

**Implementasi *Software Center* Menggunakan Metode  
*Evolutionary Prototyping* Berbasis HTML5**

Skripsi

untuk memenuhi sebagian persyaratan mencapai derajat Sarjana S-1



disusun oleh :

**ESTU FARDANI**

**NIM. 09650004**

Kepada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA**

**YOGYAKARTA**

**2014**

## **LEMBAR PENGESAHAN**

## **SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR**

Hal : Permohonan

Lamp : -

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

di Yogyakarta

*Assalamu'alaikum wr. wb.*

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Estu Fardani

NIM : 09650004

Judul Skripsi : Implementasi *Software Center* Menggunakan Metode  
*Evolutionary Prototyping* Berbasis HTML5

sudah dapat diajukan kembali kepada Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Teknik Informatika.

Dengan ini kami mengharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

*Wassalamu'alaikum wr. wb.*

Yogyakarta, April 2014  
Pembimbing

Dr. Imam Riadi M.Kom  
NIY: 60020397

## **PERNYATAAN KEASLIAN SKRIPSI**

Yang bertanda tangan di bawah ini :

Nama : Estu Fardani  
NIM : 09650004  
Program Studi : Teknik Informatika  
Fakultas : Sains dan Teknologi

Dengan ini saya menyatakan bahwa skripsi dengan judul “**Implementasi Software Center Menggunakan Metode *Evolutionary Prototyping* Berbasis HTML5**” tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 7 April 2014

Meterai

Estu Fardani

NIM. 09650004

## **MOTTO**

*“Keajaiban hanya terjadi ketika kita tak berharap,  
ketika kita tak disini lagi”*

*(Goenawan Muhamad)*

*“Talk is cheap, show me the code”*

*(Linux Trovald)*

*“Urip kui harus due itungan”*

*(Ngadikun, Ayahku)*

*“Baek-baek merantau”*

*(Sulastri, Ibuku)*

## **PERSEMBAHAN**

*Allah SWT, yang selalu melimpahkan banyak kemaafan dan kenikmatan sehingga skripsi ini dapat terselesaikan pada waktu yang tepat.*

*Nabi besar Muhammad SAW, semoga shalawat senantiasa terhatur kepadamu.*

*Emak Sulastris dan Bapak Ngadikun, terimakasih atas doa, semangat dan dorongan . Semoga aku mampu membalas semua baktimu.*

*Kakak kembarku dan adik tunggalku , Dina, Dini, Dita. Tunggu abang balek ye..*

*Dek Jauharoh Pratami, terimakasih atas kesabarannya menyemangati saya yang terbiasa malas. :) Coklatnya double yak.*

*Tim Pengembang BlankOn. Inilah yang bisa aku sumbangkan sebagai baktiku untuk Indonesia.*

## KATA PENGANTAR

*Alhamdulillah* rabbil'alamiin, Puji syukur penulis panjatkan kepada Allah SWT karena dengan restu-Nya pelaksanaan dan penyusunan skripsi yang berjudul “Implementasi *Software Center* Menggunakan Metode *Evolutionary Prototyping* Berbasis HTML5” dapat diselesaikan sebagai persyaratan menyelesaikan Sarjana Strata Satu (S1) Jurusan Teknik Informatika, Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Kalijaga Yogyakarta.

Penulisan skripsi ini tidak terlepas dari bantuan dan dukungan pelbagai pihak. Oleh karena itu, ucapan terimakasih penulis sampaikan kepada :

1. Emak Sulastri dan Bapak Ngadikun tercinta, yang senantiasa mendoakan, memberikan motivasi dan semangat kepada penulis,
2. Bapak Prof. Dr. Akh Minhaji, selaku Dekan Fakultas Sains & Teknologi UIN Sunan Kalijaga,
3. Bapak Agus Mulyanto, M.Kom, selaku Ketua Program Studi Teknik Informatika Fakultas Sains & Teknologi UIN Sunan Kalijaga,
4. Bapak Dr. Imam Riadi, M.Kom selaku Dosen Pembimbing yang telah membantu terselesaikannya skripsi ini,
5. Bapak MDAMT, Pak Haris, Pak Aji, Pak Dotovr, Pak Dhio, Pak Jasoet, Pak Syai Miff, teman-teman pengembang BlankOn, demi kemajuan sumberdaya Indonesia. Maaf jika sering merepotkan dan bertanya *mulu*,
6. Adek Jauharoh atas kesabaran memotivasi, memberi semangat dan asupan energi kepada penulis,
7. Sahabat Linuxer Jogja, Dedy Hariyadi, Iwan Setiyawan, M. Iqbal

Jalaluddin, Koespradono, Kevin dan teman-teman lain, *yuk bikin reuni* lagi,

8. Sahabat-sahabat seperjuangan Teknik Informatika Reguler 09, Pasa, Aziz, Ridho, Arum, Ismi, Ulin, Joko, Isnani, Pulung, Udin, Latief dan semua rekan-rekan. Terimakasih atas 5 tahun bersama,
9. Teman-teman Infest Jogja, Mas Khayat, Mas Ibad, Mas Lamuk, Mas Afif, terimakasih atas ruang belajar yang diberi,
10. Kawan-kawan milis vala-gnome. *It is the best place to be bigger,*
11. Semua pihak yang tidak bisa sebutkan satu per satu, terima kasih atas segala bantuannya.

Semoga Allah SWT memberikan balasan yang berlipat ganda dari segala kebaikan yang telah dilakukan. Penulis menyadari bahwa skripsi ini masih belum sempurna. Oleh karena itu, kritik dan saran yang membangun penulis harapkan sebagai masukan kearah yang lebih baik lagi.

Demikian penulis berharap agar laporan penelitian ini dapat berguna untuk kita semua.

Yogyakarta, 7 April 2014

Penulis



## DAFTAR ISI

LEMBAR PENGESAHAN.....	ii
SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR.....	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
MOTTO.....	v
PERSEMBAHAN.....	vi
KATA PENGANTAR.....	vii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
DAFTAR LISTING.....	xviii
INTISARI.....	xix
ABSTRACT.....	xx
BAB 1 Pendahuluan.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
1.6 Keaslian Penelitian.....	4
BAB 2 Tinjauan Pustaka dan Landasan Teori.....	6
2.1 Tinjauan pustaka.....	6
2.2 Landasan Teori.....	9

2.2.1 Open Source Software.....	9
2.2.2 Linux.....	10
2.2.3 Distribusi Linux.....	10
2.2.4 Linux BlankOn.....	10
2.2.5 Unified Modeling Language(UML).....	11
2.2.5.1 Use Case Diagram.....	11
2.2.5.2 Activity Diagram.....	13
2.2.5.3 Class Diagram.....	16
2.2.5.4 Sequence Diagram.....	18
2.2.6 Grafik User Interface (GUI).....	20
2.2.7 Mock-up.....	20
2.2.8 Vala.....	21
2.2.9 HTML5.....	22
2.2.10 CSS3.....	24
2.2.11 AJAX.....	24
2.2.12 DOM.....	25
2.2.13 WebKit.....	25
2.2.14 WebKitGTK+.....	26
2.2.14.1 Cara Kerja WebKitGTK+.....	28
2.2.15 JavaScript dan JavaScriptCore.....	28
2.2.16 JQuery.....	29
2.2.17 Twitter Bootstrap.....	29
2.2.18 AngularJS.....	30

2.2.19 Repository Linux.....	31
2.2.20 Packaging Tools.....	31
2.2.21 Evolutionary Prototyping.....	31
2.2.22 Evolus Pencil.....	32
BAB 3 Metode Pengembangan Sistem.....	34
3.1 Pengumpulan Data.....	34
3.2 Kebutuhan Sistem.....	35
3.3 Metodologi Pengembangan.....	35
3.3.1 Identifikasi Konsep.....	36
3.3.2 Desain dan Implementasi Prototype.....	36
3.3.3 Perbaikan Prototype Hingga Diterima.....	37
3.3.4 Penyelesaian dan Pelepasan Prototype.....	37
3.4 Metodologi Pengujian Sistem.....	37
3.4.1 Alpha Testing.....	38
3.4.2 Beta testing.....	38
BAB 4 Analisis dan Desain Sistem.....	40
4.1 Analisis Sistem.....	40
4.1.1 Analisis Masalah.....	40
4.1.1.1 Analisis Aplikasi Sejenis.....	40
4.1.2 Analisa dan Kebutuhan Fungsional.....	42
4.1.2.1 Kebutuhan Fungsional.....	42
4.1.2.2 Kebutuhan Non-Fungsional.....	43
4.2 Perancangan Sistem.....	43

4.2.1 Desain Sistem.....	43
4.2.2 Rancangan Antarmuka.....	44
4.2.2.1 Rancangan Antarmuka Halaman Utama.....	45
4.2.2.2 Rancangan Antarmuka Detail.....	46
4.2.2.3 Rancangan Antarmuka Ubah Repository.....	46
4.2.2.4 Rancangan Antarmuka Halaman Pilih Kategori.....	47
4.2.2.5 Rancangan Halaman Kategori.....	48
4.2.2.6 Rancangan Antarmuka Halaman Perbaharui.....	49
4.2.2.7 Rancangan Antarmuka Halaman Aplikasi.....	50
4.2.2.8 Rancangan Antarmuka Halaman Pencarian.....	51
4.2.3 Use Case Diagram.....	52
4.2.4 Activity Diagram.....	54
4.2.4.1 Diagram Aktifitas Cari Aplikasi.....	55
4.2.4.2 Diagram Aktifitas Manajemen Repository.....	55
4.2.5 Sequence Diagram.....	57
4.2.5.1 Squance Diagram Cari Aplikasi Berdasarkan Nama.....	57
4.2.5.2 Squance Diagram Cari Aplikasi berdasarkan Kategori.....	57
4.2.5.3 Squance Diagram Manajemen Repository.....	58
BAB 5 Implementasi dan Pengujian Sistem.....	59
5.1 Implementasi.....	59
5.1.1 Persiapan Lingkungan Kerja.....	59
5.1.1.1 Memasang Evolus Pencil.....	59
5.1.1.2 Memasang WebkitGTK+ dan dependensi.....	60

5.1.1.3 Memasang Git dan Kunci SSH.....	61
5.1.1.4 Memasang Vala dan Vala Compiler.....	62
5.1.2 Implementasi Antarmuka.....	63
5.1.2.1 Halaman Awal.....	63
5.1.2.2 Halaman Kategori.....	64
5.1.2.3 Halaman Pencarian.....	64
5.1.2.4 Halaman Manajemen Repository.....	65
5.1.2.5 Halaman Detail Aplikasi.....	66
5.1.3 Implementasi Kode.....	67
5.1.3.1 Implementasi Script Utama.....	67
5.1.3.2 Implementasi Script Inisiasi Halaman Utama.....	68
5.1.3.3 Implementasi Script Kumpulan Fungsi.....	69
5.1.3.4 Implementasi Script Perbaharui Paket.....	70
5.1.3.5 Implementasi Script Manajemen Repository.....	71
5.1.3.6 Implementasi Script Memasang dan Menghapus Aplikasi.....	72
5.1.3.7 Implementasi Script Compiler.....	72
5.2 Pengujian Sistem.....	73
5.2.1 Pengujian Alpha.....	74
5.2.2 Pengujian Beta.....	75
BAB 6 Hasil dan Pembahasan.....	78
6.1 Proses Pengembangan Software Center.....	78
6.1.1 Analisis.....	78
6.1.2 Implementasi.....	78

6.2 Hasil Pengujian Sistem.....	81
6.2.1 Hasil Pengujian Alpha Testing.....	81
6.2.2 Hasil Pengujian Beta Testing.....	83
6.2.3 Perhitungan Hasil Beta Testing.....	85
BAB 7 Kesimpulan.....	87
7.1 Kesimpulan.....	87
7.2 Saran.....	87
DAFTAR PUSTAKA.....	89
LAMPIRAN.....	91
Lampiran 1. main.vala.....	91
Lampiran 2: page.vala.....	91
Lampiran 3: app.vala.....	92
Lampiran 4: Echo Repository.....	104
Lampiran 5: index.html.....	105
Lampiran 6: app.js.....	111
Lampiran 7: appjson.js.....	112
Lampiran 8: controller.js.....	118
DAFTAR RIWAYAT HIDUP.....	123

## DAFTAR GAMBAR

Gambar 2.1: Aktor.....	12
Gambar 2.2: Use Case.....	12
Gambar 2.3: Relationship dalam UML.....	13
Gambar 2.4: Action State.....	14
Gambar 2.5: Transition.....	15
Gambar 2.6: Initial State.....	15
Gambar 2.7: Final State.....	15
Gambar 2.8: Decision State.....	16
Gambar 2.9: Synchronization Bar.....	16
Gambar 2.10: Class.....	17
Gambar 2.11: Association.....	17
Gambar 2.12: Composition.....	17
Gambar 2.13: Depedency.....	18
Gambar 2.14: Aggregation.....	18
Gambar 2.15: Genertalization.....	18
Gambar 2.16: Object.....	19
Gambar 2.17: Actor.....	19
Gambar 2.18: Lifeline.....	19
Gambar 2.19: Activation.....	20
Gambar 2.20: Message.....	20
Gambar 2.21: Cara Kerja Webkit.....	28

Gambar 2.22. Tampilan Evolus Pencil.....	33
Gambar 3.1: Evolutionary Prototyping.....	36
Gambar 4.1: Desain Sistem.....	44
Gambar 4.2: Rancangan Antarmuka Halaman Depan.....	45
Gambar 4.3: Rancangan Antarmuka Halaman Detail.....	46
Gambar 4.4: Rancangan Antarmuka Halaman Ubah Repository.....	47
Gambar 4.5: Rancangan Antarmuka Halaman Pilih Kategori.....	48
Gambar 4.6: Halaman Antarmuka Kategori.....	49
Gambar 4.7: Rancangan Antarmuka Halaman Perbaharui.....	50
Gambar 4.8: Rancangan Antarmuka Halaman Aplikasi.....	51
Gambar 4.9: Rancangan Antarmuka Halaman Pencarian.....	52
Gambar 4.10: Diagram Use case.....	53
Gambar 4.11: Diagram Aktifitas.....	54
Gambar 4.12: Diagram Aktifitas Cari Aplikasi.....	54
Gambar 4.13: Diagram Aktifitas Manajemen Repository.....	56
Gambar 4.14: Sequence Diagram Cari Aplikasi.....	57
Gambar 4.15: Sequence Diagram Pilih Kategori.....	58
Gambar 4.16: Sequence Diagram Manajemen Repository.....	58
Gambar 5.1: Implementasi Halaman Awal.....	63
Gambar 5.2: Implementasi Halaman Kategori.....	64
Gambar 5.3: Implementasi Halaman Pencarian.....	65
Gambar 5.4: Implementasi Halaman Manajemen Repository.....	66
Gambar 5.5: Implementasi Halaman Detail Aplikasi.....	67



## DAFTAR TABEL

Tabel 2.1: Perbedaan Penelitian Terdahulu.....	8
Tabel 3.1: Rancangan Item Uji Alpha Testing.....	38
Tabel 4.1: Perbandingan Aplikasi Software Center Sejenis.....	41
Tabel 5.1: Perintah Git Sederhana.....	62
Tabel 5.2: Rencana Pengujian.....	74
Tabel 5.3: Pengujian Usabilitas Aplikasi.....	76
Tabel 5.4: Pengujian Fungsionalitas Aplikasi.....	76
Tabel 6.1. Daftar Penguji Ahli.....	82
Tabel 6.2. Hasil Pengujian Alpha Testing.....	82
Tabel 6.3. Daftar Penguji Beta.....	83
Tabel 6.4: Pengujian Usabilitas Aplikasi.....	84
Tabel 6.5: Pengujian Fungsionalitas Aplikasi.....	84

## DAFTAR LISTING

Listing 5.1: Pemasangan Evolus Pencil.....	60
Listing 5.2: Pemasangan WebKitGTK+.....	60
Listing 5.3: Memasang Git.....	61
Listing 5.4: Generate Kunci SSH.....	61
Listing 5.5: Pemasangan Vala Compiler.....	62
Listing 5.6: Script main.vala.....	68
Listing 5.7: Script page.vala.....	69
Listing 5.8: Script app.vala.....	69
Listing 5.9: Script Perbaharui Repository.....	70
Listing 5.10: Script Mengganti Repository (vala).....	71
Listing 5.11: Script Mengganti Repository (javascript).....	71
Listing 5.12: Script Memasang dan Menghapus Aplikasi.....	72
Listing 5.13: Script Compiler.....	72
Listing 5.14: Menjalakan File Binari Software Center.....	73

# **Implementasi *Software Center* Menggunakan Metode *Evolutionary Prototyping* Berbasis HTML5**

Estu Fardani<sup>1</sup>

09650004

<sup>1</sup> Universitas Islam Negeri Sunan Kalijaga

surel: estu@jogja.linux.or.id

## **INTISARI**

*Software Center* adalah aplikasi wajib yang harus dimiliki oleh sistem operasi, baik sistem operasi konvensional maupun pada perangkat bergerak. *BlankOn* sebagai sistem operasi produksi dalam negeri harusnya memiliki aplikasi ini, namun dalam kenyataannya aplikasi ini belum tersedia. Trend pengembangan aplikasi mengerucut kepada teknologi HTML5 sebagai teknologi yang mudah diadaptasi hingga ke ranah desktop. Penelitian ini mencoba membuat *software center* berbasis HTML5 sebagai solusi di *BlankOn*.

Penelitian ini menggunakan metode *evolutionary prototyping* memungkinkan pengembangan aplikasi berdasarkan contoh produk yang akan dibuat. Kemudian hasil *prototyping* diimplementasikan dalam bahasa pemrograman. Setiap tahap implementasi akan dilakukan koreksi untuk mendapatkan fitur dan fungsi yang diinginkan. Proses pengembangan akan selesai jika *prototyping* yang dikerjakan telah memenuhi fitur yang diharapkan. Tahapan penelitian terbagi dalam beberapa langkah yakni analisis aplikasi sejenis, perancangan sistem, implementasi dan pengujian sistem.

Berdasarkan hasil pengujian fungsionalitas *software center*, didapatkan presentase 100% setuju sistem dapat berjalan dengan baik. Dari hasil pengujian dari segi *usability software center*, diperoleh kesimpulan bahwa responden sangat setuju sebanyak 80% dan setuju sebanyak 20%. Jadi dapat disimpulkan bahwa aplikasi *software center* dapat berjalan dengan baik dan lancar.

**Kata Kunci :** *Vala, Webkit, HTML5, Software Center, BlankOn*

# **Implementation of Software Center Using *Evolutionary Prototyping Method HTML5-Based***

Estu Fardani<sup>1</sup>

09650004

<sup>1</sup> Universitas Islam Negeri Sunan Kalijaga

email: estu@jogja.linux.or.id

## **ABSTRACT**

The Software Center is a required application in all operating system, both conventional or mobile devices. BlankOn as the local operating system should have this application but in reality this application is not yet available. HTML5 is a trend of programming because this technology easy to adapted for desktop utilization. This research tries to make software center HTML5-based as solution for BlankOn.

System development methods used in this study is evolutionary prototyping. It possible to develop application base on prototype of software. Next step is implement prototype to language programming. Every progress will be correction to get better features. Progress will be finished if prototype had all features what needs. Stages of research is divided into several steps. first analysis similar applications, second design system, third implementation and testing of the system.

Based on the results of functionality testing software center, the percentage obtained 100% agree the system can run well. From the results usability testing software center, it is concluded that the respondents strongly agreed that as much as 80% and 20% agree. It can be concluded that the software center application can run well and smoothly.

**Keywords :** *Vala, Webkit, HTML5, Software Center, BlankOn*

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Linux BlankOn merupakan sebuah distribusi Linux turunan Debian yang dikembangkan oleh Yayasan Penggerak Linux Indonesia (YPLI) dan Tim Pengembang BlankOn. Distribusi ini dirancang sesuai dengan kebutuhan pengguna komputer umum di Indonesia. Linux BlankOn sudah dipakai luas dengan pangsa utama mahasiswa, pelajar, guru, umum, hingga kalangan bisnis sekalipun. Linux BlankOn didesain dengan tetap mengedepankan budaya Indonesia, mulai dari penggunaan bahasa Indonesia, pengenalan aksara, hingga penggunaan bahasa daerah. Linux BlankOn juga mampu berjalan di komputer spesifikasi rendah yang masih banyak beredar di masyarakat Indonesia. Menurut data Distrowatch (12/3/2014) Linux BlankOn menempati posisi 37 sebagai distribusi Linux memiliki aktifitas tinggi di seluruh dunia dan merupakan satu-satunya distro di Asia Tenggara yang tercatat di DistroWatch. Menciptakan sistem operasi yang baik merupakan target dari pengembang manapun. Sisi ini mencakup bermacam-macam, mulai dari kemudahan mendapatkan informasi perangkat lunak, kemudahan memasang, serta kemudahan dalam penggunaan, termasuk kemudahan memasang perangkat lunak.

*Software Center* memiliki fungsi untuk mempermudah manajemen perangkat lunak, yaitu menambah dan menghapus perangkat lunak yang ada, keberadaannya diperlukan dalam semua sistem operasi baik desktop maupun perangkat bergerak. Aplikasi ini mulai digunakan pada Linux OpenSUSE, Ubuntu, Mac OSX,

Android, iOS, BlackBerry hingga Windows. Akan tetapi aplikasi ini belum ada di Linux BlankOn. Pemasangan aplikasi di Linux BlankOn saat ini bisa dilakukan dengan cara tradisional yaitu: pertama dengan bantuan konsol (terminal), kekurangannya adalah hanya bisa dilakukan jika mengetahui nama paket aplikasi yang ingin dipasang. Cara kedua dengan menggunakan *Manajer Paket Synaptic* sebuah manajemen paket-paket aplikasi. Meski sudah menggunakan *GUI (grafik user interface)* namun tetap hanya bisa dilakukan jika mengetahui nama paket aplikasi yang ingin dipasang dan terbatas penggunaannya. Sehingga mempersulit pengguna biasa untuk memasang aplikasi.

Aplikasi sejenis yang telah ada adalah *Ubuntu Software Center* dari distribusi Ubuntu. Aplikasi ini berlisensi *OpenSource* sehingga memungkinkan diterapkan pada Linux BlankOn. Namun karena aplikasi ini sudah menjadi *trademark* Ubuntu, dibutuhkan proses modifikasi lebih lanjut. Proses ini membutuhkan waktu dan tidak menjamin akan berjalan baik sehingga, muncul pemikiran bagaimana jika membuatnya mulai dari awal.

Trend aplikasi yang berkembang belakangan ini mengerucut pada penggunaan teknologi HTML5. Hal ini dikarenakan kemampuan adaptasi yang baik disegala platform, kemudahan dalam mempercantik desain dengan bantuan CSS3, tampilan yang responsif akibat dukungan javascript yang lebih baik dan kelebihan-kelebihan HTML5 lainnya.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas dapat dirumuskan permasalahan yang akan diselesaikan dalam penelitian ini adalah:

1. Bagaimana melakukan analisis dan mengembangkan *prototype Software Center* di Linux BlankOn?
2. Bagaimana mengimplementasikannya dalam bahasa pemrograman HTML5?

## 1.3 Batasan Masalah

Berdasarkan rumusan masalah dan dengan maksud agar pembahasan dan penyusunan penelitian dapat dilakukan secara terarah dan tercapai sesuai dengan yang diharapkan maka perlu ditetapkan batasan – batasan dari masalah yang dihadapi. Adapun batasan-batasan masalah pada penelitian ini adalah sebagai berikut :

1. Aplikasi ini hanya berjalan pada Sistem Operasi Linux BlankOn 9.0 Suroboyo 64-Bit,
2. Aplikasi ini hanya menampung perangkat lunak yang tersedia pada *repository* Linux BlankOn,
3. Kategori aplikasi dibatasi hanya 4 jenis, perkantoran, Internet, audio video, aksesoris, masing-masing kategori hanya menyediakan 4 sampel aplikasi,
4. Aplikasi masih merupakan produk contoh (*prototype*).
5. Sumber *repository* hanya dibatasi tiga tautan (Arsip BlankOn, Repo UGM, Kambing UI)

## 1.4 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut :

1. Melakukan analisis dan mengembangkan aplikasi yang dapat membantu pengguna untuk manajemen perangkat lunak yang ada di Linux BlankOn,
2. Mengimplementasikan *Software Center* di Linux BlankOn menggunakan bahasa pemrograman HTML5.

## 1.5 Manfaat Penelitian

Dengan adanya penelitian ini, diharapkan :

1. Memberi kemudahan pengguna Linux BlankOn untuk manajemen perangkat lunak yang ada dan terpasang.
2. Dapat memberikan solusi bagi pengembang BlankOn agar menjadi distribusi Linux yang lebih baik.

## 1.6 Keaslian Penelitian

Pengembangan *Software Center* sudah pernah dilakukan sebelumnya. Pada tahun 2005 penelitian dimulai oleh Matthew Paul Thomas dengan membuat prototipe "*Ubuntu Software Center*". *Software Center* ini nantinya diharapkan bisa membantu pengguna untuk memasang, membeli dan menghapus aplikasi di Ubuntu. Matthew Paul Thomas merupakan bagian tim *Canonical*, perusahaan yang mengembangkan dan penyedia dukungan komersial distribusi Linux Ubuntu. Penelitian serupa pernah dilakukan oleh Matthias Klumpp pada tahun 2012. Penelitian dilakukan dengan menerjemahkan *Ubuntu Software Center* agar



mampu berjalan di distribusi *Debian* dan *Fedora*. Bahasa yang digunakan adalah python dan C.

Penulis berkeyakinan bahwa pengembangan Sofrware Center di distribusi Linux BlankOn dengan penggunaan HTML5 sebagai bahasa pemograman belum pernah dilakukan.

## BAB 2

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Penelitian yang dilakukan oleh Matthew Paul Thomas (Paul Thomas, 2005) dengan membuat *prototype* “*Ubuntu Software Center*” pada distribusi Linux Ubuntu berbasis desktop menggunakan bahasa pemrograman python. *Software Center* ini diharapkan bisa membantu pengguna untuk memasang dan menghapus aplikasi di Ubuntu. Matthew Paul Thomas merupakan bagian tim *Canonical*, perusahaan yang mengembangkan dan penyedia dukungan komersial distribusi Linux Ubuntu.

Matthias Klumpp pada tahun 2012 melakukan penerjemahan *Ubuntu Software Center* agar mampu berjalan di distribusi *Debian* dan *Fedora*. Bahasa yang digunakan adalah python dan C (Klumpp, 2012). Penelitian ini menggunakan PackageKit sebuah tool manajemen paket aplikasi yang mampu berjalan di pelbagai distribusi Linux. Kekurangan dari penelitian ini, packagekit belum mampu menjawab tantangan perbedaan platform dan distribusi Linux (Hughes, 2008).

Selain dikembangkan oleh beberapa individu, *Software Center* sudah dikembangkan oleh beberapa perusahaan sejak lama sesuai kebutuhan masing-masing sistem operasi, diantaranya : *AppStore*, *Software Center* milik Apple Inc. *Software Center* ini berfungsi menampung aplikasi yang dibuat untuk sistem operasi OSX dan iOS. Berisi aplikasi baik yang gratis maupun yang berbayar. Kelebihan yang ada, harga memiliki nominal rupiah, dan tergantung pada lokasi

pengguna serta integrasi dengan akun apple yang dibuat pengguna (Hilderbrand and Bode, 2013).

*Google Play, Software Center* milih Google untuk menampung aplikasi untuk sistem operasi android. *Software Center* ini terintegrasi dengan akun Google. Memiliki versi mobile dan web dan tersedia sesuai rilis sistem operasi android dari 2.3 hingga versi terbaru 4.0.4 (Salim, 2013).

*BlackBerryAppWorld Software Center* milih Resech In Motion (RIM), pengembang sistem operasi *BlackBerryOS*. *Software Center* ini menampung aplikasi berbayar maupun gratis untuk perangkat mobile bersistem operasi *BlackBerry* (RIM, 2010).

*YaST (Yet another Setup Tools) Software Center* milik openSUSE sebuah distribusi Linux berbasis RedHAT memiliki kelebihan sebagai satu-satu software yang dibutuhkan pengguna openSUSE untuk manajemen aplikasi hingga pengaturan umum sistem operasi, mulai dari aplikasi, jaringan, tampilan hingga konfigurasi lain sebagainya (Mac-Vicar, 2008).

HTML5 memiliki beberapa kelebihan dalam menjawab tantangan pengembangan aplikasi. Diantaranya mampu menjawab tantangan perbedaan platform, kemudahan mempercantik tampilan dengan CSS dan kemudahan interaksi yang baik dengan javascript sehingga, pengguna menemukan *user experience* yang maksimal (Pratami, 2013).

Penelitian-penelitian yang terdahulu mengenai *Software Center* dikembangkan dengan bahasa dan konsep yang berbeda. Sebagai perbandingan dapat dilihat pada Tabel 2.1.

Tabel 2.1: Perbedaan Penelitian Terdahulu

No	Nama	Judul	Kelebihan	Kekurangan
1	Apple Inc, 2008	App Store	Integrasi dengan seluruh layanan Apple	Hanya bisa berjalan untuk sistem operasi milik Apple
2	Google Inc, 2008	Google Play	Tersedia di semua platform android, dan ponsel pintar android. Integrasi dengan akun google.	Hanya berjalan di sistem operasi android.
3	RIM, 2010	BlackBerry App World	Menyediakan aplikasi yang bisa dipasang pada blackberry	Hanya tersedia di BBOs 7 keatas, dan ponsel pintar BBOS
4	Paul Thomas, 2005	<i>Prototyping Software Center</i>	Menjelaskan secara detail apa yang harus dimiliki <i>Software Center</i>	Masih merupakan aplikasi <i>prototype</i>
5	Canonical, 2010	<i>Ubuntu Software Center</i>	Menyediakan antarmuka yang baik, kemudahan dalam penggunaan.	Terbatas hanya di bisa terpasang pada Linux Ubuntu
6	SUSE, 2010	YaST	Memudahkan pengguna untuk mengatur apapun di OpenSUSE	Hanya tersedia untuk Linux OpenSUSE
7	Šrain, 2006	YaST Architecture	Membahas bagaimana YaST dikembangkan	-
8	Klumpp, 2012	<i>Software Center for Fedora</i>	Bisa dijalankan pada distro lain, misal debian dan fedora	Masih versi beta, dikembangkan dengan bahasa C dan Python
9	Khusna Ismail & Nur Fajri, 2013	Aplikasi Permainan Casual Game “Becak	Mudah diakses melalui web	Masih sederhana

		Pursuit” Berbasis HTML 5		
10	Pratami, 2013	Rancang Bangun Game Hijaiyah Match dengan HTML5	Dapat berjalan baik pada web browser	Belum dapat berjalan baik pada mobile browser

Penelitian terdahulu *Software Center* dikembangkan spesifik dengan masing-masing sistem operasi dan distribusi yang dipakai, menggunakan bahasa pemrograman utama python, ruby dan C, maka disini penulis mencoba merancang *Software Center* menggunakan bahasa HTML5 dengan metode *Evolutionary Prototyping* sebagai salah satu solusi atas masalah yang dihadapi pada Linux BlankOn. Penggunaan HTML5 hendaknya memberikan solusi untuk desain antarmuka, tampilan dan interaksi dengan pengguna yang lebih baik.

## 2.2 Landasan Teori

Dalam landasan teori ini akan dijelaskan secara garis besar teori-teori yang menjadi dasar atau acuan dalam penelitian ini :

### 2.2.1 Open Source Software

Menurut David Wheeler, secara umum program yang dinamakan *free software* (perangkat lunak bebas) atau *open source software* (perangkat lunak sumber terbuka) adalah program yang lisensinya memberi kebebasan kepada pengguna menjalankan program untuk apa saja, mempelajari dan memodifikasi program, dan mendistribusikan penggandaan program asli atau yang sudah dimodifikasi tanpa harus membayar royalti kepada pengembang sebelumnya (Wheeler, 2000).

### **2.2.2 Linux**

Linux pada awalnya dibuat oleh seorang mahasiswa Finlandia yang bernama Linus Torvalds. Dulunya Linux merupakan proyek hobi yang diinspirasi dari Minix, yaitu sistem UNIX kecil yang dikembangkan oleh Andrew Tanenbaum. Linux versi 0.01 dikerjakan sekitar bulan Agustus 1991. Kemudian pada tanggal 5 Oktober 1991, Linus mengumumkan versi resmi Linux, yaitu versi 0.02 yang hanya dapat menjalankan shell bash (GNU Bourne Again Shell) dan gcc (GNU C Compiler).

### **2.2.3 Distribusi Linux**

Linux bisa didapatkan dalam pelbagai distribusi (sering disebut Distro). Distro adalah bundel dari kernel Linux, beserta sistem dasar Linux, program instalasi, tools basic, dan program-program lain yang bermanfaat sesuai dengan tujuan pembuatan distro (Raharja, R. Anthon dkk, 2001).

Contoh distro Linux : Debian, BlankOn, Ubuntu, Fedora, OpenSUSE dan lain-lain.

### **2.2.4 Linux BlankOn**

Linux BlankOn adalah distribusi Linux yang dikembangkan oleh Yayasan Penggerak Linux Indonesia (YPLI) dan Tim Pengembang BlankOn. Distribusi ini dirancang sesuai dengan kebutuhan pengguna komputer umum di Indonesia. Linux BlankOn dikembangkan secara terbuka dan bersama-sama untuk menghasilkan distro Linux khas Indonesia, khususnya untuk dunia pendidikan, perkantoran dan pemerintahan. Linux BlankOn merupakan bagian dari gerakan

Proyek BlankOn yang memiliki cakupan yang lebih luas lagi. Gerakan ini bertujuan untuk mengembangkan kompetensi sumber daya manusia Indonesia dalam konteks perangkat lunak bebas dan terbuka.(Purnomo Dkk, 2012)

### **2.2.5 Unified Modeling Language(UML)**

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak(Dharwiyanti, 2003).

UML cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET dapat juga digunakan untuk modeling aplikasi prosedural dalam VB atau C.

#### **2.2.5.1 Use Case Diagram**

Use case diagram menggambarkan kebutuhan sistem dari sudut pandang pengguna. Digunakan untuk menggambarkan hubungan antara internal sistem dan eksternal sistem atau hubungan antara use case dan aktor.

##### **1. Aktor**

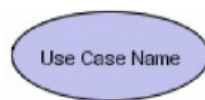
Aktor adalah pengguna sistem dan digambarkan seperti Gambar 2.1. Peran pengguna tertulis di bawah ikon. Pelaku tidak terbatas pada manusia. Jika suatu sistem berkomunikasi dengan aplikasi lain, dan akan memberikan masukan atau keluaran, maka aplikasi tersebut juga dapat dianggap sebagai aktor.



*Gambar 2.1: Aktor*

## 2. Use Case

Use case yang dibuat berdasar keperluan aktor merupakan gambaran dari “apa” yang dikerjakan oleh sistem, bukan “bagaimana” sistem mengerjakannya. Use case diberi nama yang menyatakan apa hal yang dicapai dari interaksinya dengan aktor seperti pada Gambar 2.2.



*Gambar 2.2:*

*Use Case*

## 3. Relasi

Relasi (*relationship*) digambarkan seperti pada Gambar 2.3 sebagai bentuk garis antara dua simbol dalam use case diagram. Relasi antara actor dan use case disebut juga dengan asosiasi (*association*). Asosiasi ini digunakan untuk menggambarkan bagaimana hubungan antara keduanya.

Relasi-relasi yang terjadi pada use case diagram bisa antara actor dengan use case atau use case dengan use case.

Relasi antara use case dengan use case :

- a) *Include*, pemanggilan use case oleh use case lain atau untuk menggambarkan suatu use case termasuk di dalam use case lain (diharuskan). Contohnya adalah pemanggilan sebuah fungsi program.



Digambarkan dengan garis lurus berpanah dengan tulisan <<include>>.

- b) *Extend*, digunakan ketika hendak menggambarkan variasi pada kondisi perilaku normal dan menggunakan lebih banyak kontrol form dan mendeklarasikan ekstension pada use case utama. Atau dengan kata lain adalah perluasan dari use case lain jika syarat atau kondisi terpenuhi. Digambarkan dengan garis berpanah dengan tulisan <<extend>>.
- c) *Generalization/Inheritance*, dibuat ketika ada sebuah kejadian yang lain sendiri atau perlakuan khusus dan merupakan pola berhubungan base- parent use case. Digambarkan dengan garis berpanah tertutup dari base use case ke parent use case.



*Gambar 2.3: Relationship dalam UML*

#### **2.2.5.2 Activity Diagram**

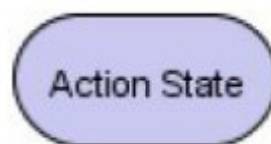
Diagram aktivitas menggambarkan proses bisnis dan urutan aktivitas-aktivitas yang mendukung penggambaran tindakan sistem baik yang bersifat kondisional maupun paralel. Tindakan kondisional dilukiskan dengan cabang (*branch*) dan penyatuan (*merge*).

Sebuah *branch* memiliki sebuah *transition* masuk atau yang disebut dengan *incoming transition* dan beberapa *transition* keluar atau yang disebut dengan *outgoing transition* dari *branch* yang berupa keputusan- keputusan. Hanya satu dari *outgoing transition* yang dapat diambil, maka keputusan-keputusan tersebut harus bersifat *mutually exclusive*. [else] digunakan sebagai keterangan singkat yang menunjukkan bahwa *transition* “else” tersebut harus digunakan jika semua keputusan yang ada pada *branch* salah.

Sebuah *merge* memiliki banyak input *transition* dan sebuah output. *Merge* menandakan akhir dari suatu kondisi yang diawali dengan sebuah *branch*. Selain *branch* dan *merge*, di dalam diagram aktivitas terdapat pula fork dan join. Fork memiliki satu *incoming transition* dan beberapa *outgoing transition*. Sedangkan pada join, *outgoing transition* diambil atau digunakan hanya ketika semua state pada *incoming transition* telah menyelesaikan aktivitasnya.

### 1. Activity State

Aktivitas ditandai tindakan oleh objek. Notasi untuk keadaan ini digambarkan seperti pada Gambar 2.4.



Gambar 2.4: Action State

### 2. Transition

Bila aktivitas sudah selesai, proses berpindah ke aktivitas yang lain.

Transisi digunakan untuk menandai gerakan ini. Transisi digambarkan

seperti pada Gambar 2.5



*Gambar 2.5: Transition*

### 3. *Initial State*

*Initial State* menandai awal masuk dan titik awal. Notasinya lingkaran yang penuh, seperti pada Gambar 2.6. Hanya ada satu initial state pada diagram.



*Gambar 2.6: Initial State*

### 4. *Final State*

*Final state* menandai akhir model alur kerja. Ada beberapa final state pada diagram, dan menggunakan model lingkaran yang penuh dikelilingi oleh lingkaran lain, seperti pada Gambar 2.7.



*Gambar 2.7: Final State*

### 5. *Decision State*

*Decision state* menandai pencabangan kondisi dari suatu proses, digambarkan seperti pada Gambar 2.8.



*Gambar 2.8: Decision State*

#### 6. *Synchronization Bar*

Kegiatan sering dapat dilakukan secara paralel. Untuk memecah proses ("fork"), atau untuk melanjutkan pemrosesan ketika beberapa kegiatan telah selesai ("join"), sinkronisasi bar digunakan. *Synchronization Bar* digambarkan seperti pada Gambar 2.9.



*Gambar 2.9: Synchronization Bar*

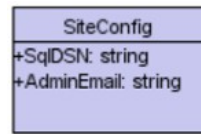
### 2.2.5.3 Class Diagram

*Class diagram* menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain (Dharwiyanti, 2003).

Diagram ini memperlihatkan himpunan kelas, antarmuka, kolaborasi serta relasi.

#### 1. *Class*

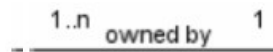
Kelas adalah bangunan blok dipemrograman berorientasi objek. Kelas digambarkan menggunakan persegi panjang dibagi menjadi tiga bagian seperti pada Gambar 2.10. Bagian atas adalah nama kelas. Bagian tengah mendefinisikan properti dari kelas. Di bagian bawah daftar metode kelas.



Gambar 2.10: Class

## 2. Association

Asosiasi adalah hubungan umum antara dua kelas, dan dimodelkan oleh baris menghubungkan dua kelas seperti pada Gambar 2.11. Baris ini dapat memenuhi syarat dengan jenis hubungan, dan juga mempunyai baragam aturan (misalnya, satu-ke-satu, satu-ke-banyak, banyak-ke- banyak) untuk hubungan.



Gambar 2.11: Association

## 3. Composition

Jika satu kelas tidak dapat berdiri dengan sendirinya, melainkan harus menjadi anggota kelas lain, maka kelas tersebut memiliki hubungan komposisi dengan kelas yang lain. Komposisi hubungan itu ditunjukkan dengan garis yang diisi dengan berlian seperti pada Gambar 2.12.

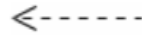


Gambar 2.12: Composition

## 4. Dependency

Bila menggunakan kelas-kelas yang lain, mungkin sebagai anggota variabel atau parameter, dan sebagainya "tergantung" dikelas itu, sebuah hubungan ketergantungan dibentuk. Ketergantungan hubungan itu

ditunjukkan dengan panah garis putus putus seperti pada Gambar 2.13.



*Gambar 2.13: Depedency*

#### 5. *Aggregation*

*Aggregation* menunjukkan seluruh bagian-hubungan, dan dikenal sebagai "memiliki-sebuah" hubungan. Penggabungan sebuah hubungan itu ditunjukkan dengan garis dengan rongga berlian seperti pada Gambar 2.14.



*Gambar 2.14: Aggregation*

#### 6. *Generalization*

Hubungan generalisasi adalah sama dengan sebuah warisan hubungan berorientasi objek. Hubungan generalisasi ditunjukkan oleh sebuah panah dengan panah yang berongga pada, atau "orang tua", kelas seperti pada Gambar 2.15.



*Gambar 2.15: Genertalization*

### **2.2.5.4 Sequence Diagram**

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) (Dharwiyanti, 2003).

### 1. *Object*

Objek adalah contoh dari kelas, dan dapat diatur horizontal ditunjukan pada Gambar 2.16. Bagian yang menggambarkan perwakilan untuk objek adalah kotak dengan nama.



*Gambar 2.16: Object*

### 2. *Actor*

Aktor juga dapat berkomunikasi dengan objek, sehingga mereka juga dapat terdaftar sebagai kolom. Aktor adalah sebuah model yang ada di mana-mana menggunakan simbol seperti Gambar 2.17.



*Gambar 2.17: Actor*

### 3. *Lifeline*

*Lifeline* yang mengidentifikasi keberadaan objek dari waktu ke waktu.

*Lifeline* menggunakan simbol seperti pada Gambar 2.18.



*Gambar 2.18: Lifeline*

### 4. *Activation*

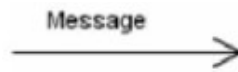
*Activation* sebagai model kotak segi empat di atas *lifeline*, menunjukkan bila objek melakukan tindakan seperti pada Gambar 2.19.



*Gambar 2.19: Activation*

## 5. Message

Pesan, dimodelkan panah horisontal antara aktifitas, menunjukkan komunikasi antar objek ditunjukkan pada Gambar 2.20.



*Gambar 2.20: Message*

### 2.2.6 Grafik User Interface (GUI)

*Graphical User Interface* (GUI) adalah suatu sistem interface pada komputer yang memberikan atau memfasilitasi tampilan pilihan pada layar, yang biasanya berbentuk ikon (simbol gambar) atau menu (daftar karakter alfanumerik) sebagai sarana yang dapat digunakan pengguna untuk memberikan perintah melalui divais input.(Hendarto, 2007)

### 2.2.7 Mock-up

Mock-up adalah *prototype* sangat awal. Mock-up bertujuan untuk memperoleh umpan balik dari pengguna tentang desain dan ide-ide desain awal dalam proses desain(Stroik, 2010).

Penggunaan yang paling umum dalam pengembangan perangkat lunak adalah untuk menciptakan *user interface* yang menunjukkan pengguna akhir *software* seperti apa yang akan terlihat tanpa harus membangun perangkat lunak atau fungsi yang mendasarinya. Desain mock-up dapat berkisar dari layout tangan



sederhana digambar layar, keterangan fungsi, dan proses yang terjadi.

Keuntungan dari mock-up (dan *prototype*) adalah:

- Fungsi mock-up sebagai media diskusi antara pengembang dan pengguna, tetapi juga antara anggota tim desain. Dengan demikian, mock-up dapat membantu memudahkan pekerjaan lintas batas disiplin tugas,
- Mock-up memungkinkan untuk melakukan pengujian kegunaan awal dalam proses pembangunan.
- Mock-up membantu eksperimen karena biaya yang murah untuk mengubah-ubahnya.

#### 2.2.8 Vala

*Vala* adalah bahasa pemrograman baru yang bertujuan untuk membawa fitur bahasa pemrograman modern pengembang *GNOME* tanpa memaksakan persyaratan runtime tambahan dan tanpa menggunakan *ABI* yang berbeda dibandingkan dengan aplikasi dan perpustakaan yang ditulis dalam C (Anwari, 2013).

*Valac*, *Vala compiler* adalah sebuah kompiler mandiri yang menerjemahkan kode sumber ke sumber Vala C dan file header. *Valac* menggunakan sistem tipe *GObject* untuk menciptakan kelas dan interface dideklarasikan pada kode sumber Vala.

Sintaks dari Vala mirip dengan C #, dimodifikasi agar lebih sesuai dengan sistem tipe *GObject*. Vala mendukung fitur bahasa modern sebagai berikut:

- Antarmuka

- Properties
- Sinyal
- Foreach
- Lambda expressions
- Type inference for local variables
- Generics
- Non-null types
- Assisted memory management
- Exception handling
- Type modules (Plugins)

Vala dirancang untuk memungkinkan akses ke perpustakaan yang ada C, terutama perpustakaan berbasis *GObject*, tanpa perlu binding runtime. Semua yang diperlukan untuk menggunakan perpustakaan dengan Vala adalah file API, yang berisi deklarasi kelas dan metode dalam sintaks Vala. Vala datang dengan binding eksperimental untuk GLib dan GTK+. Ini direncanakan untuk menyediakan hasil binding untuk GNOME platform penuh pada tahap berikutnya (Anwari, 2013).

### **2.2.9 HTML5**

HTML5 merupakan sebuah bahasa markah untuk menstrukturkan dan menampilkan isi dari World Wide Web, sebuah teknologi inti dari Internet. HTML5 adalah revisi kelima dari HTML dan hingga bulan Juni 2011 masih dalam pengembangan. Dimana tujuan utama pengembangan HTML5 adalah

untuk memperbaiki teknologi HTML agar mendukung teknologi multimedia terbaru, mudah dibaca oleh manusia dan juga mudah dimengerti oleh mesin. HTML5 merupakan hasil proyek dari W3C (World Wide Web Consortium dan WHATWG ( Web Hypertext Application Technology Working Group ). Dimana WHATWG bekerja dengan bentuk web dan aplikasi dan W3C merupakan pengembang dari XHTML 2.0 pada tahun 2006, kemudian mereka memutuskan untuk bekerja sama dan membentuk versi baru dari HTML.(Kadir, 2013)

Berikut tujuan dibuatnya HTML5 :

- a) Fitur baru harus didasarkan pada HTML, CSS, DOM, dan JavaScript,
- b) Mengurangi kebutuhan untuk plugin eksternal ( Seperti Flash ),
- c) Penanganan kesalahan yang lebih baik,
- d) Tambahan markup untuk menggantikan scripting,
- e) HTML5 merupakan perangkat mandiri,
- f) Proses pembangunan dapat terlihat untuk umum.

Fitur baru dalam HTML5 :

- a) Unsur kanvas untuk menggambar,
- b) Video dan elemen audio untuk media pemutaran,
- c) Dukungan yang lebih baik untuk penyimpanan secara offline,
- d) Elemen konten yang lebih spesifik, seperti artikel, footer, header, nav, section,
- e) Bentuk kontrol form seperti kalender, tanggal, waktu, email, url, search.

### **2.2.10 CSS3**

Cascading Style Sheets (CSS) adalah suatu bahasa stylesheet yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh World Wide Web Consortium (W3C).

CSS digunakan untuk menentukan warna, jenis huruf, tata letak, dan pelbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen (yang ditulis dengan HTML atau bahasa markup lainnya) dengan presentasi dokumen (yang ditulis dengan CSS). Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, mengurangi kompleksitas serta pengulangan pada struktur isi.

### **2.2.11 AJAX**

AJAX adalah singkatan dari Asynchronous JavaScript and XML. AJAX, terdiri dari HTML, Javascript, DHTML dan DOM yang kemudian digabungkan dengan bahasa pemrograman web di sisi server seperti PHP dan ASP, sehingga membentuk suatu aplikasi berbasis web yang interaktif. AJAX bukanlah bahasa pemrograman baru, tetapi adalah teknik baru untuk membuat aplikasi web lebih baik, lebih cepat dan lebih interaktif. Dengan AJAX, Javascript dapat langsung berkomunikasi dengan server dengan menggunakan objek XMLHttpRequest. Dengan objek ini, javascript dapat melakukan transaksi data dengan server web,

tanpa harus me-reloading halaman web tersebut secara keseluruhan (Kadir, 2009).

Berikut adalah teknologi yang termasuk dalam aplikasi AJAX :

- a) HTML yang digunakan untuk membuat Web forms dan mengidentifikasi field-field yang akan anda gunakan dalam aplikasi,
- b) JavaScript adalah kode inti untuk menjalankan aplikasi Ajax dan untuk membantu memfasilitasi komunikasi dengan aplikasi,
- c) DHTML, atau Dynamic HTML, membantu anda untuk membuat form atau web anda dinamis. Anda akan menggunakan `<div>`, `<span>` dan elemen HTML dinamis lainnya,
- d) DOM, Document Object Model, akan digunakan (melalui kode JavaScript) untuk bekerja dengan kedua struktur dari HTML dan XML anda yang dalam beberapa kasus berasal dari server.

#### **2.2.12 DOM**

*Document Object Model* (DOM) adalah platform dan bahasa-netral antarmuka yang memungkinkan program dan script untuk secara dinamis mengakses dan memperbarui konten, struktur dan gaya dokumen. Dokumen dapat diproses lebih lanjut dan hasil pengolahan yang dapat dimasukkan kembali ke dalam halaman yang disajikan. Ini merupakan ikhtisar bahan DOM terkait di sini di W3C dan di seluruh web.

#### **2.2.13 WebKit**

WebKit adalah komponen software mesin layout (layout engine software) yang dirancang untuk memungkinkan web browser untuk membuat halaman web.

Mesin ini dipakai oleh browser Safari (OS X) dan Chromium. Setiap browser yang ada memiliki mesin sendiri, sebagai contoh

1. IE → Trident,
2. Firefox/Seamonkey → Gecko,
3. Opera → Presto,
4. Safari/Chromium → WebKit.

Webkit juga digunakan sebagai dasar untuk browser eksperimental seperti pada Amazon Kindle e-book reader, serta browser default di Apple iOS, Android, BlackBerry 10, dan sistem operasi mobile Tizen. WebKit C++ API menyediakan satu set kelas untuk menampilkan konten web pada jendela, dan mengimplementasikan fitur peramban seperti berikut link ketika diklik oleh pengguna, mengelola daftar back-forward, dan mengelola riwayat halaman yang baru dikunjungi (Barth, 2012).

#### **2.2.14 WebKitGTK+**

WebKitGTK+ adalah platform port GNOME untuk mesin rendering WebKit. Menawarkan fungsionalitas penuh WebKit melalui satu set API berbasis *GObject*, kompetibel untuk proyek-proyek yang membutuhkan segala jenis integrasi web, dari aplikasi hybrid hibrida HTML / CSS hingga web browser penuh. Fungsionalitas WebKit berguna dalam pelbagai sistem dari komputer desktop untuk embedded system seperti telepon, tablet, dan televisi. WebKitGTK+ dibuat oleh komunitas pengembang dan desainer, untuk membawa platform web untuk semua orang (Barth, 2012).

Beberapa fitur dari WebKitGTK+ :

- Proses pemisahan web

Sejak menambahkan dukungan untuk WebKit2, memungkinkan untuk membangun aplikasi yang menggunakan platform web dengan peningkatan keamanan dan proses responsif. Web adalah halaman bebas, namun halaman web semestinya tidak menyebabkan galat dan freez UI. WebKitGTK+ juga menggunakan proses pemisah untuk memuluskan dukungan GTK +2 plugin (seperti Flash) di aplikasi GTK+3.

- Aksesibilitas

Akses adalah salah satu dari nilai inti, sehingga berkomitmen untuk membuat web bekerja untuk individu dari segala usia dan kemampuan, dari semua lapisan masyarakat, di seluruh dunia.

- Dukungan audio dan video

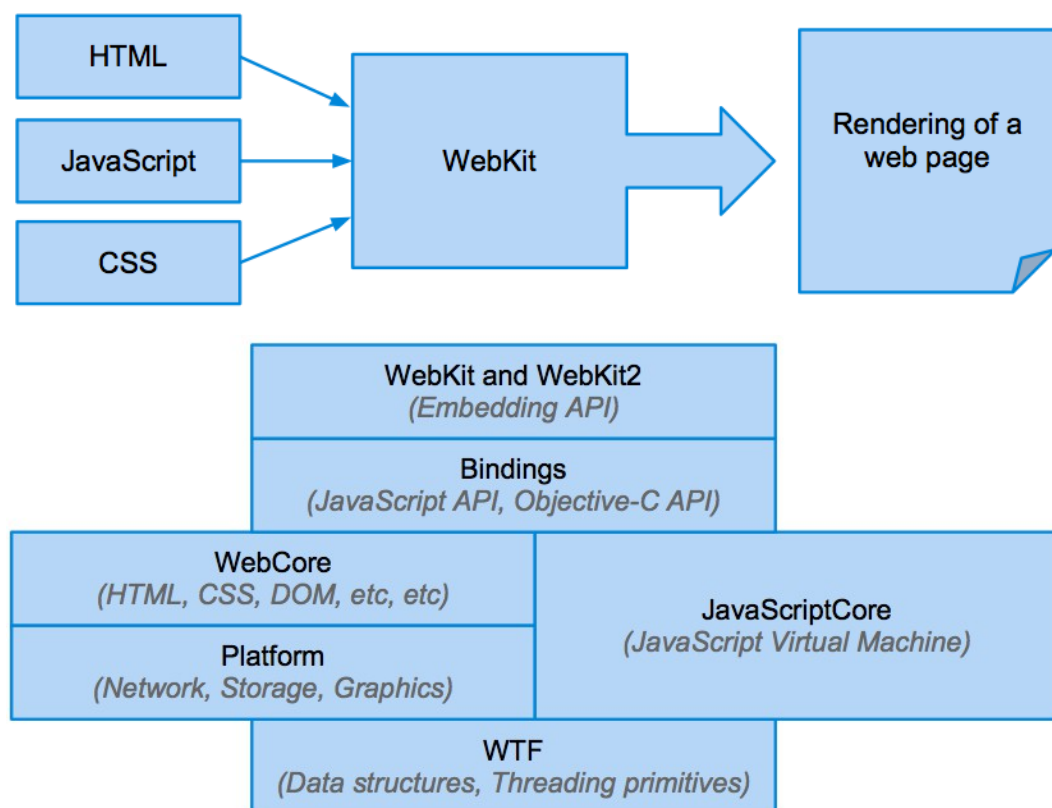
Web tidak hanya untuk membaca kata-kata, namun juga mendorong pengembangan GStreamer *back-end* WebKit dan mendukung integrasi penuh video ke konten halaman dan elemen kanvas HTML. Saat ini proses pengerjaan sedang berjalan untuk menyelesaikan dukungan untuk WebAudio dan WebRTC.

- 3D CSS dan accelerated rendering

WebKitGTK + dapat menggunakan GPU untuk memungkinkan rendering halaman web yang baik, serta transformasi 3D CSS dan 3D HTML kanvas (atau dikenal sebagai WebGL).

#### 2.2.14.1 Cara Kerja WebKitGTK+

WebKit adalah mesin rendering untuk konten web seperti pada Gambar 2.21(Barth, 2012).



*Gambar 2.21: Cara Kerja Webkit*

#### 2.2.15 JavaScript Dan JavaScriptCore

JavaScript adalah bahasa pemrograman utama yang digunakan pada halaman web. Awalnya bernama LiveScript dan sekarang juga dikenal sebagai ECMAScript, itu adalah aman, bahasa dinamis yang bisa dibilang salah satu



bahasa pemrograman yang paling terkenal di dunia. Ada standar JavaScript, yang diciptakan oleh ECMA (European Computer Manufacturers Association).

JavaScriptCore adalah mesin JavaScript WebKit dibuat berdasarkan KJS. JavaScriptCore adalah kerangka yang terpisah dari WebCore dan WebKit, dan digunakan pada Mac OS X untuk aplikasi selain halaman web JavaScript.

#### **2.2.16 JQuery**

jQuery adalah perpustakaan JavaScript yang cepat, kecil, dan kaya fitur. Adanya jQuery membuat hal-hal seperti dokumen HTML traversal dan manipulasi, penanganan event, animasi, dan Ajax jauh lebih sederhana dengan API yang mudah digunakan yang bekerja di banyak browser.

Perpustakaan jQuery berisi beberapa fitur berikut:

1. Navigasi dokumen,
2. Manipulasi HTML / DOM,
3. Manipulasi CSS,
4. Metode event HTML,
5. Efek dan animasi,
6. AJAX,
7. Utilities.

#### **2.2.17 Twitter Bootstrap**

Twitter Bootstrap adalah sebuah alat bantu untuk membuat sebuah tampilan halaman website yang dapat mempercepat pekerjaan seorang pengembang website ataupun pendesain halaman website. Sesuai namanya,

website yang dibuat dengan alat bantu ini memiliki tampilan halaman yang sama / mirip dengan tampilan halaman Twitter atau desainer juga dapat mengubah tampilan halaman website sesuai dengan kebutuhan.

Twitter Bootstrap dibangun dengan teknologi HTML dan CSS yang dapat membuat layout halaman website, tabel, tombol, form, navigasi, dan komponen lainnya dalam sebuah website hanya dengan memanggil fungsi CSS (class) dalam berkas HTML yang telah didefinisikan. Selain itu juga terdapat komponen-komponen lainnya yang dibangun menggunakan JavaScript.

### **2.2.18 AngularJS**

AngularJS adalah sebuah teknologi open source Javascript Framework dari Google yang memperluas kemampuan HTML agar bisa menangani halaman HTML dinamis. Intinya, dengan AngularJS, manipulasi DOM menjadi jauh lebih mudah dibanding harus menggunakan jQuery, tujuan dari framework ini adalah meningkatkan browser berbasis aplikasi dengan kemampuan MVC (Model View Controller) (Wahlin, 2013).

Beberapa fitur yang terdapat di dalam AngularJS di antaranya adalah :

1. Data binding, fitur ini merupakan bagian source yang dapat kita pakai untuk menampilkan view model dan memanipulasi DOM.
2. Controller, memberikan kebebasan kepada pengguna library untuk melakukan register callback dan beberapa model data.
3. Javascript Asli (Plain Javascript), AngularJS memang bukanlah framework javascript namun kemampuannya dapat dipakai untuk melakukan manipulasi fungsi dan model secara lebih mudah serta terstruktur.

### **2.2.19 Repository Linux**

*Repository* Linux adalah perpustakaan aplikasi yang bisa dipasang pada Linux. Perpustakaan ini biasa disediakan oleh komunitas atau perusahaan yang mengembangkan Distribusi Linux.

### **2.2.20 Packaging Tools**

Packaging Tools (Manajer Paket) adalah kumpulan perangkat lunak untuk mengotomatisasi proses instalasi, upgrade, mengkonfigurasi dan menghapus paket perangkat lunak untuk sistem operasi komputer secara konsisten. Ini biasanya memelihara sebuah database dependensi software dan informasi versi untuk mencegah ketidakcocokan perangkat lunak dan prasyarat yang hilang.

Paket adalah distribusi perangkat lunak, aplikasi dan data. Paket juga mengandung metadata, seperti nama perangkat lunak, deskripsi tujuan, nomor versi, vendor, checksum dan daftar dependensi yang diperlukan untuk perangkat lunak untuk berjalan dengan baik. Setelah instalasi, metadata disimpan dalam database paket lokal (Hughes, 2008).

### **2.2.21 Evolutionary Prototyping**

Evolutionary Prototyping Method merupakan salah satu dari beberapa metode yang digunakan dalam pengembangan software. Ciri khusus dari metode ini adalah melibatkan klien dalam membangun perangkat lunak (Carter Dkk, 2001).

Kelebihan Evolutionary Prototyping :

- Klien dapat melihat persyaratan sistem karena mereka ikut membicarakannya,

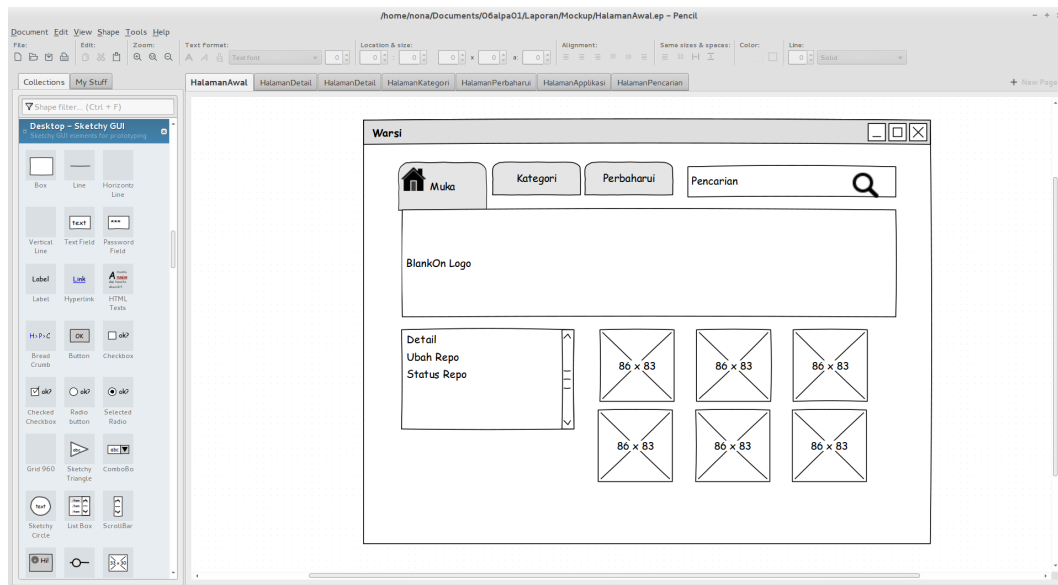
- Pengembang belajar dari klien,
- Produk akhir akan lebih akurat,
- Mampu menampung permintaan yang tak terduga,
- Memungkinkan fleksibilitas desain dan pengembangan,
- Proses kemajuan pengembangan dapat dipantau dengan baik

Kekurangan Evolutionary Prototyping :

- Membutuhkan keterlibatan klien lebih dekat,
- Perencanaan waktu akan terganggu,
- Proses dapat terus selamanya.

#### **2.2.22 Evolus Pencil**

Evolus Pencil adalah aplikasi untuk membuat sketsa tampilan, mockup maupun purwarupa lainnya. Memiliki fitur yang banyak, diantaranya ekspor hasil desain menjadi gambar(PNG), pdf, maupun halaman web. Memiliki koleksi icon sesuai kebutuhan. Tampilan Evolus Pencil seperti pada Gambar 2.22.



*Gambar 2.22. Tampilan Evolus Pencil*

## **BAB 3**

### **METODE PENGEMBANGAN SISTEM**

Suatu penelitian tidak akan berjalan dengan baik apabila tidak dilakukan dalam suatu proses yang teratur dan searah. Oleh karena itu diperlukan suatu metodologi untuk melaksanakan suatu penelitian.

#### **3.1 Pengumpulan Data**

Metode pengumpulan data yang digunakan dalam penelitian tugas akhir ini, yaitu:

1. Studi Kepustakaan

Pengumpulan data dengan cara mempelajari referensi-referensi dan teori-teori yang terkait dengan pengembangan *Software Center* dipelbagai platform maupun sistem operasi lain.

2. Wawancara

Wawancara dilakukan kepada pihak yang memiliki kaitan langsung dengan masalah yang diteliti, dalam hal ini adalah stakeholder atau klien yang menjadi sumber utama dalam mendapatkan informasi, data dan mencari kebutuhan sistem yang akan dibangun dalam penelitian ini. Stakeholder tersebut adalah Bapak Ahmad Haris selaku Manajer Rilis Linux BlankOn dan Mohammad Anwari selaku Manajer Proyek Linux BlankOn.

### 3.2 Kebutuhan Sistem

Perangkat keras yang dibutuhkan untuk mendukung pengembangan aplikasi *Software Center* dengan spesifikasi sebagai berikut :

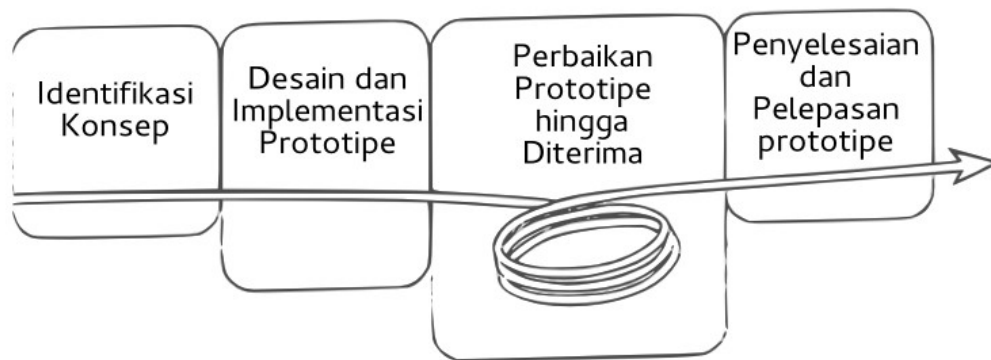
1. Procesor : CPU Intel Core 2 Duo Processor (2CPUs), ~2.4GHz.
2. Memory : 4GB.
3. Display : VGA NVIDIA GeForce 320M 256 MB
4. Hardisk : 128 GB

Sedangkan perangkat lunak yang digunakan :

1. Sistem Operasi : Linux BlankOn Suroboyo (develop Branch)
2. Sketsa Editor : Evolus Pencil 2.0.5
3. Text Editor : Sublime Text 3 Stable Build 3059
4. Back-end Programing : Vala 0.20 dan Vala Compiler 0.2037
5. Front-end Programing : HTML5
6. Rendering HTML5 : WebKitGtk Stable 2.0
7. Subversioning : Git
8. Web Browser : Google Chrome Ver 27.0.1453.110

### 3.3 Metodologi Pengembangan

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah dengan metode *Evolutionary Prototyping*. Metode evolutionary prototyping mempunyai tahapan-tahapan seperti pada Gambar 3.1.



*Gambar 3.1: Evolutionary Prototyping*

### 3.3.1 Identifikasi Konsep

Pada tahapan ini akan ditentukan apa saja yang akan dikerjakan untuk mewujudkan *Software Center*. Hal ini meliputi kebutuhan sistem, kriteria sistem, fungsi yang ingin dicapai. Tahapan ini penting agar gambaran detail aplikasi bisa dijelaskan dari awal. Tahap ini juga biasanya dikerjakan oleh kedua pihak, pengembang dan klien.

### 3.3.2 Desain Dan Implementasi *Prototype*

Dalam tahapan ini dikerjakan desain aplikasi standar dengan fungsi minimal untuk menjawab point-point penting yang telah dideskripsikan pada langkah sebelumnya. Desain *prototype* dikerjakan dalam bentuk mockup desain awal dan fungsi-fungsi yang bisa dijalankan aplikasi menggunakan bantuan Evolus Pencil.

Dari hasil desain awal *prototype* mulai dikerjakan dalam bentuk kode-kode program. Dimulai dari desain antarmuka menggunakan HTML5, CSS3,



javascript. Desain yang dicapai tidak harus persis sama dengan yang dirancang.

Hal selanjutnya adalah menggabungkan dengan pemograman Vala, *webkit* sebagai *back-end Software Center*. Pemasangan kontainer dikerjakan pada tahap ini agar beberapa fungsi aplikasi bisa berjalan. Setelah ini *prototype* awal *Software Center* didapat. Proses ini juga meliputi apakah aplikasi bisa dijalankan di Linux BlankOn dengan baik, fungsional dan usability.

### 3.3.3 Perbaikan *Prototype* Hingga Diterima

*Prototype* yang telah dikerjakan awal didiskusikan kembali kepada klien untuk dimintai masukan, baik fungsional, usability, maupun desain. Segala masukan yang memungkinkan untuk dikerjakan ditampung. Kemudian *prototype* diperbaiki lagi. Proses ini terus berulang hingga *prototype* tersebut disetujui atau memenuhi kriteria kebutuhan.

### 3.3.4 Penyelesaian Dan Pelepasan *Prototype*

Setelah *prototype* disetujui klien, aplikasi memasuki tahap rilis, dilanjutkan dengan penyempurnaan dokumentasi dan produk siap dipaketkan sesuai dengan distribusi Linux BlankOn.

## 3.4 Metodologi Pengujian Sistem

Pada tahap ini dilakukan pengujian akhir pada software yang sudah jadi sebagai bentuk evaluasi terhadap pengembangan sistem. Model uji validasi dan uji verifikasi yang digunakan adalah pengujian *Black Box*. Pengujian *Black Box* adalah pengujian yang dilakukan dengan mengabaikan bagaimana sistem berjalan.

Pengujian ini fokus kepada output yang dikeluarkan tanpa peduli bagaimana input itu diproses pada sistem hingga mengeluarkan output tersebut. Pengujian *Black Box* dibagi menjadi dua berdasarkan tingkat kedalaman dan waktu pengujian yaitu *Alpha Testing* dan *Beta Testing*. *Alpha Testing* meliputi bagaimana pengujian dilakukan pada sistem oleh pakar dengan item-item tertentu. Pengujian dilakukan sebelum aplikasi layak diuji oleh pengguna umum. *Beta Testing* meliputi pengujian yang dilakukan oleh *end-user*. Pengujian ini dilakukan setelah Alpha Testing selesai.

#### 3.4.1 Alpha Testing

Uji validasi *Alpha Testing* terfokus pada fungsionalitas sistem. Pengujian mulai dilakukan ketika proses pembuatan sedang berlangsung. Berikut akan dijelaskan rencana pengujian dalam Tabel 3.1.

*Tabel 3.1: Rancangan Item Uji Alpha Testing*

Item Uji	Detail Uji
Memasang Aplikasi	Aplikasi berhasil dipasang
Menghapus Aplikasi	Aplikasi berhasil dihapus
Mengubah Sumber <i>Repository</i>	Sumber <i>repository</i> berhasil diubah
Mencari Aplikasi berdasarkan kategori	Memilik list aplikasi berdasarkan katergori
Mencari Aplikasi berdasarkan kata kunci	Memasukkan key pencarian di kolom pencarian

#### 3.4.2 Beta Testing

Dalam perencanaan pengujian *Beta Testing* ini, akan ditetapkan beberapa parameter untuk mengetahui sejauh mana sistem dapat diterima oleh para pengguna. Adapun parameter-parameter tersebut adalah:

1. Tingkat efektivitas *Software Center* sebelum dan sesudah diimplementasikan di Linux BlankOn,
2. Kemudahan pengguna dalam memahami dan menggunakan sistem yang dibangun.

## **BAB 4**

### **ANALISIS DAN DESAIN SISTEM**

Pada tahap analisis dan perancangan diperlukan tahapan sistematis agar aplikasi yang dibuat sesuai dengan kegunaan dan tujuannya. Tahap pertama dari analisis adalah melakukan analisis sistem, mulai dari analisis masalah, analisis fungsional dan analisis non-fungsional. Sedangkan untuk tahap perancangan di mulai dengan melakukan perancangan sistem dan perancangan antarmuka yang akan digunakan untuk diterapkan pada aplikasi.

#### **4.1 Analisis Sistem**

Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke bagian- bagian komponen yang dimaksudkan untuk mengidentifikasi dan mengevaluasi segala macam permasalahan dan hambatan apa saja yang bisa terjadi dan kebutuhan-kebutuhan apa saja yang diharapkan dapat menjadi acuan untuk diusulkannya perbaikan-perbaikan.

##### **4.1.1 Analisis Masalah**

Analisis masalah dilakukan untuk memperoleh definisi permasalahan dan penggambaran yang tepat dari apa yang akan dilakukan oleh aplikasi, yang bertujuan untuk mengetahui seluk beluk aplikasi yang akan diteliti.

##### **4.1.1.1 Analisis Aplikasi Sejenis**

Analisis *Software Center* sejenis dilakukan sebagai pembanding dalam

merancang *Software Center* dengan HTML5. *Software Center* yang dibahas dan menjadi acuan disajikan dalam Tabel 4.1:

*Tabel 4.1: Perbandingan Aplikasi Software Center Sejenis*

<b>Item Banding</b>	<b>Kemudahan Akses</b>	<b>Update</b>	<b>Kategorisasi</b>
<b>Ubuntu Software Center</b>	Ubuntu Software Center dapat diakses melalui desktop bersistem operasi ubuntu bisa juga diakses melalui website <a href="https://apps.ubuntu.com/">https://apps.ubuntu.com/</a>	Untuk mengupdate suatu aplikasi sistem tidak melakukan download aplikasi secara keseluruhan tetapi hanya melengkapi paket apa yang belum ada pada aplikasi versi sebelumnya.	Aplikasi dibagi menjadi beberapa kategori yaitu Provided by Ubuntu, Canonical Partners, dan For Purchase.
<b>App Store</b>	Dalam mengakses App Store diperlukan gadget atau setidaknya sebuah PC dengan iTunes yang terinstall untuk download aplikasi	Untuk mengupdate suatu aplikasi, pengguna harus mendownload keseluruhan aplikasi tersebut.	Memiliki pembagian aplikasi yang rapi dan terorganisir. Kategori terbagi atas aplikasi yang terbaru, aplikasi yang terpopuler, aplikasi berbayar, dan aplikasi gratis.
<b>Google Play</b>	Google Play mudah untuk diakses karena dapat diakses tanpa memerlukan gadget dan bisa diakses melalui web browser.	Sama halnya dengan app store, untuk mengupdate suatu aplikasi, pengguna harus mendownload keseluruhan aplikasi tersebut.	Pembagian aplikasi pada google play dibagi menjadi 3 yaitu apps, games, dan books. Pembagian kategori apps dan games berdasarkan Top Paid, Top Free, Top Grossing, Top New Paid, dan Top New Free.
<b>WarSi BlankOn</b>	Tampilan berbasis HTML5 dan interaktif.	Untuk mengupdate pengguna hanya cukup menekan tombol update.	Ada 4 kategori utama; Internet, Aksesoris, Office, Audio. Semua aplikasi gratis.

#### 4.1.2 Analisa Dan Kebutuhan Fungsional

Hasil dari wawancara dengan stakeholder didapatkan beberapa analisis utama dalam pengembangan sistem ini yaitu mengenai pengguna dan fitur apa saja yang akan dikembangkan. Berikut merupakan pembagian kategori pengguna dalam sistem ini:

1. Pengguna BlankOn, merupakan pengguna utama. Pengguna ini dapat melakukan apapun pada *Software Center* seperti melihat software terbaru, melihat software yang terpasang, memasang software baru, menghapus software, mengubah sumber *repository*.
2. Pemaket, merupakan pengguna secara tidak langsung, bertugas merawat konten *Software Center*, sehingga interaksi langsung tidak dilakukan.

##### 4.1.2.1 Kebutuhan Fungsional

Kebutuhan Pengguna :

- Pengguna dapat mengetahui software terbaru,
- Pengguna dapat memasang software,
- Pengguna dapat menghapus software yang sudah terpasang,
- Pengguna dapat memperbaharui software yang sudah terpasang,
- Pengguna dapat menggunakan fitur pencarian software sesuai nama / kategori,
- Pengguna dapat mengubah sumber *repository*.

#### 4.1.2.2 Kebutuhan Non-Fungsional

##### 1. Kemudahan Pengguna

Dalam penggunaan sistem ini, pengguna tidak terlalu kesulitan untuk menggunakan secara langsung sistem ini, karena sistem memiliki interface yang menarik serta mudah dipelajari. Kemudahan dalam penggunaan sistem ini, karena menggunakan HTML5 sebagai antarmuka, sehingga tampilan dari aplikasi lebih interaktif.

##### 2. Kemudahan implementasi

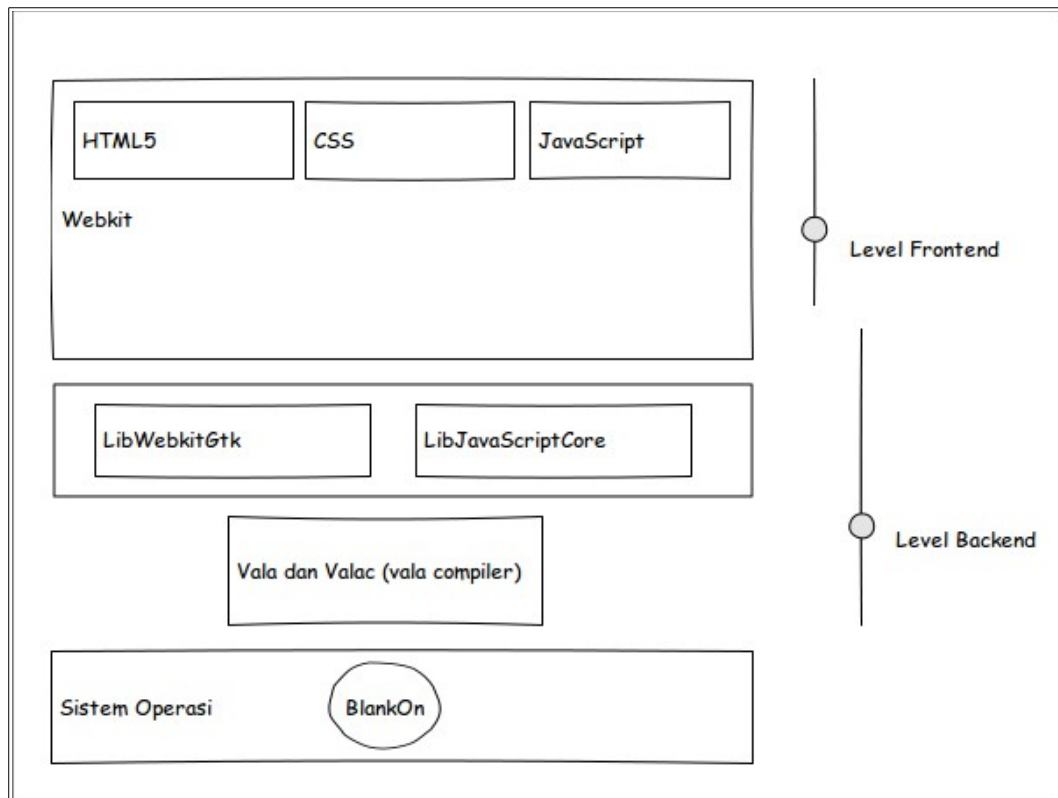
Sistem ini dikembangkan dengan disertai kode sumber terbuka (open source) sehingga memungkinkan untuk dikembangkan dalam distribusi lain.

#### 4.2 Perancangan Sistem

Hasil dari analisis kebutuhan fungsional sebelumnya kemudian dirancang dengan menggunakan UML (Unified Modelling Language).

##### 4.2.1 Desain Sistem

Berdasarkan kondisi sistem operasi dan rencana bahasa yang digunakan untuk membangun *Software Center* maka didesain bagaimanakah bentuk desain sistem *Software Center* ini bekerja. Desain itu dapat dilihat pada Gambar 4.1.



Gambar 4.1: Desain Sistem

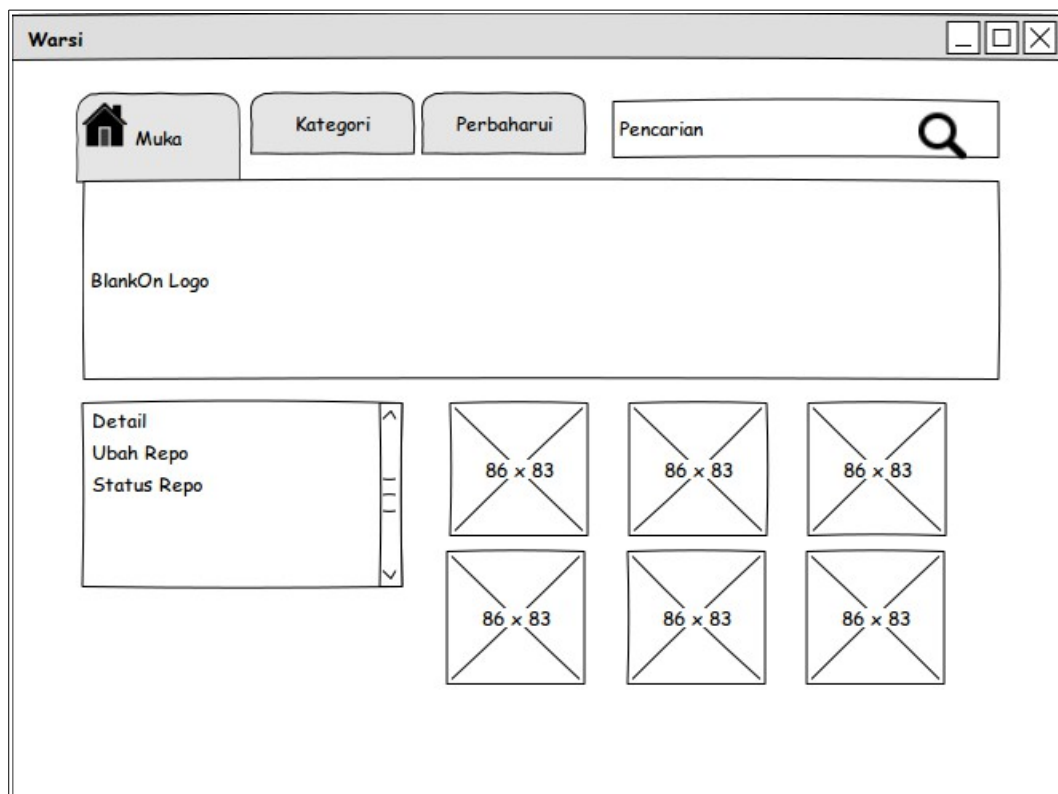
#### 4.2.2 Rancangan Antarmuka

Dalam pembuatan sistem, diperlukan rancangan Antarmuka untuk memudahkan pengguna dalam melakukan proses interaksi terhadap sistem. Rancangan antarmuka dibuat dalam bentuk mockup untuk menyelesaikan permasalahan *Software Center*, rancangan ini termasuk fitur apa saja yang bisa dikerjakan oleh *Software Center*. Rancangan ini buat dengan bantuan tools Evolus Pencil. Rancangan Antarmuka untuk sistem yang akan dibuat adalah sebagai berikut :



#### 4.2.2.1 Rancangan Antarmuka Halaman Utama

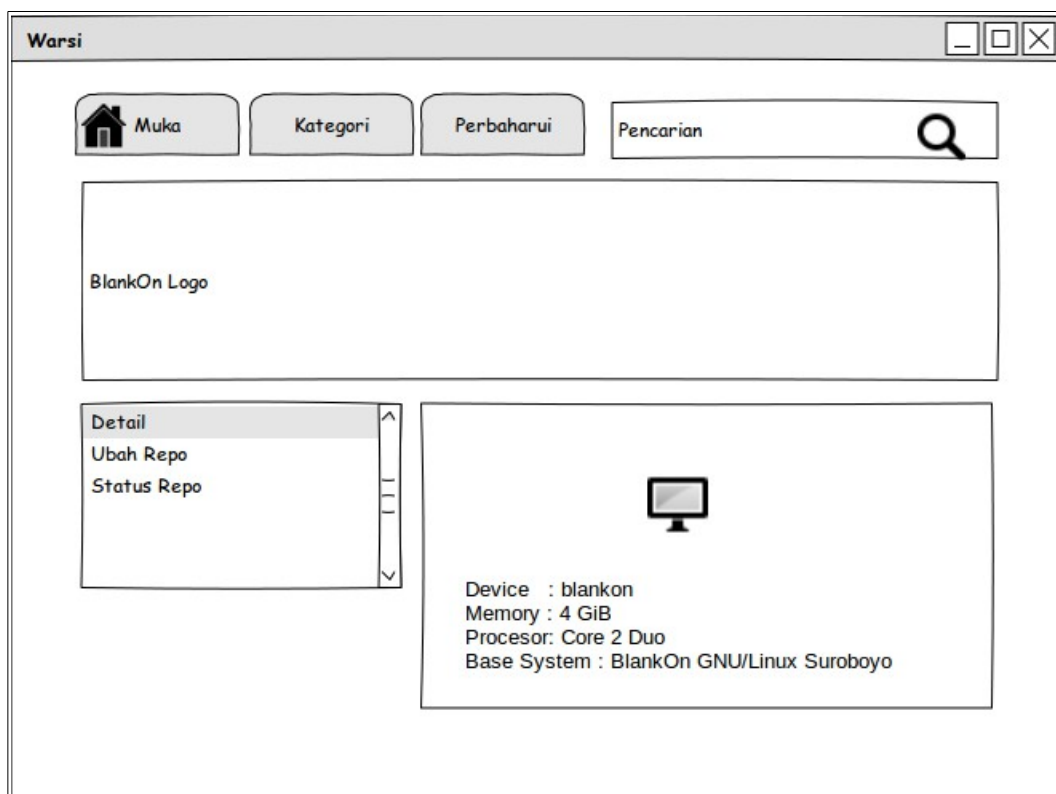
Rancangan halaman depan ini akan didapat ketika pertama kali menjalankan aplikasi. Halaman awal nantinya akan menyajikan informasi aplikasi yang bisa dipasang, status pengguna dan lain-lain. Terdapat tiga menu utama; Muka, Katogori dan Perbaharui. Tiga menu tambahan; Detail, Ubah Repo dan Status Repo. Rancangan antarmuka halaman depan dapat dilihat pada Gambar 4.2.



*Gambar 4.2: Rancangan Antarmuka Halaman Depan*

#### 4.2.2.2 Rancangan Antarmuka Detail

Rancangan antarmuka detail akan ditampilkan jika pengguna mengklik menu detail. Rancangan tersebut seperti pada Gambar 4.3. Antar muka ini akan menampilkan informasi mengenai sitem yang digunakan. Informasi ini meliputi informasi perangkat keras seperti tipe prosesor dan besar ram, perangkat lunak meliputi seri sistem operasi yang digunakan.



*Gambar 4.3: Rancangan Antarmuka Halaman Detail*

#### 4.2.2.3 Rancangan Antarmuka Ubah Repository

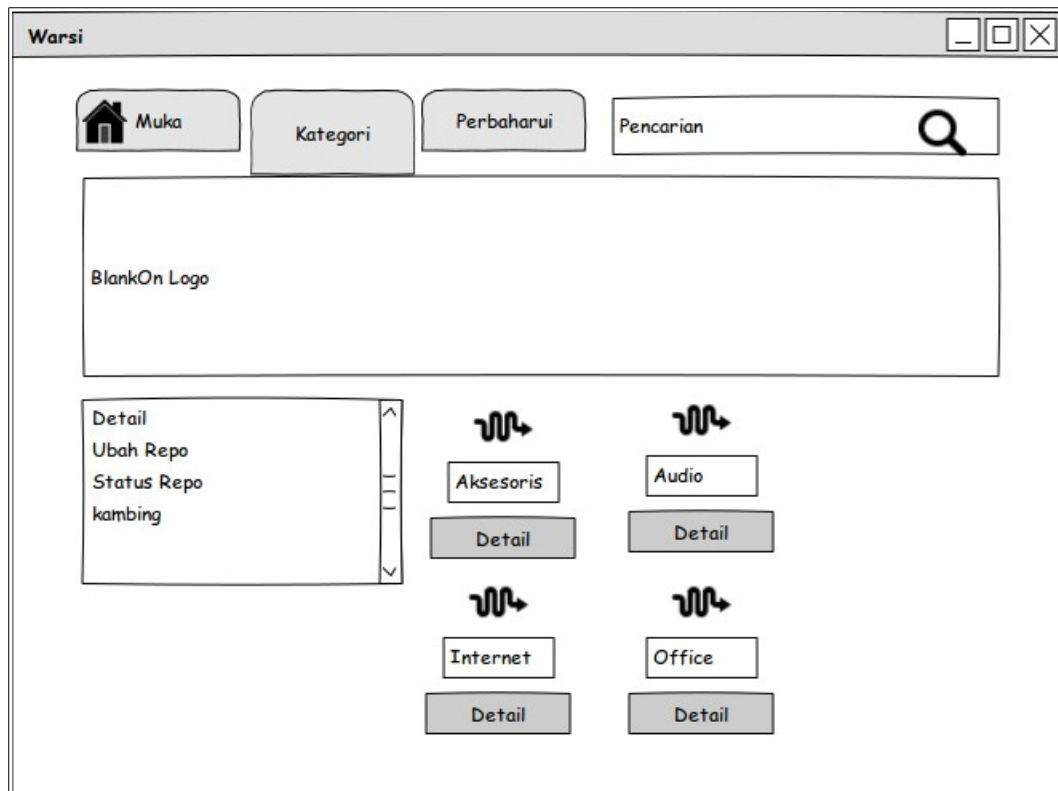
Rancangan antarmuka halaman ubah *repository* dapat dilihat pada Gambar 4.4. Halaman ini berfungsi untuk melakukan pengubahan sumber *repository* sesuai keinginan. Daftar sumber *repository* disediakan oleh sistem. Pengguna cukup memilih dari daftar yang telah ada.



*Gambar 4.4: Rancangan Antarmuka Halaman Ubah Repository*

#### **4.2.2.4 Rancangan Antarmuka Halaman Pilih Kategori**

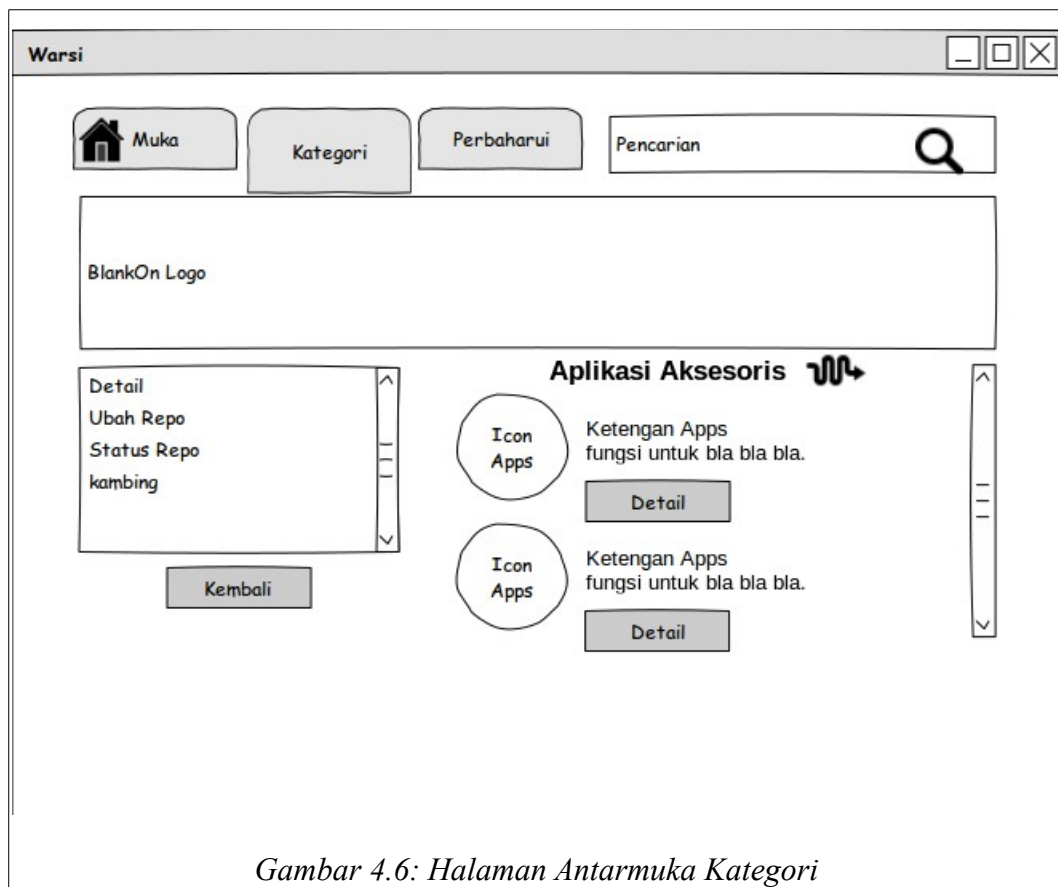
Rancangan antarmuka halaman pilih kategori dapat dilihat pada Gambar 4.5. Halaman ini berfungsi untuk memilih aplikasi sesuai kategori. Kategori yang tersedia adalah aksesoris, audio, Internet, dan office. Ketika halaman masing masing kategori dipilih, akan menuju halaman daftar aplikasi sesuai kategorinya.



*Gambar 4.5: Rancangan Antarmuka Halaman Pilih Kategori*

#### **4.2.2.5 Rancangan Halaman Kategori**

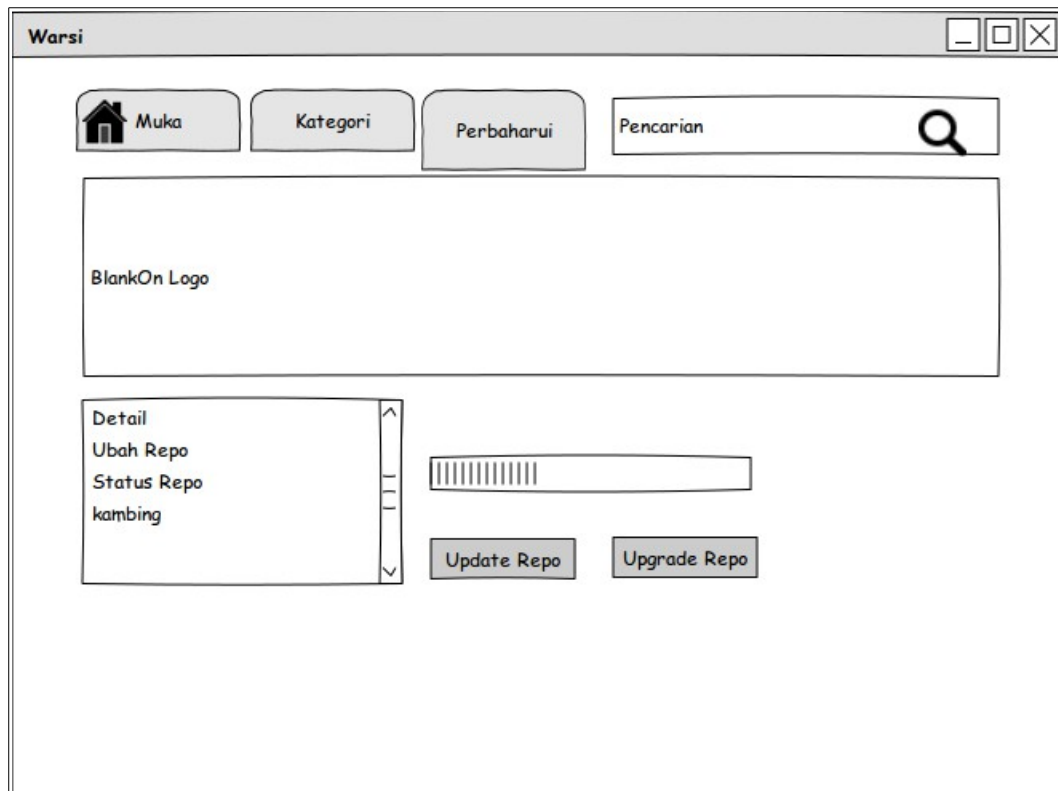
Rancangan antarmuka halaman kategori dapat dilihat pada Gambar 4.6. Halaman ini akan menampilkan daftar aplikasi sesuai kategori yang dipilih. Halaman ini dilengkapi tombol untuk kembali ke menu sebelumnya.



*Gambar 4.6: Halaman Antarmuka Kategori*

#### 4.2.2.6 Rancangan Antarmuka Halaman Perbaharui

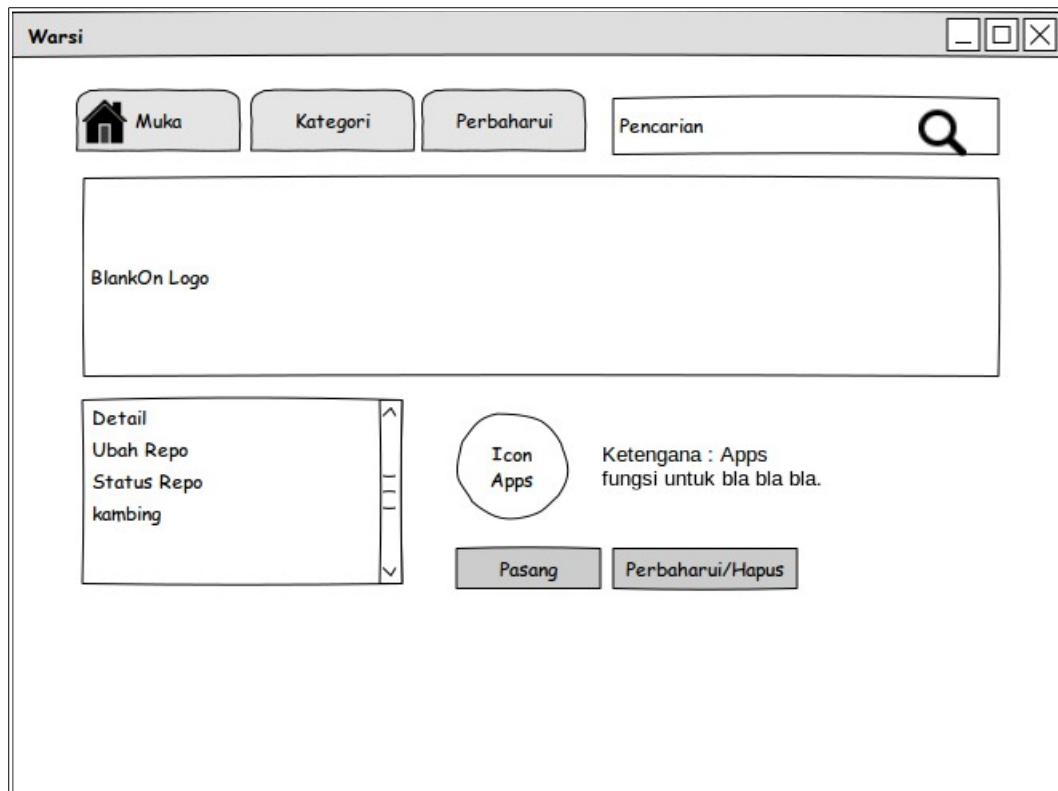
Rancangan antarmuka halaman perbaharui dapat dilihat pada Gambar 4.7. Halaman ini memiliki dua aksi utama, yaitu perbaharui *repository* untuk mengunduh daftar aplikasi terbaru dari sumber *repository* dan upgrade untuk memperbaharui semua aplikasi yang telah terpasang.



*Gambar 4.7: Rancangan Antarmuka Halaman Perbaharui*

#### **4.2.2.7 Rancangan Antarmuka Halaman Aplikasi**

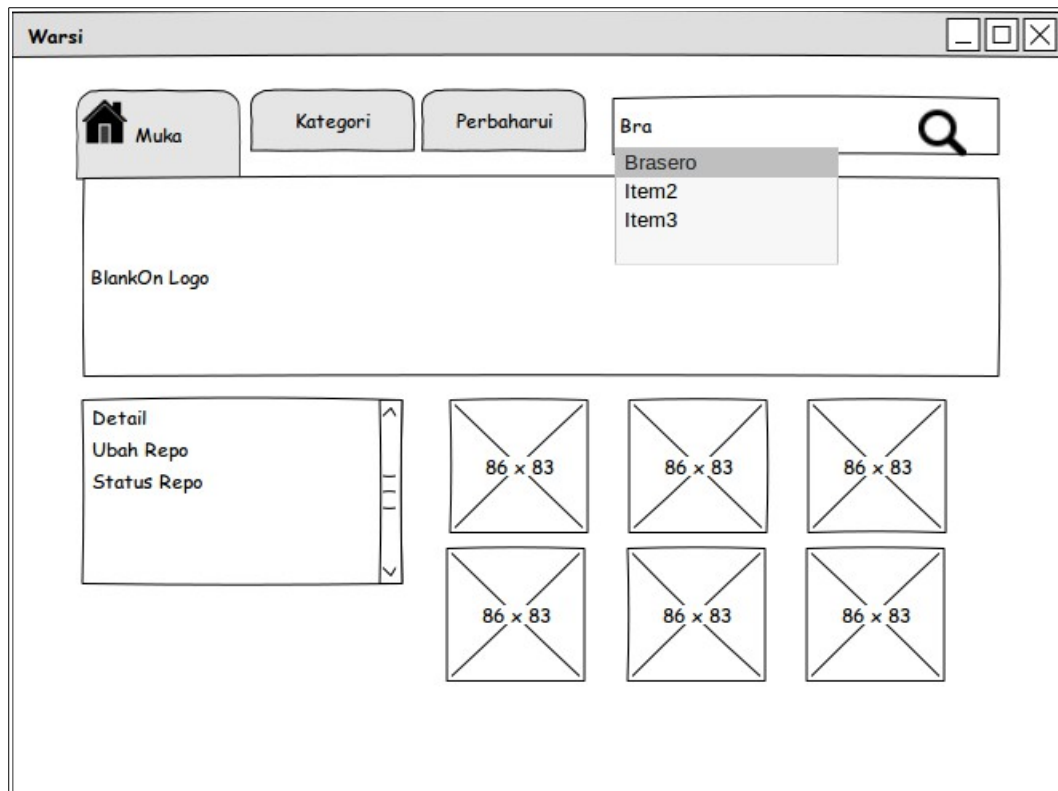
Rancangan antarmuka halaman aplikasi dapat lihat pada Gambar 4.8. Halaman ini menampilkan informasi mengenai aplikasi yang ada, ditambah opsi untuk memasang, menghapus atau memperbaharui aplikasi.



*Gambar 4.8: Rancangan Antarmuka Halaman Aplikasi*

#### **4.2.2.8 Rancangan Antarmuka Halaman Pencarian**

Rancangan antarmuka halaman pencarian dapat lihat pada Gambar 4.9. Halaman ini menampilkan bagaimana mencari aplikasi berdasarkan nama atau kata kunci. Hasil pencarian membawa pengguna menuju halaman aplikasi.

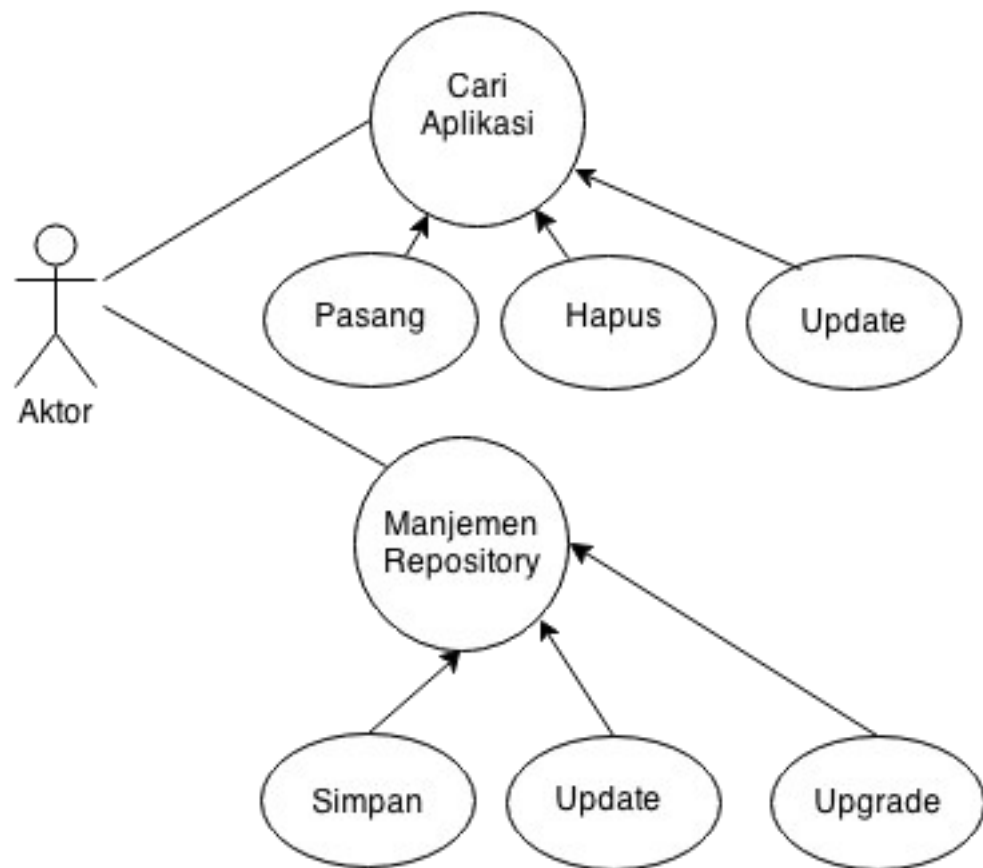


Gambar 4.9: Rancangan Antarmuka Halaman Pencarian

#### 4.2.3 Use Case Diagram

*Use case* Diagram merupakan konstruksi untuk mendeskripsikan hubungan-hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan sistem yang akan dibangun. Dari hasil analisis aplikasi yang ada maka *use case diagram* untuk aplikasi *Software Center* dapat di lihat pada Gambar 4.10.





*Gambar 4.10: Diagram Use case*

Selanjutnya, berikut ini merupakan penjelasan proses yang dipaparkan dalam *use case* diagram:

1. Cari Aplikasi

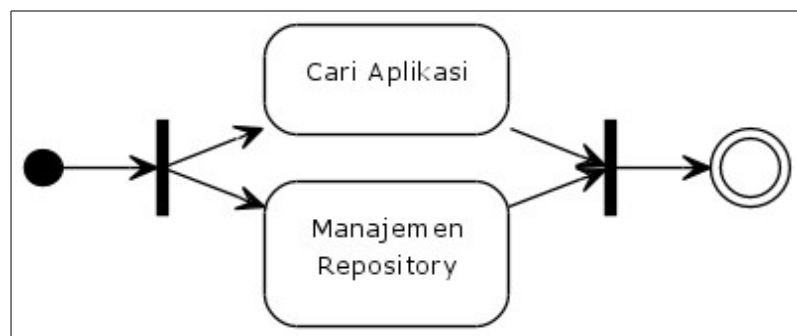
Proses ini dilakukan untuk mencari aplikasi sesuai keinginan pengguna. Setelah pencarian selesai pengguna dapat memasang/menghapus aplikasi maupun melakukan upgrade aplikasi.

2. Manajemen Repository

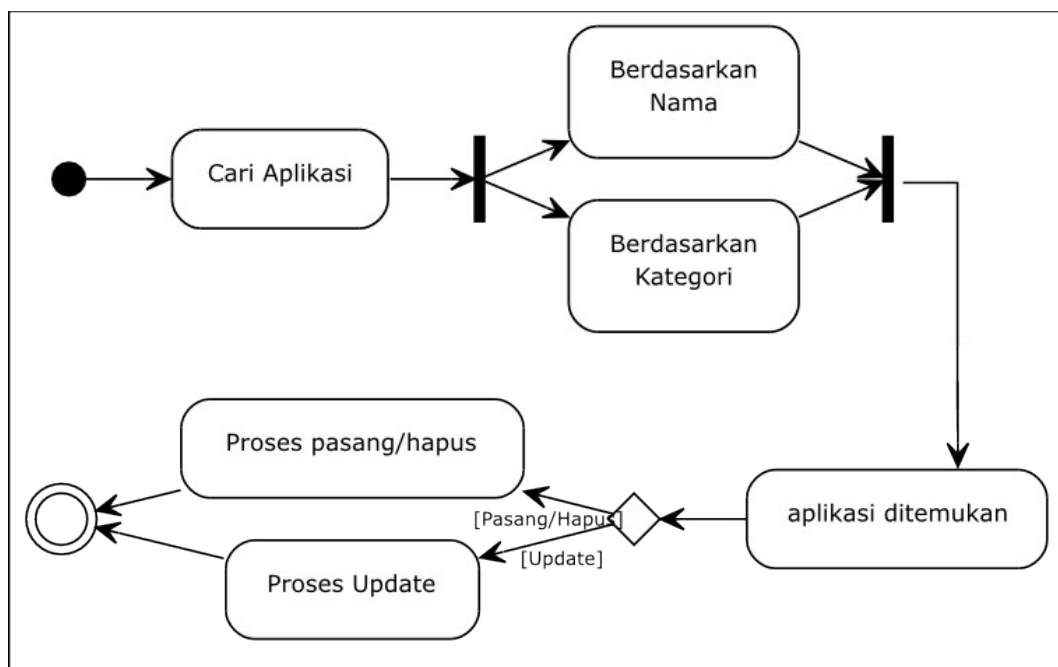
Dalam proses ini pengguna dapat mengubah repository; menyimpan, mengupdate dan mengupgrade repository.

#### 4.2.4 Activity Diagram

Activity diagram digunakan untuk memodelkan proses yang terjadi pada sistem yang akan menjelaskan aktifitas pada setiap fungsi yang ada. Diagram Aktifitas secara umum dapat dilihat pada Gambar 4.11.



*Gambar 4.11: Diagram Aktifitas*



*Gambar 4.12: Diagram Aktifitas Cari Aplikasi*

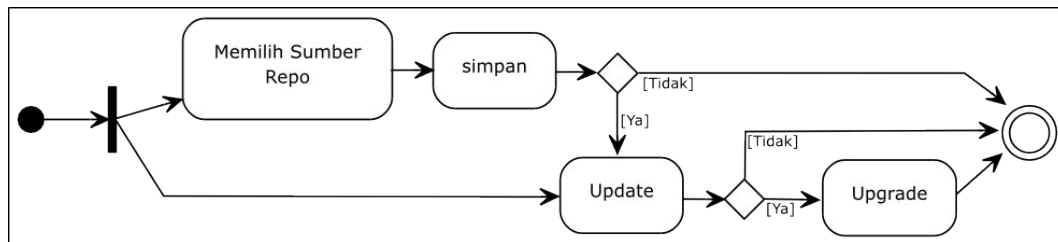
#### 4.2.4.1 Diagram Aktivitas Cari Aplikasi

Gambar 4.12 merupakan diagram aktifitas untuk menampilkan halaman pencarian aplikasi. Pencarian Aplikasi terbagi dua, yaitu dengan memasukkan kata kunci di kolom pencarian dan dengan mencari aplikasi berdasarkan kategori. Proses untuk Gambar 4.12 yaitu:

1. Pengguna memilih mencari aplikasi berdasarkan nama atau berdasarkan kategori aplikasi,
2. Jika memilih berdasarkan nama, pengguna memasukkan kata kunci atau nama aplikasi yang dicari pada kolom pencarian hingga aplikasi ditemukan,
3. Jika memilih berdasarkan kategori aplikasi, pengguna memilih kategori yang ada hingga aplikasi ditemukan,
4. Aplikasi ditemukan, pengguna memilih untuk memasanga/menghapus atau memperbaharui aplikasi,
5. Proses pemasangan, dan proses selesai.

#### 4.2.4.2 Diagram Aktivitas Manajemen *Repository*

Gambar 4.13 merupakan diagram aktifitas untuk menampilkan halaman manajemen *repository*. Pengguna bisa memilih *repository* sesuai daftar yang ada.



Gambar 4.13: Diagram Aktivitas Manajemen Repository

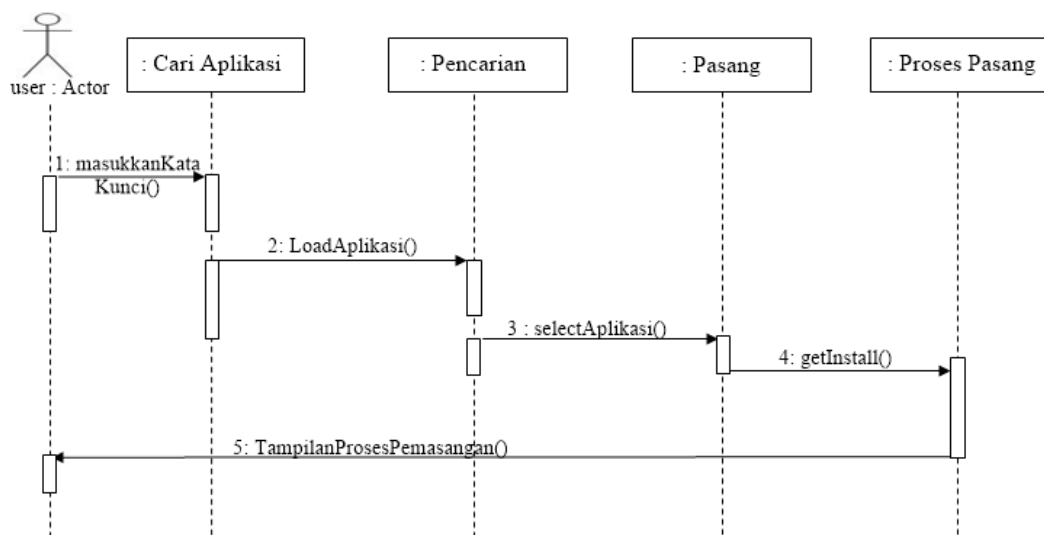
Proses untuk gambar 4.13 yaitu:

1. Pengguna memilih menu *repository* dari menu utama selanjutnya sistem akan menampilkan halaman *repository*, terdapat dua pilihan, memilih sumber *repository* atau langsung melakukan update,
2. Pengguna dapat melakukan perubahan *repository* dengan memilih daftar *repository* yang telah tersedia,
3. Apabila pengguna melakukan perubahan *repository* maka sistem akan melakukan penyimpanan pada perubahan,
4. Apabila pengguna melakukan update terhadap *repository* maka update terhadap *repository* akan dilakukan dan sistem akan menampilkan proses update,
5. Setelah melakukan update pengguna bisa melakukan upgrade terhadap sistem maka upgrade akan dilakukan dan sistem akan menampilkan proses update.
6. Proses selesai.

### 4.2.5 Sequence Diagram

*Sequence Diagram* menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan pada sistem sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu.

#### 4.2.5.1 *Squence Diagram* Cari Aplikasi Berdasarkan Nama

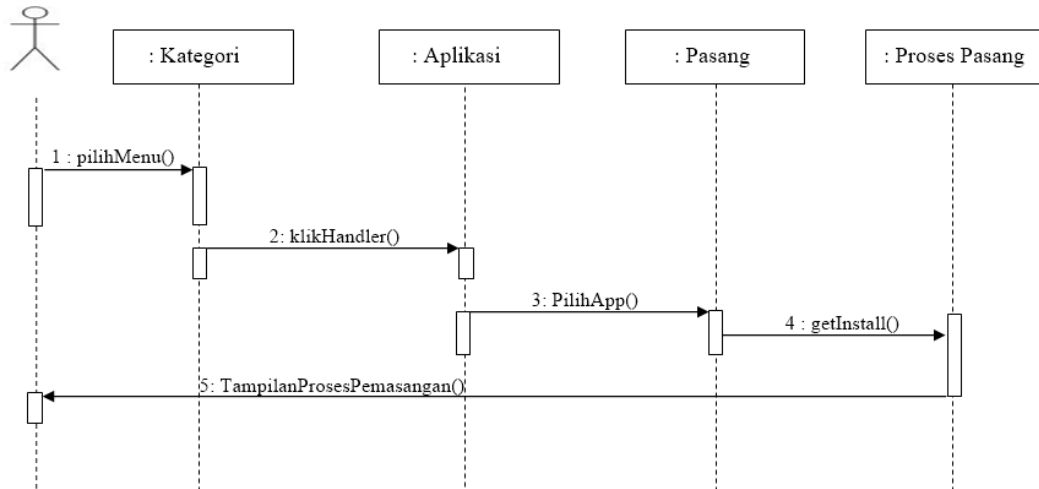


Gambar 4.14: *Sequence Diagram* Cari Aplikasi

Gambar 4.14 merupakan *Squence Diagram* cari aplikasi. Diagram ini menangani proses untuk menampilkan halaman pencarian.

#### 4.2.5.2 *Squence Diagram* Cari Aplikasi Berdasarkan Kategori

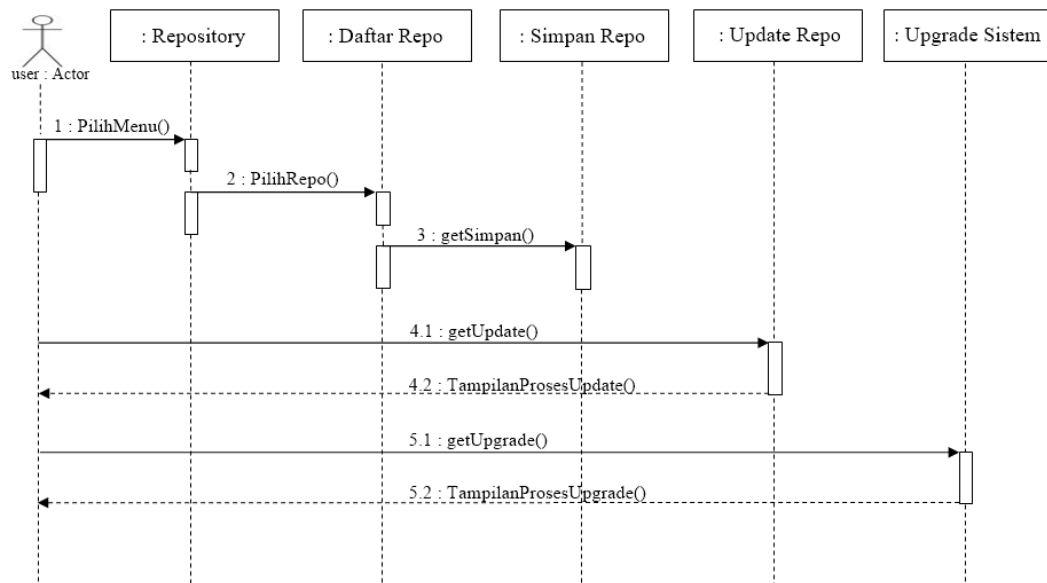
*Sequence* diagram pilih aplikasi berdasarkan kategori dapat dilihat pada Gambar 4.15. Diagram ini menangani proses untuk menampilkan halaman pilahan kategori aplikasi.



Gambar 4.15: Sequence Diagram Pilih Kategori

#### 4.2.5.3 Sequence Diagram Manajemen Repository

Sequence Diagram manajemen repository terdapat pada Gambar 4.16. Diagram ini menangani proses untuk menampilkan halaman manajemen repository.



Gambar 4.16: Sequence Diagram Manajemen Repository

## **BAB 5**

### **IMPLEMENTASI DAN PENGUJIAN SISTEM**

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya diimplementasikan pada bahasa pemrograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem untuk mengetahui apakah aplikasi yang dikembangkan sudah sesuai dengan perancangan dan mengetahui kekurangan-kekurangan aplikasi.

#### **5.1 Implementasi**

Tahap implementasi merupakan kelanjutan dari kegiatan perancangan aplikasi berdasarkan hasil analisis pada bab sebelumnya.

##### **5.1.1 Persiapan Lingkungan Kerja**

Sebelum program ditulis, diperlukan lingkungan kerja yang sesuai, yaitu meliputi tools dan aplikasi pendukung. Beberapa tool dan aplikasi belum dikenal luas dan memerlukan pemasangan khusus.

###### **5.1.1.1 Memasang Evolus Pencil**

Evolus Pencil tersedia untuk versi Linux BlankOn dari web resminya. Cukup diunduh dari situs Evolus (<http://pencil.evolus.vn/Downloads.html>) kemudian dipasang dengan perintah seperti pada Listing 5.1.

*Listing 5.1: Pemasangan Evolus Pencil*

```
$ sudo dpkg -i evoluspencil_2.0.5_all.deb
```

Setelah *Evolus Pencil* terpasang penulis mulai membuat rancangan (*mock-up*) antarmuka beserta rencana fungsi yang akan bisa dilakukan oleh *software center*. Rancangan (*mock-up*) ini sebagai gambaran bagaimana tampilan dari *software center*. Rancangan ini diusahakan sesederhana mungkin untuk mempermudah implementasi kedalam bahasa pemograman.

#### 5.1.1.2 Memasang WebkitGTK+ Dan Dependensi

WebKitGtk+ telah tersedia dalam *repository* Linux BlankOn dan bisa dipasang melalui konsole. Berikut adalah list paket *webkitgtk+* yang harus terpasang beserta dependensinya :

- *libglib2.0-dev, libglib2.0-doc*
- *libgtk-3-dev, libgtk-3-doc*
- *libwebkitgtk-dev, libwebkitgtk-3.0-dev, libwebkitgtk-3.0-doc*

Paket dipasang melalui konsole seperti pada listing 5.2:

*Listing 5.2: Pemasangan WebKitGTK+*

```
$ sudo apt-get install libglib2.0-dev libglib2.0-doc libgtk-3-dev
libgtk-3-doc libwebkitgtk-dev libwebkitgtk-3.0-dev libwebkitgtk-
3.0-doc libjavascriptcoregtk-dev libjavascriptcoregtk-3.0-dev
```

WebKit dibutuhkan untuk merender halaman HTML5 ke halaman aplikasi desktop. Webkit sendiri akan dipanggil dari program *back-end*, dalam hal ini adalah Vala compiler.



### 5.1.1.3 Memasang Git Dan Kunci SSH

Git dibutuhkan agar pengerjaan aplikasi bisa dipantau oleh orang lain dan aplikasi memiliki riwayat pengerjaan yang baik. Penulis menggunakan github sebagai server upload kode sumber. Berikut perintah pemasangan git melalui konsole seperti pada Listing 5.3:

*Listing 5.3: Memasang Git*

```
$ sudo apt-get install git xclip  
$ git config --global user.name "tuanpembual"  
$ git config --global user.email "andro.medh4@gmail.com"
```

Setelah git terpasang dilakukan pengaturan kunci ssh agar lebih mudah dalam melakukan sinkronisasi kode di komputer lokal dengan kode sumber di situs <http://github.com>. Perintah untuk melakukan pengaturan kunci ssh terdapat pada Listing 5.4.

*Listing 5.4: Generate Kunci SSH*

```
$ ssh-keygen -t rsa -C "andro.medh4@gmail.com"  
$ xclip -sel clip < ~/.ssh/id_rsa.pub
```

Penulis melakukan login pada situs github, kemudian pada pengaturan akun memilih ssh-key. Kunci RSA yang sudah disalin dengan xclips tadi penulis masukkan sebagai kunci ssh yang dikenali.

Penulis membuat *repository* baru dengan nama `tuanpembual.github.io`. Setelah *repository* selesai dibuat. Penulis melakukan cloning *repository* ke dalam komputer lokal dengan melakukan perintah seperti pada Listing 5.5.

```
$ git clone git@github.com:tuanpembual/tuanpembual.github.io.git
```

```
$ git init
$ git status
```

Perintah yang biasa digunakan dalam pengerjaan aplikasi software center menggunakan git seperti pada Tabel 5.1.

*Tabel 5.1: Perintah Git Sederhana*

Perintah	Fungsi
\$ git init	Untuk melakukan inisiasi git pertama kali
\$ git status	Untuk mengetahui status pengembangan, lokasi pengembangan, perubahan pada file/kode
\$ git add	Untuk menambahkan file yang akan dimonitor oleh git
\$ git commit	Untuk melakukan persetujuan terhadap perubahan, sehingga perubahan dicatat dalam git
\$ git push	Untuk mengunggah perubahan setelah commit
\$ git pull	Untuk menarik perubahan yang ada di awan, ini dilakukan jika pengembangan dilakukan secara kolaborasi atau dilakukan pada beberapa komputer lokal.

#### 5.1.1.4 Memasang Vala Dan Vala Compiler

Vala yang digunakan versi 0.20. Berikut perintah pemasangan Vala dan Vala compiler melalui konsole tertuang pada Listing 5.5 :

*Listing 5.5: Pemasangan Vala Compiler*

```
$ sudo apt-get install valac
```

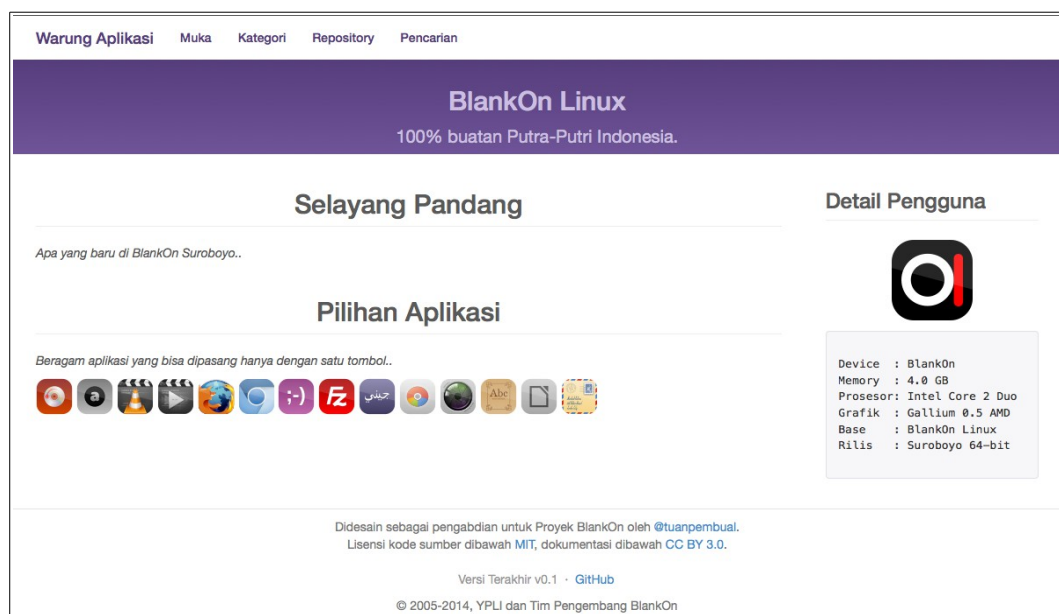
Valac adalah compiler Vala, yang akan mengkompilasi kode Vala ke biner. Biner yang dihasilkan akan memiliki nama yang sama dengan file sumber dan kemudian dapat langsung dijalankan pada mesin.

## 5.1.2 Implementasi Antarmuka

Implementasi antarmuka dilakukan untuk setiap tampilan aplikasi yang dibangun menggunakan HTML5. Pada bagian ini akan ditampilkan implementasi antarmuka halaman utama, halaman kategori, halaman pencarian, halaman manajemen *repository*.

### 5.1.2.1 Halaman Awal

Gambar 5.1 merupakan tampilan halaman yang pertama kali akan muncul saat aplikasi dibuka. Halaman ini menampilkan menu utama, ikon aplikasi yang bisa dipasang dan informasi sederhana mengenai sistem operasi dan hardware yang terpasang. Jika ikon aplikasi dipilih akan menuju ke halaman detail aplikasi.



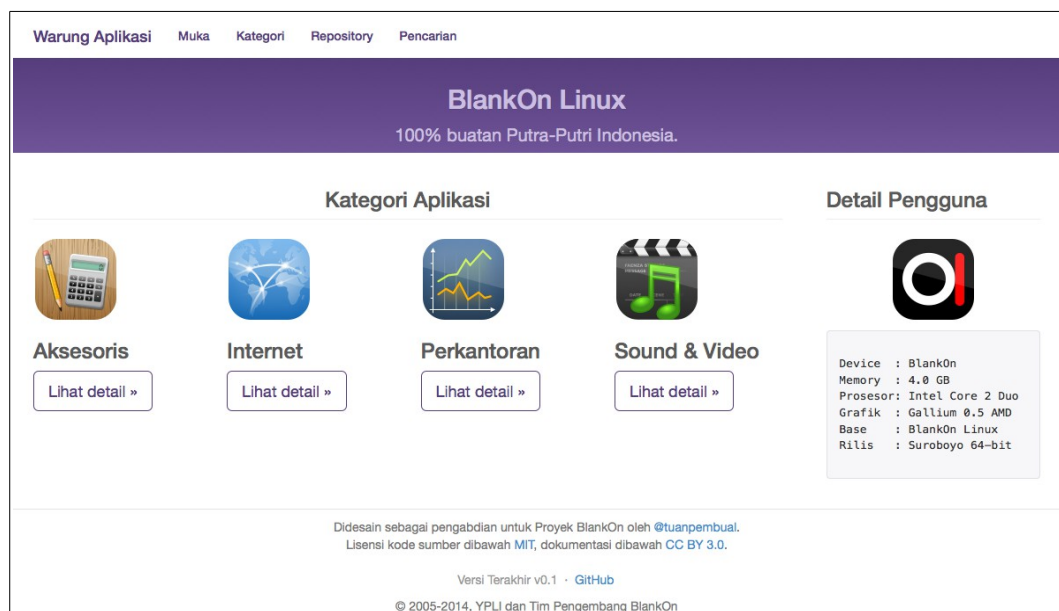
Gambar 5.1: Implementasi Halaman Awal

### 5.1.2.2 Halaman Kategori

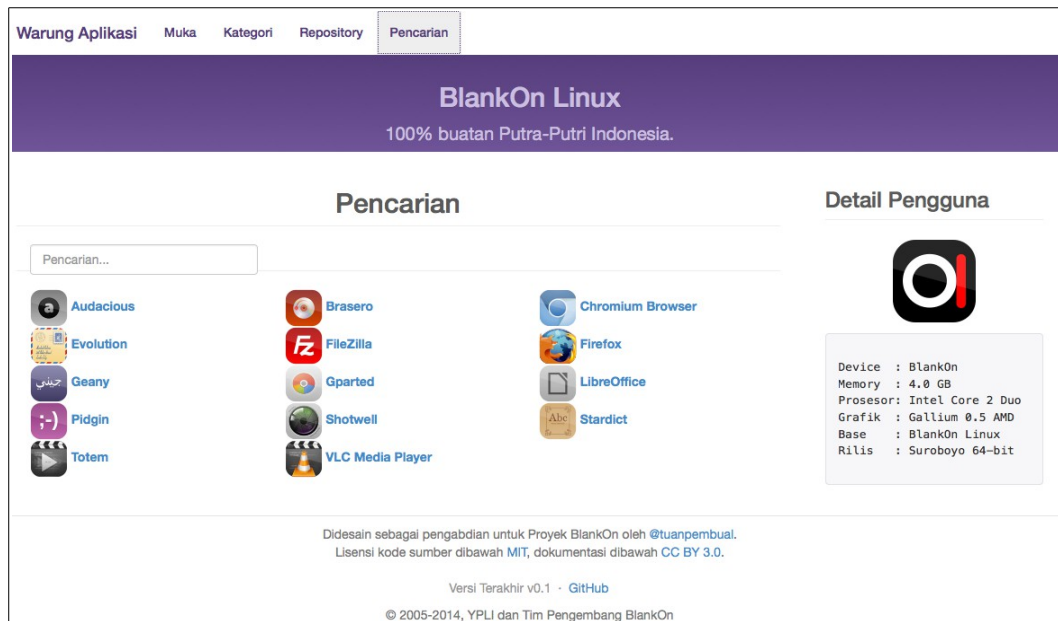
Gambar 5.2 merupakan tampilan halaman pembagian aplikasi berdasarkan kategori. Halaman ini menampilkan 4 contoh kategori aplikasi: Aksesoris, Internet, Perkantoran dan Sound & Audio. Masing-masing kategori berisi aplikasi berdasarkan kriteria aplikasi. Jika salah satu kategori dipilih, pengguna akan diarahkan menuju daftar aplikasi berdasarkan kategori yang telah dipilih.

### 5.1.2.3 Halaman Pencarian

Halaman pencarian dapat dilihat pada Gambar 5.3. Halaman ini memiliki form isian untuk memasukkan kata kunci. Setelah kata kunci dimasukkan maka akan keluar daftar aplikasi yang sesuai atau yang mendekati kata kunci.



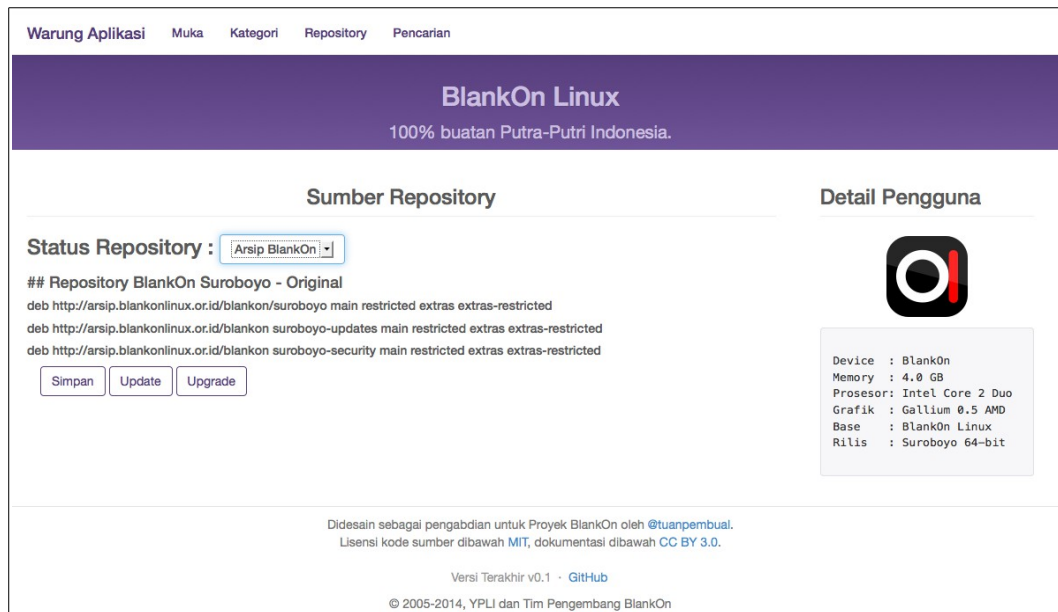
*Gambar 5.2: Implementasi Halaman Kategori*



Gambar 5.3: Implementasi Halaman Pencarian

#### 5.1.2.4 Halaman Manajemen *Repository*

Gambar 5.4 memuat antarmuka manajemen *repository*. Halaman ini berfungsi untuk mempermudah pengguna mengganti sumber *repository*. Ada tiga sumber *repository* utama yang biasa dipakai oleh pengguna BlankOn. Repo UGM, Kambing UI dan arsip BlankOn. Ada tiga tombol aksi yang bisa dipilih sesuai keinginan pengguna. Setelah sumber diubah, pengguna dipersilahkan menyimpan sumber *repository*, mengupdate, dan mengupgrade.



*Gambar 5.4: Implementasi Halaman Manajemen Repository*

#### 5.1.2.5 Halaman Detail Aplikasi

Halaman detail aplikasi menampilkan informasi tentang aplikasi beserta tautan pengembang. Logo/ikon aplikasi berasal dari tema yang disediakan dari pengembang kesenian BlankOn. Saat ini versi ikon yang digunakan adalah versi Komodo. Halaman detail aplikasi muncul jika ikon atau nama aplikasi dipilih oleh pengguna. Tampilan Halaman Aplikasi seperti pada Gambar 5.5. Ada tiga tombol aksi yang bisa dilakukan seperti :

- Pasang

Tombol ini dipilih jika aplikasi belum terpasang.

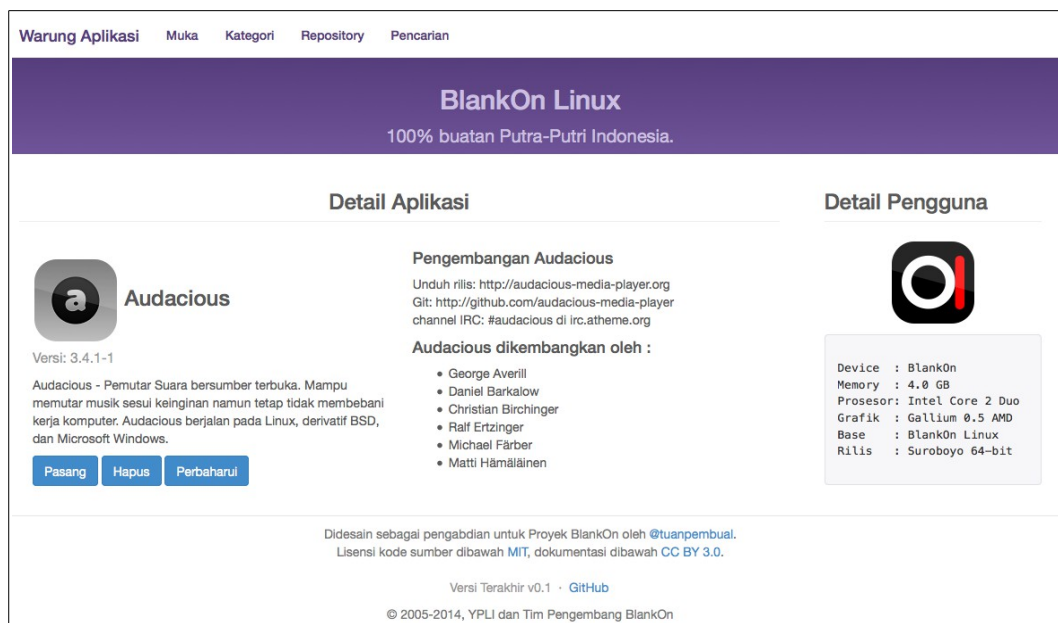
- Hapus

Tombol ini bisa dipilih jika aplikasi sudah terpasang dan pengguna ingin

menghapus aplikasi.

- Perbaharui

Tombol ini hanya bisa dipilih jika memang terdapat pembaharuan aplikasi terpilih dari lumbung *repository*.



Gambar 5.5: Implementasi Halaman Detail Aplikasi

### 5.1.3 Implementasi Kode

Tahap ini merupakan kelanjutan dari perancangan pada bab sebelumnya yang merupakan implementasi pada bahasa pemrograman. Beberapa script yang dikerjakan adalah untuk menjalankan proses di balik layar (*back-end*).

#### 5.1.3.1 Implementasi Script Utama

Script dibawah ini merupakan script untuk menginisiasi program utama

ketika program pertama kali dijalankan.. Diberi nama main.vala sesuai dengan aturan pemrograman Vala, tertuang dalam Listing 5.6.

*Listing 5.6: Script main.vala*

```
using GLib;
using Gtk;
using WebKit;
using JSCore;

namespace Warung {
    private const string TITLE = "Warung Aplikasi BlankOn";
    int main (string[] args) {
        Gtk.init (ref args);
        var webView = new Main ();
        var window = new Gtk.Window();
        window.add(webView);
        window.show_all ();
        window.resize(1024, 600);
        window.title = TITLE;
        window.window_position = WindowPosition.CENTER;
        try {
            window.icon = new Gdk.Pixbuf.from_file
(Environment.get_current_dir() + "/public/favicon.ico");
        } catch (Error e) {
            stderr.printf ("Could not load application icon: %s\n",
e.message);
        }
        window.destroy.connect(Gtk.main_quit);
        Gtk.main ();
        return 0;
    }
}
```

### 5.1.3.2 Implementasi Script Inisiasi Halaman Utama

Script dibawah ini merupakan script untuk membangun halaman utama program disimpan dengan nama page.vala. Script ini memanggil script app.vala dan membangun tampilan dengan memanggil file index.html. Script tertuang pada Listing 5.7.



*Listing 5.7: Script page.vala*

```

using GLib;
using Gtk;
using WebKit;
using JSCore;

namespace Warung {
    public class Main : WebKit.WebView {
        public void * context { get; set; }
        public Main ()
        {
            var test = new App();
            load_uri ("file://" + Environment.get_current_dir() +
"/public/index.html");
            window_object_cleared.connect ((frame, context) => {
                test.setup_js_class ((JSCore.GlobalContext) context);
            });
        }
    }
}

```

### 5.1.3.3 Implementasi Script Kumpulan Fungsi

Script dibawah ini merupakan script yang berisi kumpulan fungsi(*function*) yang akan digunakan untuk berkomunikasi dengan halaman html, Script disimpan dengan nama app.vala. Script tertuang pada Listing 5.8.

*Listing 5.8: Script app.vala*

```

static const JSCore.StaticFunction[] js_funcs = {
    { "ugm", repougm, PropertyAttribute.ReadOnly },
    { "arsip", repoarsip, PropertyAttribute.ReadOnly },
    { "kambing", repokambing, PropertyAttribute.ReadOnly },
    { "update", update, PropertyAttribute.ReadOnly },
    { "upgrade", upgrade, PropertyAttribute.ReadOnly },
    { "installapp", installapp, PropertyAttribute.ReadOnly },
    { null, null, 0 }
};

static const ClassDefinition js_class = {
    0, // version
    ClassAttribute.None, // attribute
    "Apps", // className
    null, // parentClass
}

```

```

null, // static values
js_funcs, // static functions
null, // initialize
null, // finalize
null, // hasProperty
null, // getProperty
null, // setProperty
null, // deleteProperty
null, // getPropertyNames
null, // callAsFunction
null, // callAsConstructor
null, // hasInstance
null // convertToType
};

public void setup_js_class (GlobalContext context) {
    var theClass = new Class (js_class);
    var theObject = new JSCore.Object (context, theClass, context);
    var theGlobal = context.get_global_object ();
    var id = new String.with_utf8_c_string ("Apps");
    theGlobal.set_property (context, id, theObject,
        PropertyAttribute.None, null);
}

```

#### 5.1.3.4 Implementasi Script Perbaharui Paket

Script untuk memperbaharui paket dilakukan dengan memasukkan paksa (injeksi) perintah terminal pada bahasa Vala dengan bantuan API Vala. Implementasi kode tersebut dapat dilihat pada Listing 5.9.

*Listing 5.9: Script Perbaharui Repository*

```

public static JSCore.Value updaterepo (Context ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {
    exception = null;
    //its need root previlage
    try {
        Process.spawn_command_line_async ("sudo apt-get update");
    } catch (SpawnError e) {
        stdout.printf ("Error: %s\n", e.message);
    }
    return new JSCore.Value.undefined (ctx); }

```

### 5.1.3.5 Implementasi Script Manajemen *Repository*

Manajemen *repository* memiliki beberapa fitur yaitu :

1. Mengganti sumber *repository*,
2. Menyimpan *repository* yang sudah diganti.

Script untuk mengganti *repository* terbagi dua, yang pertama script dari Vala yang bisa dilihat pada Listing 5.10.

*Listing 5.10: Script Mengganti Repository (vala)*

```
public static JSCore.Value repoarsip (Context ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

    exception = null;
    try {
        Process.spawn_command_line_sync ("sudo sh
"+Environment.get_current_dir()+"/belajar/lib/echoarsip");
    } catch (SpawnError e) {
        stdout.printf ("Error: %s\n", e.message);
    }
    return new JSCore.Value.undefined (ctx); }
```

Script yang kedua adalah script javascript untuk memanggil fungsi Vala.

Script ini tertuang dalam Listing 5.11.

*Listing 5.11: Script Mengganti Repository (javascript)*

```
$scope.simpanrepo = function(app) {

    if (app === "repougum") {
        (Apps.ugm())
    } else if (app === "kambing") {
        (Apps.kambing())
    } else {
        (Apps.arsip())
    };
}
```

### 5.1.3.6 Implementasi Script Memasang Dan Menghapus Aplikasi

Script untuk menjalankan perintah pasang atau hapus aplikasi hanya berupa perintah konsole apt yang kemudian dimudahkan dengan tampilan GUI. Perintah pemasangan ini tertuang dalam Listing 5.12.

*Listing 5.12: Script Memasang dan Menghapus Aplikasi*

```
exception = null;
//script pasang
try {
    Process.spawn_command_line_async ("sudo apt-get install
evolution -y");
} catch (SpawnError e) {
    stdout.printf ("Error: %s\n", e.message);
}
return new JSCore.Value.undefined (ctx); }
//script hapus
try {
    Process.spawn_command_line_async ("sudo apt-get remove
brasero -y");
} catch (SpawnError e) {
    stdout.printf ("Error: %s\n", e.message);
}
return new JSCore.Value.undefined (ctx); }
```

### 5.1.3.7 Implementasi Script Compiler

Script ini berfungsi untuk mengkompilasi program-program Vala agar mampu dijalankan dengan mesin render webkit. Hasil akhir dari script ini adalah file binari dari aplikasi *software center*. Oleh karena itu, script ini tidak termasuk dalam kode aplikasi *software center*. Script disimpan dengan nama MakeFile seperti tertuang dalam Listing 5.13.

*Listing 5.13: Script Compiler*

```
valac main.vala app.vala page.vala util.vala \
```

```
--pkg webkit-1.0 --pkg glib-2.0 --pkg libsoup-2.4 \
--pkg javascriptcore --thread --vapidir=./vapi \
--header=./vapi/warung.h \
--vapi=./vapi/warung.vapi -X -I/usr/include/webkitgtk-3.0
```

Setelah script MakeFile selesai dijalankan dan menghasilkan file binari, maka aplikasi software center bisa dieksekusi melalui konsol dengan perintah seperti pada Listing 5.14.

*Listing 5.14: Menjalakan File Binari Software Center*

```
$ sudo ./main
```

## 5.2 Pengujian Sistem

Pengujian sistem dilakukan bertujuan untuk menemukan kesalahan atau kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

Model uji validasi dan uji verifikasi yang digunakan adalah pengujian *Black Box*. Pengujian *Black Box* adalah pengujian yang dilakukan dengan mengabaikan bagaimana sistem berjalan. Pengujian ini fokus kepada output yang dikeluarkan tanpa peduli bagaimana input itu diproses pada sistem hingga mengeluarkan output tersebut. Pengujian *Black Box* dibagi menjadi dua berdasarkan tingkat kedalaman dan waktu pengujian yaitu *Alpha Testing* dan *Beta Testing*.

### 5.2.1 Pengujian Alpha

*Alpha Testing* meliputi bagaimana pengujian dilakukan pada sistem oleh pakar dengan item-item tertentu. Pengujian dilakukan sebelum aplikasi layak diuji oleh pengguna umum. Pengujian alpha dilakukan pada lingkungan pengembang software untuk memastikan program yang dibuat dapat berjalan dengan baik.

Rencana pengujian yang akan dilakukan dapat dilihat pada tabel 5.2.

*Tabel 5.2: Rencana Pengujian*

<b>Masukan</b>	<b>Hasil yang diharapkan</b>	<b>Pengamatan</b>	<b>Hasil Uji</b>
Menekan tombol menu muka	Menampilkan halaman muka	Dapat menampilkan halaman muka	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol menu kategori	Menampilkan halaman kategori aplikasi	Dapat menampilkan halaman kategori	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Memasukkan kata kunci dikolom pencari	Menampilkan halaman hasil pencarian	Dapat menampilkan halaman hasil pencarian	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol menu kategori Aksesoris	Menampilkan halaman aplikasi dengan kategori Aksesoris	Dapat menampilkan halaman kategori Aksesoris	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol menu kategori Internet	Menampilkan halaman aplikasi dengan kategori Internet	Dapat menampilkan halaman kategori Internet	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol menu kategory Audio Video	Menampilkan halaman aplikasi dengan kategori Audio Video	Dapat menampilkan halaman kategori Audio Video	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol	Menampilkan	Dapat	<input type="checkbox"/> Berhasil

menu kategori Office	halaman aplikasi dengan kategori Office	menampilkan halaman kategori Office	<input type="checkbox"/> Tidak berhasil
Menekan ikon aplikasi	Menampilkan keterangan aplikasi	Dapat menampilkan halaman aplikasi terpilih.	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan ikon pasang/hapus aplikasi	Menampilkan halaman persetujuan pemasangan/penghapusan aplikasi	Dapat memasang/menghapus aplikasi	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol menu perbaharui	Menampilkan halaman menu perbaharui	Dapat menampilkan halaman perbaharui	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil
Menekan tombol ubah <i>repository</i>	Menampilkan halaman pengubah <i>repository</i>	Dapat mengubah repostori dan menyimpannya	<input type="checkbox"/> Berhasil <input type="checkbox"/> Tidak berhasil

### 5.2.2 Pengujian Beta

Pengujian beta dilakukan pada lingkungan pengguna aplikasi. Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan kuesioner mengenai tanggapan pengguna terhadap aplikasi yang telah dibuat..

Pengujian dilakukan terhadap *usability* dan fungsionalitas sistem. Pengujian *usability* sistem dapat dilihat pada tabel 5.3 dan pengujian fungsional sistem dapat dilihat pada tabel 5.4.

Tabel 5.3: Pengujian Usabilitas Aplikasi

No	Pertanyaan	Penilaian				
		SS	S	N	TS	STS
1	Saya memahami cara menggunakan aplikasi <i>software center</i>					
2	Saya merasa mudah memasang dan menghapus aplikasi di Linux BlankOn setelah menggunakan <i>software center</i>					
3	Saya setuju <i>software center</i> ini memiliki tampilan yang menarik					
4	Saya berkeinginan menggunakan <i>software center</i> ini kembali					

**Keterangan**

SS : Sangat Setuju

TS : Tidak Setuju

S : Setuju

STS : Sangat Tidak Setuju

N : Netral

Tabel 5.4: Pengujian Fungsionalitas Aplikasi

No.	Pertanyaan	Penilaian	
		Ya	Tidak
1	Software Center dapat menampilkan halaman muka		
2	Software Center dapat mencari aplikasi berdasarkan kata kunci		
3	Software Center dapat menampilkan halaman kategori		
4	Software Center dapat menampilkan halaman kategori aksesoris		
5	Software Center dapat menampilkan halaman kategori office		
6	Software Center dapat menampilkan halaman kategori Internet		
7	Software Center dapat menampilkan halaman kategori		



	audio video		
8	Software Center dapat menampilkan halaman manajemen <i>repository</i>		
9	Software Center dapat memasang aplikasi terpilih		
10	Software Center dapat menghapus aplikasi terpilih		
11	Software Center dapat mengubah sumber <i>repository</i>		

## **BAB 6**

### **HASIL DAN PEMBAHASAN**

#### **6.1 Proses Pengembangan *Software Center***

Pengembangan aplikasi *Software Center* ini menggunakan metode *Evolutionary Prototyping*. Dengan metode ini, pengembangan berawal dari desain sederhana (mockup), kemudian desain tadi dipresentasikan kepada klien untuk diminta saran dan koreksi sebelum dibuat rancang dalam bentuk kode pemograman.

##### **6.1.1 Analisis**

Pada tahap awal pengembangan, informasi didapat dengan melakukan wawancara dengan klien (stakeholder) dalam hal ini bapak Ahmad Harus selaku manajer rilis BlankOn Suroboyo dan bapak Muhammad Anwari selaku manajer Proyek BlankOn. Hasil wawancara adalah pengembang mengetahui fitur – fitur apa saja yang akan dikerjakan pada *Software Center*. Kemudian bagaimana pengembangan *Software Center* sejalan dengan pengembangan desktop BlankOn kedepan.

Setelah mendapatkan fitur-fitur dari aplikasi, pengembang melakukan analisis lebih lanjut yaitu merancang mockup antarmuka aplikasi sebagai modal awal pengembangan. Mockup ini kemudian dipresentasikan kepada stakeholder untuk mendapatkan saran dan koreksi.

##### **6.1.2 Implementasi**

Dari hasil mockup yang telah dikoreksi, pengembang melanjutkan tahap

implementasi. Pada tahap ini dilakukan penerjemahan kode menggunakan bahasa pemrograman HTML5, CSS dan Javascript. Penulis memilih menggunakan platform ini berdasarkan petunjuk Manajer Proyek BlankOn agar sesuai dengan pengembangan desktop BlankOn. Setelah pengkodean tampilan (*front-end*) selesai, proses selanjutnya adalah mengerjakan pemograman di balik layar (*back-end*). Pengembang memilih menggunakan bahasa Vala karena bahasa ini mampu menjembatani HTML5 dengan sistem operasi BlankOn.

Antarmuka didesain dengan bantuan bootstrap css agar tampilan menarik dan mudah dimodifikasi. Bootstrap mendukung penggunaan elemen yang dinamis sesuai dengan aturan HTML5. Interaksi dinamis sangat dibutuhkan agar nilai user experience (UX) semakin baik. Javascript dan JQuery mampu menjawab tantangan ini.

Penulis menggunakan AngularJS sebagai framework javascript karena kemudahan teknologi dan hasil user experience (UX) yang baik dari framework ini. Namun dalam perjalanan pengembangan ini penulis menemukan hambatan yang cukup besar. AngularJS membutuhkan akses file secara langsung dan template halaman html yang sangat dinamis. Sayangnya proses ini terbentur dengan aturan webkit yaitu pembatasan akses langsung terhadap file (MDN, 2014). Masalah ini biasa dikenal sebagai *HTTP access control* atau *Cross-Origin Resource Sharing (CORS)*. Akibatnya akses langsung terhadap file html dan json terganggu.

Penulis memutuskan tetap menggunakan angularJS dengan melakukan beberapa perubahan, mencari alternatif untuk memanipulasi aturan *CORS*.

Jawabannya dengan memasukan file yang akan diakses langsung ke dalam halaman sebelumnya. Misal saat mengakses halaman detail aplikasi, halaman ini penulis masukkan kedalam file index.html sebagai script kategori template. File index.html bisa dilihat pada bagian lampiran.

File json sebagai objek yang akan diakses penulis masukkan ke dalam file javascript secara langsung agar tetap bisa diakses oleh angularJS. Contoh file yang berisi data json dapat dilihat pada lampiran AppJson.js.

Setelah desain antarmuka dapat berjalan baik, pengembangan menuju bagian sistem (*back-end*). Pengerjaan dimulai dari mencari cara agar Vala mampu berkomunikasi dengan halaman html. Komunikasi ini berupa saling kirim pengembalian setiap aksi yang dilakukan masing-masing bagian. Penulis menemukan kendala lain akibat kurangnya publikasi program kombinasi Vala dan webkit.

Satu persatu kendala terselesaikan kemudian dipresentasikan kepada pengembang BlankOn untuk dimintai koreksi kemudian dilakukan perbaikan sesuai saran yang diberikan.

Koreksi dan perbaikan ini mencakup desain antarmuka maupun proses *back-end* yang dilakukan demi mencapai aplikasi yang lebih baik. Perbaikan antarmuka pertama adalah anjuran menggunakan bootstarp agar sesuai dengan pengembangan destop BlankOn. Perbaikan antarmuka kedua adalah migrasi bootstrap agar menggunakan versi 3(tiga). Perbaikan ketiga adalah agar mampu menampung perubahan halaman yang lebih dinamis dengan memanfaatkan fitur template dari angularJS.

Perbaikan dari sisi *back-end* juga tidak luput. Perbaikan pertama adalah saran untuk menggunakan pustaka API Vala yang lebih terstruktur sesuai pustaka yang telah disediakan oleh Vala. Perbaikan kedua adalah penggunaan JSCore yang mengacu pada standar pengkodean sesuai dengan dokumentasi pengembangan aplikasi Gnome. Perbaikan ketiga adalah perbaikan bentuk program angularJS yaitu `app.js` dan `controller.js` dibuat terpisah sesuai manual pemograman angularJS. Perbaikan keempat meski belum sempurna meliputi komunikasi antara JSCore dengan angular agar lebih dinamis dan mudah dirawat oleh pengembang lainnya.

Setelah tiga kali revisi antarmuka dan empat kali revisi model pengkodean *back-end* akhirnya *prototype* aplikasi *software center* dinyatakan layak digunakan oleh klien (dalam hal ini manajer rilis dan manajer proyek BlankOn) sehingga status pengembangan selesai. Dilanjutkan menuju state selanjutnya yaitu proses implementasi secara langsung pada BlankOn untuk dilakukan pengujian selanjutnya.

## **6.2 Hasil Pengujian Sistem**

### **6.2.1 Hasil Pengujian Alpha Testing**

Pengujian Alpha *testing* dilakukan dengan mengajukan kuesione berisi 5 poin penting mengenai sistem kepada responden untuk disini, tentunya setelah sistem didemokan di depan para responden. Daftar penguji ahli dapat dilihat pada Tabel 6.1.

Tabel 6.1. Daftar Penguji Ahli

No	Nama	Pekerjaan
1	Ahmad Haris	Manajer Rilis BlankOn
2	M. Anwari	Manajer Proyek BlankOn
3	Mahyudin Idam Ramli	Pemaket BlankOn
4	Kukuh Syafaat	Kesenian BlankOn

Adapun lembar kuesioner dan hasil dari pengujian *Alpha testing* terhadap sistem dapat dilihat pada Tabel 6.2.

Tabel 6.2. Hasil Pengujian Alpha Testing

KASUS DAN HASIL UJI				
Parameter Uji	Yang diharapkan	Pengamatan	Ya	Tidak
Menekan tombol menu muka	Menampilkan halaman muka	Dapat menampilkan halaman muka	4	0
Menekan tombol menu kategori	Menampilkan halaman kategori aplikasi	Dapat menampilkan halaman kategori	4	0
Memasukkan kata kunci dikolom pencari	Menampilkan halaman hasil pencarian	Dapat menampilkan halaman hasil pencarian	4	0
Menekan tombol menu kategori Aksesoris	Menampilkan halaman aplikasi dengan kategori Aksesoris	Dapat menampilkan halaman kategori Aksesoris	4	0
Menekan tombol menu kategori Internet	Menampilkan halaman aplikasi dengan kategori Internet	Dapat menampilkan halaman kategori Internet	4	0
Menekan tombol menu kategori Audio Video	Menampilkan halaman aplikasi dengan kategori Audio Video	Dapat menampilkan halaman kategori Audio Video	4	0
Menekan tombol menu kategori Office	Menampilkan halaman aplikasi dengan kategori Office	Dapat menampilkan halaman kategori Office	4	0
Menekan ikon aplikasi	Menampilkan keterangan aplikasi	Dapat menampilkan halaman aplikasi terpilih.	4	0

Menekan ikon pasang/hapus aplikasi	Menampilkan halaman persetujuan pemasangan/penghapusan aplikasi	Dapat memasang/menghapus aplikasi	4	0
Menekan tombol menu perbaharui	Menampilkan halaman menu perbaharui	Dapat menampilkan halaman perbaharui	4	0
Menekan tombol ubah <i>repository</i>	Menampilkan halaman pengubah <i>repository</i>	Dapat mengubah repostori dan menyimpannya	4	0
<b>Jumlah</b>			44	0
<b>Keterangan : B = Berhasil    TB= Tidak Berhasil</b>				

Berdasarkan hasil pengujian *alpha* oleh para ahli, secara keseluruhan sistem yang dibangun dapat dikatakan berhasil atau diterima, karena secara fungsional sistem sudah dapat bekerja dan menghasilkan keluaran (output) yang diharapkan.

### 6.2.2 Hasil Pengujian Beta Testing

Pengujian *Beta testing* dilakukan dengan mengajukan kuesioner berisi 5 poin penting mengenai sistem kepada responden untuk disini, tentunya setelah sistem didemokan di depan para responden. Daftar penguji *Beta testing* dapat dilihat pada Tabel 6.3.

*Tabel 6.3. Daftar Penguji Beta*

No	Nama	Pekerjaan
1	Muhammad Khayat	Staff IT Infest Yogyakarta
2	Muhammad Amirudin Latief	Heldesk Jogja Goes Open Source
3	Yudi Setiyadi	Blogger Pengguna Linux
4	Cindy Prastica Adhi	Mahasiswa STIE YKPN Yogyakarta
5	Jauharoh Pratami	Programer Web

6	Wahyu Bimo S	IT Instruktur, Sys Admin
7	Dedy Hariyadi	Mahasiswa Magister T. Informatika UII
8	Jauhari Ramadhan	Siswa Kelas X SMK Muh 1 Yogyakarta
9	Rizqa Gardha M	Guru TIK SMA Muh 3 Yogyakarta
10	Aziz Ardiansyah W.	Java Developer

Adapun lembar kuesioner dan hasil dari pengujian *Beta testing* terhadap sistem dapat dilihat pada Tabel 6.4. dan Tabel 6.5

*Tabel 6.4: Pengujian Usabilitas Aplikasi*

No	Pertanyaan	Penilaian				
		SS	S	N	TS	STS
1	Saya memahami cara menggunakan aplikasi <i>software center</i>	10	0	0	0	0
2	Saya merasa mudah memasang dan menghapus aplikasi di Linux BlankOn setelah menggunakan <i>software center</i>	8	2	0	0	0
3	Saya setuju <i>software center</i> ini memiliki tampilan yang menarik	8	2	0	0	0
4	Saya berkeinginan menggunakan <i>software center</i> ini kembali	6	4	0	0	0
<b>Jumlah</b>		32	8	0	0	0

*Tabel 6.5: Pengujian Fungsionalitas Aplikasi*

No.	Pertanyaan	Penilaian	
		Ya	Tidak
1	Software Center dapat menampilkan halaman muka	10	0
2	Software Center dapat mencari aplikasi berdasarkan kata kunci	10	0
3	Software Center dapat menampilkan halaman kategori	10	0
4	Software Center dapat menampilkan halaman kategori aksesoris	10	0
5	Software Center dapat menampilkan halaman kategori office	10	0



6	Software Center dapat menampilkan halaman kategori Internet	10	0
7	Software Center dapat menampilkan halaman kategori audio video	10	0
8	Software Center dapat menampilkan halaman perbaharui	10	0
9	Software Center dapat memasang aplikasi terpilih	10	0
10	Software Center dapat menghapus aplikasi terpilih	10	0
11	Software Center dapat mengubah sumber <i>repository</i>	10	0
<b>Jumlah</b>		110	0

### 6.2.3 Perhitungan Hasil Beta Testing

Presentase hasil pengujian *usability* sistem :

- Sangat setuju :  $(32/40) * 100 = 80\%$
- Setuju :  $(8/40) * 100 = 20\%$
- Netral :  $(0/40) * 100 = 0\%$
- Tidak setuju :  $(0/40) * 100 = 0\%$
- Sangat tidak setuju :  $(0/40) * 100 = 0\%$

Presentase hasil pengujian fungsionalitas sistem :

- Menjawab YA :  $(110 / 110) * 100 \% = 100\%$
- Menjawab TIDAK :  $(0/ 110) * 100 \% = 0\%$

Berdasarkan hasil pengujian yang melibatkan 10 responden atau pengguna di atas, dapat diketahui bahwa sebagian besar pengguna menyatakan penilaian yang baik terhadap aplikasi *software center* yang telah dibuat, maka didapat hasil pengujian yang menunjukkan bawa 100% pengguna menyatakan fungsionalitas sistem telah berjalan dengan baik dan 0% responden menyatakan fungsional

sistem tidak berjalan dengan baik.

Berdasarkan pengujian dari segi *usability software center*, diperoleh kesimpulan bahwa sebagian besar responden puas dengan aplikasi yang dibuat. Data hasil pengujian *usability* bahwa responden sangat setuju sebanyak 80%, setuju sebanyak 20%, netral 0%, yang menyatakan tidak setuju sebanyak 0% dan yang menyatakan sangat tidak setuju sebanyak 0%.

Berdasarkan hasil pengujian tersebut, dapat disimpulkan bahwa *software center* yang telah dibuat ini layak untuk digunakan. Akan tetapi perlu adanya pengembangan aplikasi yang lebih lanjut untuk mendapatkan aplikasi yang optimal.

## **BAB 7**

### **KESIMPULAN**

#### **7.1 Kesimpulan**

Berdasarkan kegiatan yang telah dilakukan selama pengembangan *Software Center* menggunakan metode *Evolutionary Prototyping* berbasis HTML5, maka dapat diambil kesimpulan sebagai berikut :

1. Analisis dan pengembangan *Software Center* yang dapat membantu pengguna untuk manajemen aplikasi di Linux BlankOn sudah berhasil dilakukan,
2. Dalam penelitian ini bahasa pemrograman HTML5 mampu digunakan untuk pengembangan aplikasi desktop,
3. Berdasarkan hasil pengujian menunjukkan bahwa *Software Center* dapat berjalan di Linux BlankOn.

#### **7.2 Saran**

*Software Center* ini tentunya tidak terlepas dari kekurangan dan kelemahan. Oleh karena itu, untuk kebaikan pengembangan sistem selanjutnya, maka penulis menyarankan beberapa hal, diantaranya:

1. Perlu adanya penambahan data daftar aplikasi beserta informasi detail lainnya. Sebagai alternatif bisa dimulai oleh tim pemaket aplikasi *repository* untuk mulai melakukan pendataan aplikasi,
2. Perlu adanya metode komunikasi antara JSCore dan HTML5 yang lebih baik agar kode *Software Center* bisa menjadi sangat dinamis,

3. Kendala awal dalam pengembangan ini adalah adanya keterbatasan CORS. Perlu solusi yang lebih baik agar akses file langsung tidak menemukan kendala,
4. Perlu adanya otomasi untuk memperbaharui daftar aplikasi beserta versi terbaru tiap aplikasi,
5. Perlu adanya perubahan lebih lanjut agar *Software Center* ini bisa diimplementasikan pada distribusi Linux lainnya,
6. Penambahan status bar progres pemasangan aplikasi agar *Software Center* lebih mudah digunakan.

## DAFTAR PUSTAKA

- Anwari, Muhammad. (2013). *GNOME 3 Application Development Beginner*. Birmingham: Packt Publishing Ltd.
- Barth, Adam. (2012). *How WebKit Works*. WebKit.
- Carter, Ryan A., Antón, Annie I., Dagnino, Aldo., Williams, Laurie. (2001). *Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model*. North Carolina State University, .
- Dharwiyanti, S., & Wahono, R. S. (2003). *Pengantar Unified Modeling Language (UML)*. Jakarta: IlmuKomputer.
- Hendarto, Handri. (2007). *GUI: Pengembangan Graphical User Interface (GUI) Untuk Operator Training Simulator (OTS)*. Bandung: Institut Teknologi Bandung.
- Hildebrand, Nikolaus & Bode, Nicolai . (2013). *The Apple App Store - Mechanism of Technology Adoption*. Munchen: Parel To.
- Hughes, Ricard. (2008, Februari 24). *PackageKit: Package Abstraction Framework*. Mei 24, 2013. <http://hughsie.com/public/introduction-to-packagekit.pdf>.
- Ismail, Adhi Kusna & Nur Fajri, Aufar. (2013). *Aplikasi Permainan Casual Game Black Pursuit Berbasis HTML 5*. Jakarta: Universitas Bina Nusantara.
- Kadir, Abdul. (2009). *Mastering Ajax dan PHP*. Yogyakarta: Andi Yogyakarta.
- Kadir, Abdul. (2013). *From Zero To A Pro HTML5*. Yogyakarta: Andi Publisher.
- Klumpp, Matthias. (2012, Agustus 30). *GSOC Appstream Final Report*. April 17, 2013. <http://blog.tenstral.net/2012/08/gsoc-appstream-final-report.html>.
- Mac-Vicar P., Duncan . (2008). *What you should know about YaST*. Novell, Inc.
- Mozilla Developer Network. (2014, April 18). *HTTP access control (CORS)*. April 20, 2014. [https://developer.mozilla.org/en-US/docs/Web/HTTP/Access\\_control\\_CORS](https://developer.mozilla.org/en-US/docs/Web/HTTP/Access_control_CORS).
- Paul Thomas, Mathew. (2005, Agustus 29). *Ubuntu Software Center*. April 18, 2013. <http://wiki.ubuntu.com/SoftwareCenter>.
- Pratami, Jauharoh. (2013). *Rancang Bangun Game Hijaiyah Match dengan HTML5*. Yogyakarta: Universitas Islam Negeri Sunan Kalijaga.
- Purnomo, Agus Dkk. (2012). *Panduan BlankOn Rote*. Jakarta: YLPI.
- Raharja, R. Anthon dkk. (2001). *Modul Pelatihan Pengenalan Linux*. Bandung: Telemactics Indonesia.

- Research In Motion. (2010). *BlackBerry App World Storefront Version: 2.0 User Guide*. Research In Motion Limited.
- Salim, Hendri. (2013). *App Store vs Google Play, Ronde 1 Fight!* . Desember, 18 2013. <http://id.gamesinasia.com/app-store-vs-google-play-ronde-1-fight/>.
- Stroik, Brian . (2010). *Why a Mock Up, Because the Owner Expects it Done Right* . Oscar J. Boldt Construction.
- Wahlin, Dan. (2013). *AngularJS in 60 Minutes*. Arizona: Wahlin Consulting.
- Wheeler, David A.. (2000). *Why Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!*. California: Self-Publisher.
- Šrain, Jiří. (2006). *YaST Architecture*. :Novel, Inc.

## LAMPIRAN

### *Lampiran 1. Main.vala*

```
using GLib;
using Gtk;
using WebKit;
using JSCore;

namespace Warung {
    private const string TITLE = "Warung Aplikasi BlankOn";
    int main (string[] args) {
        Gtk.init (ref args);
        var webView = new Main ();
        var window = new Gtk.Window();
        window.add(webView);
        window.show_all ();
        window.resize(1024, 600);
        window.title = TITLE;
        window.window_position = WindowPosition.CENTER;
        try {
            window.icon = new Gdk.Pixbuf.from_file
(Environment.get_current_dir() + "/public/favicon.ico");
        } catch (Error e) {
            stderr.printf ("Could not load application icon: %s\n",
e.message);
        }
        window.destroy.connect(Gtk.main_quit);
        Gtk.main ();
        return 0;
    }
}
```

### *Lampiran 2: Page.vala*

```
using GLib;
using Gtk;
using WebKit;
using JSCore;

namespace Warung {
    public class Main : WebKit.WebView {
        public void * context { get; set; }
        public Main ()
        {
            var test = new App();
            load_uri ("file://" + Environment.get_current_dir() +
"/public/index.html");
            window_object_cleared.connect ((frame, context) => {
```

```

        test.setup_js_class ((JSCore.GlobalContext) context);
    });
}
}}
```

### ***Lampiran 3: App.vala***

```

using GLib;
using Gtk;
using WebKit;
using JSCore;

namespace Warung {

    public class App : GLib.Object {

        // Functions

        public static JSCore.Value repougum (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            //its need root previlage
            try {
                Process.spawn_command_line_sync ("sudo sh
        "+Environment.get_current_dir()+"/belajar/lib/echougum");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value repokambing (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            //its need root previlage
            try {
                Process.spawn_command_line_sync ("sudo sh
        "+Environment.get_current_dir()+"/belajar/lib/echokambing");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }
    }
}
```



```

        public static JSCore.Value repoarsip (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            //its need root previlage
            try {
                Process.spawn_command_line_sync ("sudo sh
"+Environment.get_current_dir()+"/belajar/lib/echoarsip");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value update (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {
            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get update -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value upgrade (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {
            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get dist-upgrade -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installapp (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;

```

```

        //var nama = "braserero";
        var text = "braserero";
        var nama = text;
        //run program with root previlage or run
gksudo, its show a box root
        try {
            Process.spawn_command_line_async (@"sudo
apt-get install $nama -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    /*
    ===== static function =====
    */

    public static JSCore.Value installbraserero (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get install braserero -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value installaudacious (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get install audacious -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value installvlc (Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,

```

```

        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install vlc -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installtotem (Context ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install totem -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installfirefox (Context
ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install firefox -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installchromium (Context
ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install chromium -y");

```

```

        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value installpidgin (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get install pidgin -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value installfilezilla (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get install filezilla -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value installgeany (Context ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get install geany -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }
}

```

```

        public static JSCore.Value installgparted (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install gparted -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installshotwell (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install shotwell -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installstardict (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install stardict -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installlibreoffice
(Context ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,

```

```

        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install libreoffice -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value installevolution (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get install evolution -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        /*
        ===== static remove =====
        */

        public static JSCore.Value removebrasero (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get remove brasero -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value removeaudacious (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

```

```

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove audacious -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value removevlc (Context ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove vlc -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value removetotem (Context ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove totem -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value removefirefox (Context
ctx,
JSCore.Object function,
JSCore.Object thisObject,
JSCore.Value[] arguments,
out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove firefox -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);

```

```

    }
    return new JSCore.Value.undefined (ctx);
}

public static JSCore.Value removechromium (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

    exception = null;
    try {
        Process.spawn_command_line_async ("sudo
apt-get remove chromium -y");
    } catch (SpawnError e) {
        stdout.printf ("Error: %s\n", e.message);
    }
    return new JSCore.Value.undefined (ctx);
}

public static JSCore.Value removepidgin (Context ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

    exception = null;
    try {
        Process.spawn_command_line_async ("sudo
apt-get remove pidgin -y");
    } catch (SpawnError e) {
        stdout.printf ("Error: %s\n", e.message);
    }
    return new JSCore.Value.undefined (ctx);
}

public static JSCore.Value removefilezilla (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

    exception = null;
    try {
        Process.spawn_command_line_async ("sudo
apt-get remove filezilla -y");
    } catch (SpawnError e) {
        stdout.printf ("Error: %s\n", e.message);
    }
    return new JSCore.Value.undefined (ctx);
}

public static JSCore.Value removegeany (Context ctx,
    JSCore.Object function,

```



```

        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get remove geany -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value removegparted (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get remove gparted -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value removeshotwell (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;
            try {
                Process.spawn_command_line_async ("sudo
apt-get remove shotwell -y");
            } catch (SpawnError e) {
                stdout.printf ("Error: %s\n", e.message);
            }
            return new JSCore.Value.undefined (ctx);
        }

        public static JSCore.Value removestardict (Context
ctx,
        JSCore.Object function,
        JSCore.Object thisObject,
        JSCore.Value[] arguments,
        out JSCore.Value exception) {

            exception = null;

```

```

        try {
            Process.spawn_command_line_async ("sudo
apt-get remove stardict -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value removelibreoffice (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove libreoffice -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    public static JSCore.Value removeevolution (Context
ctx,
    JSCore.Object function,
    JSCore.Object thisObject,
    JSCore.Value[] arguments,
    out JSCore.Value exception) {

        exception = null;
        try {
            Process.spawn_command_line_async ("sudo
apt-get remove evolution -y");
        } catch (SpawnError e) {
            stdout.printf ("Error: %s\n", e.message);
        }
        return new JSCore.Value.undefined (ctx);
    }

    /*
    ===== end of static function
    */

    static const JSCore.StaticFunction[] js_funcs = {
        { "ugm", repoug, PropertyAttribute.ReadOnly },
        { "arsip", repoarsip,
PropertyAttribute.ReadOnly },
        { "kambing", repokambing,
PropertyAttribute.ReadOnly },
        { "update", update,
PropertyAttribute.ReadOnly },
        { "upgrade", upgrade,

```

```

PropertyAttribute.ReadOnly },
    { "installapp", installapp,
PropertyAttribute.ReadOnly },
    { "installbrasero", installbrasero,
PropertyAttribute.ReadOnly },
    { "installaudacious", installaudacious,
PropertyAttribute.ReadOnly },
    { "installvlc", installvlc,
PropertyAttribute.ReadOnly },
    { "installtotem", installtotem,
PropertyAttribute.ReadOnly },
    { "installfirefox", installfirefox,
PropertyAttribute.ReadOnly },
    { "installchromium", installchromium,
PropertyAttribute.ReadOnly },
    { "installpidgin", installpidgin,
PropertyAttribute.ReadOnly },
    { "installgeany", installgeany,
PropertyAttribute.ReadOnly },
    { "installgparted", installgparted,
PropertyAttribute.ReadOnly },
    { "installshotwell", installshotwell,
PropertyAttribute.ReadOnly },
    { "installstardict", installstardict,
PropertyAttribute.ReadOnly },
    { "installlibreoffice", installlibreoffice,
PropertyAttribute.ReadOnly },
    { "installevolution", installevolution,
PropertyAttribute.ReadOnly },
    { "removebrasero", removebrasero,
PropertyAttribute.ReadOnly },
    { "removeaudacious", removeaudacious,
PropertyAttribute.ReadOnly },
    { "removevlc", removevlc,
PropertyAttribute.ReadOnly },
    { "removetotem", removetotem,
PropertyAttribute.ReadOnly },
    { "removefirefox", removefirefox,
PropertyAttribute.ReadOnly },
    { "removechromium", removechromium,
PropertyAttribute.ReadOnly },
    { "removepidgin", removepidgin,
PropertyAttribute.ReadOnly },
    { "removegeany", removegeany,
PropertyAttribute.ReadOnly },
    { "removegparted", removegparted,
PropertyAttribute.ReadOnly },
    { "removeshotwell", removeshotwell,
PropertyAttribute.ReadOnly },
    { "removestardict", removestardict,
PropertyAttribute.ReadOnly },
    { "removelibreoffice", removelibreoffice,
PropertyAttribute.ReadOnly },
    { "removeevolution", installevolution,
PropertyAttribute.ReadOnly },
    { null, null, 0 }

```

```

};

static const ClassDefinition js_class = {
    0, // version
    ClassAttribute.None, // attribute
    "Apps", // className
    null, // parentClass
    null, // static values
    js_funcs, // static functions
    null, // initialize
    null, // finalize
    null, // hasProperty
    null, // getProperty
    null, // setProperty
    null, // deleteProperty
    null, // getPropertyNames
    null, // callAsFunction
    null, // callAsConstructor
    null, // hasInstance
    null // convertToType
};

public void setup_js_class (GlobalContext context) {
    var theClass = new Class (js_class);
    var theObject = new JSCore.Object (context,
theClass, context);
    var theGlobal = context.get_global_object ();
    var id = new String.with_utf8_c_string
("Apps");
    theGlobal.set_property (context, id, theObject,
PropertyAttribute.None, null);
}
}
}

```

#### ***Lampiran 4: Echo Repository***

```

cp /etc/apt/sources.list /etc/apt/sources.list.ori
echo "## Repository BlankOn Suroboyo - Original" | tee
/etc/apt/sources.list
echo "deb http://arsip.blankonlinux.or.id/blankon suroboyo main
restricted extras extras-restricted" | tee -a
/etc/apt/sources.list
echo "deb http://arsip.blankonlinux.or.id/blankon suroboyo-
updates main restricted extras extras-restricted" | tee -a
/etc/apt/sources.list
echo "deb http://arsip.blankonlinux.or.id/blankon suroboyo-
security main restricted extras extras-restricted" | tee -a
/etc/apt/sources.list

```

## Lampiran 5: Index.html

```
<!DOCTYPE html>
<html lang="en" ng-app="WarsiApp">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <meta name="description" content="Warung Aplikasi, Sedia
software untuk BlankOn Linux">
    <meta name="keywords" content="HTML, CSS, JS, JavaScript,
Vala, Webkit, BlankOn, Package, Manager, WarSi">
    <meta name="author" content="Estu Fardani
<andro.medh4@gmail.com">">
    <link rel="shortcut icon" href="favicon.ico">

    <title>WarSi | Warung Aplikasi BlankOn</title>
    <!-- Bootstrap core CSS -->
    <link href="css/bootstrap.css" rel="stylesheet">
    <link href="css/carousel.css" rel="stylesheet">
    <link href="css/docs.min.css" rel="stylesheet">
    <link href="css/xeditable.css" rel="stylesheet">
  </head>
  <body>

<!-- Docs master nav header
===== -->
    <a class="sr-only" href="#content">Skip to main content</a>

    <!-- Docs master nav -->
    <header class="navbar navbar-static-top bs-docs-nav" id="top"
role="banner">
      <div class="container">
        <div class="navbar-header">
          <button class="navbar-toggle" type="button" data-
toggle="collapse" data-target=".bs-navbar-collapse">
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
          </button>
          <a class="navbar-brand" href="#/home">Warung Aplikasi</a>
        </div>

        <nav class="collapse navbar-collapse bs-navbar-collapse"
role="navigation">
          <ul class="nav navbar-nav navbar-left">
            <li><a href="#/home">Muka</a></li>
            <li><a href="#/kategori">Kategori</a></li>
            <li><a href="#/repo">Repository</a></li>
            <li><a href="#/cari">Pencarian</a></li>
          </ul>
```

```

    </nav>
  </div>
</header>

<!-- Docs page layout
===== -->
<div class="bs-docs-header" id="content">
  <div class="container">
    <center><h2>BlankOn Linux</h2>
    100% buatan Putra-Putri Indonesia.
    </center>
  </div>
</div>
<div class="container bs-docs-container">
  <div class="row">
    <div class="col-md-9" role="main">
      <div ng-view></div>
    </div>

<!-- Sidebar
===== -->
    <div class="col-md-3">
      <div class="bs-docs-sidebar hidden-print"
role="complementary">
        <div class="bs-docs-section">
          <h3 id="info" class="page-header">Detail Pengguna</h3>
          <center></center><p></p>
          <div class="highlight"><pre><code class="bash">
Device   : BlankOn
Memory   : 4.0 GB
Prosesor: Intel Core 2 Duo
Grafik   : Gallium 0.5 AMD
Base     : BlankOn Linux
Rilis    : Suroboyo 64-bit
          </code></pre></div>

          </script>
        </div>
      </div>
    </div>

  </div>
</div>

<!-- Footer
===== -->
<footer class="bs-docs-footer" role="contentinfo">
  <div class="container">
    <p>Didesain sebagai pengabdian untuk Proyek BlankOn oleh <a
href="http://twitter.com/tuanpembual"
target="_blank">@tuanpembual</a>.</p>
    <p>Lisensi kode sumber dibawah <a
href="https://github.com/tuanpembual/06alpa01/blob/master/LICENSE
" target="_blank">MIT</a>, dokumentasi dibawah <a

```

```

href="http://creativecommons.org/licenses/by/3.0/">CC BY
3.0</a>.</p>
  <ul class="bs-docs-footer-links muted">
    <li>Versi Terakhir v0.1</li>
    <li>&middot;</li>
    <li><a
href="https://github.com/tuanpembual/06alpa01">GitHub</a></li>
  </ul>
  <p>&copy; 2005-2014, YPLI dan Tim Pengembang BlankOn</p>
</div>
</footer>

<!-- Kumpulan Script -->

<!-- Script Home
===== -->
  <script type="text/ng-template" id="home.html">
    <div class="bs-docs-section">
      <h2 align="center" class="page-header">Selayang
Pandang</h2>
      <p><i>Apa yang baru di BlankOn Suroboyo..</i></p>
      <p></p>
    </div>

    <!-- Aplikasi
===== -->
    <div class="bs-docs-section" ng-controller="AppListCtr">
      <h2 align="center" class="page-header">Pilihan
Aplikasi</h2>
      <p><i>Beragam aplikasi yang bisa dipasang hanya dengan satu
tombol..</i></p>
      <div ng-repeat="app in apps | filter:filters">
        <a href="#/detail/{{app.id}}" class="thumb"></a>
      </div>
    </div>
  </script>

<!-- Script Kategori
===== -->
  <script type="text/ng-template" id="kategori.html">
    <div class="bs-docs-section">
      <h3 align="center" class="page-header">Kategori
Aplikasi</h3>
      <div class="row">
        <div class="col-md-3">
          <a href="#/kategori/aksesoris"></a>
          <h3>Aksesoris</h3>
          <a type="button" class="btn btn-outline btn-lg"
href="#/kategori/aksesoris">Lihat detail >></a>
        </div>
        <div class="col-md-3">
          <a href="#/kategori/aksesoris"></a>
    <h3>Internet</h3>
    <a type="button" class="btn btn-outline btn-lg"
href="#/kategori/internet">Lihat detail >></a>
</div>
    <div class="col-md-3">
        <a href="#/kategori/aksesoris"></a>
        <h3>Perkantoran</h3>
        <a type="button" class="btn btn-outline btn-lg"
href="#/kategori/office">Lihat detail >></a>
    </div>
    <div class="col-md-3">
        <a href="#/kategori/aksesoris"></a>
        <h3>Sound & Video</h3>
        <a type="button" class="btn btn-outline btn-lg"
href="#/kategori/audio">Lihat detail >></a>
    </div>
</div>
</script>

<!-- Script Aksesoris
===== -->
    <script type="text/ng-template" id="aksesoris.html">
        <div class="bs-docs-section">
            <h2 align="center" class="page-header">Kategori
Aksesoris</h2>
            <div ng-repeat="app in apps | filter:{genre:'aksesoris'}
| orderBy:'name'" class="thumb">
                <a href="#/detail/{{app.id}}" class="thumb"></a>
                <a href="#/detail/{{app.id}}"><b>{{app.name}}</b></a>
                <p>{{app.info}}</p>
            </div>
        </div>
    </script>

<!-- Script Internet
===== -->
    <script type="text/ng-template" id="internet.html">
        <div class="bs-docs-section">
            <h2 align="center" class="page-header">Kategori
Internet</h2>
            <div ng-repeat="app in apps | filter:{genre:'internet'} |
orderBy:'name'" class="thumb">
                <a href="#/detail/{{app.id}}" class="thumb"></a>
                <a href="#/detail/{{app.id}}"><b>{{app.name}}</b></a>
                <p>{{app.info}}</p>
            </div>
        </div>
    </script>

```



```

<!-- Script Office
===== -->
<script type="text/ng-template" id="office.html">
  <div class="bs-docs-section">
    <h2 align="center" class="page-header">Kategori
Perkantoran</h2>
    <div ng-repeat="app in apps | filter:{genre:'office'} |
orderBy:'name'" class="thumb">
      <a href="#/detail/{{app.id}}" class="thumb"></a>
      <a href="#/detail/{{app.id}}"><b>{{app.name}}</b></a>
      <p>{{app.info}}</p>
    </div>
  </div>
</script>

<!-- Script Audio
===== -->
<script type="text/ng-template" id="audio.html">
  <div class="bs-docs-section">
    <h2 align="center" class="page-header">Kategori
Perkantoran</h2>
    <div ng-repeat="app in apps | filter:{genre:'audio'} |
orderBy:'name'" class="thumb">
      <a href="#/detail/{{app.id}}" class="thumb"></a>
      <a href="#/detail/{{app.id}}"><b>{{app.name}}</b></a>
      <p>{{app.info}}</p>
    </div>
  </div>
</script>

<!-- Script Repository
===== -->
<script type="text/ng-template" id="repo.html">
  <div class="bs-docs-section">
    <h3 align="center" class="page-header">Sumber
Repository</h3>
    <div>
      <h4>Status Repository : <a href="#" editable-
select="user.status" buttons="no" e-ng-options="s.value as s.text
for s in statuses">
        {{showStatus()}}
      </a></h4>
    </div>
    <h4>{{ itemS.konten.komentar }}</h4>
    <h5>{{ itemS.konten.isimain }}</h5>
    <h5>{{ itemS.konten.isiupdate }}</h5>
    <h5>{{ itemS.konten.isisecurity }}</h5>
    <div>
      <button class="btn btn-outline" ng-
click="simpanrepo(itemS.kode)">Simpan</button>
      <button class="btn btn-outline" ng-
click="updaterepo(itemS.kode)">Update</button>
      <button class="btn btn-outline" ng-

```

```

click="upgraderepo(itemS.kode)">Upgrade</button>
</div>
</div>
</script>

<!-- Script Detail Aplikasi | disable state :)
http://getbootstrap.com/css/#buttons-disabled
===== -->
<script type="text/ng-template" id="app-detail.html">
  <div class="bs-docs-section">
    <h3 align="center" class="page-header">Detail Aplikasi</h3>
    <div class="bs-docs-section col-md-6">
      <h3>
      {{app.name}}<br><small>Versi: {{app.versi}}</small></h3>
      <p>{{app.info}}</p>
      <button class="btn btn-primary" ng-
click="installapp(app.id)">Pasang</button>
      <button class="btn btn-primary" ng-
click="removeapp(app.id)">Hapus</button>
      <button class="btn btn-primary" ng-
click="installapp(app.id)">Upgrade</button>
    </div>

    <div class="bs-docs-section col-md-6">
      <h4>Pengembangan {{app.name}}</h4>
      <div ng-repeat="url in app.url">{{url}}</div>
      <h4>{{app.name}} dikembangkan oleh :</h4>
      <ul><li ng-repeat="author in app.authors">
        {{author}}
      </li></ul>
    </div>
  </div>
</script>

<!-- Script Pencarian || Mentah show only tree kode, how?
===== -->
<script type="text/ng-template" id="cari.html">
  <div class="bs-docs-section">
    <h2 align="center" class="page-header">Pencarian</h2>
    <div>
      <form class="col-md-4">
        <input ng-model="query" type="text" typeahead="deret for
deret in derets | filter:$viewValue" class="form-control input-
normal" placeholder="Pencarian...">
      </form>
    </div>
    <h3 align="center" class="page-header"></h3>
    <div ng-repeat="app in apps | filter:query |
orderBy:'name'" class="thumb col-md-4">
      <a href="#/detail/{{app.id}}" class="thumb"></a>
      <a href="#/detail/{{app.id}}"><b>{{app.name}}</b></a>
    </div>
  </div>
</script>
<!-- Bootstrap core JavaScript

```

```

===== -->
<!-- Placed at the end of the document so the pages load faster
-->
<script src="lib/jquery-1.11.0.js"></script>
<script src="lib/bootstrap.js"></script>
<script src="lib/docs.min.js"></script>

<script src="lib/angular.js"></script>
<script src="lib/angular-route.js"></script>
<script src="lib/xeditable.js"></script>
<script src="lib/ui-bootstrap-tpls-0.10.0.js"></script>
<script src="lib/underscore.js"></script>

<script src="js/app.js"></script>
<script src="js/appjson.js"></script>
<script src="js/controller.js"></script>

<script src="js/vala.js"></script>
</body>
</html>

```

### ***Lampiran 6: App.js***

```

'use strict';

/* App Module
Change
phonecatApp => WarsiApp
phonecatControllers => WarsiCtr
*/

var WarsiApp = angular.module('WarsiApp', [
  'ngRoute',
  'WarsiCtr'
]);

WarsiApp.config(['$routeProvider',
  function($routeProvider) {
    $routeProvider.
      when('/home', { //done
        templateUrl: 'home.html',
        controller: 'AppListCtr'
      }).
      when('/kategori', { //done
        templateUrl: 'kategori.html'
      }).
      when('/repo', { //done
        templateUrl: 'repo.html',
        controller: 'RepoCtr'
      }).
      when('/kategori/aksesoris', { //done

```

```

        templateUrl: 'aksesoris.html',
        controller: 'AppListCtr'
    }).
    when('/kategori/internet', { //done
        templateUrl: 'internet.html',
        controller: 'AppListCtr'
    }).
    when('/kategori/office', { //done
        templateUrl: 'office.html',
        controller: 'AppListCtr'
    }).
    when('/kategori/audio', { //done
        templateUrl: 'audio.html',
        controller: 'AppListCtr'
    }).
    when('/detail/:appId', { //done
        templateUrl: 'app-detail.html',
        controller: 'AppDetailCtr'
    }).
    when('/cari', { //done
        templateUrl: 'cari.html',
        controller: 'AppListCtr'
    }).
    otherwise({
        redirectTo: '/home'
    });
}
]);

```

## Lampiran 7: Appjson.js

```

var WarsiJson = angular.module('WarsiJson', []);

WarsiJson.factory('DaftarApp', function() {
    return {
        name : [
            {
                "age": 0,
                "id": "brasero",
                "genre": "audio",
                "imageUrl": "img/Komodo/apps/scalable/brasero.svg",
                "name": "Brasero",
                "snipset": "Pembakar Cakram",
                "info": "Brasero adalah sebuah aplikasi untuk membakar CD /
                DVD untuk Gnome Desktop. Hal ini dirancang untuk menjadi
                sesederhana mungkin dan memiliki beberapa fitur unik yang
                memungkinkan pengguna untuk membuat cakram mereka dengan mudah
                dan cepat.",
                "url" : [
                    "Unduh rilis:
                    http://download.gnome.org/sources/brasero/",

```

```

        "Git: http://git.gnome.org/browse/brasero/",
        "Mailing List: brasero-list@gnome.org",
        "channel IRC: #brasero di irc.gimp.org"],
    "authors" : [
        "Philippe Rouquier",
        "Luis Medinas",
        "Joshua Lock"],
    "versi" : "3.8.0-2"
},
{
    "age": 1,
    "id": "audacious",
    "genre": "audio",
    "imageUrl": "img/Komodo/apps/scalable/audacious.svg",
    "name": "Audacious",
    "snipset": "",
    "info": "Audacious - Pemutar Suara bersumber terbuka. Mampu
memutar musik sesuai keinginan namun tetap tidak membebani kerja
komputer. Audacious berjalan pada Linux, derivatif BSD, dan
Microsoft Windows.",
    "url" :[
        "Unduh rilis: http://audacious-media-player.org",
        "Git: http://github.com/audacious-media-player",
        "channel IRC: #audacious di irc.atheme.org"],
    "authors" : [
        "George Averill",
        "Daniel Barkalow",
        "Christian Birchinger",
        "Ralf Ertzinger",
        "Michael Färber",
        "Matti Hämmäläinen"],
    "versi" : "3.4.1-1"
},
{
    "age": 2,
    "id": "vlc",
    "genre": "audio",
    "imageUrl": "img/Komodo/apps/scalable/vlc.svg",
    "name": "VLC Media Player",
    "snipset": "",
    "info": "VLC media player adalah pemutar video cross-
platform bebas dan opensource. VLC mampu memainkan semua jenis
tipe video, DVD, Audio CD, VCD, dan pelbagai protokol
streaming.",
    "url" :[
        "Unduh rilis: http://www.videolan.org/vlc/",
        "Git: http://git.videolan.org/",
        "channel IRC: IRC on #videolan di irc.videolan.org"],
    "authors" : [
        "VideoLAN non-profit Organization."],
    "versi" : "2.1.2-2"
},
{
    "age": 3,
    "id": "totem",
    "genre": "audio",

```

```

        "imageUrl": "img/Komodo/apps/scalable/totem.svg",
        "name": "Totem",
        "snipset": "",
        "info": "Pemutar Video Original dari GNOME.",
        "url" :[
            "Laman Web: http://wiki.gnome.org/Apps/Totem",
            "Git: http://git.gnome.org/browse/totem/",
            "channel IRC: #totem di irc.gimp.org"],
        "authors" : [
            "The Telestrator mode was sponsored by Paul Cooper",
            "The video zooming feature was sponsored by Lyndon
Drake",
            "The Totem Mozilla plugin was sponsored by Ryan
Thiessen",
            "Fluendo have sponsored Ronald Bultje and Tim-Philipp",
            "Müller's work on Totem's GStreamer backend",
            "Collabora Multimedia sponsors Tim-Philipp Müller's work
on Totem's GStreamer backend now"],
        "versi" : "3.10.1-1"
    },
    {
        "age": 4,
        "id": "firefox",
        "genre": "internet",
        "imageUrl": "img/Komodo/apps/scalable/firefox.svg",
        "name": "Firefox",
        "snipset": "",
        "info": "Firefox sebuah perambah internet dikembangkan oleh
Mozilla, tersedia di semua platform sistem operasi seperti
windows, OS x dan Linux. Pengembangan dibawah naungan Mozilla
Foundation.",
        "url" :[
            "Unduh: http://mozilla.org/firefox"],
        "authors" : [
            "Mozilla.org"],
        "versi" : "3.10.1-1"
    },
    {
        "age": 5,
        "id": "chromium",
        "genre": "internet",
        "imageUrl": "img/Komodo/apps/scalable/chromium.svg",
        "name": "Chromium Browser",
        "snipset": "",
        "info": "Chromium salah satu projek open source perambah
internet yang cepat, aman dan stabil.",
        "url" :[
            "Laman Web: http://chromium.org"],
        "authors" : [
            "Chromium Project"],
        "versi" : "32.0.1700"
    },
    {
        "age": 6,
        "id": "pidgin",
        "genre": "internet",

```

```

        "imageUrl": "img/Komodo/apps/scalable/pidgin.svg",
        "name": "Pidgin",
        "snipset": "",
        "info": "Pidgin - Aplikasi chatting Universal, mampu
menjalankan layanan chatting dari platform lain semisal facebook
messenger, gtalk, yahoo! messenger dan lain-lain.",
        "url" :[
            "Laman Web: http://www.pidgin.im"],
        "authors" : [
            "Pidgin Dev"],
        "versi" : "2.10.9-1"
    },
    {
        "age": 7,
        "id": "filezilla",
        "genre": "internet",
        "imageUrl": "img/Komodo/apps/scalable/filezilla.svg",
        "name": "FileZilla",
        "snipset": "",
        "info": "Open-source (GNU GPL) File Transfer Protokol (FTP)
client untuk Windows, Mac OS X and GNU/Linux.",
        "url" :[
            "Laman Web: http://filezilla-project.org"],
        "authors" : [
            "Tim Kosse"],
        "versi" : "3.7.3"
    },
    {
        "age": 8,
        "id": "geany",
        "genre": "aksesoris",
        "imageUrl": "img/Komodo/apps/scalable/geany.svg",
        "name": "Geany",
        "snipset": "",
        "info": "Geany adalah teks editor dengan toolkit GTK2 yang
yang terhubung dengan integrated development environment(IDE).
Hal ini dikembangkan untuk memberikan IDE kecil dan cepat, hanya
memiliki beberapa dependensi dari paket lain. Geany mendukung
banyak tipe file dengan beberapa fitur yang bagus.",
        "url" :[
            "Laman Web: http://www.geany.org"],
        "authors" : [
            "Colomban Wendling <lists.ban@herbesfolles.org>
(maintainer, developer)",
            "Nick Treleaven <nick.treleaven@btinternet.com>
(developer)",
            "Matthew Brush <matt@geany.org> (developer)",
            "Enrico Tröger <enrico.troeger@uvena.de> (developer)",
            "Frank Ianitz <frank@frank.uvena.de> (translation
maintainer)",
            "Lex Trotman <elextr@gmail.com> (Regular Contributors)",
            "Eugene Arshinov <earshinov@gmail.com> (Regular
Contributors)"],
        "versi" : "1.23.1"
    },
    {

```

```

    "age": 9,
    "id": "gparted",
    "genre": "aksesoris",
    "imageUrl": "img/Komodo/apps/scalable/gparted.svg",
    "name": "Gparted",
    "snipset": "",
    "info": "GParted adalah partisi editor GUI untuk mengatur
partisi hardisk anda. Dengan GParted Anda dapat mengubah ukuran,
menyalin, dan memindahkan partisi tanpa kehilangan data,
memungkinkan Anda untuk: Menambah atau mengurangi ukuran drive C:
Menciptakan ruang untuk sistem operasi baru, Mencoba penyelamatan
data dari partisi yang hilang",
    "url" :[
        "Laman Web: http://gparted.org/",
        "Git : http://gparted.org/git.php"],
    "authors" : [
        "Curtis Gedak (GParted Application Manager)"],
    "versi" : "0.18.0"
},
{
    "age": 10,
    "id": "shotwell",
    "genre": "aksesoris",
    "imageUrl": "img/Komodo/apps/scalable/shotwell.svg",
    "name": "Shotwell",
    "snipset": "",
    "info": "Photo organizer untuk desktop GNOME. Fitur :
(1)Impor dari disk atau kamera,(2)Mengatur dengan berbasis waktu
Events, Tag (kata kunci), Folders, dan lebih,(3)Lihat foto Anda
dalam full-window atau mode fullscreen,(4)Crop, rotate, warna
menyesuaikan, meluruskan, dan meningkatkan foto slideshow,
(5)Video dan dukungan foto RAW, (6)Berbagi ke layanan Web utama,
termasuk Facebook, Flickr, dan YouTube",
    "url" :[
        "Laman Web: http://www.yorba.org/projects/shotwell"],
    "authors" : [
        "Jim Nelson - Executive Director",
        "Charles - ",
        "Adam -",
        "Dave Neary"],
    "versi" : "0.15.1"
},
{
    "age": 11,
    "id": "stardict",
    "genre": "aksesoris",
    "imageUrl": "img/Komodo/apps/scalable/accessories-
dictionary.svg",
    "name": "Stardict",
    "snipset": "",
    "info": "Program kamus penterjemah terbaik di Linux dan
Windows.",
    "url" :[
        "Laman Web: http://code.google.com/p/stardict-3/"],
    "authors" : [
        "Stardict3 Team"],

```



```

        "versi" : "3.0.4"
    },
    {
        "age": 12,
        "id": "libreoffice",
        "genre": "office",
        "imageUrl": "img/Komodo/apps/scalable/libreoffice-
main.svg",
        "name": "LibreOffice",
        "snipset": "",
        "info": "LibreOffice adalah 'powerful office suite',
antarmuka yang bersih, tool yang lengkap. LibreOffice terdiri
dari beberapa aplikasi, diantaranya: Writer, pengolah kata, Calc,
aplikasi spreadsheet, Impress, mesin presentasi, Menggambar,
menggambar dan aplikasi flowcharting, Base, database kami dan
frontend database, dan matematika untuk mengedit matematika.",
        "url" : [
            "Laman Web: http://www.libreoffice.org"],
        "authors" : [
            "The Document Foundation"],
        "versi" : "4.1.4"
    },
    {
        "age": 13,
        "id": "evolution",
        "genre": "office",
        "imageUrl": "img/Komodo/apps/scalable/evolution.svg",
        "name": "Evolution",
        "snipset": "",
        "info": "Aplikasi manajemen informasi pribadi terintegrasi
dengan surel, kalender dan buku kontak.",
        "url" : [
            "Laman Web: http://wiki.gnome.org/Apps/Evolution"],
        "authors" : [
            "Matthew Barnes <mbarnes@redhat.com>",
            "Milan Crha <mcrha@redhat.com>",
            "Fabiano Fidêncio <fabiano@fidencio.org>"],
        "versi" : "3.10.3"
    }
    ]});
});

WarsiJson.factory('DaftarRepo', function() {
    return {
        name : [
            {
                "kode": "arsip",
                "konten" : {
                    "komentar": "## Repository BlankOn Suroboyo - Original",
                    "isimain": "deb
http://arsip.blankonlinux.or.id/blankon/suroboyo main restricted
extras extras-restricted",
                    "isiupdate": "deb http://arsip.blankonlinux.or.id/blankon
suroboyo-updates main restricted extras extras-restricted",
                    "isisecurity": "deb
http://arsip.blankonlinux.or.id/blankon suroboyo-security main

```

```

restricted extras extras-restricted"
    }
  },
  {
    "kode": "kambing",
    "konten" : {
      "komentar": "## Repository BlankOn Suroboyo - Kambing",
      "isimain": "deb http://kambing.ui.ac.id/blankon/suroboyo
main restricted extras extras-restricted",
      "isiupdate": "deb http://kambing.ui.ac.id/blankon
suroboyo-updates main restricted extras extras-restricted",
      "isisecurity": "deb http://kambing.ui.ac.id/blankon
suroboyo-security main restricted extras extras-restricted"
    }
  },
  {
    "kode": "repougum",
    "konten" : {
      "komentar": "## Repository BlankOn Suroboyo - repougum",
      "isimain": "deb http://repo.ugm.ac.id/blankon/suroboyo
main restricted extras extras-restricted",
      "isiupdate": "deb http://repo.ugm.ac.id/blankon suroboyo-
updates main restricted extras extras-restricted",
      "isisecurity": "deb http://repo.ugm.ac.id/blankon
suroboyo-security main restricted extras extras-restricted"
    }
  }
  ]};
});

```

### ***Lampiran 8: Controller.js***

```

'use strict';

function install0() {
  (Apps.install0());
}

/* Controllers */

var WarsiCtr = angular.module('WarsiCtr',
['WarsiJson','xeditable','ui.bootstrap']);

WarsiCtr.run(function(editableOptions) {
  editableOptions.theme = 'bs3'; // bootstrap3 theme. Can be also
  'bs2', 'default'
});

WarsiCtr.controller('History', ['$scope',
function($scope) {
  $scope.$back = function() {

```

```

        window.history.back();
    }
}]);

WarsiCtr.controller('AppListCtr', ['$scope', 'DaftarApp',
function($scope, DaftarApp) {
    $scope.selected = undefined;
    $scope.apps = DaftarApp.name;
    $scope.derets
= ['brasero', 'audacious', 'vlc', 'totem', 'firefox', 'chromium', 'pidgi
n', 'filezilla', 'geany', 'gparted', 'shotwell', 'stardict', 'libreoffi
ce', 'evolution'
    ];
    $scope.orderProp = 'age';
    $scope.aksi = $scope.selected;
}
]);

// hack cors | where the json
WarsiCtr.controller('AppDetailCtr', ['$scope',
'$routeParams', 'DaftarApp',
function($scope, $routeParams, DaftarApp) {
    $scope.user = $routeParams.appId;
    $scope.isis = DaftarApp.name;
    $scope.apps = "";
    var itemSelected = _.find($scope.isis, function(item) {
        return item.id === $routeParams.appId;
    });
    $scope.app = itemSelected;
    $scope.install = function(app) {
        alert(app)
    }
    $scope.installapp = function(app) {
        if (app === "brasero") {
            (Apps.installbrasero())
        } else if (app === "audacious") {
            (Apps.installaudacious())
        } else if (app === "vlc") {
            (Apps.installvlc())
        } else if (app === "totem") {
            (Apps.installtotem())
        } else if (app === "firefox") {
            (Apps.installfirefox())
        } else if (app === "chromium") {
            (Apps.installchromium())
        } else if (app === "pidgin") {
            (Apps.installpidgin())
        } else if (app === "geany") {
            (Apps.installgeany())
        } else if (app === "gparted") {
            (Apps.installgparted())
        } else if (app === "shotwell") {
            (Apps.installshotwell())
        } else if (app === "stardict") {
            (Apps.installstardict())
        } else if (app === "libreoffice") {

```

```

        (Apps.installlibreoffice())
    } else {
        (Apps.installevolution())
    };
}

$scope.removeapp = function(app){
    if (app === "brasero") {
        (Apps.removebrasero())
    } else if (app === "audacious") {
        (Apps.removeaudacious())
    } else if (app === "vlc") {
        (Apps.removevlc())
    } else if (app === "totem") {
        (Apps.removetotem())
    } else if (app === "firefox") {
        (Apps.removefirefox())
    } else if (app === "chromium") {
        (Apps.removechromium())
    } else if (app === "pidgin") {
        (Apps.removepidgin())
    } else if (app === "geany") {
        (Apps.removegeany())
    } else if (app === "gparted") {
        (Apps.removegparted())
    } else if (app === "shotwell") {
        (Apps.removeshotwell())
    } else if (app === "stardict") {
        (Apps.removestardict())
    } else if (app === "libreoffice") {
        (Apps.removelibreoffice())
    } else {
        (Apps.removeevolution())
    };
}
}
});

WarsiCtr.controller('RepoCtr',
function($scope, $filter) {
    $scope.debhttps = [
        {
            "kode": "arsip",
            "konten" : {
                "komentar": "### Repository BlankOn Suroboyo - Original",
                "isimain": "deb
http://arsip.blankonlinux.or.id/blankon/suroboyo main restricted
extras extras-restricted",
                "isiupdate": "deb http://arsip.blankonlinux.or.id/blankon
suroboyo-updates main restricted extras extras-restricted",
                "isisecurity": "deb
http://arsip.blankonlinux.or.id/blankon suroboyo-security main
restricted extras extras-restricted"
            }
        },
        {

```

```

        "kode": "kambing",
        "konten" : {
            "komentar": "## Repository BlankOn Suroboyo - Kambing",
            "isimain": "deb http://kambing.ui.ac.id/blankon/suroboyo
main restricted extras extras-restricted",
            "isiupdate": "deb http://kambing.ui.ac.id/blankon
suroboyo-updates main restricted extras extras-restricted",
            "isisecurity": "deb http://kambing.ui.ac.id/blankon
suroboyo-security main restricted extras extras-restricted"
        }
    },
    {
        "kode": "repougum",
        "konten" : {
            "komentar": "## Repository BlankOn Suroboyo - repougum",
            "isimain": "deb http://repo.ugm.ac.id/blankon/suroboyo
main restricted extras extras-restricted",
            "isiupdate": "deb http://repo.ugm.ac.id/blankon suroboyo-
updates main restricted extras extras-restricted",
            "isisecurity": "deb http://repo.ugm.ac.id/blankon
suroboyo-security main restricted extras extras-restricted"
        }
    }
    ]];
    $scope.user = {
        status: 'arsip'
    };
    //ini status akan dapat kiriman dari JSCore, but how?

    $scope.$watch('user.status',function(newVal,oldVal) {
        $scope.user.status=newVal;
        $scope.itemS = "";
        var itemSelected = _.find($scope.debhttps,function(item) {
            return item.kode === newVal;
        });
        $scope.itemS = itemSelected;
    });

    $scope.statuses = [
        {value: 'arsip', text: 'Arsip BlankOn'},
        {value: 'kambing', text: 'Kambing UI'},
        {value: 'repougum', text: 'Repo UGM'}];

    $scope.showStatus = function() {
        var selected = $filter('filter')($scope.statuses, {value:
$scope.user.status});
        return ($scope.user.status && selected.length) ?
selected[0].text : 'Not set';
    };
    $scope.simpanrepo = function(app) {

        if (app === "repougum") {
            (Apps.ugm())
        } else if (app === "kambing") {
            (Apps.kambing())
        } else {
            (Apps.arsip())
        }
    }

```

```
    };  
  }  
  $scope.updaterepo = function(app) {  
    (Apps.update())  
  }  
  $scope.upgraderepo = function(app) {  
    (Apps.upgrade())  
  }  
});
```

## DAFTAR RIWAYAT HIDUP



Nama : Estu Fardani  
Tempat, tanggal lahir : Tanjung Samak, 10 Mei 1991  
Jenis Kelamin : Laki-laki  
Agama : Islam  
Alamat Asal : Jalan Hang Nadim, RT/RW : 02/03 Sejangat, Bukit Batu, Bengkalis Riau  
No. HP : 085643214898  
Email : estu@jogja.linux.or.id

### Riwayat Pendidikan

1. SD YKPP Sungai Pakning (1996-2002)
2. MTS Bukit Batu (2002-2005)
3. MAN 2 Model Pekanbaru (2005-2008)
4. S1 Teknik Informatika UIN Sunan Kalijaga Yogyakarta (2009-2014)