar *Sequence Diagram*

No.	<i>Activity Diagram</i>	<i>Sequence Diagram</i>
1.	Melihat penjelasan CLI	Melihat penjelasan CLI
2.	Mencari perintah CLI Linux	Mencari perintah CLI Linux
3.	Mencari perintah CLI Windows	Mencari perintah CLI Windows
4.	Melihat informasi aplikasi	Melihat informasi aplikasi
5.	Melihat tata cara menggunakan aplikasi	Melihat tata cara menggunakan aplikasi
6.	Mengerjakan kuis	Mengerjakan kuis

Berikut *sequence diagram* melihat penjelasan CLI pada Gambar 16. *Sequence*

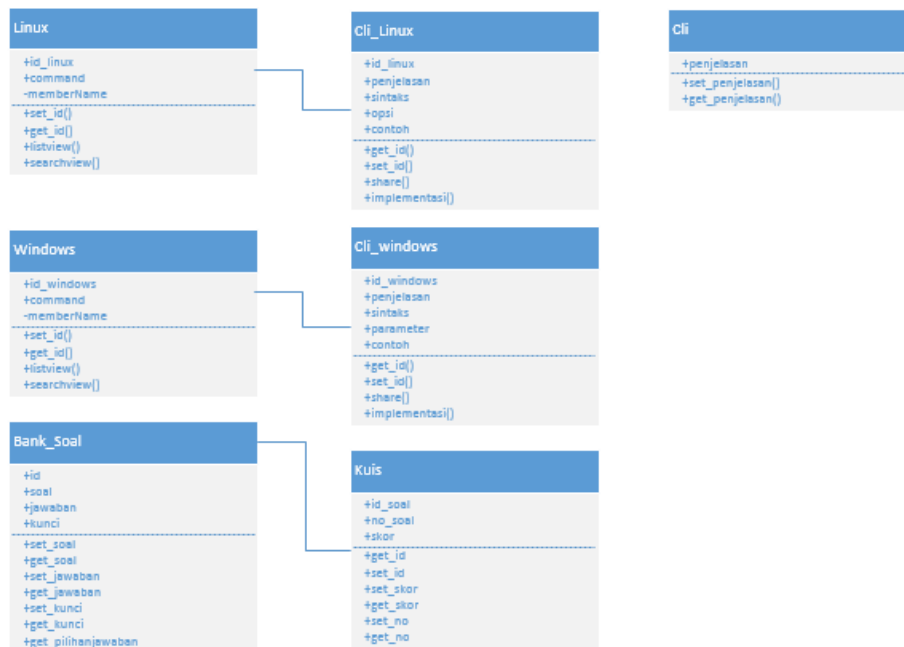
diagram nomor 2 sampai 6 terdapat pada Lampiran 15.



Gambar 16. *Sequence Diagram* Melihat Penjelasan CLI

d) *Class Diagram*

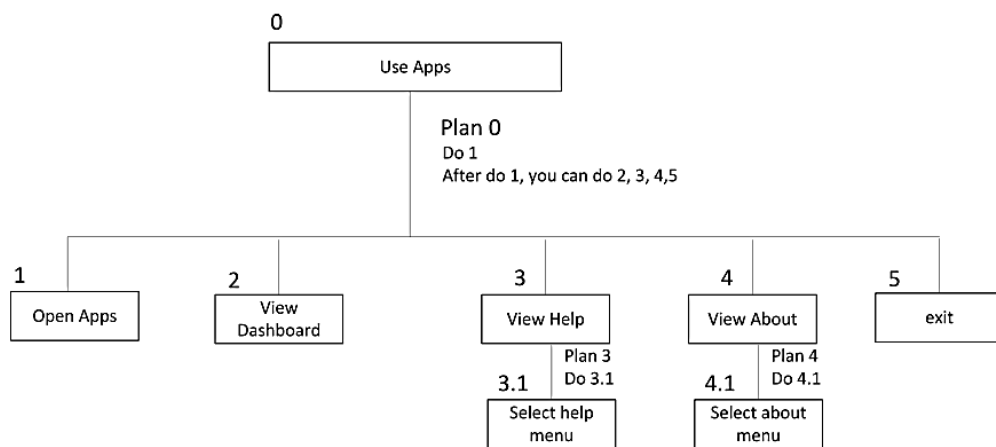
Setelah mengetahui setiap fungsi yang berjalan dengan menggunakan *sequence diagram*, kemudian dibuat *class diagram*. Pada *class diagram* digambarkan *class* apa saja dan bagaimana hubungan antar *class* pada saat tahap konstruksi desain ke kode program. *Class diagram* dapat dilihat pada Gambar 17.



Gambar 17. *Class Diagram* Aplikasi

e) *Hierarki Task Analysis*

Hierarki Task Analysis (HTA) digunakan untuk menjabarkan *task-task* yang akan terjadi pada aplikasi yang dikembangkan sehingga memudahkan dalam menentukan desain yang sesuai dengan aplikasi yang memudahkan pengguna. HTA aplikasi Android *mobile* kamus perintah dasar *command line interface* (CLI) Linux dan Windows terdapat pada Lampiran 16.



Gambar 18. Potongan HTA

f) *Storyboard*

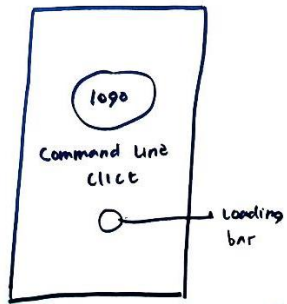
Storyboard digunakan untuk memvisualisasikan ide dari aplikasi yang akan dibangun, sehingga dapat memberikan Gambaran dari aplikasi yang akan dihasilkan. *Storyboard* dapat dikatakan juga *visual script* yang akan dijadikan *outline* dari sebuah proyek. *Storyboard* yang dihasilkan terdapat pada Tabel 17.

4) *Implementation*

Implementasi dalam proses pembuatan aplikasi dalam proses *elaboration* salah satunya dapat digambarkan dengan *mockup*. Aplikasi yang digunakan oleh pengembang untuk membuat *mockup* adalah *Coreldraw X7*. *Mockup* digunakan

untuk menggambarkan desain *user interface* berdasarkan *storyboard* yang telah dibuat mulai dari tata letak, *font*, warna, sampai Gambar sehingga fungsi-fungsi yang telah dibuat dalam proses *analysis & design* dapat digambarkan sebelum proses coding. Hasil desain *mockup* terlampir pada Lampiran 18.

Tabel 17. *Storyboard*

No.	Sketsa	Visual
1.	<p><i>Splashscreen</i></p> 	<p>Ketika aplikasi pertama kali dijalankan, akan tampil halaman <i>splashscreen</i> sebelum masuk ke menu utama. Halaman ini berisi logo aplikasi dan nama aplikasi.</p>
Data <i>storyboard</i> lengkap ada di Lampiran 17		

5) Test

Berikut hasil konfirmasi terkait hasil desain aplikasi *dictionary and emulator* perintah CLI Linux dan Windows bersama Bapak Muhammad Izzuddin M, S.Pd.T, M.Cs.

Tabel 18. Konfirmasi Desain *Mockup*

No.	Mockup	Konfirmasi
		Ya/Tidak
1.	<i>Splash Screen</i>	Ya

No.	Mockup	Konfirmasi
		Ya/Tidak
6.	Menu detail CLI Linux	Ya
7.	Menu <i>share</i>	Ya
8.	Menu implementasi CLI Linux	Ya
9.	Menu daftar CLI Windows	Ya
11.	Menu <i>share</i>	Ya
12.	Menu implementasi CLI Windows	Ya
13.	Menu informasi <i>command line interface</i>	Ya
14.	Menu beranda kuis	Ya
15.	Menu mulai kuis	Ya
16.	Menu soal	Ya
17.	Menu submit	Ya
18.	Menu hasil skor kuis	Ya

6) *Deployment*

Pada tahap ini tidak dilakukan *deployment* karena tahap *elaboration* fokus untuk pembuatan *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram* dan desain *interface*.

7) *Configuration & Change Management*

Pada tahap ini tidak dilakukan karena hanya dikerjakan oleh satu orang sehingga tidak memerlukan pergantian *management*.

8) *Project Management*

Pada tahap ini tidak ada perubahan *project management*, karena masih sama dengan *project management* pada tahap *inception*.

9) *Environment*

Pada tahap ini tidak ada perubahan *environment*, karena masih sama dengan *environment* pada tahap *inception*.

c. Tahap *Construction*

Kegiatan yang dilakukan pada tahap *construction* lebih difokuskan pada implementasi kode program dan pengujian. Tetapi tidak menutup kemungkinan untuk melakukan perbaikan perancangan UML maupun desain *user interface*.

1) *Business Modelling*

Pada tahap ini *business modelling* sudah tidak dilakukan karena informasi serta deskripsi produk yang akan dikembangkan sudah sesuai kebutuhan berdasarkan tahap sebelumnya.

2) *Requirements*

Pada tahap ini kegiatan *requirements* tidak dilakukan karena analisis kebutuhan fungsional aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sudah sesuai dengan kebutuhan pengguna.

3) *Analysis & Design*

Pada tahap ini kegiatan *analysis & design* tidak dilakukan karena *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram*, dan desain *user interface* sudah sesuai dengan kebutuhan.

4) *Implementation*

a) Implementasi Desain *User Interface*

Pada tahap ini, *mockup* desain *user interface* aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows yang telah dibuat pada tahap *elaboration* diimplementasikan pada *layout* Android Studio. Pembuatan *layout* aplikasi menggunakan aturan *Material Design* yang ada pada IDE Android Studio 3.0.1.

Tabel 19. *Layout* Aplikasi

No.	Nama Layout	Deskripsi
1.	Activity_splashscreen.xml	<i>Layout</i> untuk menampilkan halaman <i>splash screen</i> .
2.	Activity_main.xml	<i>Layout</i> pendukung untuk menampilkan <i>fragment home, about</i> , dan petunjuk.
3.	Activity_linux.xml	<i>Layout</i> untuk menampilkan daftar perintah CLI Linux.
4.	Detail_linux.xml	<i>Layout</i> pendukung activity_linux.xml untuk menampilkan <i>listview</i> perintah CLI Linux.
5.	Activity_cli_linux.xml	<i>Layout</i> untuk menampilkan detail penjelasan perintah CLI Linux yang dicari.
6.	Implementasi_linux.xml	<i>Layout</i> untuk menampilkan terminal Linux.
7.	Activity_windows.xml	<i>Layout</i> untuk menampilkan daftar perintah CLI Windows.

No.	Nama Layout	Deskripsi
8.	Detail_windows.xml	<i>Layout</i> pendukung activity_windows.xml untuk menampilkan <i>listview</i> perintah CLI Windows.
9.	Activity_cli_windows.xml	Layout untuk menampilkan detail penjelasan perintah CLI Windows yang dicari.
10.	Activity_implementasi_windows.xml	<i>Layout</i> untuk menampilkan <i>command prompt</i> windows.
12.	Fragment_home.xml	<i>Layout</i> untuk menampilkan halaman beranda yang berisi menu utama aplikasi.
13.	Fragment_about.xml	<i>Layout</i> untuk menampilkan penjelasan tentang aplikasi.
14.	Fragment_petunjuk.xml	<i>Layout</i> untuk menampilkan penjelasan petunjuk penggunaan aplikasi.
15.	Activity_bantu_cli.xml	<i>Layout</i> untuk menampilkan penjelasan tentang CLI.
16.	Activity_quiz.xml	<i>Layout</i> untuk menampilkan halaman awal kuis.
17.	Activity_soal.xml	<i>Layout</i> untuk menampilkan soal pilihan ganda pada kuis.
18.	Activity_skor.xml	<i>Layout</i> untuk menampilkan skor yang diperoleh hasil dari mengerjakan soal kuis.

Berikut potongan kode *layout* yang dapat dilihat pada Gambar 19.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/container"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:orientation="vertical">
9
10     <FrameLayout
11         android:id="@+id/content"
12         android:layout_width="match_parent"
13         android:layout_height="0dp"
14         android:layout_weight="1">
15     </FrameLayout>
16
17     <android.support.design.widget.BottomNavigationView
18         android:id="@+id/navigation"
19         android:layout_width="match_parent"
20         android:layout_height="wrap_content"
21         android:layout_gravity="bottom"
22         android:background="?android:attr/windowBackground"
23         app:menu="@menu/navigation" />
24

```

Gambar 19. Potongan *Script* XML pada *activity_main.xml*

b) Implementasi Pemrograman

Setelah membuat *layout* pada Android Studio, kemudian dikonfigurasi pada setiap *layout* yang dibuat. Bahasa pemrograman yang digunakan adalah Java. Berikut masing-masing file Java dan deskripsi fungsionalitasnya tersaji dalam Tabel 20.

Tabel 20. File Java

No.	Nama File	Deskripsi
1.	Splashscreen.java	Menangani halaman <i>splashscreen</i> aplikasi.
2.	MainActivity.java	Menangani menu utama aplikasi, <i>Bottom Navigation</i> , dan menu opsi.
3.	LinuxActivity.java	Menangani <i>listview</i> dan proses pencarian perintah CLI Linux.

No.	Nama File	Deskripsi
4.	Cli_linux.java	Menangani fragment perintah CLI Linux yang dicari untuk menampilkan penjelasan, parameter, sintaks, dan contoh tentang perintah tersebut serta menangani fungsi <i>share</i> .
5.	Implementasi_linux.java	Menangani terminal Linux.
6.	WindowsActivity.java	Menangani listview dan proses pencarian perintah CLI Windows.
7.	Cli_windows.java	Menangani <i>fragment</i> perintah CLI Windows yang dicari untuk menampilkan penjelasan, opsi, sintaks, dan contoh tentang perintah tersebut serta menangani fungsi <i>share</i> .
8.	Implementasi_windows.java	Menangani <i>command prompt</i> Windows.
9.	AboutFragment.java	Menangani halaman tentang aplikasi.
10.	HomeFragment.java	Menangani halaman beranda aplikasi.
11.	PetunjukFragment.java	Menangani halaman petunjuk aplikasi.
12.	QuizActivity.java	Menangani halaman kuis aplikasi.
13.	KuisActivity.java	Menangani <i>fragment</i> Kuisfragment.java.
14.	KuisFragment.java	Menangani halaman kuis untuk menampilkan soal dan pilihan ganda.

No.	Nama File	Deskripsi
15.	ScoreFragment.java	Menangani hasil pengerjaan soal kuis dan menampilkan skor yang diperoleh.
16.	Bank_Soal.java	Menangani kumpulan soal, jawaban, dan kunci untuk kuis.

Berikut potongan kode file java yang dapat dilihat pada Gambar 20.

```

1 package dian.clict.dianvika.clict;
2
3 import ...
13
14 public class MainActivity extends AppCompatActivity {
15     HomeFragment homeFragment = new HomeFragment();
16
17     private long backPressedTime = 0;
18     // private TextView mTextMessage;
19
20     private BottomNavigationView.OnNavigationItemSelectedListener mOnNavigationItemSelectedListener
21     = (item) -> {
22         switch (item.getItemId()) {
23             case R.id.navigation_home:
24                 HomeFragment homeFragment = new HomeFragment();
25                 FragmentTransaction fragmentTransaction = getSupportFragmentManager().beginTransaction();
26                 fragmentTransaction.replace(R.id.content, homeFragment);
27                 fragmentTransaction.commit();
28                 return true;
29             case R.id.navigation_dashboard:
30                 PetunjukFragment petunjukFragment= new PetunjukFragment();
31                 FragmentTransaction fragmentTransaction = getSupportFragmentManager().beginTransaction();
32                 fragmentTransaction.replace(R.id.content, petunjukFragment);
33                 fragmentTransaction.commit();
34                 return true;
35             case R.id.navigation_notifications:
36                 AboutFragment aboutFragment = new AboutFragment();
37                 FragmentTransaction fragmentTransaction = getSupportFragmentManager().beginTransaction();
38                 fragmentTransaction.replace(R.id.content, aboutFragment);
39
40
41

```

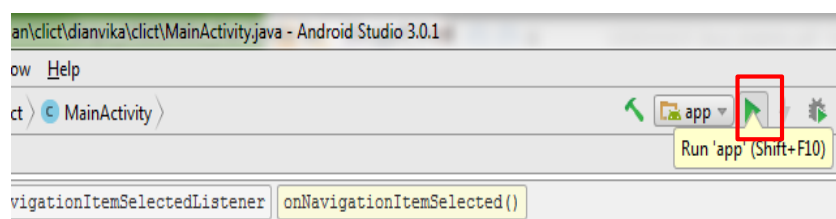
Gambar 20. Potongan File Java pada Class MainActivity.java

5) Test

Testing debugging aplikasi bertujuan untuk menguji secara internal apakah tiap fungsi aplikasi berjalan dengan baik. Terdapat dua cara dalam *debugging* aplikasi. Pertama, menggunakan *device* Android secara langsung. Komunikasi datanya menggunakan kabel data atau dengan wifi. Kedua, menggunakan Emulator Android antara lain *Android Virtual Device* (AVD), Genymotion, dan lain-lain. Pengembang menggunakan cara pertama karena laptop yang digunakan

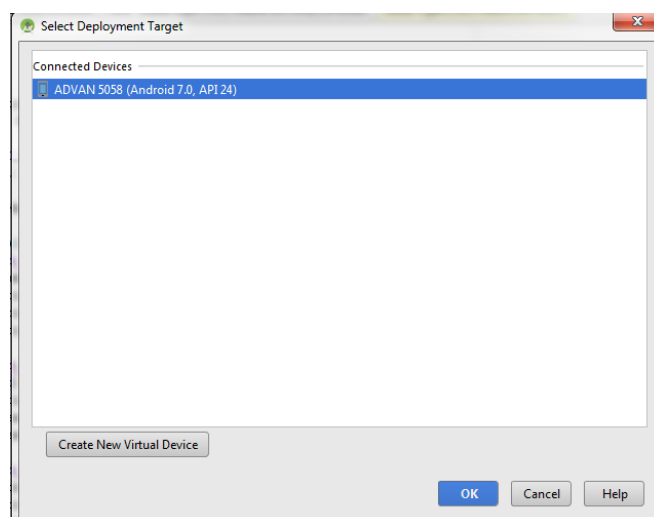
untuk mengembangkan aplikasi ini tidak mendukung *debugging* dengan Emulator Android. Berikut langkah-langkah dalam menjalankan *debugging* dengan *device* Android secara langsung.

- a) Menghubungkan laptop dengan *device* Android yang akan digunakan dengan kabel data.
- b) Membuka menu Run App.

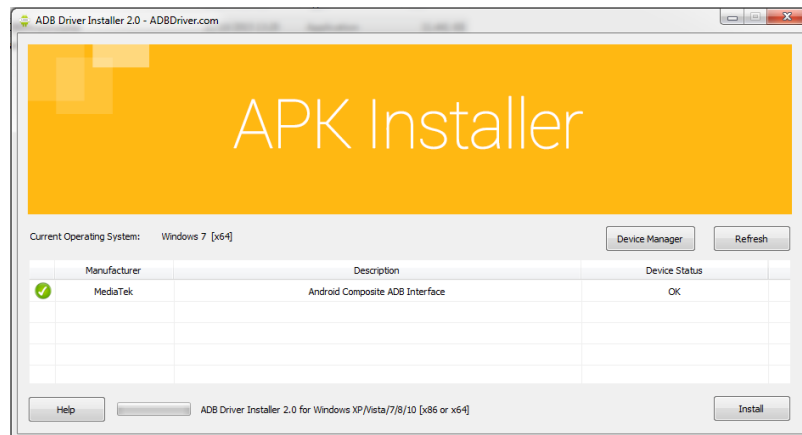


Gambar 21. Icon Run 'app'

- c) Jika *driver* adb sudah terinstal maka ketika di klik Run akan muncul jendela yang menampilkan *device* Android yang terhubung dengan Android Studio. Jika belum terinstal, maka harus diinstal terlebih dahulu agar *debugging* Android Studio dapat mendeteksi *device* Android yang terhubung.

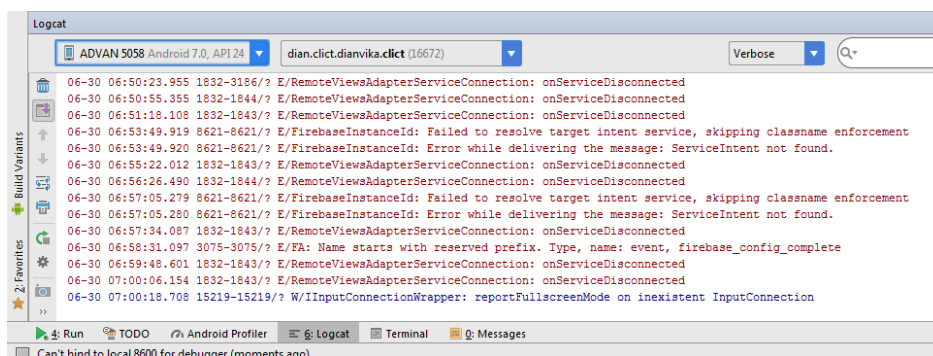


Gambar 22. Pilihan *Device* Android yang terhubung



Gambar 23. Install driver adb

- d) Memilih *device* Android yang terhubung, kemudian klik OK untuk memulai proses *debugging*.
- e) Menunggu proses *gradle build* aplikasi dan menginstal aplikasi ke *device* Android.
- f) Ketika proses *debugging*, dapat dilihat *error*/kesalahan dalam pengkodean di Logcat pada Android Monitor.



Gambar 24. Layar Logcat pada Android Monitor

6) Deployment

Pada tahap ini tidak dilakukan *deployment* karena tahap *construction* fokus pada implementasi aplikasi.

7) *Configuration & Change Management*

Pada tahap ini tidak dilakukan karena hanya dikerjakan oleh satu orang sehingga tidak memerlukan pergantian *management*.

8) *Project Management*

Pada tahap ini tidak ada perubahan *project management*, karena masih sama dengan *project management* pada tahap *inception*.

9) *Environment*

Pada tahap ini tidak ada perubahan *environment*, karena masih sama dengan *environment* pada tahap *inception*.

d. *Transition*

Kegiatan yang dilakukan pada tahap *transition* lebih difokuskan pada *deployment* atau instalasi aplikasi agar dimengerti oleh *user* dan pengujian aplikasi apakah sudah memenuhi harapan *user*.

1) *Business Modelling*

Pada tahap ini *business modelling* sudah tidak dilakukan karena informasi serta deskripsi produk yang akan dikembangkan sudah sesuai kebutuhan berdasarkan tahap sebelumnya.

2) *Requirements*

Pada tahap ini kegiatan *requirements* tidak dilakukan karena analisis kebutuhan fungsional aplikasi *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sudah sesuai dengan kebutuhan pengguna.

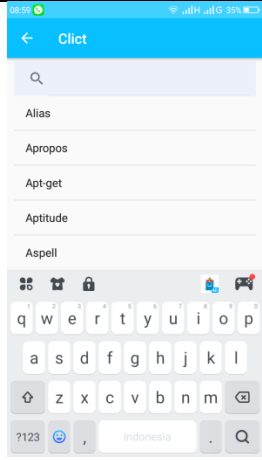
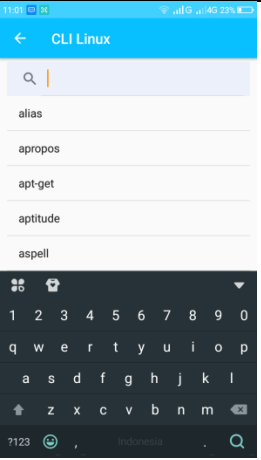
3) *Analysis & Design*

Pada tahap ini kegiatan *analysis & design* tidak dilakukan karena *use case diagram*, *class diagram*, *activity diagram*, *sequence diagram*, dan desain *user interface* sudah sesuai dengan kebutuhan.

4) *Implementation*

Pada tahap ini dilakukan iterasi implementasi setelah proses validasi instrumen terdapat beberapa saran untuk diperbaiki yaitu dari sisi implementasi desain *user interface* dan fungsionalitas program. Berikut implementasi setelah dilakukan perbaikan terdapat pada Tabel 21.

Tabel 21. Hasil Perbaikan

No	Saran	Sebelum Perbaikan	Sesudah Perbaikan
1	Nama-nama perintah CLI pada aplikasi sebaiknya ditulis sesuai dengan perintah aslinya saat diimplementasikan pada terminal yaitu menggunakan huruf kecil semua.		
Data lengkap perbaikan ada di Lampiran 19.			

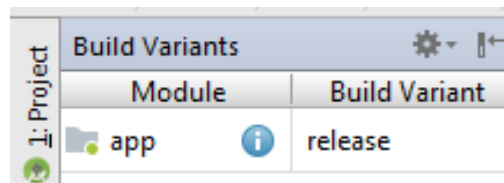
5) *Test*

Pada tahap ini pengujian dilakukan dengan menggunakan standar ISO 25010 yaitu pada aspek *functional suitability*, *usability*, *compatibility*, dan *performance efficiency* yang akan dijelaskan pada sub bab bagian pengujian aplikasi.

6) *Deployment*

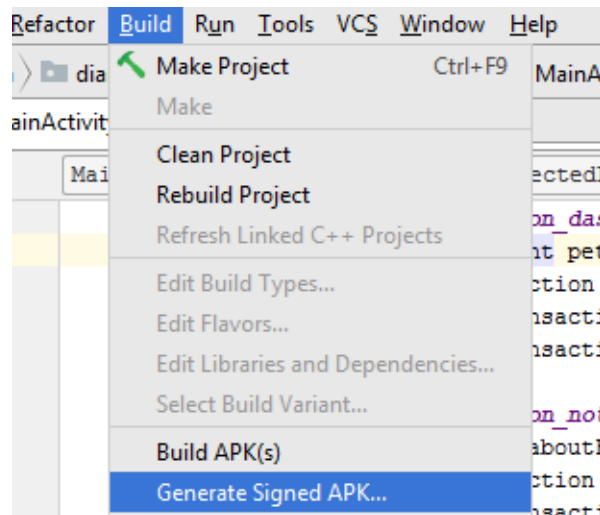
Pada tahap ini dilakukan distribusi menggunakan *play store*. Dalam proses *deployment* di fase *transition* terdapat beberapa tahap.

a) Tahapan *Build Variant*

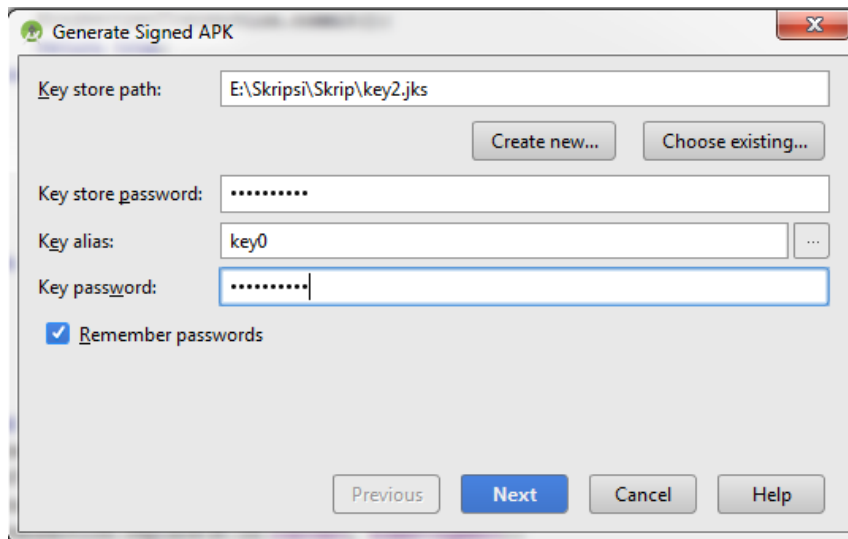


Gambar 25. *Build Variant release*

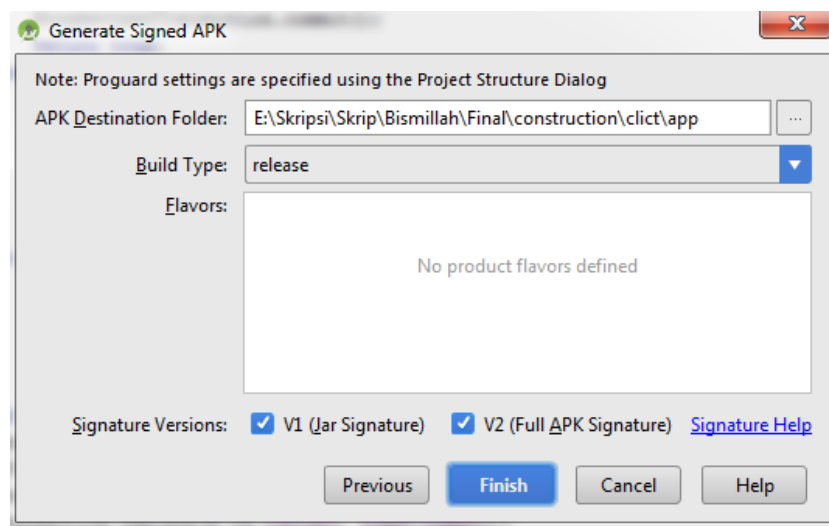
b) Tahapan *Generate Signed APK*



Gambar 26. *Build APK*



Gambar 27. Proses Pengisian *Generate Signed APK*

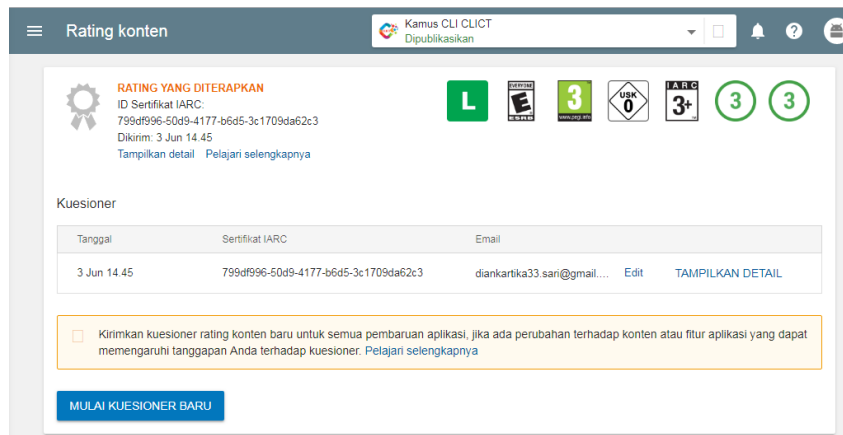


Gambar 28. Pengaturan Penyimpanan

Name	Date modified	Type	Size
build	28/06/2018 16:15	File folder	
libs	01/03/2018 0:31	File folder	
src	10/01/2010 0:41	File folder	
.gitignore	01/03/2018 0:31	GITIGNORE File	1 KB
app.iml	30/06/2018 0:45	IML File	12 KB
build.gradle	27/06/2018 2:41	GRADLE File	2 KB
clict.apk	10/01/2010 0:30	APK File	4.366 KB
proguard-rules.pro	01/03/2018 0:31	PRO File	1 KB

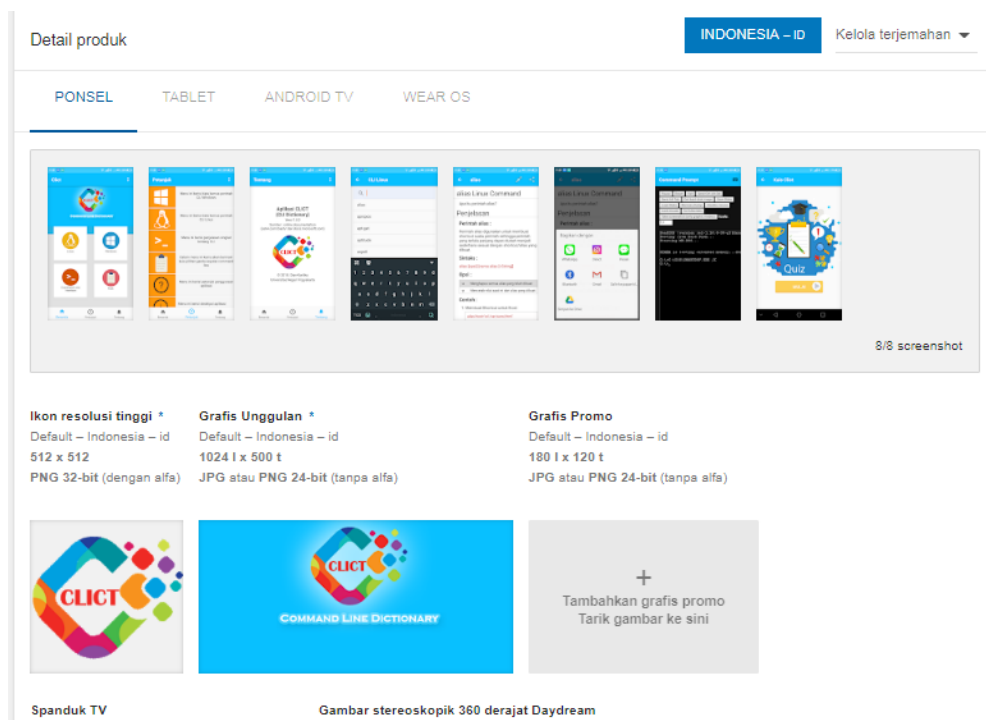
Gambar 29. Hasil *Build Release*

c) *Content Rating*



Gambar 30. Hasil *Rating* Aplikasi

d) *Pengisian Product Details*



Gambar 31. Detail Produk Aplikasi

7) *Configuration & Change Management*

Pada tahap ini tidak dilakukan karena hanya dikerjakan oleh satu orang sehingga tidak memerlukan pergantian *management*.

8) *Project Management*

Pada tahap ini tidak ada perubahan *project management*, karena masih sama dengan *project management* pada tahap *inception*.

9) *Environment*

Pada tahap ini tidak ada perubahan *environment*, karena masih sama dengan *environment* pada tahap *inception*.

2. Pengujian Aplikasi

Aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows diuji berdasarkan standar ISO 25010 yang meliputi aspek *functional suitability*, *compatibility*, *usability*, dan *performance efficiency*. Berikut hasil pengujian aplikasi yang telah dilakukan.

1) Hasil Uji Materi

Pada pengujian materi, materi yang disampaikan pada aplikasi dinyatakan valid dan aplikasi layak digunakan untuk penelitian oleh ahli materi dalam jaringan dan administrasi server yaitu Ibu Dariyati, S.Kom. Hasil uji coba terlampir pada Lampiran 20.

2) *Functional Suitability*

Pengujian *functional suitability* dilakukan oleh empat orang ahli dalam bidang *software development*. Daftar ahli yang melakukan *functional suitability* dapat dilihat pada Tabel 22.

Tabel 22. Daftar Ahli Pengujian *Functional Suitability*

No.	Nama	Profesi	Instansi
1.	Catur Adi Nugroho	<i>Software Analysis</i>	PT Qiscus Tekno Indonesia
2.	Rahardyan Bisma	<i>IOS Developer</i>	PT Qiscus Tekno Indonesia
3.	Fauzi Sholicin	<i>IOS Developer</i>	PT Qiscus Tekno Indonesia
4.	Heru Setiawan	<i>Backend Developer</i>	PT Qiscus Tekno Indonesia

Pengujian dilakukan untuk memvalidasi fungsi aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows berjalan dengan benar atau tidak. Berikut hasil pengujian *functional suitability* yang dilakukan oleh empat orang ahli.

Tabel 23. Hasil Uji *Functional Suitability*

No	Fitur	Skor Penguji					Skor Maksimal
		1	2	3	4	Jumlah	
1	Beranda	1	1	1	1	4	4
2	Petunjuk	1	1	1	1	4	4
3	Tentang aplikasi	1	1	1	1	4	4
4	CLI Linux	1	1	1	1	4	4
5	CLI Windows	1	1	1	1	4	4
6	Pencarian CLI Linux	1	1	1	1	4	4
7	Menampilkan papan keyboard	1	1	1	1	4	4

No	Fitur	Skor Penguji					Skor Maksimal
		1	2	3	4	Jumlah	
9	Rincian perintah CLI Linux	1	1	1	1	4	4
10	Berbagi perintah CLI Linux	1	1	1	1	4	4
11	Implementasi terminal Linux	1	1	1	1	4	4
12	Menampilkan papan keyboard pada terminal Linux	1	1	1	1	4	4
13	Pencarian CLI Windows	1	1	1	1	4	4
14	Rincian perintah CLI Windows	1	1	1	1	4	4
15	Berbagi perintah CLI Windows	1	1	1	1	4	4
16	Implementasi command prompt (cmd) Windows	1	1	1	1	4	4
17	Menampilkan papan keyboard pada cmd	1	1	1	1	4	4
18	Kuis	1	1	1	1	4	4
19	Skor	1	1	1	1	4	4
20	Konfirmasi pengerjaan kuis	1	1	1	1	4	4
21	Keluar	1	1	1	1	4	4
Total		21	21	21	21	84	84

Berdasarkan data diatas dapat diukur kelayakannya sebagai berikut :

$$\begin{aligned}
 \text{Persentase Kelayakan (\%)} &= \frac{\text{Skor yang diobservasi}}{\text{Skor yang diharapkan}} \times 100\% \\
 &= \frac{84}{84} \times 100\% \\
 &= 100\%
 \end{aligned}$$

Hasil dari pengujian *functional suitability* yaitu persentase kelayakan sebesar 100% yang berarti bahwa seluruh fitur yang ada pada aplikasi dapat berjalan dengan baik. Berdasarkan hasil perhitungan persentase kelayakan tersebut dapat disimpulkan bahwa kualitas aplikasi dari sisi *functional suitability* memiliki nilai “**Sangat Layak**”. Walaupun 100% fitur dapat berjalan dengan baik, terdapat saran dari ahli pengembang perangkat lunak agar aplikasi yang

dikembangkan dapat lebih sempurna. Berikut saran dari ahli pengembangan perangkat lunak Mas Fauzi (Qiscus), ketika klik *back* pada *list* CLI Linux dan Windows langsung keluar. Sebaiknya ketika di klik *back* maka akan kembali ke halaman sebelumnya bukan langsung keluar aplikasi.

3) *Usability*

Pengujian *usability* dilakukan pada oleh 30 siswa kelas XI TKJ 2 SMK N 1 Bantul. Siswa mencoba aplikasi dan mengisi kuesioner yang telah disediakan. Hasil pengujian *usability* dapat dilihat pada Lampiran 21.

Berdasarkan kuesioner hasil pengujian *usability* dapat diketahui bahwa total jawaban sangat setuju (SS) berjumlah 425, setuju (S) berjumlah 418, ragu-ragu (RG) berjumlah 54, tidak setuju (TS) berjumlah tiga, dan sangat tidak setuju (STS) berjumlah nol. Hasil pengujian *usability* dihitung menggunakan rumus sebagai berikut :

$$\begin{aligned}\text{Skor total} &= (J_{ss} \times 5) + (J_s \times 4) + (J_{rg} \times 3) + (J_{ts} \times 2) + (J_{sts} \times 1) \\ &= (425 \times 5) + (418 \times 4) + (54 \times 3) + (3 \times 2) + (0 \times 1) \\ &= 3965\end{aligned}$$

$$\begin{aligned}\text{Persentase skor}(\%) &= \frac{\text{Skor total}}{\text{ixrx5}} \times 100\% \\ &= \frac{3965}{30 \times 30 \times 5} \times 100\% \\ &= 88,1\%\end{aligned}$$

Hasil perhitungan persentase kelayakan adalah 88,1%, sehingga dapat disimpulkan bahwa aplikasi memiliki standar *usability* yang “**Sangat Layak**”.

4) *Compatibility*

Hasil uji *compatibility* terdiri dari aspek *coexistence*, pengujian pada sistem operasi, dan pengujian pada berbagai tipe perangkat. Berikut hasil uji *compatibility* dari ketiga aspek tersebut.

a) Hasil Uji *Coexistence*

Pengujian *coexistence* dilakukan dengan cara observasi. Berikut hasil uji *coexistence* pada Tabel 24.

Tabel 24. Hasil Uji *Coexistence*

No.	Aplikasi yang dijalankan		Berhasil	Gagal
1.	Aplikasi “clicl”	<i>Play store</i>	1	0
2.	Aplikasi “clicl”	<i>Google chrome</i>	1	0
3.	Aplikasi “clicl”	<i>Whatsapp</i>	1	0
4.	Aplikasi “clicl”	<i>Instagram</i>	1	0
5.	Aplikasi “clicl”	<i>Facebook Lite</i>	1	0
6.	Aplikasi “clicl”	<i>WPS Office</i>	1	0
Total			6	0

Berdasarkan hasil uji tersebut, maka dapat diperoleh kelayakannya sebagai

berikut :

$$\begin{aligned} \text{Persentase Kelayakan (\%)} &= \frac{\text{Skor yang diobservasi}}{\text{Skor yang diharapkan}} \times 100\% \\ &= \frac{6}{6} \times 100\% \\ &= 100\% \end{aligned}$$

Hasil perhitungan persentase kelayakan yaitu sebesar 100% yang berarti bahwa aplikasi dapat berjalan bersamaan dengan aplikasi yang lain dengan baik. Berdasarkan hasil perhitungan persentase kelayakan tersebut dapat disimpulkan bahwa kualitas aplikasi dari sisi *compatibility* memiliki nilai “**Sangat Layak**”.

b) Hasil Pengujian pada Berbagai Sistem Operasi dan Tipe Perangkat

Pengujian aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows pada berbagai sistem operasi dan berbagai tipe perangkat menggunakan *tools* Firebase Test Lab. Aplikasi ini akan diuji dengan 15 perangkat Android yang disediakan oleh Firebase Test Lab. Perangkat yang digunakan dalam pengujian menggunakan sistem operasi Android 4.4 Kitkat sampai dengan Android 8.0 Oreo. Berikut hasil pengujian dan daftar perangkat yang digunakan dalam pengujian *compatibility* menggunakan Firebase Test Lab pada Tabel 25.

Tabel 25. Hasil dan Perangkat Pengujian *Compatibility*

No	Nama Perangkat	Sistem Operasi	Hasil
1	Moto G Play (4th Gen) XT1607, Tingkat API 23	Android 6.0 Marshmallow	Berhasil
2	LG G3, Tingkat API 19	Android 4.4 KitKat	Berhasil
3	Nexus 6, Virtual, Tingkat API 24	Android 7.0 Nougat	Berhasil
4	Nexus 5, Virtual, Tingkat API 22	Android 5.1 Lollipop	Berhasil
5	Nexus 7 (2012), Virtual, Tingkat API 19	Android 4.4 KitKat	Berhasil

No	Nama Perangkat	Sistem Operasi	Hasil
6	Huawei Mate 9, Tingkat API 24	Android 7.0 Nougat	Berhasil
7	Nexus 7 (2013), Tingkat API 21	Android 5.1 Lollipop	Berhasil
8	Nexus 4, Virtual, Tingkat API 22	Android 5.1 Lollipop	Berhasil
9	Nexus 5X, Virtual, Tingkat API 26	Android 8.0 Oreo	Berhasil
10	Nexus 7 (2012), Virtual, Tingkat API 21	Android 5.0 Lollipop	Berhasil
11	Galaxy J1 ace SM-J111M, API 22	Android 5.1 Lollipop	Berhasil
12	Low-resolution MDPI phone, Virtual, Tingkat API 23	Android 6.0 Marshmallow	Berhasil
13	Nexus 10, Virtual, Tingkat API 21	Android 5.0 Lollipop	Berhasil
14	Nexus 9, Virtual, Tingkat API 24	Android 7.0 Nougat	Berhasil
15	Nexus 9, Virtual, Tingkat API 25	Android 7.1 Nougat	Berhasil

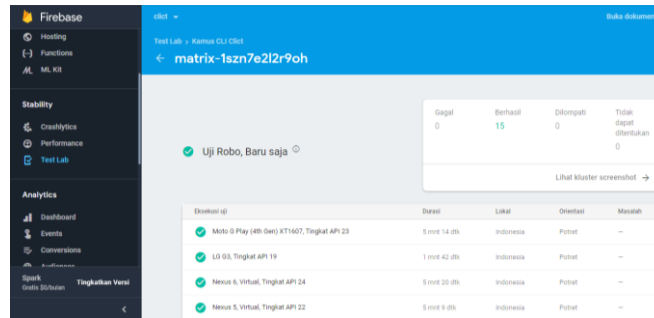
Berdasarkan hasil pengujian, aplikasi *Android mobile dictionary and emulator* perintah dasar CLI Linux dan Windows berhasil dijalankan pada 15 perangkat Android dengan berbagai sistem operasi. Hasil pengujian dengan Firebase Test Lab dapat dilihat pada Gambar 32.

Dari hasil uji tersebut, dapat diperoleh persentase kelayakan sebagai berikut :

$$\begin{aligned}
 \text{Persentase Kelayakan (\%)} &= \frac{\text{Skor yang didapat}}{\text{Skor Maksimal}} \times 100\% \\
 &= \frac{15}{15} \times 100\% \\
 &= 100\%
 \end{aligned}$$

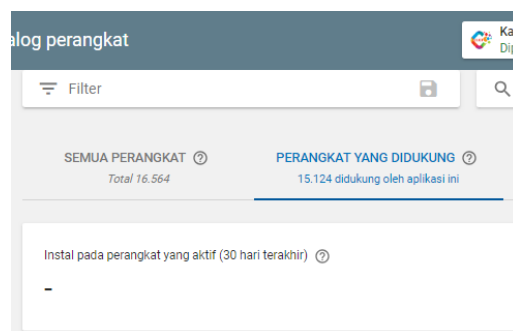
Hasil perhitungan persentase kelayakan dengan pengujian menggunakan Firebase Test Lab adalah 100%, sehingga dapat disimpulkan bahwa kualitas

aplikasi memenuhi standar *compatibility* dari sisi *devices* dan sistem operasi dengan kategori “**Sangat Layak**”.



Gambar 32. Hasil Pengujian *Compatibility* dengan Firebase Test Lab

Selain menggunakan data dari Firebase Test Lab, Peneliti juga menggunakan data dari Google Play Store. Data dari Google Play Store menyatakan bahwa aplikasi kompatibel dengan **15.124 perangkat** dari total 16.564 perangkat yang tersedia. Berikut hasil uji aplikasi dengan menggunakan Google Play Store pada Gambar 33.



Gambar 33. Hasil Pengujian *Compatibility* dengan Google Play Store

Dari hasil uji tersebut, dapat diperoleh persentase kelayakan sebagai berikut:

$$\text{Persentase Kelayakan (\%)} = \frac{\text{Skor yang didapat}}{\text{Skor Maksimal}} \times 100\%$$

$$= \frac{15.124}{16.564} \times 100\%$$

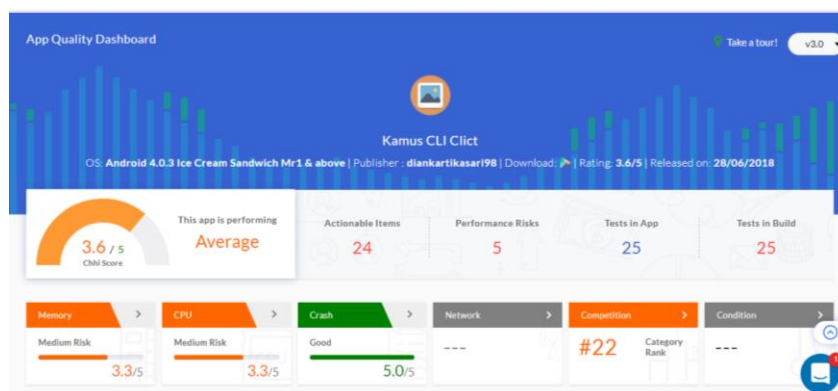
$$= 91,3\%$$

Hasil perhitungan persentase kelayakan dengan pengujian menggunakan Google Play Store adalah 91,3%, sehingga dapat disimpulkan bahwa kualitas aplikasi memenuhi standar *compatibility* dari sisi *devices* dan sistem operasi dengan kategori “**Sangat Layak**”.

Kesimpulan dari hasil pengujian *compatibility* menggunakan *tools* Firebase Test Lab dan Google Play Store, aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memenuhi standar *compatibility* dari sisi *devices* dan sistem operasi dengan kategori “**Sangat Layak**”.

4) *Performance Efficiency*

Pengujian *performance efficiency* menggunakan *tools* pengujian dari Appachi. Pengujian ini menggunakan enam perangkat virtual yang disediakan oleh Appachi. Hasil pengujian dengan Appachi dapat dilihat pada Gambar 34.



Gambar 34. Hasil Pengujian *Performance Efficiency*

Pada hasil pengujian dikelompokkan menjadi tiga, dari segi penggunaan CPU, memori dan eksekusi *thread*. Berikut ringkasan hasil pengujian *performance efficiency* yang terdapat pada Tabel 26.

Tabel 26. Ringkasan Uji *Performance Efficiency*

No.	Perangkat	CPU(%)	Memori(MB)	Thread(/s)
1.	Micromax Canvas AQ4502	49	19.45	29
2.	InFocus	70	18.55	20
3.	LYF-WATER	45	27.49	27
4.	Moto E (2nd Gen) With 4G LTE	55	21.23	21
5.	Samsung Galaxy J7	29	27.06	29
Rata-rata		49,6	22,8	25,2
Minimal		29	18,55	20
Maksimal		70	27,49	29

Tabel ringkasan uji *performance efficiency* di atas menunjukkan performa aplikasi dengan menggunakan berbagai perangkat dengan kecepatan CPU, kapasitas memori dan kemampuan eksekusi *thread* per detik yang berbeda. Penjelasan detail dari hasil uji *performance efficiency* adalah sebagai berikut :

a) Hasil Pengujian *Time Behaviour*

Time behaviour dihitung dengan membagi satu detik dengan rata-rata *thread* per detik. Dari hasil pengujian *performance efficiency* pada Tabel 30 di atas *time behaviour* yang dihasilkan adalah sebagai berikut.

$$\begin{aligned}
\textit{Time Behaviour} &= \frac{1}{\textit{Thread per Second}} \\
&= \frac{1}{25,2} \\
&= 0,039 \textit{ second/thread}
\end{aligned}$$

Berdasarkan hasil perhitungan dapat disimpulkan bahwa waktu eksekusi sebuah *thread* pada aplikasi *Android mobile dictionary and emulator* perintah dasar CLI Linux dan Windows adalah 0,039 detik/*thread*.

b) Hasil Pengujian *Resource Utilization* pada CPU

CPU Utilization adalah aktivitas perangkat dalam menggunakan sumber daya CPU ketika menjalankan aplikasi dalam rentang waktu tertentu. Berdasarkan hasil pengujian dari *tools* Appachhi aplikasi *Android mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memiliki rata-rata penggunaan CPU sebesar 49,6%.

c) Hasil Pengujian *Resource Utilization* pada Memori

Memory Utilization adalah aktivitas perangkat dalam menggunakan sumber daya memori ketika menjalankan aplikasi dalam rentang waktu tertentu. Berdasarkan hasil pengujian dari *tools* Appachhi aplikasi *Android mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memiliki rata-rata penggunaan memori sebesar 22,8 MB.

Berdasarkan seluruh pengujian *performance efficiency* yang sudah dilakukan, dapat disimpulkan secara umum bahwa aplikasi dapat berjalan dengan baik tanpa

mengalami *memory leak* yang mengakibatkan *force close* dan *launch fail*, sehingga aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dikatakan “**Layak**”.

B. Pembahasan

Aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows memberikan informasi tentang penjelasan fungsi, sintaks, opsi, dan contoh penggunaan perintah. Aplikasi ini dikembangkan menggunakan Android Studio dengan telah melalui seluruh tahapan pengembangan mulai dari tahap *inception*, *elaboration*, *construction*, dan *transition*. Di setiap tahapan pengembangan, terdapat *core process workflows* dan *core supporting workflows*. *Core process workflows* dibagi menjadi enam tahap yaitu *business modelling*, *requirements*, *analysis & design*, *implementation*, *testing*, dan *deployment*. *Core supporting workflows* dibagi menjadi tiga tahap yaitu *configuration & change management*, *project management*, dan *environment*.

Pada tahap *inception* pengembangan lebih berkonsentrasi pada analisis dan kebutuhan *user*. Pada tahap *elaboration* pengembangan dari hasil analisis dan kebutuhan di detailkan ke dalam rancangan desain dari sistem tersebut. Pada tahap *construction* rancangan-rancangan desain sistem tersebut diimplementasikan ke dalam proses *coding*/pemrograman. Pada tahap *transition* pengembangan ditujukan kepada migrasi data dan pengujian terakhir dari aplikasi yang telah dikembangkan. Pengujian terakhir berdasarkan ISO/IEC 25010 meliputi

functional suitability, usability, compatibility, dan performance efficiency. Berikut hasil pengujian terakhir aplikasi berdasarkan ISO/IEC 25010 pada Tabel 27.

Tabel 27. Hasil Pengujian Aplikasi Berdasarkan ISO/IEC 25010

No.	Aspek	Hasil	Kategori
1.	<i>Functional Suitability</i>	Seluruh fungsi dapat berjalan 100%.	Sangat Layak
2.	<i>Usability</i>	Aplikasi memperoleh skor <i>usability</i> sebesar 88,1% setelah diuji oleh 30 siswa kelas XI TKJ 2 SMK N 1 Bantul.	Sangat Layak
3.	<i>Compatibility</i>	Tahap <i>coexistence</i> aplikasi dapat berjalan berdampingan dengan aplikasi yang lain dengan presentase kelayakan 100%.	Sangat Layak
		Aplikasi dapat berjalan diberbagai tipe perangkat dan sistem operasi Android. Aplikasi dapat berhasil berjalan pada 15 perangkat yang diujikan pada <i>Firebase Tesat Lab</i> dengan persentase kelayakan 100% dan kompatibel dengan 15.124	Sangat Layak

		perangkat dari total 16.564 perangkat yang tersedia pada <i>Google Play Store</i> dengan persentase kelayakan 91,3%.	
4.	<i>Performance Efficiency</i>	Aplikasi <i>mobile dictionary and emulator</i> perintah dasar CLI Linux dan Windows dapat berjalan dengan baik tanpa terjadi <i>memory leak</i> yang mengakibatkan <i>launch fail</i> dan <i>force stop</i> .	Layak
		Aplikasi <i>mobile dictionary and emulator</i> perintah dasar CLI Linux dan Windows memiliki time behavior rata-rata 0,039 <i>seconds/thread</i> .	
		Aplikasi <i>mobile dictionary and emulator</i> perintah dasar CLI Linux dan Windows menggunakan CPU rata-rata sebesar 49,6%.	
		Aplikasi <i>mobile dictionary and</i>	

		<i>emulator</i> perintah dasar CLI Linux dan Windows menggunakan <i>memory</i> rata-rata 22,8 MB	
--	--	---	--

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan dalam pengembangan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dapat disimpulkan bahwa :

1. Penelitian ini menghasilkan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows yang telah memenuhi kebutuhan *user*. Aplikasi yang dikembangkan berisi perintah-perintah dasar CLI Linux dan Windows yang memuat tentang fungsi, sintaks, parameter, dan contoh penggunaan perintah. Aplikasi ini memiliki fitur utama yaitu, fitur *search* untuk mencari perintah CLI yang diinginkan, fitur *sharing* untuk membagikan data perintah yang dicari ke *user* lain melalui perantara aplikasi lain seperti media sosial, fitur implementasi untuk mengimplementasikan perintah yang dicari pada emulator OS baik untuk OS Linux maupun Windows sehingga *user* tidak perlu membuka laptop/PC untuk mengimplementasikan perintah tersebut, dan fitur kuis untuk mengukur sejauh mana pemahaman *user* tentang perintah CLI.
2. Aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows telah memenuhi standar kualitas perangkat lunak ISO 25010. Hasil pengujian aspek *functional suitability* memperoleh skor 100% atau sangat layak, karena seluruh fungsi yang ada di aplikasi dapat berjalan

dengan baik. Pengujian aspek *compatibility* juga memperoleh hasil sangat layak karena aplikasi dapat berjalan bersamaan dengan aplikasi lain tanpa merugikan salah satu aplikasi dan aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows dapat dipasang dan dijalankan dengan baik pada berbagai tipe perangkat dan berbagai jenis sistem operasi Android. Pada pengujian aspek *performance efficiency* aplikasi dapat berjalan dengan baik tanpa terjadi *memory leak* atau *force close* dengan *time behavior* 0,039 *seconds/thread* dengan penggunaan CPU rata-rata 49,6% dan konsumsi *memory* rata-rata 22,8 MB. Sedangkan untuk aspek *usability* aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows mendapatkan skor 88,1 % dengan kategori sangat layak. Sehingga secara umum aplikasi Android *mobile dictionary and emulator* perintah dasar CLI Linux dan Windows sangat layak untuk digunakan sebagai sumber belajar tambahan siswa TKJ.

B. Keterbatasan Produk

Aplikasi yang dikembangkan masih memiliki keterbatasan yakni sebagai berikut :

1. Data perintah CLI masih bersifat lokal, jadi untuk melakukan *update* data harus dilakukan dengan memperbarui aplikasi melalui Google Play Store.
2. Aplikasi hanya dapat digunakan pada sistem operasi Android dengan versi minimal Android 4.0.3-4.0.4 Ice Cream Sandwich atau API 15.

3. Pada emulator OS Linux dan Windows ada perintah-perintah tertentu yang tidak bisa dijalankan pada emulator karena membutuhkan hak akses oleh sistem.

C. Pengembangan Lebih Lanjut

Pengembangan aplikasi selanjutnya adalah dengan menambahkan fitur-fitur aplikasi yang belum ada seperti data perintah CLI dibuat *online* dengan disimpan di *cloud database* jadi jika ada pembaharuan pada perintah CLI dapat langsung diperbarui melalui *cloud database* yang digunakan tanpa melalui *update* aplikasi melalui Play Store, sehingga pengguna dapat memaksimalkan penggunaan aplikasi.

D. Saran

Berdasarkan dari simpulan dan temuan dari penelitian yang telah dilakukan, maka peneliti memberikan saran sebagai berikut :

1. Data perintah CLI dibuat *online* dengan disimpan di *cloud database* seperti Firebase jadi jika ada pembaharuan pada perintah CLI dapat langsung diperbarui melalui *cloud database* tanpa harus *update* aplikasi di Play Store.
2. Mengembangkan aplikasi pada *platform* selain Android seperti iOS agar aplikasi dapat menjangkau pengguna yang lebih banyak.
3. Memperbarui emulator OS Linux dan Windows yang memungkinkan semua perintah CLI dapat dijalankan pada sistem.

DAFTAR PUSTAKA

- APJII.(2017).Profil Pengguna Internet Indonesia 2017. Jakarta: Asosiasi Penyelenggara Jasa Internet Indonesia.
- Buyens, Jim. 2001. Web Database Development. Elex Media Komputindo. Jakarta
- Chaer, Abdul.(2007). Linguistik Umum. Jakarta: PT.Rineka Cipta.
- Counter, Stat. (2018). *Mobile Operating System Market Share Indonesia*. Diakses dari <http://gs.statcounter.com/os-market-share/mobile/indonesia/>. pada tanggal 5 Februari 2018, Jam 16.50 WIB.
- David, Assaf Ben.(2011).*Mobile Application Testing (Best Practices to Ensure Quality)*.Amdocs, 2.
- Developers, Android. (2018). *Platform Versions*. Diakses dari <https://developer.Android.com/about/dashboards/index.html>. pada tanggal 5 Februari 2018, Jam 18.30 WIB
- DeveloperWorks. (2004). *Linux Today*. Diakses dari developerWorks: Write Emulator-Friendly Linux Code pada tanggal 12 Agustus 2018. Jam 20.30.
- Enterprise, Jubilee. (2015). *Mengenal Dasar-Dasar Pemrograman Android*. Jakarta : PT Elex Media Komputindo.
- Hutomo, Afied. (2014). *Pengembangan Aplikasi Android Kamus Command Line (Fycomm) Sebagai Media Bantu Belajar Siswa SMK Negeri 1 Bantul Kompetensi Keahlian Teknik Komputer Dan Jaringan*. Jurnal Elektronik Pendidikan Teknik Informatika.
- Irsyad, Hanif. (2016). *Aplikasi Android dalam 5 menit cara cepat membuat aplikasi Android tanpa coding*. Jakarta : PT Elex Media Komputindo.
- ISO/IEC. (2011, Maret 1). *Systems and software engineering – systems and software quality requirements and evaluation (SquaRE) – systems and software quality models*, 1. (I. J. 7, Editor) Dipetik Februari, 2018, dari ISO: <http://iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>
- KBBI Daring.(2018). Pengertian Kamus. Diakses dari <http://kbbi.web.id/media>. Pada tanggal 5 Februari 2018 jam 16.17 WIB.
- Lund, A.M. (2001). *Measuring Usability with the USE Questionnaire*. STC Usability SIG Newsletter, p.2.
- Munir, Muhammad. 2013. Analisis Pengembangan Media Pembelajaran Pengolah Angka (Spreadsheet) Berbasis Video Screencast. Jurnal Pendidikan Teknologi dan Kejuruan, Volume 21, Nomor 4.

- Nazruddin ,Safaat H. (2014). *Pengembangan Aplikasi Mobile Smartphone dan Tablet PC berbasis Android*. Bandung : Informatika Bandung.
- Olivia, Femi.(2008).Teknik Membaca Efektif: Menciptakan Kebiasaan Belajar yang Efektif dengan Keterampilan Membaca Formula 5S. Jakarta : PT Elex Media Komputindo.
- Pressman, R.(2012). *Software Engineering: A Practitioner's Approach, Seventh Edition*.(A. Nugroho, G. Nikijuluw, T. Rochadiani, & I. Wijaya, Trans.) Yogyakarta: Penerbit Andi.
- Ramadhina, Syahrina. 2015. Pembuatan Sistem Informasi Manajemen Bengkel Di Sekolah Menengah Kejuruan Negeri 3 Yogyakarta. Jurnal JPTK,22,328
- Rosa A. S., & Shalahuddin, M. (2011). Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek).Bandung: Modula.
- Sommerville, Ian.(2003).*Software Engineering*.Penerjemah: Dra. Yuhliza Hanum, M.Eng. Jakarta : Erlangga.
- Sudaryono. (2011). *Metodologi Riset di Bidang TI (Panduan Praktis Teori dan Contoh Kasus)*. Yogyakarta: Penerbit Andi.
- Sugiyono. (2014). Metode Penelitian Pendidikan Pendekatan Kuantitatif,Kualitatif, dan R&D. Bandung: Alfabeta.
- Sugiono, Dendy, Alwi, Hasan. (2002). *Telaah Bahasa dan Sastra*. Jakarta : Yayasan Obor Indonesia.
- Sunarso, A., Patoni, A., & Basiroh U. (1990). *Pedoman Penyusunan Kamus Dwi Bahasa*. Jakarta : Departemen Pendidikan dan Kebudayaan.
- Wardhana, S.R, Purwitasari, D. & Rochimah, S. (2017). Analisis sentimen pada *review* pengguna aplikasi *mobile* untuk evaluasi faktor *usability*. Jurnal Sistem dan Informatika.
- Wagner, S. (2013). *Software Product Quality Control*. Springer, 60-64.
- Widianti, Sri. 2000. Pengantar Basis Data. Penerbit Fajar: Jakarta
- Wulandari, B., Suparman, Santoso, D., Muslikhin, Utami, A.D.W. 2015. Pengembangan Trainer Equalizer Grafis Dan Parametris Sebagai Media Pembelajaran Mata Kuliah Praktik Sistem Audio. Jurnal Pendidikan Teknologi dan Kejuruan, Volume 22, Nomor 4.

LAMPIRAN

Lampiran 1. Surat Keputusan Dosen Pembimbing

**KEPUTUSAN DEKAN FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
NOMOR : 137/PINF/PB/VI/2018**

**TENTANG
PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) MAHASISWA
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

DEKAN FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

- Menimbang : a. bahwa untuk kelancaran pelaksanaan kegiatan Tugas Akhir Skripsi (TAS) mahasiswa, dipandang perlu mengangkat dosen pembimbingnya;
b. bahwa untuk keperluan sebagaimana dimaksud pada huruf a perlu menetapkan Keputusan Dekan Tentang Pengangkatan Dosen Pembimbing Tugas Akhir Skripsi (TAS) Fakultas Teknik Universitas Negeri Yogyakarta.
- Mengingat : 1. Undang-undang RI Nomor 20 Tahun 2003 Tentang Sistem Pendidikan Nasional (Lembaran Negara Tahun 2003 Nomor 78, Tambahan Lembaran Negara Nomor 4301);
2. Peraturan Pemerintah Republik Indonesia Nomor 4 Tahun 2014 Tentang Penyelenggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi (Lembaran Negara Tahun 2014 Nomor 16, Tambahan Lembaran Negara Republik Indonesia Nomor 5500);
3. Keputusan Presiden Republik Indonesia Nomor 93 Tahun 1999 Tentang Perubahan Institut Keguruan dan Ilmu Pendidikan menjadi Universitas;
4. Peraturan Mendiknas RI Nomor 23 Tahun 2011 Tentang Organisasi dan Tata Kerja Universitas Negeri Yogyakarta;
5. Peraturan Mendiknas RI Nomor 34 Tahun 2011 Tentang Statuta Universitas Negeri Yogyakarta;
6. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 98/MPK.A4/KP/2013 Tentang Pengangkatan Rektor Universitas Negeri Yogyakarta;
7. Peraturan Rektor Nomor 2 Tahun 2014 tentang Peraturan Akademik;
8. Keputusan Rektor Nomor 800/UN.34/KP/2016 tahun 2016 tentang Pengangkatan Dekan Fakultas Teknik Universitas Negeri Yogyakarta.

MEMUTUSKAN

Menetapkan : **KEPUTUSAN DEKAN TENTANG PENGANGKATAN DOSEN PEMBIMBING TUGAS AKHIR SKRIPSI (TAS) FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA.**

PERTAMA : Mengangkat Saudara :
Nama : Nurkhamid, S.Si., M.Kom., Ph.D.
NIP : 19680707 199702 1 001
Pangkat/Golongan : Penata Muda, III/a
Jabatan Akademik : Asisten Ahli



sebagai Dosen Pembimbing Untuk mahasiswa penyusun Tugas Akhir Skripsi (TAS) :

Nama : Dian Kartika Sari
NIM : 14520244012
Prodi Studi : Pend. Teknik Informatika - S1
Judul Skripsi/TA : PENGEMBANGAN APLIKASI KAMUS PERINTAH DASAR LINUX DAN WINDOWS BERBASIS COMMAND LINE INTERFACE PADA PLATFORM ANDROID

Lampiran 2. Surat Ijin Penelitian dari Fakultas Teknik UNY

KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI UNIVERSITAS NEGERI YOGYAKARTA FAKULTAS TEKNIK Alamat : Kampus Karangmalang, Yogyakarta, 55281 Telp. (0274) 586168 psw. 276,289,292 (0274) 586734 Fax. (0274) 586734 Laman: ft.uny.ac.id E-mail: ft@uny.ac.id, teknik@uny.ac.id	
Nomor : 451/UN34.15/LT/2018	25 Mei 2018
Lamp. : 1 Bendel Proposal	
Hal : Izin Penelitian	
 Yth . 1. Gubernur Daerah Istimewa Yogyakarta c.q. Ka. Badan Kesatuan Bangsa dan Politik DIY 2. Kepala Sekolah SMK Negeri 1 Bantul	
 Kami sampaikan dengan hormat, bahwa mahasiswa tersebut di bawah ini:	
Nama :	Dian Kartika Sari
NIM :	14520244012
Program Studi :	Pend. Teknik Informatika - S1
Judul Tugas Akhir :	PENGEMBANGAN APLIKASI MOBILE KAMUS PERINTAH DASAR LINUX DAN WINDOWS BERBASIS COMMAND LINE INTERFACE PADA PLATFORM ANDROID
Tujuan :	Memohon izin mencari data untuk penulisan Tugas Akhir Skripsi (TAS)
Waktu Penelitian :	Senin - Kamis, 4 - 7 Juni 2018
 Untuk dapat terlaksananya maksud tersebut, kami mohon dengan hormat Bapak/Ibu berkenan memberi izin dan bantuan seperlunya.	
Demikian atas perhatian dan kerjasamanya kami sampaikan terima kasih.	
	 Dekan Fakultas Teknik  Dr. Drs. Widarto, M.Pd. NIP. 19631230 198812 1 001
Tembusan	
1 Sub. Bagian Pendidikan dan Kemahasiswaan ;	
2 Mahasiswa yang bersangkutan.	

Lampiran 3. Surat Ijin Penelitian dari Badan Kesatuan Bangsa dan Politik DIY

	<p>PEMERINTAH DAERAH DAERAH ISTIMEWA YOGYAKARTA BADAN KESATUAN BANGSA DAN POLITIK Jl. Jenderal Sudirman No 5 Yogyakarta – 55233 Telepon : (0274) 551136, 551275, Fax (0274) 551137</p>
	<p>Yogyakarta, 28 Mei 2018</p>
	<p>Kepada Yth. :</p>
Nomor : 074/6497/Kesbangpol/2018	Kepala Dinas Pendidikan, Pemuda, dan Olahraga DIY
Perihal : Rekomendasi Penelitian	di Yogyakarta
<p>Memperhatikan surat :</p>	
Dari : Dekan Fakultas Teknik Universitas Negeri Yogyakarta	
Nomor : 451/UN34.15/LT/2018	
Tanggal : 25 Mei 2018	
Perihal : Izin Penelitian	
<p>Setelah mempelajari surat permohonan dan proposal yang diajukan, maka dapat diberikan surat rekomendasi tidak keberatan untuk melaksanakan riset/penelitian dalam rangka penyusunan skripsi dengan judul proposal : "PENGEMBANGAN APLIKASI MOBILE KAMUS PRINTAH DASAR LINUX DAN WINDOWS BERBASIS COMMAND LINE INTERFACE PADA PLATFORM ANDROID" kepada:</p>	
Nama : DIAN KARTIKA SARI	
NIM : 14520244012	
No.HP/Identitas : 0895362328906/3318216612950003	
Prodi/Jurusan : Pendidikan Teknik Informatika/Pendidikan Teknik Elektronika dan Informatika	
Fakultas : Fakultas Teknik Universitas Negeri Yogyakarta	
Lokasi Penelitian : SMK Negeri 1 Bantul	
Waktu Penelitian : 28 Mei 2018 s.d 30 Juni 2018	
<p>Sehubungan dengan maksud tersebut, diharapkan agar pihak yang terkait dapat memberikan bantuan / fasilitas yang dibutuhkan.</p>	
<p>Kepada yang bersangkutan diwajibkan:</p>	
<ol style="list-style-type: none">1. Menghormati dan mentaati peraturan dan tata tertib yang berlaku di wilayah riset/penelitian;2. Tidak dibenarkan melakukan riset/penelitian yang tidak sesuai atau tidak ada kaitannya dengan judul riset/penelitian dimaksud;3. Menyerahkan hasil riset/penelitian kepada Badan Kesbangpol DIY selambat-lambatnya 6 bulan setelah penelitian dilaksanakan.4. Surat rekomendasi ini dapat diperpanjang maksimal 2 (dua) kali dengan menunjukkan surat rekomendasi sebelumnya, paling lambat 7 (tujuh) hari kerja sebelum berakhirnya surat rekomendasi ini.	
<p>Rekomendasi Ijin Riset/Penelitian ini dinyatakan tidak berlaku, apabila ternyata pemegang tidak mentaati ketentuan tersebut di atas.</p>	
<p>Demikian untuk menjadikan maklum.</p>	
<p style="text-align: right;"> KERAJAAN DAERAH DAERAH ISTIMEWA YOGYAKARTA BADAN KESATUAN BANGSA DAN POLITIK AGUS SUPRIYONO, SH NIP. 196010261992031004</p>	
<p><u>Tembusan disampaikan Kepada Yth.</u></p> <ol style="list-style-type: none">1. Gubernur DIY (sebagai laporan)2. Dekan Fakultas Teknik Universitas Negeri Yogyakarta;3. Yang bersangkutan	

Lampiran 4. Surat Ijin Penelitian dari DISPORA Yogyakarta



PEMERINTAH DAERAH DAERAH ISTIMEWA YOGYAKARTA
DINAS PENDIDIKAN, PEMUDA, DAN OLAHRAGA
Jalan Cendana No. 9 Yogyakarta, Telepon (0274) 541322, Fax. 541322
web : www.dikpora.jogjaprovo.go.id, email : dikpora@jogjaprovo.go.id, Kode Pos 55166

Yogyakarta, 28 Mei 2018

Nomor : 070 / 5955
Lamp : -
Hal : Rekomendasi Penelitian

Kepada Yth.
Kepala SMK Negeri 1 Bantul

Dengan hormat, memperhatikan surat dari Badan Kesatuan Bangsa dan Politik Pemerintah Daerah Daerah Istimewa Yogyakarta nomor: 074/6497/Kesbangpol/2018 tanggal 28 Mei 2018 perihal Rekomendasi Penelitian, kami sampaikan bahwa Dinas Pendidikan, Pemuda, dan Olahraga DIY memberikan ijin rekomendasi penelitian kepada:

Nama : Dian Kartika Sari
NIM : 14520244012
Prodi/Jurusan : Pendidikan Teknik Informatika/Pendidikan Teknik Elektronika dan Informatika
Fakultas : Fakultas Teknik Universitas Negeri Yogyakarta
Judul : PENGEMBANGAN APLIKASI *MOBILE* KAMUS PRINTAH DASAR LINUX DAN WINDOWS BERBASIS *COMMAND LINE INTERFACE* PADA PLATFORM ANDROID
Lokasi : SMK Negeri 1 Bantul
Waktu : 28 Mei 2018 s.d 30 Juni 2018

Dengan ketentuan sebagai berikut :

1. Ijin ini hanya dipergunakan untuk keperluan ilmiah, dan pemegang ijin wajib mentaati ketentuan yang berlaku di lokasi penelitian.
2. Ijin yang diberikan dapat dibatalkan sewaktu-waktu apabila pemegang ijin ini tidak memenuhi ketentuan yang berlaku.

Atas perhatian dan kerjasama yang baik, kami menyampaikan terimakasih.

a.n Kepala
Kepala Bidang Perencanaan dan Standarisasi

Didik Wardaya, SE., M.Pd.
NIP.19660530 198602 1 002

Tembusan Yth :
1 Kepala Dinas Dikpora DIY
2 Kepala Bidang Dikmenti Dikpora DIY

Lampiran 5. Surat Selesai Penelitian di SMK N 1 Bantul

	PEMERINTAH DAERAH DAERAH ISTIMEWA YOGYAKARTA DINAS PENDIDIKAN PEMUDA DAN OLARHAGA SMK NEGERI 1 BANTUL Jl Parangliris Km 11 Sabdodadi Bantul 55702 Telp. 367156 web : www.smkn1bantul.sch.id e-mail : smeanbl@yahoo.com
<u>SURAT KETERANGAN</u> Nomor : 070/547	
Yang bertanda tangan pada surat ini PLH Kepala Sekolah:	
Nama	: Drs. Warohman, M.S.I
NIP	: 195907291979121005
Pangkat / Gol	: Pembina/Iva
Jabatan	: Guru Madya / Sek. Kepala Sekolah
Instansi	: SMKN 1 Bantul
Menerangkan bahwa :	
Nama	: Dian Kartika Sari
NIM	: 14520244012
Prodi/Fak/Universitas	: Pendidikan Teknik Informatika/Fakultas Teknik/UNY
Benar-benar telah melaksanakan penelitian Dengan Judul " Pengembangan Aplikasi Mobile Kamus Perintah Dasar Linux Dan Windows Berbasis Command Line Interface Pada Platform Android di SMK N 1 Bantul ", yang dilaksanakan pada pada tanggal 28 Mei 2018 s.d. 30 Juni 2018.	
Demikian surat keterangan ini dibuat untuk dapat digunakan sebagaimana mestinya.	
 Bantul, 6 Juni 2018 PLH. Kepala Sekolah, Drs. Warohman, M.S.I NIP. 195907291979121005	

Lampiran 6. Surat Permohonan Validasi Instrumen TAS

Hal : Permohonan Validasi Instrumen TAS

Lampiran : 1 Bandel

Kepada Yth,

Bapak Nurkhamid S.Si., M.Kom., Ph.D.

Dosen Jurusan Pendidikan Teknik Elektronika

di Fakultas Teknik UNY

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

Nama : Dian Kartika Sari

NIM : 14520244012

Program Studi : Pendidikan Teknik Informatika

Judul TAS : PENGEMBANGAN APLIKASI *MOBILE* PERINTAH DASAR
LINUX DAN WINDOWS BERBASIS *COMMAND LINE*
INTERFACE PADA *PLATFORM 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, dan (2) draf instrumen penelitian TAS.

Demikian permohonan saya, atas bantuan dan perhatian Bapak diucapkan terimakasih.

Yogyakarta,

Pemohon,



Dian Kartika Sari

NIM 14520244012

Mengetahui,

Kaprodi Pendidikan Teknik Informatika,

Pembimbing TAS,



Handaru Jati, Ph.D.

NIP. 19740511 199903 1 002



Nurkhamid S.Si., M.Kom., Ph.D.

NIP. 19680707 199702 1 001

Lampiran 7. Surat Pernyataan Instrumen Penelitian TAS

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

Saya yang bertanda tangan dibawah ini:

Nama : NURKHAMID S.Si., M.Kom., Ph.D.
NIP : 19680707 199702 1 001
Jurusan : DOSEN PEJID - TEKNIK INFORMATIKA

Menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Dian Kartika Sari
NIM : 14520244002
Program Studi : PENDIDIKAN TEKNIK INFORMATIKA
Judul TAS : PENGEMBANGAN APLIKASI *MOBILE* PERINTAH
DASAR LINUX DAN WINDOWS BERBASIS
COMMAND LINE INTERFACE PADA PLATFORM
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, Mei 2018

Validator,



Nurkhamid S.Si., M.Kom., Ph.D.

NIP. 19680707 199702 1 001

Catatan:

- Beri tanda ✓

Hasil Validasi Instrumen Penelitian TAS

Nama Mahasiswa : Dian Kartika Sari

NIM : 14520244012

Judul TAS : PENGEMBANGAN APLIKASI *MOBILE* PERINTAH DASAR LINUX DAN WINDOWS BERBASIS *COMMAND LINE INTERFACE* PADA *PLATFORM ANDROID*

No.	Variabel	Saran/Tanggapan
	Komentar Umum/Lain-lain	Usability memahar USE (Lund)-

Yogyakarta, Mei 2018

Validator,



Nurkhamid S.Si., M.Kom., Ph.D.

NIP. 19680707 199702 1 001

Lampiran 8. Hasil Uji *Functional Suitability*

LEMBAR PENGUJIAN *FUNCTIONAL SUITABILITY*

**PENGEMBANGAN APLIKASI *MOBILE* KAMUS PERINTAH DASAR LINUX DAN WINDOWS
BERBASIS *COMMAND LINE INTERFACE* PADA *PLATFORM ANDROID***

A. Identitas Responden

1. Nama responden : Heru Setiawan
 2. Status : Karyawan di PT Aneka Tekn
 3. Program keahlian : Backend Developer

B. Petunjuk Pengisian Kuisiner

- Mengisi identitas Anda dengan lengkap.
- Download aplikasi pada *play store* dengan *key word* aplikasi kamus CLI Clict dan *install* pada *smartphone* Anda.
- Pengisian dilakukan dengan cara memberikan tanda centang (V) pada salah satu jawaban yang menurut Anda paling tepat.
- Setiap pertanyaan hanya memiliki satu jawaban.
- Setelah mengisi jawaban pada kuisiner ini, mohon diperiksa kembali agar pertanyaan yang belum terisi tidak terlewat (kosong).

C. Contoh Pengisian Kuisiner

No	Fitur	Deskripsi	Langkah	Kegiatan	Hasil yang diharapkan	Hasil Keluaran	
						Berhasil	Gagal
1.	Beranda	Fitur untuk menampilkan halaman utama (dashboar d) yang berisi menu – menu yang ada di aplikasi.	1	Sentuh icon beranda yang ada di menu bagian bawah.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.	v	

Pada contoh diatas jika fitur berhasil dijalankan sesuai dengan hasil yang diharapkan maka berilah tanda (v) pada kolom tabel Berhasil.

D. Instrumen *Functional Suitability*

No.	Fitur	Deskripsi	Langkah	Kegiatan	Hasil yang diharapkan	Hasil Keluaran	
						Berhasil	Gagal
1.	Beranda	Fitur untuk menampilkan halaman utama (dashboard) yang berisi menu – menu yang ada di aplikasi.	1	Sentuh icon beranda yang ada di menu bagian bawah.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.	✓	
2.	Petunjuk	Fitur untuk menampilkan petunjuk mengenai menu – menu yang ada pada aplikasi.	1	Sentuh icon petunjuk yang ada di menu bagian bawah.	Halaman petunjuk muncul dan menampilkan petunjuk mengenai menu – menu yang ada pada aplikasi.	✓	
3.	Tentang aplikasi	Fitur untuk memunculkan informasi aplikasi yang meliputi kredit, versi dll.	1	Sentuh icon tentang yang ada di menu bagian bawah.	Halaman tentang tampil.	✓	
4.	CLI Linux	Fitur untuk menampilkan daftar perintah CLI Linux.	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.	✓	
			2	Pilih menu Linux.	Halaman daftar perintah CLI Linux tampil.	✓	
5.	CLI Windows	Fitur untuk menampilkan daftar perintah CLI Windows.	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.	✓	
			2	Pilih menu Windows.	Halaman daftar perintah CLI Windows tampil.	✓	
6.	Pencarian CLI	Fitur untuk	1	Lakukan kegiatan	Halaman daftar perintah CLI	✓	

	Linux	menampilkan hasil pencarian dibawah kolom pencarian		4. Ketikkan kata/perintah yang ingin dicari.	Linux tampil. Aplikasi menampilkan daftar perintah hasil pencarian.	✓	
7.	Menampilkan papan keyboard	Fitur untuk menampilkan papan keyboard.	1	Sentuh kolom pencarian.	Papan keyboard akan tampil dan kursor fokus pada kolom pencarian.	✓	
8.	Filter karakter	Fitur untuk menampilkan hasil pencarian hanya pada perintah yang mengandung karakter yang di ketikan pada kolom search.	1	Lakukan kegiatan 7.	Aplikasi hanya akan menampilkan daftar perintah yang mengandung karakter yang dimasukkan.	✓	
			2	Masukkan 2 karakter atau lebih dengan <i>soft input keyboard</i> .		✓	
9.	Rincian Perintah CLI Linux	Fitur untuk menampilkan penjelasan perintah CLI yang berupa fungsi, sintaks, dan contoh penerapan perintah.	1	Lakukan kegiatan 6.	Aplikasi menampilkan daftar perintah hasil pencarian.	✓	
			2	Sentuh perintah yang dicari.	Aplikasi akan membuka halaman baru yang berisi tentang penjelasan dari	✓	

					peirntah yang dipilih.		
10.	Berbagi perintah CLI Linux	Fitur untuk membagikan konten yang terdapat di halaman penjelasan rincian perintah CLI Linux melalui aplikasi lain.	1	Lakukan kegiatan 9.	Halaman penjelasan perintah CLI Linux akan muncul.	✓	
			2	Sentuh icon "share" disudut kanan atas.	Jendela opsi aplikasi yang mendukung berbagi konten muncul.	✓	
			3	Pilih salah satu aplikasi dari daftar.	Konten yang akan dibagikan masuk kedalam kolom "konten" suatu Aplikasi.	✓	
			4	Kirim sesuai dengan metode kirim tiap – tiap aplikasi.	Konten dikirimkan kepada penerima.	✓	
11.	Implementasi terminal Linux	Fitur untuk menerapkan/mengimplem entasikan perintah CLI ke emulator terminal Linux secara langsung.	1	Lakukan kegiatan 9.	Halaman penjelasan perintah CLI Linux akan muncul.	✓	
			2	Sentuh icon "implementasi" disudut kanan atas.	Aplikasi akan menampilkan emulator terminal Linux.	✓	
12.	Menampilkan papan keyboard pada terminal Linux	Fitur untuk menampilkan papan keyboard pada terminal linux.	1	Lakukan kegiatan 11.	Halaman emulator terminal tampil.	✓	
			2	Sentuh icon "keyboard" di sudut kanan atas.	Aplikasi menampilkan papan keyboard dan fokus pada emulator.	✓	
13.	Pencarian CLI Windows	Fitur untuk menampilkan	1	Lakukan kegiatan 5.	Halaman daftar perintah CLI Windows tampil.	✓	

		hasil pencarian dibawah kolom pencarian	2	Ketikkan kata/perintah yang ingin dicari.	Aplikasi menampilkan daftar perintah hasil pencarian.	✓	
14.	Rincian Perintah CLI Windows	Fitur untuk menampilkan penjelasan perintah CLI yang berupa fungsi, sintaks, dan contoh penerapan perintah.	1	Lakukan kegiatan 13.	Aplikasi menampilkan daftar perintah hasil pencarian.	✓	
			2	Sentuh perintah yang dicari.	Aplikasi akan membuka halaman baru yang berisi tentang penjelasan dari perintah yang dipilih.	✓	
15.	Berbagi perintah CLI Windows	Fitur untuk membagikan konten yang terdapat di halaman penjelasan rincian perintah CLI Windows melalui aplikasi lain.	1	Lakukan kegiatan 14.	Halaman penjelasan perintah CLI Windows akan muncul.	✓	
			2	Sentuh icon "share" disudut kanan atas.	Jendela opsi aplikasi yang mendukung berbagi konten muncul.	✓	
			3	Pilih salah satu aplikasi dari daftar.	Konten yang akan dibagikan masuk kedalam kolom "konten" suatu Aplikasi.	✓	
			4	Kirim sesuai dengan metode kirim tiap - tiap aplikasi.	Konten dikirimkan kepada penerima.	✓	
16.	Implementasi <i>command</i>	Fitur untuk menerapkan/mengimplem	1	Lakukan kegiatan 14.	Halaman penjelasan perintah CLI windows akan	✓	

	<i>prompt (cmd) Windows</i>	entaskan perintah CLI ke emulator cmd secara langsung..			muncul.		
			2	Sentuh icon "implementasi" disudut kanan atas.	Aplikasi akan menampilkan emulator cmd.	✓	
17.	Menampilkan papan keyboard pada cmd	Fitur untuk menampilkan papan keyboard pada cmd.	1	Lakukan kegiatan 16.	Halaman emulator terminal tampil.	✓	
			2	Sentuh icon "keyboard" di sudut kanan atas.	Aplikasi menampilkan papan keyboard dan fokus pada emulator.	✓	
18.	Kuis	Fitur untuk menampilkan kuis atau latihan soal bagi <i>user</i> .	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI..	✓	
			2	Pilih menu Kuis.	Halaman kuis akan tampil.	✓	
19.	Skor	Fitur yang digunakan untuk menampilkan skor kuis.	1	Lakukan kegiatan 18.	Halaman kuis akan tampil.	✓	
			2	Kerjakan soal dengan cara meng-klik pada opsi yang dianggap benar.	Aplikasi dapat merekam/menyimpan sementara daftar jawaban <i>user</i> sampai pengerjaan soal selesai secara keseluruhan.	✓	
			3	Klik submit.	Aplikasi akan menampilkan jumlah skor yang diperoleh <i>user</i> dan bagian konfirmasi.	✓	
20.	Konfirmasi pengerjaan kuis	Fitur untuk mengkonfirmasi apakah akan keluar dari kuis atau ingin mengerjakan kuis lagi.	1	Lakukan kegiatan 19.	Aplikasi akan menampilkan jumlah skor yang diperoleh <i>user</i> dan bagian konfirmasi.	✓	
			2	Klik tombol "✓"	Aplikasi menampilkan soal kuis.	✓	
			3	Klik tombol "X"	Aplikasi akan kembali ke halaman beranda.	✓	
21.	Keluar	Fitur untuk keluar dari	1	Klik menu	Keluar dari aplikasi.	✓	

		aplikasi.		"keluar" dibagian opsi menu kanan atas.			
--	--	-----------	--	---	--	--	--

Sleman, 4 Juni 2018



Heru Setiawan

Lampiran 9. Hasil *Usability*

30

LEMBAR PENGUJIAN *USABILITY*

PENGEMBANGAN APLIKASI *MOBILE* KAMUS PERINTAH DASAR LINUX DAN WINDOWS BERBASIS *COMMAND LINE INTERFACE* PADA *PLATFORM ANDROID*

A. Identitas Responden

- 1. Nama : *Bagus Nurcahyo*.....
- 2. Kelas : *XI IPA 2*.....
- 3. Sekolah : *SMK N 1 BANTUL*.....

B. Petunjuk Pengisian Kuisisioner

- 1. Mengisi identitas Anda dengan lengkap.
- 2. Download aplikasi pada *play store* dengan *key word* aplikasi kamus CLI Clict dan *install* pada *smartphone* Anda.
- 3. Bacalah dengan teliti setiap pertanyaan dalam kuisisioner ini sebelum Anda memilih jawaban.
- 4. Pengisian dilakukan dengan cara memberikan tanda centang (V) pada salah satu jawaban yang menurut Anda paling tepat.
Untuk *usability* memilih dari : Sangat Tidak Setuju (STS), Tidak Setuju (TS), Ragu-Ragu (RG), Setuju (S), dan Sangat Setuju (SS).
- 5. Setiap pertanyaan hanya memiliki satu jawaban.
- 6. Setelah mengisi jawaban pada kuisisioner ini, mohon diperiksa kembali agar pertanyaan yang belum terisi tidak terlewat (kosong).
- 7. Jika ada yang tidak mengerti, bertanyalah pada Guru atau Peneliti.

C. Contoh Pengisian Kuisisioner

No	Pernyataan	Skala Penilaian				
		SS	S	RG	TS	STS
<i>Usefulness</i>						
1	Aplikasi ini membantu saya menjadi lebih efektif		v			

Pada contoh diatas jika **Setuju** aplikasi dapat membantu responden lebih efektif makan bertilah tanda (v) pada kolom tabel S.

D. Instrumen Usability

No	Pertanyaan	Skala Penilaian				
		SS	S	RG	TS	STS
Usefulness						
1	Aplikasi ini membantu saya menjadi lebih efektif	✓				
2	Aplikasi ini membantu saya menjadi lebih produktif	✓				
3	Aplikasi ini bermanfaat	✓				
4	Aplikasi ini memberikan dampak yang besar terhadap tugas yang saya lakukan dalam hidup saya	✓				
5	Aplikasi ini memudahkan saya dalam mencapai hal – hal yang saya inginkan	✓				
6	Aplikasi ini menghemat waktu saya ketika saya menggunakannya		✓			
7	Aplikasi ini sesuai dengan kebutuhan saya	✓				
8	Aplikasi ini sesuai dengan apa yang saya harapkan	✓				
Ease of Use						
9	Aplikasi ini mudah digunakan	✓				
10	Aplikasi ini praktis untuk digunakan	✓				
11	Aplikasi ini mudah dipahami		✓			
12	Aplikasi ini hanya memerlukan langkah-langkah singkat dalam menggunakannya		✓			
13	Aplikasi ini dapat disesuaikan dengan kebutuhan	✓				
14	Tidak ada kesulitan dalam menggunakan Aplikasi ini		✓			
15	Saya dapat menggunakan tanpa instruksi tertulis			✓		
16	Saya melihat Aplikasi ini sudah konsisten ketika digunakan		✓			
17	Baik pengguna yang jarang maupun yang	✓				

	rutin akan suka menggunakan Aplikasi ini					
18	Saya dapat kembali dari kesalahn dengan cepat dan mudah	✓				
19	Saya dapat menggunakan Aplikasi ini dengan hasil setiap saya gunakan		✓			
<i>Ease of Learning</i>						
20	Saya belajar menggunakan Aplikasi ini dengan cepat	✓				
21	Saya mengingat penggunaan Aplikasi ini dengan mudah	✓				
22	Penggunaan Aplikasi ini mudah dipelajari	✓				
23	Saya mahir menggunakan Aplikasi ini dengan cepat		✓			
<i>Satisfication</i>						
24	Saya puas dengan Aplikasi ini	✓				
25	Saya merekomendasikan Aplikasi ini kepada teman-teman	✓				
26	Aplikasi ini menyenangkan untuk digunakan	✓				
27	Aplikasi ini bekerja sesuai dengan yang saya inginkan	✓				
28	Aplikasi ini sangat bagus	✓				
29	Saya merasa harus menggunakan Aplikasi ini	✓				
30	Aplikasi ini nyaman untuk digunakan	✓				

Lampiran 10. Karakteristik dan Sub Karakteristik Standar 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>Coexistence</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> <i>Accessibility</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>
7	<i>Maintainability</i>	<i>Modularity</i>
		<i>Reusability</i>
		<i>Analysability</i>
		<i>Modified Ability</i>
		<i>Testability</i>
8	<i>Portability</i>	<i>Adaptability</i>
		<i>Installability</i>
		<i>Replaceability</i>

Lampiran 11. Instrumen Pengujian Aspek *Functional Suitability*

LEMBAR PENGUJIAN *FUNCTIONAL SUITABILITY*

PENGEMBANGAN APLIKASI ANDROID *MOBILE* KAMUS PERINTAH DASAR *COMMAND LINE INTERFACE* LINUX DAN WINDOWS UNTUK SISWA TKJ SMK N 1 BANTUL

A. Identitas Responden

1. Nama responden :
2. Status :
3. Program keahlian :

B. Petunjuk Pengisian Kuesioner

1. Mengisi identitas Anda dengan lengkap.
2. Download aplikasi pada *play store* dengan *key word* aplikasi kamus CLI Clict dan *install* pada *smartphone* Anda.
3. Pengisian dilakukan dengan cara memberikan tanda centang (V) pada salah satu jawaban yang menurut Anda paling tepat.
4. Setiap pertanyaan hanya memiliki satu jawaban.
5. Setelah mengisi jawaban pada kuesioner ini, mohon diperiksa kembali agar pertanyaan yang belum terisi tidak terlewat (kosong).

C. Contoh Pengisian Kuesioner

No	Fitur	Deskripsi	Langkah	Kegiatan	Hasil yang diharapkan	Hasil Keluaran	
						Berhasil	Gagal
1.	Beranda	Fitur untuk menampilkan halaman	1	Sentuh icon beranda yang ada di menu	Halaman utama muncul dan menampilkan menu –	v	

		utama (dashboar d) yang berisi menu – menu yang ada di aplikasi.		bagian bawah.	menu yang ada pada aplikasi kamus CLI.		
--	--	--	--	---------------	--	--	--

Pada contoh diatas jika fitur berhasil dijalankan sesuai dengan hasil yang diharapkan maka **berilah tanda (v)** pada kolom Tabel **Berhasil**.

D. Instrumen *Functional Suitability*

No.	Fitur	Deskripsi	Langkah	Kegiatan	Hasil yang diharapkan	Hasil Keluaran	
						Berhasil	Gagal
1.	Beranda	Fitur untuk menampilkan halaman utama (dashboard) yang berisi menu – menu yang ada di aplikasi.	1	Sentuh icon beranda yang ada di menu bagian bawah.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.		
2.	Petunjuk	Fitur untuk menampilkan petunjuk mengenai menu – menu yang ada pada aplikasi.	1	Sentuh icon petunjuk yang ada di menu bagian bawah.	Halaman petunjuk muncul dan menampilkan petunjuk mengenai menu – menu yang ada pada aplikasi.		
3.	Tentang aplikasi	Fitur untuk memunculkan informasi aplikasi yang meliputi kredit, versi dll.	1	Sentuh icon tentang yang ada di menu bagian bawah.	Halaman tentang tampil.		
4.	CLI Linux	Fitur untuk menampilkan daftar perintah CLI Linux.	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.		
			2	Pilih menu Linux.	Halaman daftar perintah CLI Linux tampil.		
5.	CLI Windows	Fitur untuk menampilkan daftar perintah CLI Windows.	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.		
			2	Pilih menu Windows.	Halaman daftar perintah CLI Windows tampil.		

6.	Pencarian CLI Linux	Fitur untuk menampilkan hasil pencarian dibawah kolom pencarian	1	Lakukan kegiatan 4.	Halaman daftar perintah CLI Linux tampil.		
			2	Ketikkan kata/perintah yang ingin dicari.	Aplikasi menampilkan daftar perintah hasil pencarian.		
7.	Menampilkan papan keyboard	Fitur untuk menampilkan papan keyboard.	1	Sentuh kolom pencarian.	Papan keyboard akan tampil dan kursor fokus pada kolom pencarian.		
8.	Filter karakter	Fitur untuk menampilkan hasil pencarian hanya pada perintah yang mengandung karakter yang di ketikan pada kolom search.	1	Lakukan kegiatan 7.	Aplikasi hanya akan menampilkan daftar perintah yang mengandung karakter yang dimasukkan.		
			2	Masukkan 2 karakter atau lebih dengan <i>soft input keyboard</i> .			
9.	Rincian Perintah CLI Linux	Fitur untuk menampilkan penjelasan perintah CLI yang berupa fungsi, sintaks, dan contoh penerapan perintah.	1	Lakukan kegiatan 6.	Aplikasi menampilkan daftar perintah hasil pencarian.		

			2	Sentuh perintah yang dicari.	Aplikasi akan membuka halaman baru yang berisi tentang penjelasan dari perintah yang dipilih.		
10.	Berbagi perintah CLI Linux	Fitur untuk membagikan konten yang terdapat di halaman penjelasan rincian perintah CLI Linux melalui aplikasi lain.	1	Lakukan kegiatan 9.	Halaman penjelasan perintah CLI Linux akan muncul.		
			2	Sentuh icon “share” di sudut kanan atas.	Jendela opsi aplikasi yang mendukung berbagi konten muncul.		
			3	Pilih salah satu aplikasi dari daftar.	Konten yang akan dibagikan masuk kedalam kolom “konten” suatu Aplikasi.		
			4	Kirim sesuai dengan metode kirim tiap – tiap aplikasi.	Konten dikirimkan kepada penerima.		
11.	Implementasi terminal Linux	Fitur untuk menerapkan/mengimplementasikan perintah CLI ke emulator terminal Linux secara langsung.	1	Lakukan kegiatan 9.	Halaman penjelasan perintah CLI Linux akan muncul.		
			2	Sentuh icon “implementasi” di sudut kanan atas.	Aplikasi akan menampilkan emulator terminal Linux.		
12.	Menampilkan papan keyboard pada terminal Linux	Fitur untuk menampilkan papan keyboard pada terminal linux.	1	Lakukan kegiatan 11.	Halaman emulator terminal tampil.		
			2	Sentuh icon “keyboard” di	Aplikasi menampilkan papan keyboard dan fokus pada		

				sudut kanan atas.	emulator.		
13.	Pencarian CLI Windows	Fitur untuk menampilkan hasil pencarian dibawah kolom pencarian	1	Lakukan kegiatan 5.	Halaman daftar perintah CLI Windows tampil.		
			2	Ketikkan kata/perintah yang ingin dicari.	Aplikasi menampilkan daftar perintah hasil pencarian.		
14.	Rincian Perintah CLI Windows	Fitur untuk menampilkan penjelasan perintah CLI yang berupa fungsi, sintaks, dan contoh penerapan perintah.	1	Lakukan kegiatan 13.	Aplikasi menampilkan daftar perintah hasil pencarian.		
			2	Sentuh perintah yang dicari.	Aplikasi akan membuka halaman baru yang berisi tentang penjelasan dari perintah yang dipilih.		
15.	Berbagi perintah CLI Windows	Fitur untuk membagikan konten yang terdapat di halaman penjelasan rincian perintah CLI Windows melalui aplikasi lain.	1	Lakukan kegiatan 14.	Halaman penjelasan perintah CLI Windows akan muncul.		
			2	Sentuh icon “share” di sudut kanan atas.	Jendela opsi aplikasi yang mendukung berbagi konten muncul.		
			3	Pilih salah satu aplikasi dari daftar.	Konten yang akan dibagikan masuk kedalam kolom “konten” suatu Aplikasi.		
			4	Kirim sesuai dengan metode	Konten dikirimkan kepada penerima.		

				irim tiap – tiap aplikasi.		
16.	Implementasi <i>command prompt</i> (cmd) Windows	Fitur untuk menerapkan/mengimplementasikan perintah CLI ke emulator cmd secara langsung.	1	Lakukan kegiatan 14.	Halaman penjelasan perintah CLI windows akan muncul.	
			2	Sentuh icon “implementasi” di sudut kanan atas.	Aplikasi akan menampilkan emulator cmd.	
17.	Menampilkan papan keyboard pada cmd	Fitur untuk menampilkan papan keyboard pada cmd.	1	Lakukan kegiatan 16.	Halaman emulator terminal tampil.	
			2	Sentuh icon “keyboard” di sudut kanan atas.	Aplikasi menampilkan papan keyboard dan fokus pada emulator.	
18.	Kuis	Fitur untuk menampilkan kuis atau latihan soal bagi <i>user</i> .	1	Lakukan kegiatan 1.	Halaman utama muncul dan menampilkan menu – menu yang ada pada aplikasi kamus CLI.	
			2	Pilih menu Kuis.	Halaman kuis akan tampil.	
19.	Skor	Fitur yang digunakan untuk menampilkan skor kuis.	1	Lakukan kegiatan 18.	Halaman kuis akan tampil.	
			2	Kerjakan soal dengan cara mengklik pada opsi yang dianggap benar.	Aplikasi dapat merekam/menyimpan sementara daftar jawaban <i>user</i> sampai pengerjaan soal selesai secara keseluruhan.	
			3	Klik submit.	Aplikasi akan menampilkan jumlah skor yang diperoleh <i>user</i> dan bagian konfirmasi.	
20.	Konfirmasi pengerjaan kuis	Fitur untuk mengkonfirmasi apakah akan keluar dari kuis atau ingin mengerjakan kuis	1	Lakukan kegiatan 19.	Aplikasi akan menampilkan jumlah skor yang diperoleh <i>user</i> dan bagian konfirmasi.	

		lagi.	2	Klik tombol “√”	Aplikasi menampilkan soal kuis.		
			3	Klik tombol “X”	Aplikasi akan kembali ke halaman beranda.		
21.	Keluar	Fitur untuk keluar dari aplikasi.	1	Klik menu “keluar” di bagian opsi menu kanan atas.	Keluar dari aplikasi.		

LEMBAR PENGUJIAN *USABILITY*

PENGEMBANGAN APLIKASI *MOBILE* KAMUS PERINTAH DASAR
LINUX DAN WINDOWS BERBASIS *COMMAND LINE INTERFACE* PADA
PLATFORM ANDROID

A. Identitas Responden

1. Nama :.....
2. Kelas :.....
3. Sekolah :.....

B. Petunjuk Pengisian Kuesioner

1. Mengisi identitas Anda dengan lengkap.
2. Download aplikasi pada *play store* dengan *key word* aplikasi kamus CLI Clict dan *install* pada *smartphone* Anda.
3. Bacalah dengan teliti setiap pertanyaan dalam kuesioner ini sebelum Anda memilih jawaban.
4. Pengisian dilakukan dengan cara memberikan tanda centang (V) pada salah satu jawaban yang menurut Anda paling tepat.
Untuk *usability* memilih dari : Sangat Tidak Setuju (STS), Tidak Setuju (TS), Ragu-Ragu (RG), Setuju (S), dan Sangat Setuju (SS).
5. Setiap pertanyaan hanya memiliki satu jawaban.
6. Setelah mengisi jawaban pada kuesioner ini, mohon diperiksa kembali agar pertanyaan yang belum terisi tidak terlewat (kosong).
7. Jika ada yang tidak mengerti, bertanyalah pada Guru atau Peneliti.

C. Contoh Pengisian Kuesioner

No	Pernyataan	Skala Penilaian
----	------------	-----------------

		SS	S	R G	TS	STS
<i>Usefulness</i>						
1	Aplikasi ini membantu saya menjadi lebih efektif		v			

Pada contoh diatas jika **Setuju** aplikasi dapat membantu responden lebih efektif makan **berilah tanda (v)** pada kolom Tabel S.

D. Instrumen *Usability*

No	Pernyataan	Skala Penilaian				
		SS	S	R G	TS	STS
<i>Usefulness</i>						
1	Aplikasi ini membantu saya menjadi lebih efektif					
2	Aplikasi ini membantu saya menjadi lebih produktif					
3	Aplikasi ini bermanfaat					
4	Aplikasi ini memberikan dampak yang besar terhadap tugas yang saya lakukan dalam hidup saya					
5	Aplikasi ini memudahkan saya dalam mencapai hal-hal yang saya inginkan					
6	Aplikasi ini menghemat waktu saya ketika					

	saya menggunakannya					
7	Aplikasi ini sesuai dengan kebutuhan saya					
8	Aplikasi ini sesuai dengan apa yang saya harapkan					
<i>Ease of Use</i>						
9	Aplikasi ini mudah digunakan					
10	Aplikasi ini praktis untuk digunakan					
11	Aplikasi ini mudah dipahami					
12	Aplikasi ini hanya memerlukan langkah – langkah singkat dalam menggunakannya					
13	Aplikasi ini dapat disesuaikan dengan kebutuhan					
14	Tidak ada kesulitan dalam menggunakan Aplikasi ini					
15	Saya dapat menggunakan tanpa instruksi tertulis					
16	Saya melihat Aplikasi ini sudah konsisten ketika digunakan					
17	Baik pengguna yang jarang maupun yang rutin akan suka menggunakan Aplikasi ini					
18	Saya dapat kembali dari kesalahan dengan cepat dan mudah					
19	Saya dapat menggunakan Aplikasi ini dengan					

	hasil setiap saya gunakan					
<i>Ease of Learning</i>						
20	Saya belajar menggunakan Aplikasi ini dengan cepat					
21	Saya mengingat penggunaan Aplikasi ini dengan mudah					
22	Penggunaan Aplikasi ini mudah dipelajari					
23	Saya mahir menggunakan Aplikasi ini dengan cepat					
<i>Satisfaction</i>						
24	Saya puas dengan Aplikasi ini					
25	Saya merekomendasikan Aplikasi ini kepada teman-teman					
26	Aplikasi ini menyenangkan untuk digunakan					
27	Aplikasi ini bekerja sesuai dengan yang saya inginkan					
28	Aplikasi ini sangat bagus					
29	Saya merasa harus menggunakan Aplikasi ini					
30	Aplikasi ini nyaman untuk digunakan					

Lampiran 13. *Expanded Use Case*

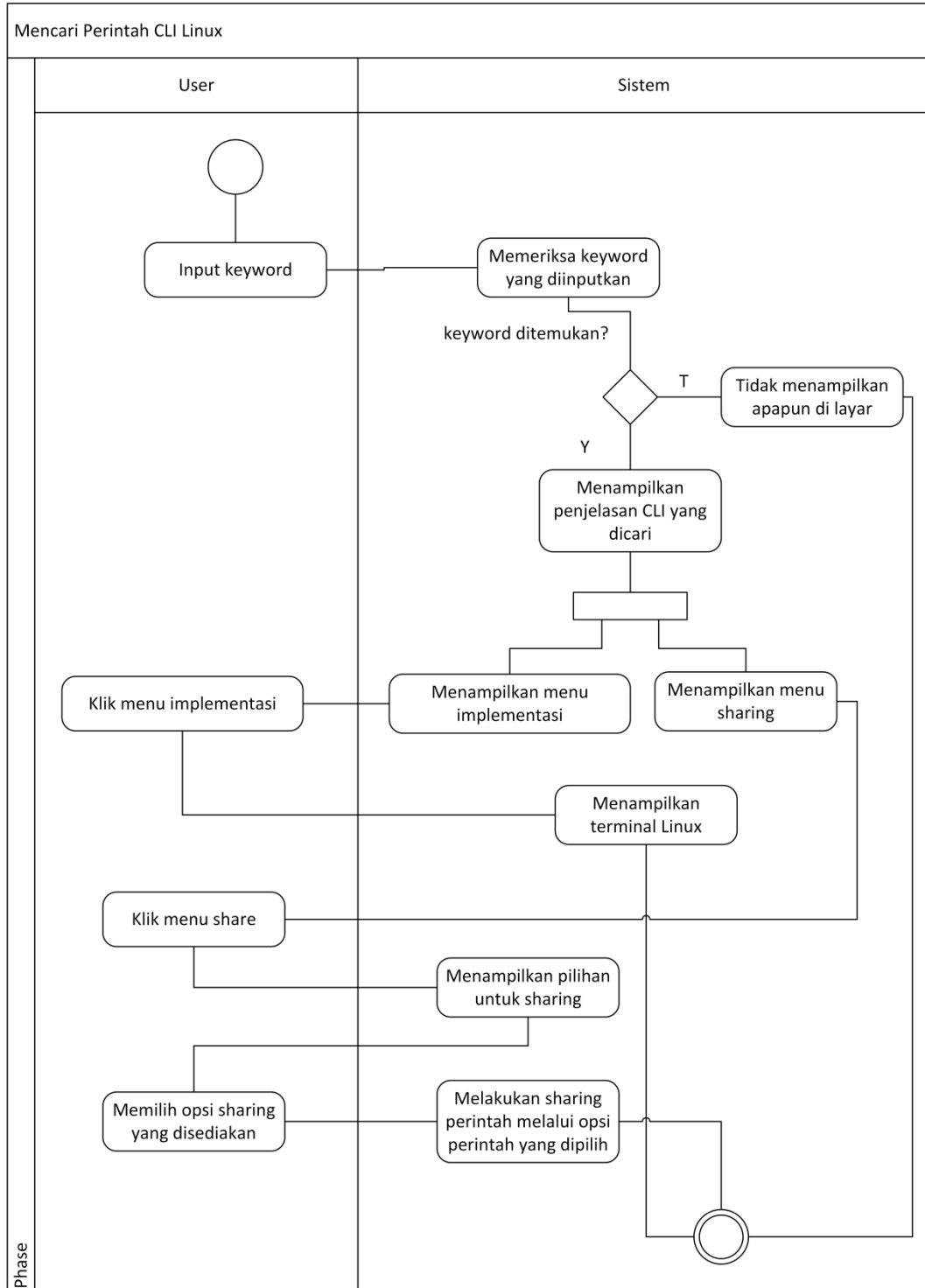
<i>Use case 1 : Melihat penjelasan CLI</i>	
No.	Deskripsi
1.	<i>User</i> memilih penjelasan <i>command line interface</i>
2.	Sistem menampilkan penjelasan <i>command line interface</i>
<i>Use case 2 : Mencari perintah CLI Linux</i>	
1.	<i>User</i> memasukkan <i>keyword</i> untuk menemukan perintah CLI Linux yang dicari.
2.	Sistem menampilkan penjelasan perintah CLI Linux sesuai dengan <i>keyword</i> yang dimasukkan.
3.	<i>Include fragment 2a: Implementasi perintah CLI Linux</i>
4.	<i>User</i> memilih untuk : Membagikan perintah yang dicari ke <i>user</i> lain : <i>variant 2.4a</i> .
<i>Fragment 2a : Implementasi perintah CLI Linux</i>	
1.	Sistem menampilkan halaman yang digunakan untuk <i>user</i> mengimplementasikan perintah CLI Linux yang dicari.
2.	<i>User</i> memasukkan perintah CLI.
3.	Sistem mengoreksi perintah yang diinputkan dan menampilkan <i>feedback</i> apakah perintah yang diinputkan <i>user</i> benar atau tidak.
4.	<i>User</i> menerima <i>feedback</i> .
<i>Variant 2.4a</i>	
1.	Sistem menampilkan perintah CLI Linux yang dicari.
2.	<i>User</i> memilih menu <i>sharing</i> .

3.	Sistem menampilkan pilihan <i>share</i> yang dapat dipilih <i>user</i> .
4.	<i>User</i> memilih salah satu pilihan <i>share</i> yang disediakan sistem.
5.	Sistem melakukan fungsi <i>sharing</i> .
<i>Use case 3 : Mencari perintah CLI Windows</i>	
1.	<i>User</i> memasukkan <i>keyword</i> untuk menemukan perintah CLI Windows yang dicari.
2.	Sistem menampilkan penjelasan perintah CLI Windows sesuai dengan <i>keyword</i> yang dimasukkan.
3.	<i>Include fragment 3a: Implementasi perintah CLI Windows.</i>
4.	<i>User</i> memilih untuk : Membagikan perintah yang dicari ke <i>user</i> lain : <i>variant 3.4a</i> .
<i>Fragment 3a : Implementasi perintah CLI Windows</i>	
1.	Sistem menampilkan halaman yang digunakan untuk <i>user</i> mengimplementasikan perintah CLI Windows yang dicari.
2.	<i>User</i> memasukkan perintah CLI.
3.	Sistem mengoreksi perintah yang diinputkan dan menampilkan <i>feedback</i> apakah perintah yang diinputkan <i>user</i> benar atau tidak.
4.	<i>User</i> menerima <i>feedback</i> .
<i>Variant 3.4a</i>	
1.	Sistem menampilkan perintah CLI Windows yang dicari.
2.	<i>User</i> memilih menu <i>sharing</i> .
3.	Sistem menampilkan pilihan <i>share</i> yang dapat dipilih <i>user</i> .

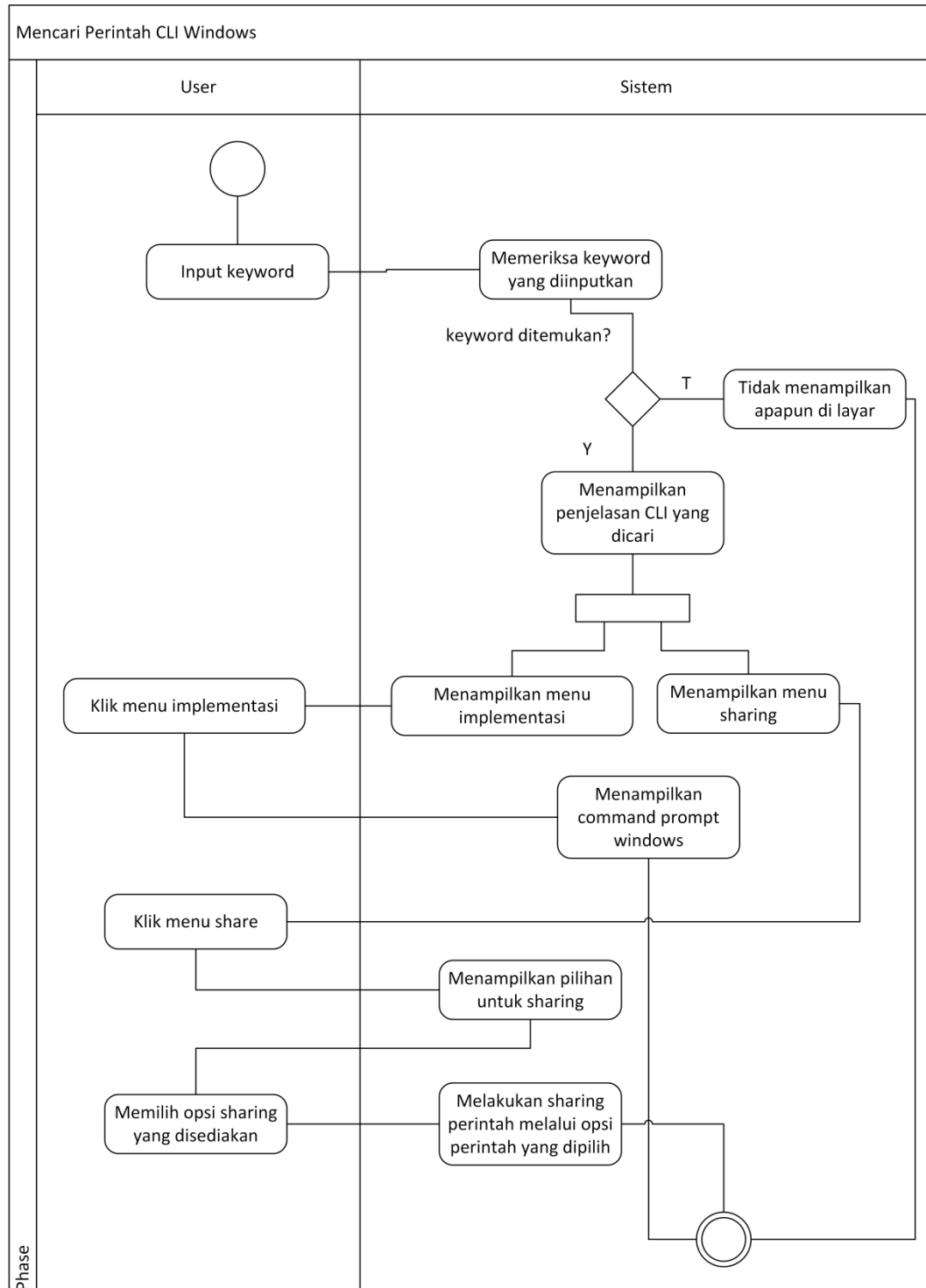
4.	<i>User</i> memilih salah satu pilihan <i>share</i> yang disediakan sistem.
5.	Sistem melakukan fungsi <i>sharing</i> .
<i>Use case 4 : Melihat informasi aplikasi</i>	
1.	<i>User</i> memilih informasi aplikasi.
2.	Sistem menampilkan informasi tentang aplikasi.
<i>Use case 5 : Melihat tata cara menggunakan aplikasi</i>	
1.	<i>User</i> memilih bantuan.
2.	Sistem menampilkan informasi tentang bantuan atau tata cara menggunakan aplikasi.
<i>Use case 6 : Mengerjakan kuis</i>	
1.	<i>User</i> memilih menu kuis.
2.	Sistem menampilkan kuis yang akan dikerjakan <i>user</i> .
3.	<i>User</i> mengerjakan kuis yang disediakan sistem.
4.	Sistem merekam jawaban yang diinputkan <i>user</i> .
5.	<i>Include fragment 5a: melihat skor</i>
<i>Fragment 5a : melihat skor</i>	
1.	<i>User</i> menyimpan jawaban yang telah dikerjakan.
2.	Sistem menampilkan skor yang diperoleh <i>user</i> .

Lampiran 14. Activity Diagram

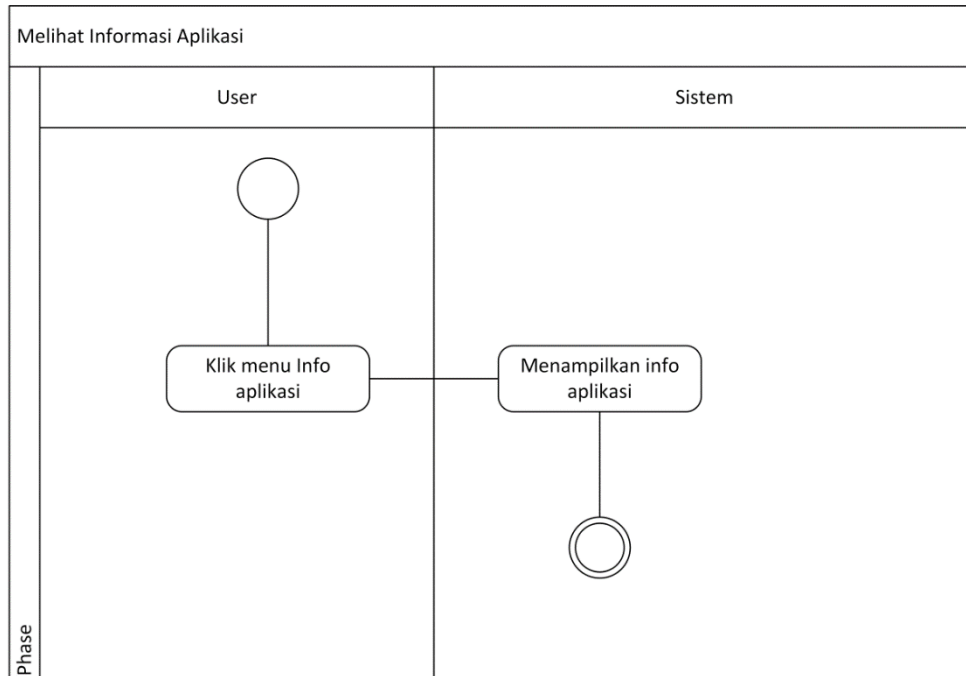
2. Mencari Perintah CLI Linux



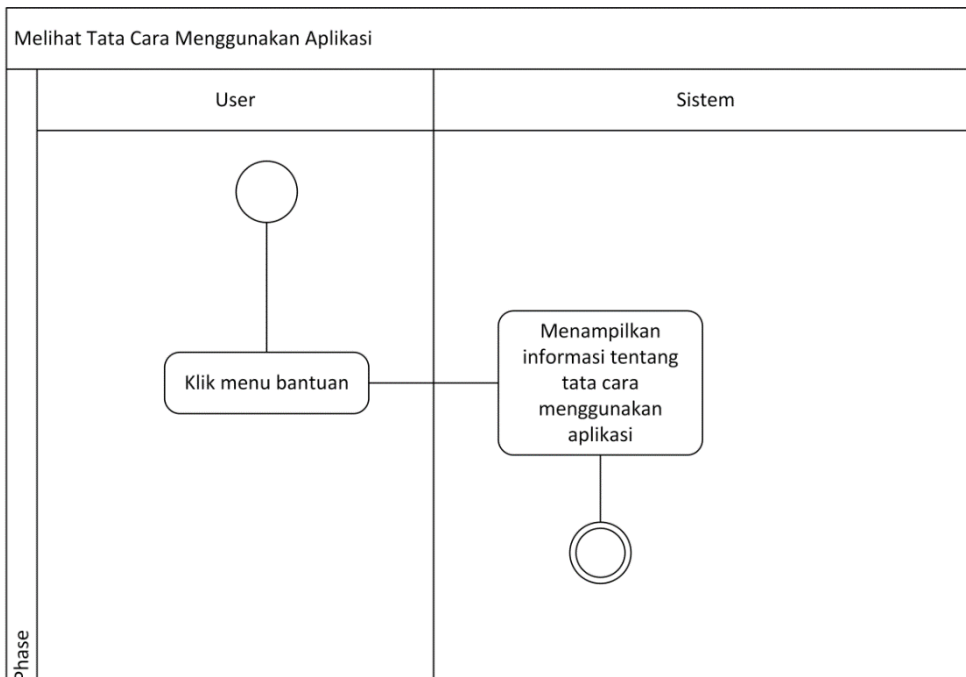
3. Mencari Perintah CLI Windows



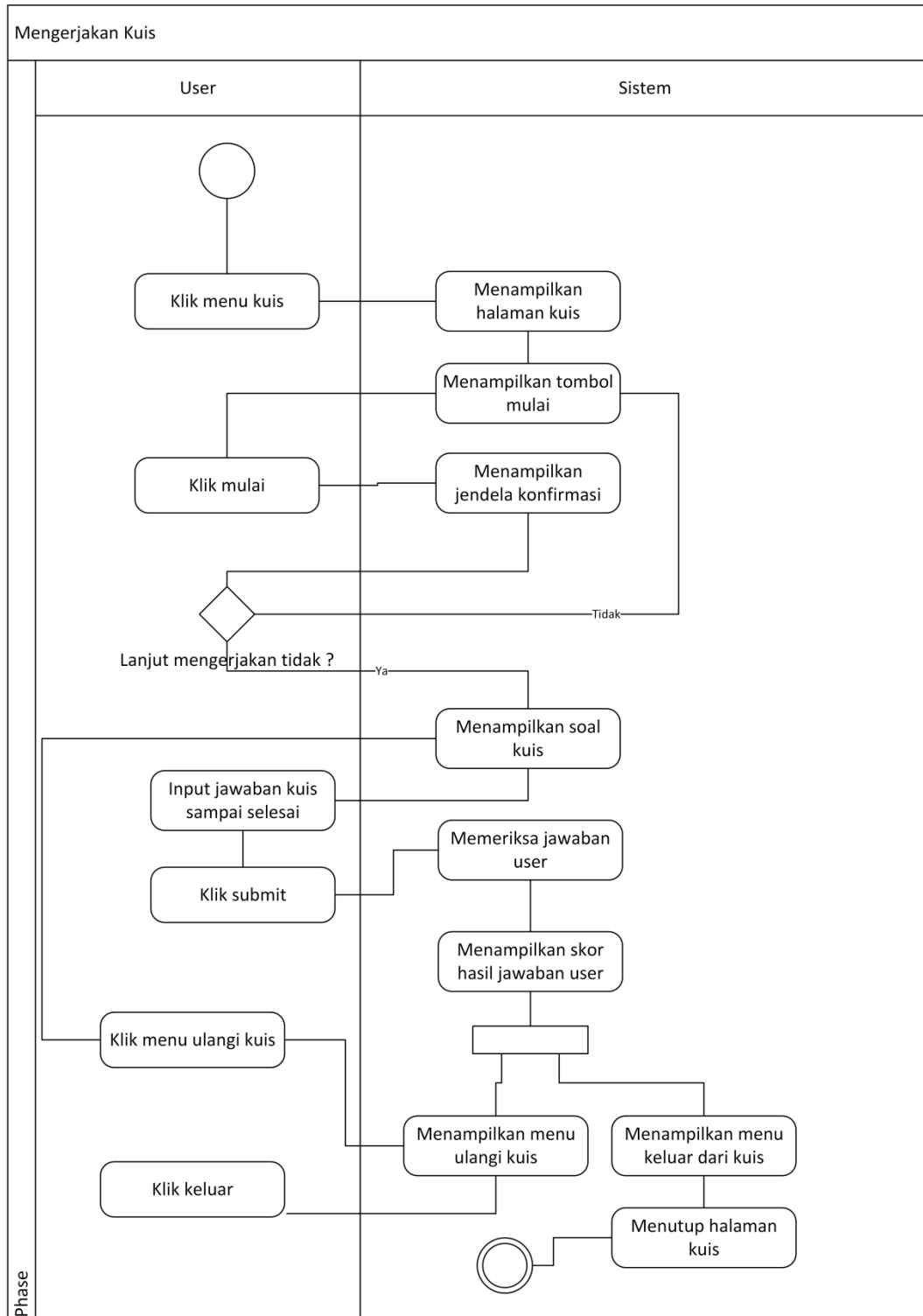
4. Melihat Informasi Aplikasi



5. Melihat Tata Cara Menggunakan Aplikasi

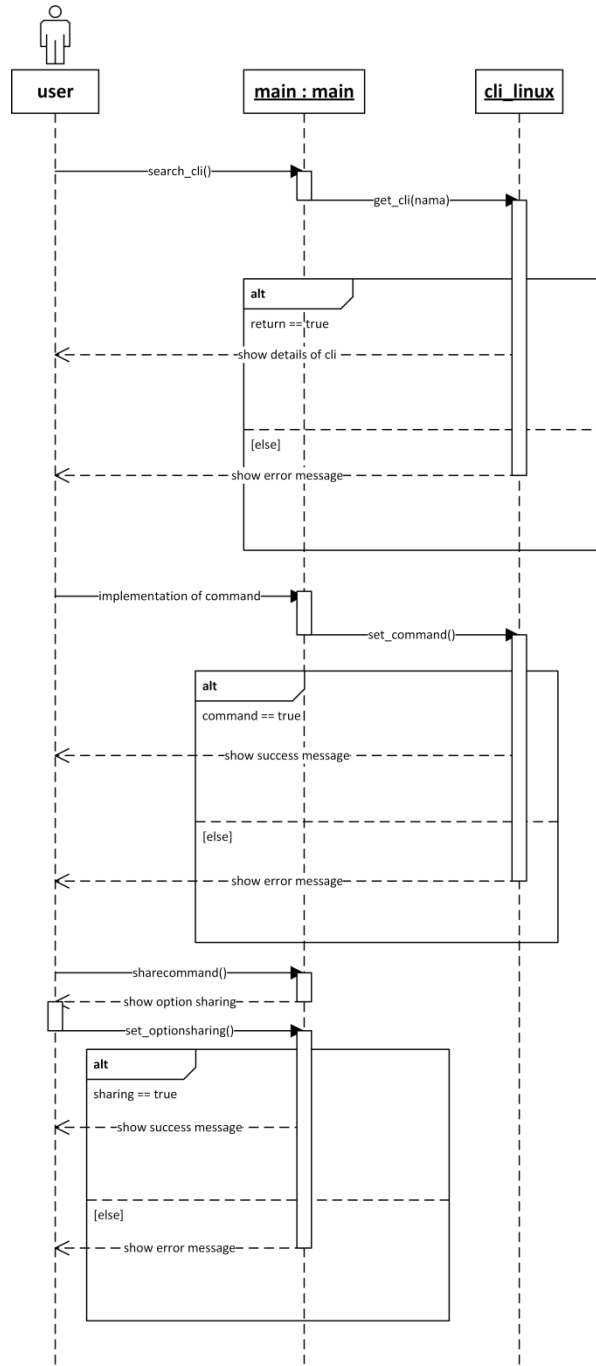


6. Mengerjakan Kuis

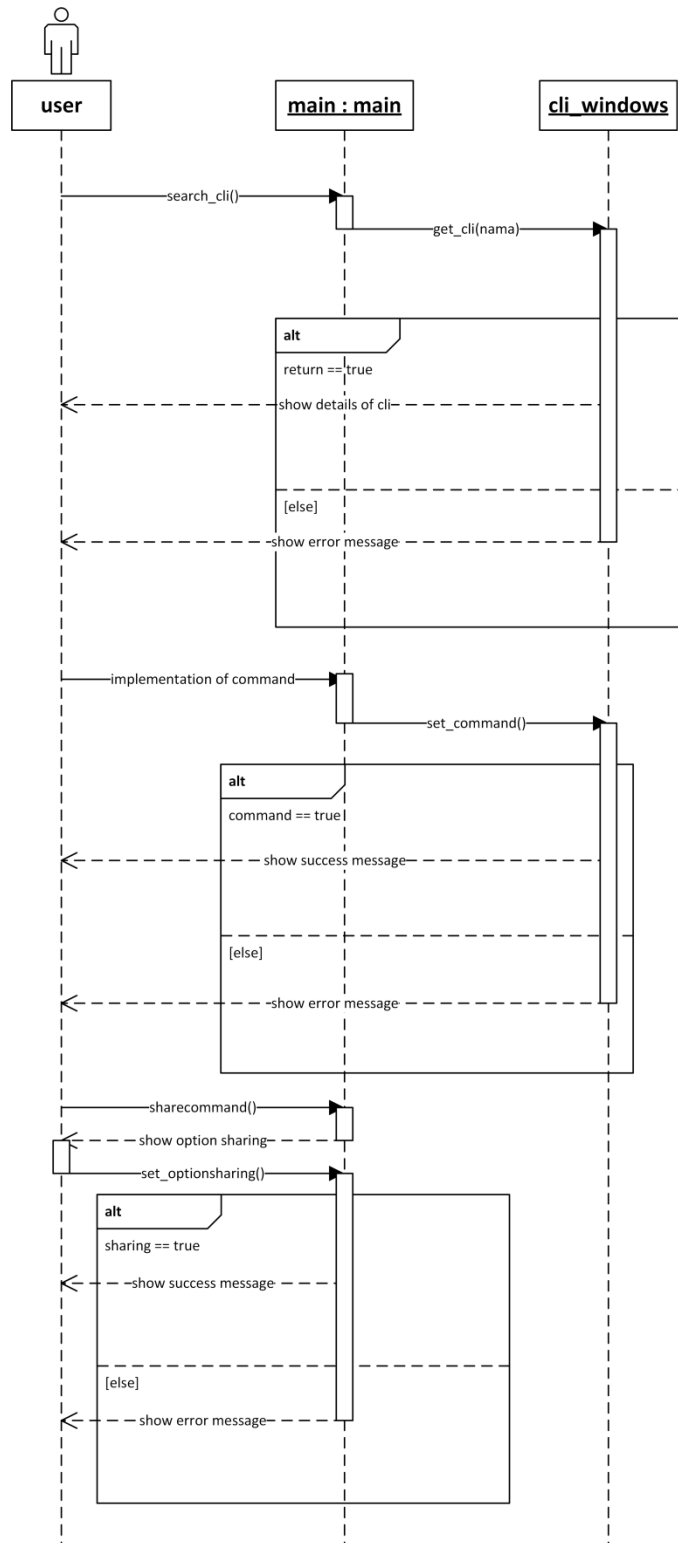


Lampiran 15. Sequence Diagram

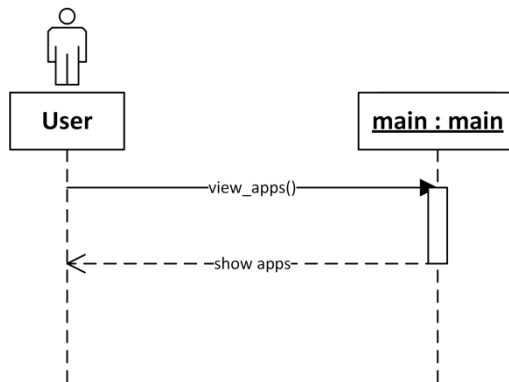
➤ Sequence diagram search, share, dan implementasi CLI Linux



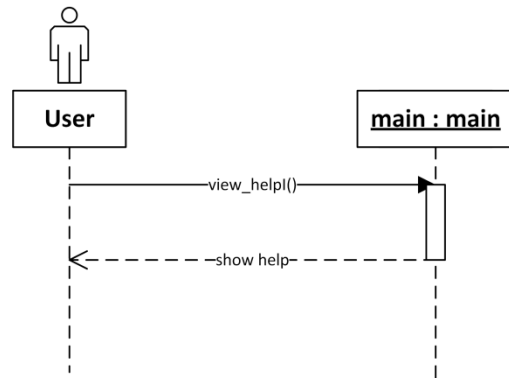
➤ *Sequence diagram search, share, dan implementasi CLI Windows*



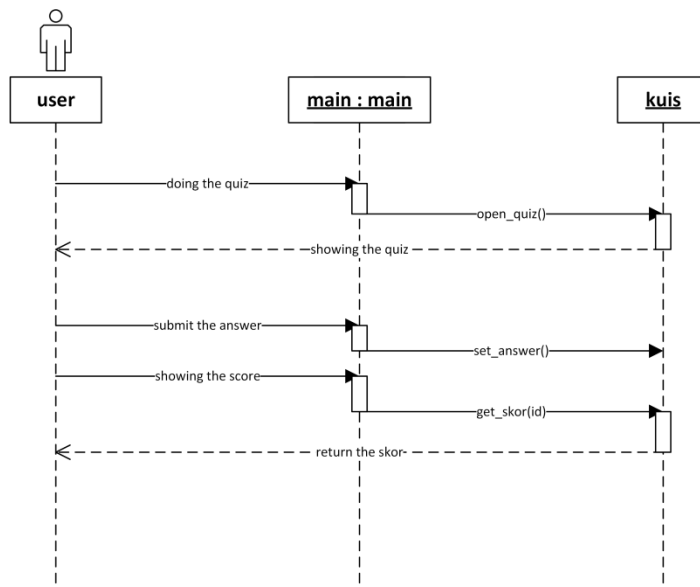
➤ *Sequence diagram* informasi tentang aplikasi



➤ *Sequence diagram* petunjuk aplikasi

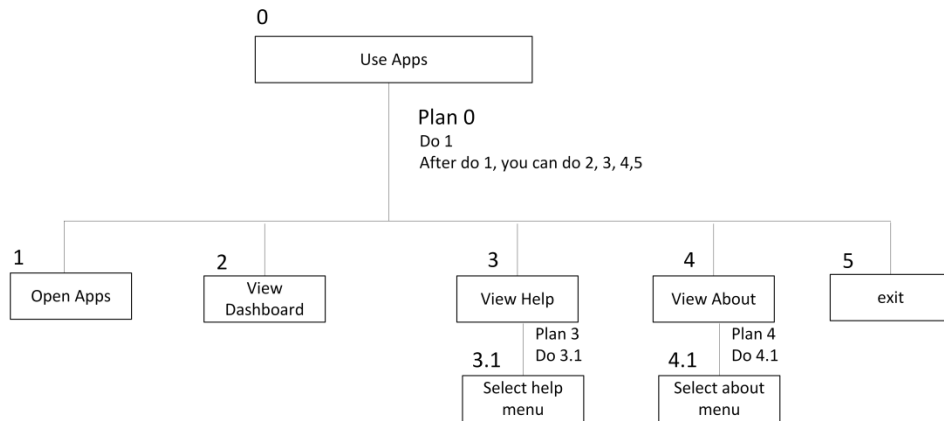


➤ *Sequence diagram* mengerjakan kuis

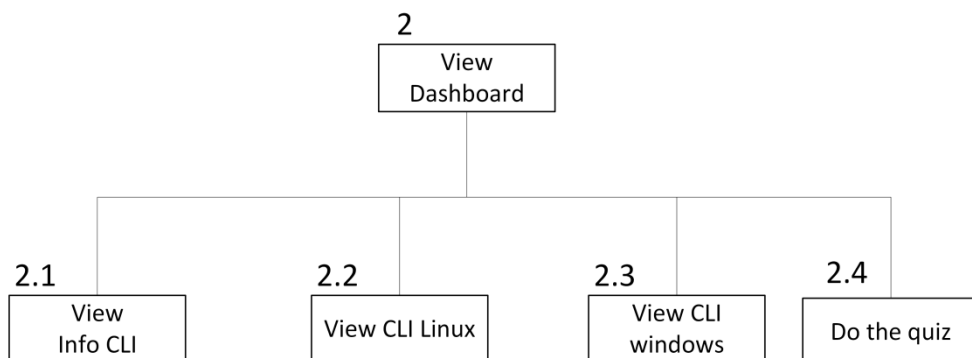


Berikut HTA akan dipecah per *activity* agar lebih mudah dipahami.

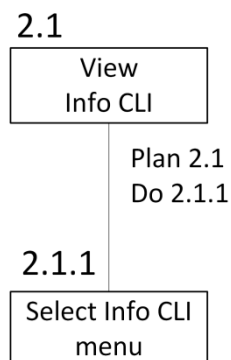
➤ Activity 0, menggunakan aplikasi



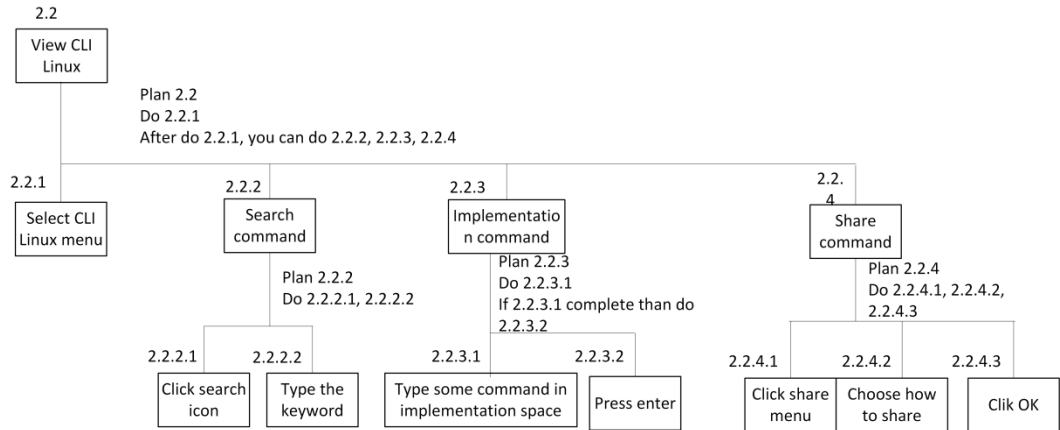
➤ Activity 2, View Dashboard dibagi dalam 4 task.



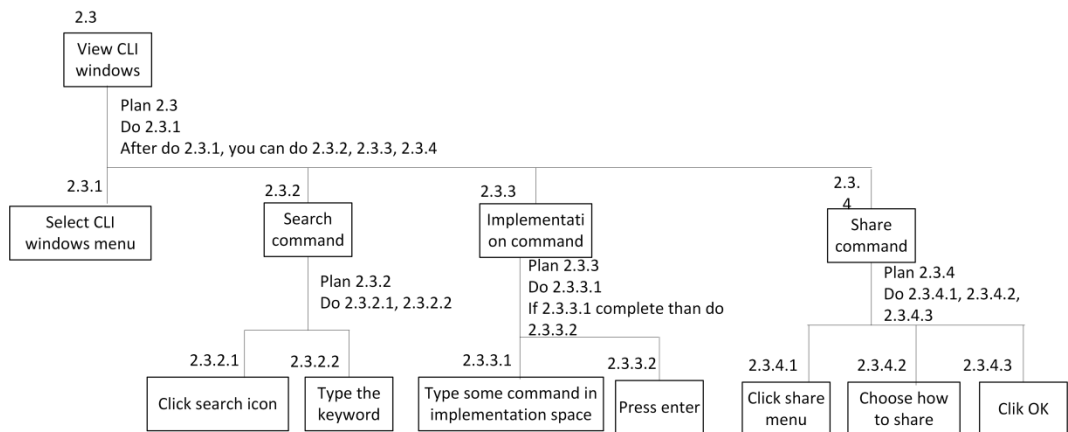
➤ Task activity 2.1, melihat informasi CLI



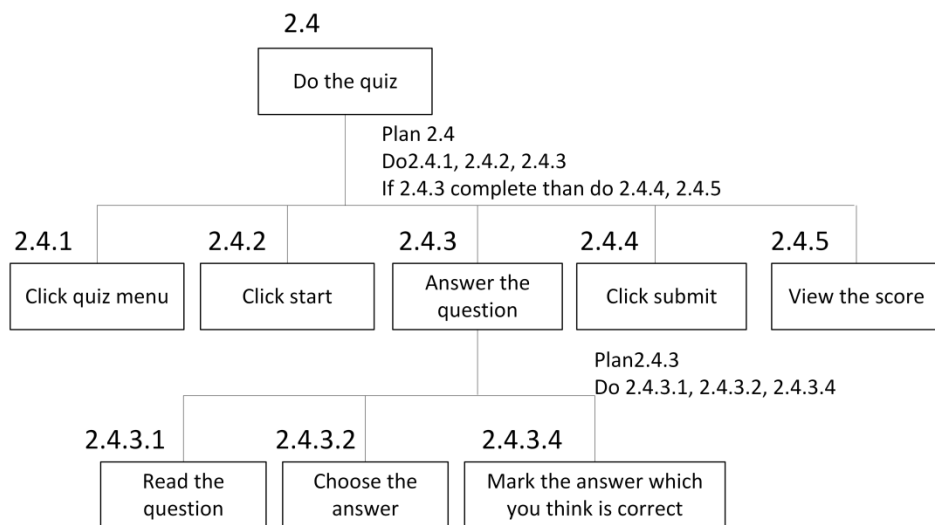
➤ *Task activity 2.2, melihat perintah CLI Linux*



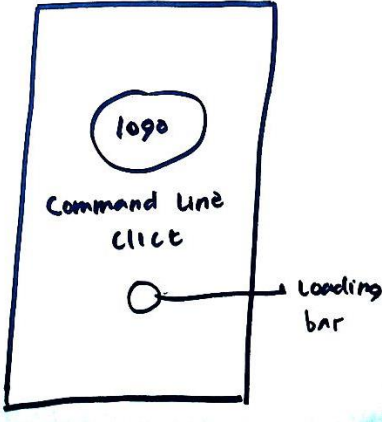
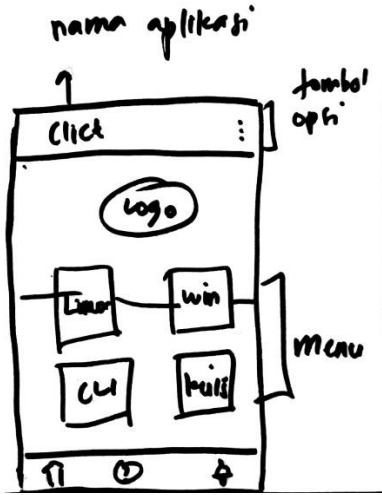
➤ *Task activity 2.3, melihat perintah CLI Windows*

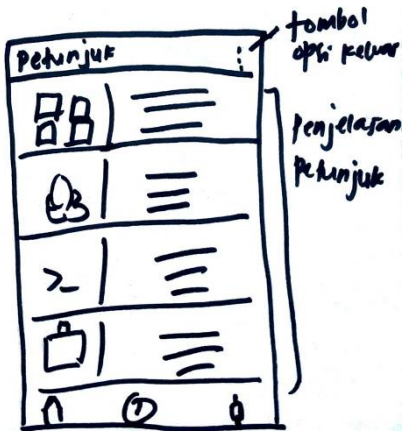



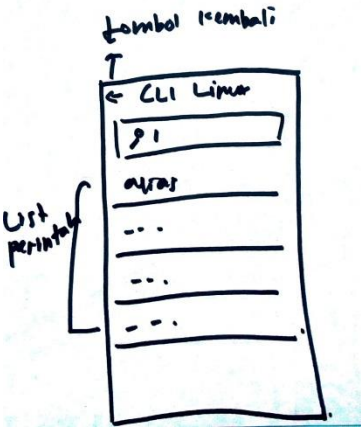
➤ *Task activity 2.4, Mengerjakan kuis*

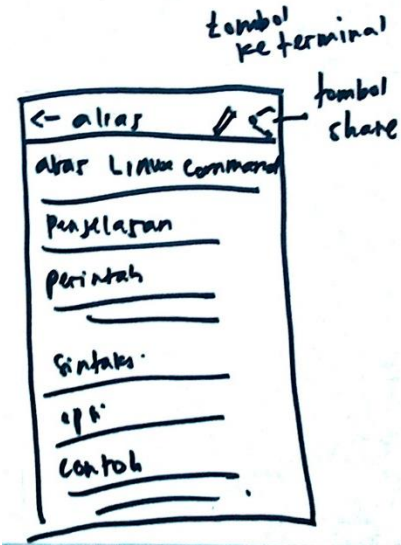



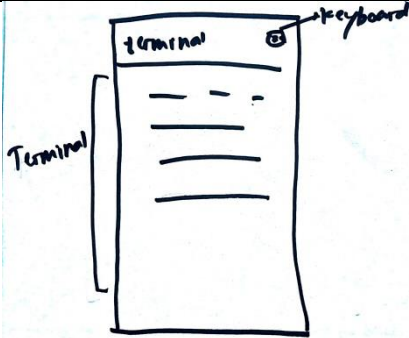
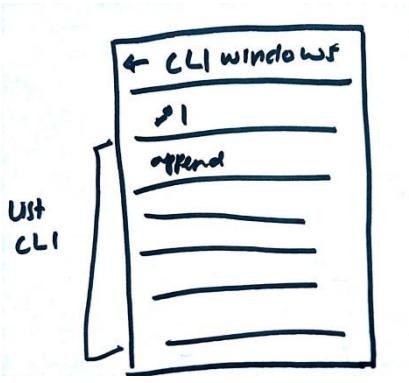
Lampiran 17. Hasil *Storyboard*

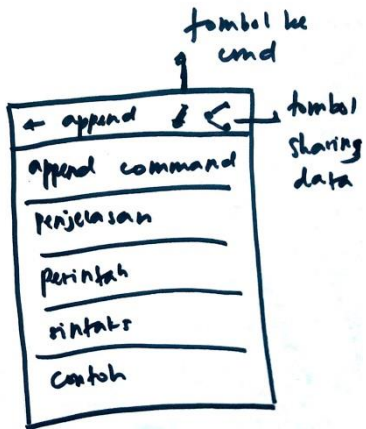

No.	Sketsa	Visual
1.	<p><i>Splashscreen</i></p> 	<p>Ketika aplikasi pertama kali dijalankan, akan tampil halaman <i>splash screen</i> sebelum masuk ke menu utama. Halaman ini berisi logo aplikasi dan nama aplikasi.</p>
2.	<p>Menu utama (beranda)</p> 	<p>Pada menu utama (beranda) terdapat logo, nama aplikasi, tombol opsi yang berfungsi untuk keluar aplikasi, menu utama aplikasi, tombol beranda, tombol petunjuk, dan tombol tentang. Menu utama terdiri dari : 1. Linux, 2. Windows, 3. <i>Command Line Interface</i>, 4. Kuis. Menu Linux berfungsi untuk membuka daftar</p>

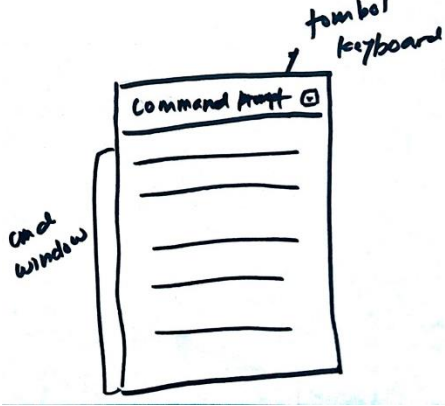
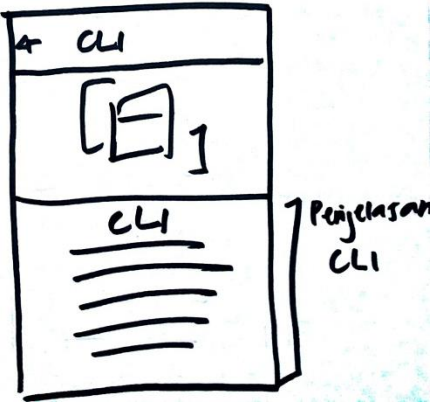
		<p>perintah CLI Linux. Menu Windows berfungsi untuk membuka daftar perintah CLI Windows. Menu <i>Command Line Interface</i> berfungsi untuk membuka halaman informasi tentang CLI. Menu kuis berfungsi untuk memulai kuis.</p>
3.	<p>Petunjuk</p> 	<p>Pada halaman petunjuk menjelaskan tentang fungsi-fungsi <i>icon</i> yang ada pada aplikasi. Dibagian kanan atas terdapat opsi menu yang berguna untuk keluar dari aplikasi. Dibagian bawah terdapat <i>bottomNavigation</i> yang berisi tombol beranda, petunjuk, dan tentang.</p>
4.	<p>Tentang</p>	<p>Pada halaman tentang terdapat logo, nama aplikasi, versi aplikasi,</p>

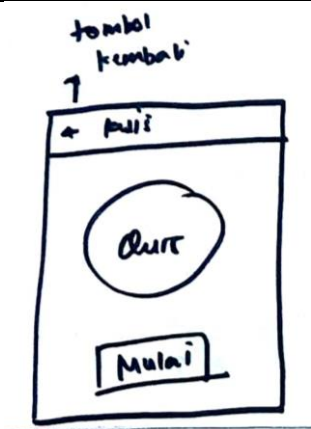

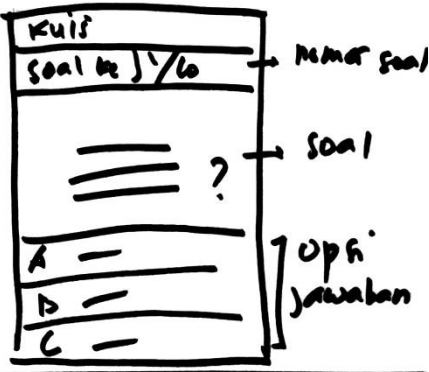
		<p>sumber materi yang digunakan, dan <i>copyright</i> pengembang. Dibagian kanan atas terdapat opsi menu yang berguna untuk keluar dari aplikasi. Dibagian bawah terdapat <i>buttomNavigation</i> yang berisi tombol beranda, petunjuk, dan tentang.</p>
5.	<p>CLI Linux</p> 	<p>Pada halaman CLI Linux terdapat daftar CLI Linux, tombol <i>search</i>, dan tombol <i>back</i>. Tombol <i>search</i> digunakan untuk mencari perintah CLI yang diinginkan. Tombol <i>back</i> digunakan untuk kembali ke halaman beranda.</p>
6.	<p>Detail Perintah CLI Linux</p>	<p>Pada halaman detail perintah berisi tentang penjelasan dari perintah yang dicari, meliputi : arti</p>

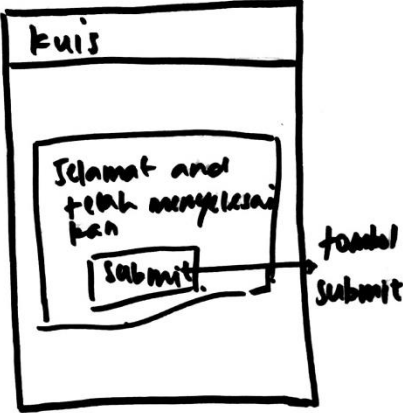
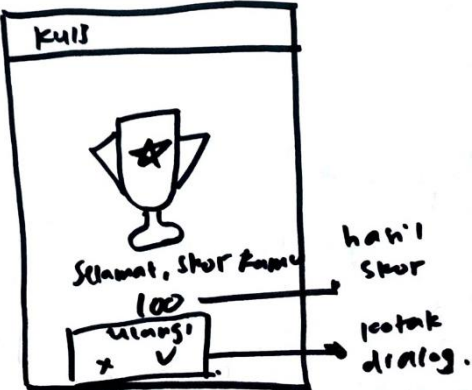
		<p>dan fungsi perintah, sintaks, opsi, dan contoh penerapan perintah. Dibagian kanan atas pada app bar terdapat tombol implementasi untuk membuka terminal Linux, tombol <i>share</i> untuk berbagi penjelasan perintah ke <i>user</i> lainnya, dan tombol <i>back</i> untuk kembali ke halaman sebelumnya.</p>
7.	<p><i>Share</i></p> 	<p>Ketika tombol <i>share</i> pada halaman detail CLI Linux di klik, akan muncul jendela “Berbagi dengan” dimana terdapat kumpulan aplikasi lain yang bisa digunakan untuk <i>sharing</i> data.</p>
8.	Implementasi CLI Linux	<p>Ketika tombol implementasi pada halaman detail CLI Linux di klik, akan tampil halaman terminal Linux. Halaman ini berisi emulator</p>

		<p>terminal Linux yang dapat digunakan untuk mengimplementasikan perintah yang dicari secara langsung pada OS Linux. Dibagian kanan atas pada app bar terdapat <i>icon</i> keyboard yang berfungsi untuk menampilkan <i>soft keyboard</i>.</p>
9.	<p>CLI Windows</p> 	<p>Pada halaman CLI Windows terdapat daftar CLI Windows, tombol <i>search</i>, dan tombol <i>back</i>. Tombol <i>search</i> digunakan untuk mencari perintah CLI yang diinginkan. Tombol <i>back</i> digunakan untuk kembali ke halaman beranda.</p>
10.	<p>Detail CLI Windows</p>	<p>Pada halaman detail perintah berisi tentang penjelasan dari perintah</p>

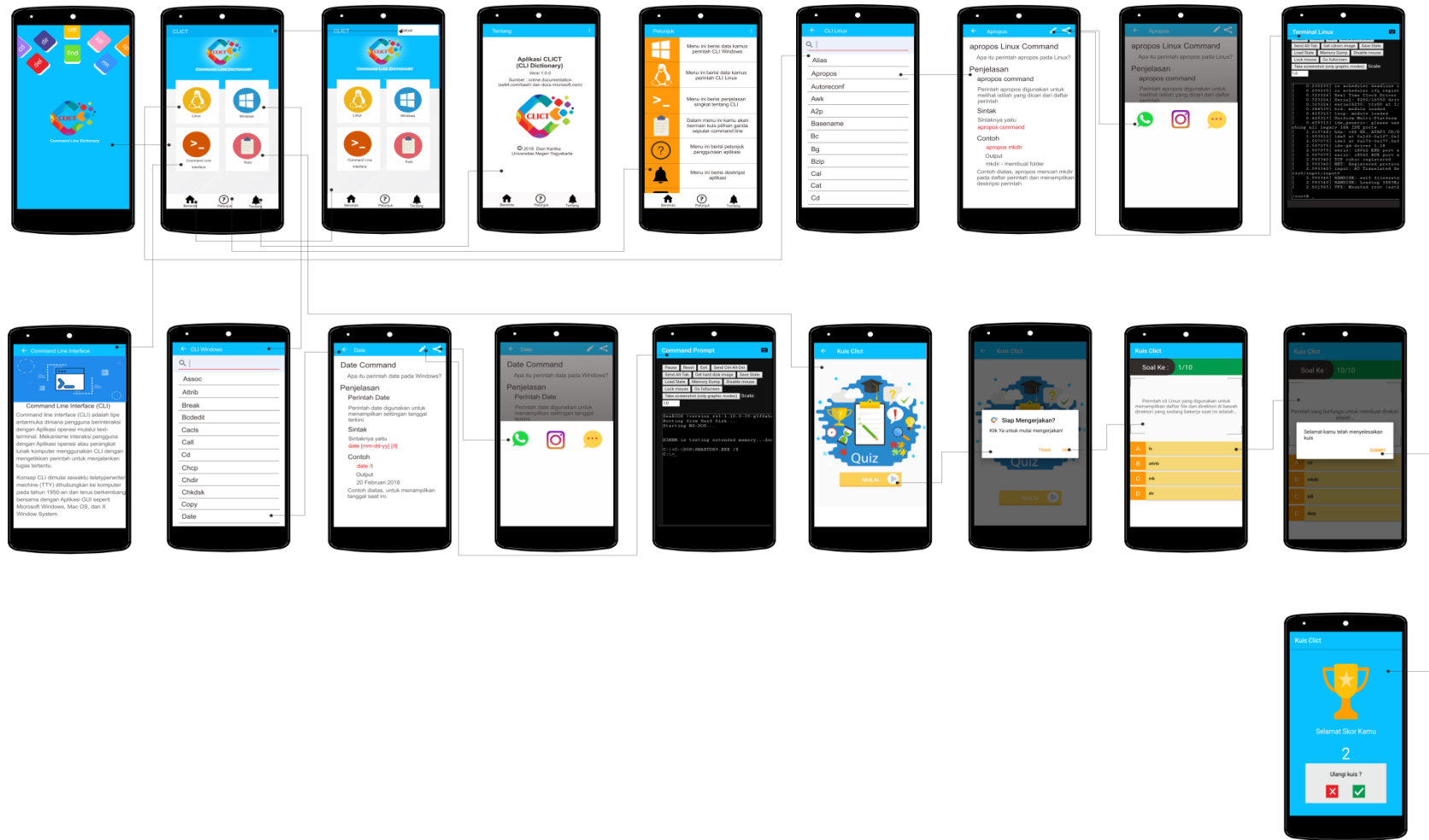
		<p>yang dicari, meliputi : arti dan fungsi perintah, sintaks, parameter, dan contoh penerapan perintah. Dibagian kanan atas pada app bar terdapat tombol implementasi untuk membuka <i>command prompt</i>, tombol <i>share</i> untuk berbagi penjelasan perintah ke <i>user</i> lainnya, dan tombol <i>back</i> untuk kembali ke halaman sebelumnya.</p>
11.	<p><i>Share</i></p> 	<p>Ketika tombol <i>share</i> pada halaman detail CLI windows di klik, akan muncul jendela “Berbagi dengan” dimana terdapat kumpulan aplikasi lain yang bisa digunakan untuk <i>sharing data</i>.</p>
12.	implementasi CLI Windows	Ketika tombol implementasi pada halaman detail CLI

		<p>Windows di klik, akan tampil halaman <i>command prompt</i> windows. Halaman ini berisi emulator <i>command prompt</i> yang dapat digunakan untuk mengimplementasikan perintah yang dicari secara langsung pada OS Windows. Dibagian kanan atas pada app bar terdapat <i>icon</i> keyboard yang berfungsi untuk menampilkan <i>soft keyboard</i>.</p>
13.	<p>Informasi <i>command line interface</i></p> 	<p>Halaman ini menjelaskan tentang apa itu <i>command line interface</i>. Pada bagian kiri atas terdapat tombol <i>back</i> untuk kembali ke halaman beranda.</p>
14.	<p>Kuis</p>	<p>Pada halaman ini terdapat logo kuis, tombol mulai</p>

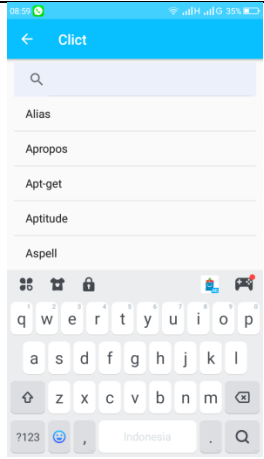
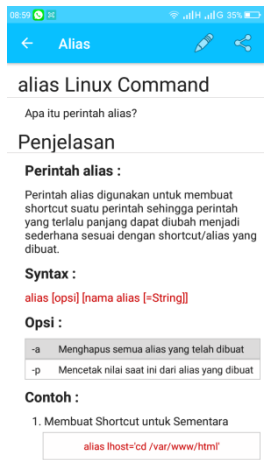
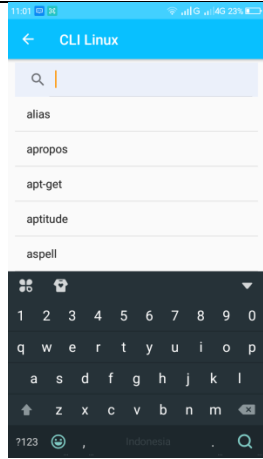
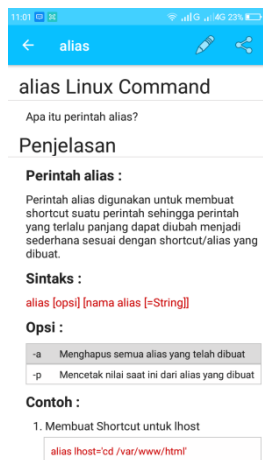
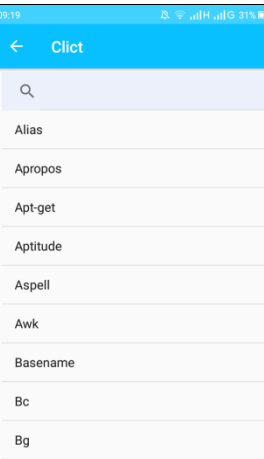
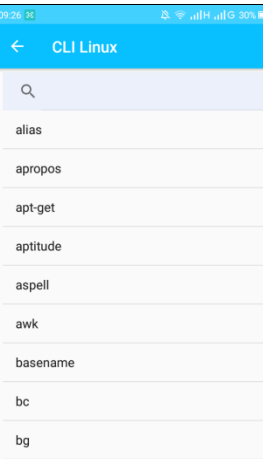
		<p>untuk memulai kuis, dan tombol <i>back</i> untuk kembali ke halaman beranda.</p>
15.	<p>Menu mulai kuis</p> 	<p>Ketika tombol mulai pada halaman kuis diklik, akan muncul jendela konfirmasi untuk lanjut mengerjakan kuis atau tidak. Jendela konfirmasi berisi tombol ya dan tidak.</p>
16.	<p>Soal</p> 	<p>Halaman kuis terdapat nomor soal, soal, dan opsi jawaban.</p>
17.	<p>Submit Kuis</p>	<p>Ketika soal kuis sudah terjawab semua maka akan muncul jendela konfirmasi</p>

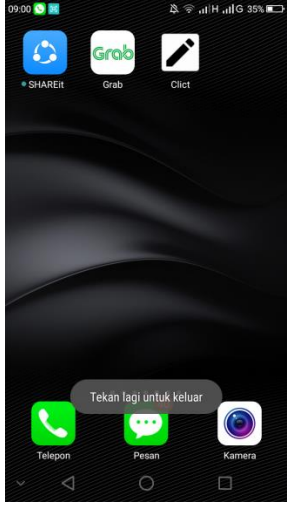
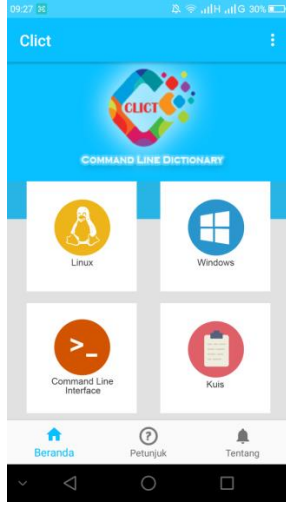
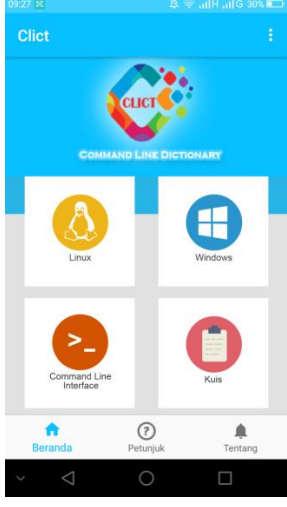

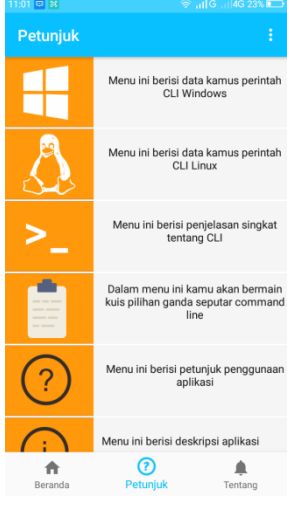

		<p>submit untuk melihat skor yang diperoleh.</p>
<p>18.</p>	<p>Skor Kuis</p> 	<p>Pada halaman ini terdapat Gambar piala, skor yang diperoleh, dan kotak konfirmasi apakah ingin mengerjakan kuis lagi atau tidak. Kotak konfirmasi tersebut terdapat tombol silang (X) dan centang (V). Silang (X) berfungsi untuk kembali ke halaman awal kuis. Tombol (V) berfungsi untuk menampilkan soal kuis lagi.</p>



Lampiran 17. Mockup Design User Interface



Lampiran 19. Hasil Perbaikan Aplikasi

No	Saran	Sebelum Perbaikan	Sesudah Perbaikan
1	<p>Nama-nama perintah CLI pada aplikasi sebaiknya ditulis sesuai dengan perintah aslinya saat diimplementasikan pada terminal yaitu menggunakan huruf kecil semua.</p>	 	 
2	<p>Ketika tombol back pada halaman list perintah CLI Linux dan Windows diklik lebih baik kalau kembali ke halaman sebelumnya tidak langsung keluar dari</p>		

	<p>aplikasi.</p>		
<p>3</p>	<p>Menu <i>command line interface</i> dihapus, kemudian penjelasan tentang CLI dimasukkan dalam menu tentang.</p>		
<p>4</p>	<p>Menu petunjuk dijabarkan lebih detail lagi.</p>		

5	<p>Menu tentang dilengkapi dengan keterangan tugas akhir skripsi, dosen pembimbing, universitas.</p>		
---	--	--	---

Lampiran 20. Hasil Uji Materi

**SURAT PERNYATAAN VALIDASI
UJI MATERI**

Saya yang bertanda tangan dibawah ini:

Nama : DARIYATI, S. Kom

NIP : 197812072014062002

Menyatakan bahwa instrumen penelitian TAS atas nama mahasiswa:

Nama : Dian Kartika Sari

NIM : 14520244002

Program Studi : PENDIDIKAN TEKNIK INFORMATIKA

Judul TAS : PENGEMBANGAN APLIKASI *MOBILE* PERINTAH
DASAR LINUX DAN WINDOWS BERBASIS
COMMAND LINE INTERFACE PADA *PLATFORM*
ANDROID

Setelah dilakukan kajian materi tersebut dapat dinyatakan:

Layak digunakan untuk penelitian

Layak digunakan dengan perbaikan

Tidak layak digunakan untuk penelitian yang bersangkutan dengan saran.

Saran: Materi sudah valid namun perlu
dikembangkan lagi.

Demikian agar dapat digunakan sebagaimana mestinya.

Bantul, Juni 2018

Validator,



Dariyati, S. Kom

NIP. 19781207 201406 2 002

Catatan:

Beri tanda ✓

Lampiran 21. Hasil Uji Usability

No Responden	Pertanyaan																														
	USEFULNESS					EASE OF USE					EASE OF LEARNING					SATISFACTION															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
1	5	5	5	5	4	5	5	5	5	4	4	5	4	3	4	5	5	4	4	5	5	4	4	5	5	5	5	5	5		
2	4	4	5	4	3	4	4	4	5	4	4	3	3	4	3	3	4	4	4	3	4	4	4	5	4	4	3	4	4		
3	5	5	4	4	4	4	3	4	4	3	4	3	4	4	4	4	5	5	5	5	4	4	4	5	5	5	4	4	4		
4	5	5	5	5	4	5	5	5	5	4	4	5	4	3	4	5	5	5	4	5	5	5	4	5	5	5	5	5	5		
5	5	5	5	5	4	5	5	5	5	4	4	5	4	3	4	5	5	5	4	5	5	5	4	5	5	5	5	5	5		
6	4	5	5	4	4	5	4	3	4	5	4	5	4	4	5	4	4	4	5	5	5	4	5	5	4	4	4	4	4		
7	5	5	5	5	5	4	5	5	5	5	5	5	5	4	5	5	5	5	5	5	4	5	5	5	5	5	5	5	5		
8	5	5	5	5	5	4	5	5	5	4	4	5	4	3	4	5	5	5	4	5	5	4	5	5	5	5	5	5	5		
9	4	4	5	3	2	5	3	3	5	5	5	3	5	4	4	4	3	3	3	3	4	4	4	4	5	3	3	5	5		
10	4	4	4	4	4	5	4	4	5	5	4	4	5	4	4	5	3	3	2	2	3	3	4	4	4	5	5	3	4		
11	4	4	5	4	4	4	4	3	4	4	4	4	3	5	3	4	4	4	4	5	5	4	4	4	4	4	4	4	4		
12	5	5	5	5	5	4	5	5	5	4	4	5	4	3	4	5	5	5	4	5	5	4	4	5	5	5	5	5	5		
13	4	4	4	4	4	4	5	5	5	4	5	5	4	3	4	4	5	5	4	5	5	4	4	5	5	5	5	5	5		
14	4	3	5	4	3	5	4	4	5	4	5	3	5	4	4	4	4	3	4	4	4	4	4	4	4	5	4	4	5		
15	5	4	5	4	4	4	5	5	5	4	4	5	4	3	4	4	4	5	5	4	5	5	4	4	5	5	5	5	5		
16	5	4	5	5	4	4	5	5	5	4	4	4	5	4	3	4	5	5	4	5	5	4	5	4	5	5	5	4	5		
17	4	4	4	5	4	4	4	4	5	5	4	4	4	4	5	4	4	4	4	5	5	4	5	4	4	4	4	4	4		
18	5	4	5	4	4	4	4	4	5	4	5	4	5	4	5	3	3	4	4	4	5	3	4	4	5	4	4	4	5		
19	5	5	4	4	4	4	4	4	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
20	5	5	5	5	5	4	4	4	4	5	5	4	4	4	4	4	5	5	4	5	5	4	4	5	5	5	5	5	5		
21	5	5	4	4	4	4	4	3	4	4	3	4	4	3	4	4	4	5	5	5	4	4	4	4	4	4	4	4	4		
22	5	5	4	4	4	4	4	3	4	4	3	4	4	3	4	4	4	4	5	5	4	4	4	4	4	4	4	4	4		
23	5	5	4	4	4	4	4	4	4	5	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
24	5	5	5	4	4	4	4	5	4	4	5	5	4	5	4	4	4	5	5	5	4	5	5	5	5	5	4	5	5		
25	4	5	4	4	5	4	4	4	5	5	5	4	5	4	4	4	4	4	5	5	5	5	5	5	5	5	4	5	5		
26	5	5	4	4	4	4	4	4	4	4	5	4	3	4	4	4	4	4	4	4	4	4	4	4	4	5	4	5	5		
27	5	5	4	4	4	4	4	4	5	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
28	5	5	4	4	4	4	4	4	4	5	4	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4		
29	5	5	5	5	5	4	5	5	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	5	3	4	4	4		
30	5	5	5	5	4	5	4	5	5	4	4	5	4	4	3	4	5	5	4	5	5	4	4	5	5	5	5	5	5		
Total SS	21	21	27	13	10	7	11	12	20	20	13	10	16	5	3	3	11	15	7	19	18	14	6	16	20	13	20	16	18	425	
Total S	9	8	3	16	17	23	18	12	10	10	14	20	9	20	18	24	16	12	21	8	10	15	23	14	9	9	15	10	13	418	
Total RG	0	1	0	1	2	0	1	6	0	0	3	0	5	5	9	3	3	3	1	2	2	1	1	0	1	1	2	0	1	0	54
Total TS	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	3	
Total STS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Total

Usefulness

- It helps me be more effective.
- It helps me be more productive.
- It is useful.
- It gives me more control over the activities in my life.
- It makes the things I want to accomplish easier to get done.
- It saves me time when I use it.
- It meets my needs.
- It does everything I would expect it to do.

Ease of Use

- It is easy to use.
- It is simple to use.
- It is user friendly.
- It requires the fewest steps possible to accomplish what I want to do with it.
- It is flexible.
- Using it is effortless.
- I can use it without written instructions.
- I don't notice any inconsistencies as I use it.
- Both occasional and regular users would like it.
- I can recover from mistakes quickly and easily.
- I can use it successfully every time.

Ease of Learning

- I learned to use it quickly.
- I easily remember how to use it.
- It is easy to learn to use it.
- I quickly became skillful with it.

Satisfaction

- I am satisfied with it.
- I would recommend it to a friend.
- It is fun to use.
- It works the way I want it to work.
- It is wonderful.
- I feel I need to have it.
- It is pleasant to use.

Lampiran 23. Dokumentasi Penelitian



