BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi di era globalisasi sangatlah maju dan cepat, hal ini membuat para individu untuk terus berusaha mengikuti perkembangan teknologi yang ada, sehingga tidak dikatakan sebagai seseorang yang ketinggalan zaman.

Di bidang teknologi komputer membutuhkan sistem operasi untuk menjalankan aplikasi–aplikasi yang telah tersedia. Windows merupakan salah satu sistem operasi, sehingga banyak kalangan atas yang menggunakan Window sebagai sistem operasi untuk menjalankan aplikasi–aplikasi yang ingin digunakan. Tetapi banyak juga kalangan yang tidak mampu membayar Windows untuk menggunakannya sebagai sistem operasi dikarenakan masalah ekonomi. Sebagai salah satu sistem operasi berbasis *open source* saat ini GNU Linux diperuntukkan bagi kalangan yang tidak mampu membayar sistem operasi berbayar dikarenakan GNU Linux berlisensi *free*.

Bukan hanya sistem operasi yang bisa didapatkan secara *free* , tetapi aplikasi–aplikasinya dapat diperoleh sebagian besar secara *free*. GNU Linux memiliki bermacam–macam jenis seperti, Ubuntu, Mandriva, Debian, Redhat, Mandrake, Caldera Open, Slackware, Suse, Corel, Turbo dan Fedora. Setiap jenis GNU Linux memiliki kekurangan dan kelebihan masing–masing. (http://id.wikipedia.org/wiki/Linux)

Musik Player merupakan aplikasi dasar yang didapat pada saat menginstal sistem operasi di komputer. Tapi beberapa *vendor* menginginkan bayaran untuk mendapatkan aplikasi tersebut, saat ini dibuat aplikasi pemutar musik mp3 di GNU Linux jenis Ubuntu berbasis *open source* sehingga tidak perlu membayar untuk mendapatkan aplikasi tersebut karena berlisensi *free*.

Selain masalah lisensi, memori yang digunakan dalam penginstalan aplikasi MP3 yang sudah ada terbilang relatife besar dan membutuhkan waktu yang lama dalam penginstalannya. *Interface* yang dihasilkan kadang termasuk cukup membingungkan para *user* dalam menggunakan aplikasi itu sendiri. Banyaknya paket tambahan yang diperlukan untuk menjalankan aplikasi yang sudah ada, kadang membuat *user* kesulitan dalam meng *update* paket – paket tambahan tersebut.

GNU Linux Ubuntu dipilih sebagai sistem operasi dalam pengerjaannya dikarenakan tampilan desktop nya yang sangat familiar dengan tampilan Windows dan memiliki aplikasi program seperti Gambas *(*Gambas *Almost*

Means BASic) sebagai IDE *(Integrated Development Environment)* dengan basisnya menggunakan bahasa Basic yang jalan di atas *platform* GNU Linux dan bisa didapatkan secara *free*. Judul yang diambil di dalam penelitian ini adalah "Rancang Bangun MP3 Player di GNU LINUX Berbasis Open Source".

1.2 Rumusan Masalah

Di dalam penelitian MP3 Player ini terdapat tiga permasalahan :

1. Aplikasi dibuat agar dapat memutar *file* berfomat MP3 di GNU Linux tanpa menggunakan *space* harddisk yang besar .
2. Cara *input file*, menyimpan *playlist* sesuai dengan *output* yang dikeluarkan.
3. *Implementasi Software* pemrograman *Gambas* (*Gambas Almost Means BASic*) sebagai IDE *(Integrated Development Environment)* dalam membuat program pemutar musik MP3 player

1.3 Batasan Masalah

Di dalam penelitian ini didapat batasan masalah sebagai berikut :

1. Program diperuntukan hanya bisa memutar *file* musik berformat MP3 dan OGG.
2. Bahasa pemrograman yang digunakan adalah *Gambas (*Gambas *Almost Means BASic)* sebagai IDE *(Integrated Development Environment).*
3. Aplikasi ini memiliki fasilitas dasar dalam menjalankan sebuah *file* MP3.
4. Sistem Operasi yang digunakan adalah GNU Linux Ubuntu 9.10

1.4 Tujuan

Program aplikasi ini dibuat bertujuan untuk :

1. *User* dapat memutar *file* berformat MP3 dan OGG di sistem operasi GNU Linux.
2. Program berbasis *open source* sehingga dapat dengan mudah untuk dikembangkan.

3. Program tidak berbayar alias (atau) *free*, sehingga *user* tidak perlu mengeluarkan dana untuk mendapatkannya.
4. Membuat *user* menjadi lebih nyaman dalam menggunakan aplikasi MP3 *player*, karena tidak berurusan dengan hukum yang ada dikarenakan aplikasi tersebut bersifat *legal*.

1.5 **Manfaat**

Di dalam membuat program ini diharapkan agar dapat memperoleh beberapa manfaat sebagai berikut :

1. *User* menjadi lebih nyaman dalam menggunakan aplikasi MP3 *player*, karena aplikasi tersebut bersifat *legal*.
2. Tampilan *interface* yang sangat familiar supaya *user* dapat dengan mudah menggunakan program tersebut.
3. *Space* harddisk yang digunakan relatife kecil maka *user* dapat mendapatkan *free space* hardisk yang lebih besar.
4. Diharapkan program menjadi lebih baik karena program berbasis *open source* di dalam mengembangkannya.

BAB II

TINJAUAN PUSTAKA

2.1 Multimedia

Multimedia di dalam bidang komputer disebut sebagai alat yang berfungsi untuk menjalankan suatu *file* berupa *audio, video* dan *grafis* hingga hal tersebut dapat berfungsi sebagaimana mestinya. Sebuah *player* biasa digunakan di perangkat lunak atau keras multimedia, hal ini dikarenakan di dalam menjalankan suatu alat yang berhubungan dengan multimedia membutuhkan sebuah *player*, sebagai contohnya untuk memutar *file* berformat *avi*. *User* membuthkan media *player classic* untuk menjalankan *file* tersebut.

Menurut IBM, multimedia adalah gabungan video, audio, grafik dan teks dalam suatu produksi bertingkat berbasis komputer yang dapat dialami secara interaktif. Menurut McCormick multimedia secara umum merupakan kombinasi tiga elemen yaitu suara, gambar dan teks atau menurut Robin dan Linda multimedia

merupakan alat yang dapat menciptakan presentasi yang dinamis dan interaktif yang mengkombinasikan teks, grafik, animasi, audio dan gambar video.

2.2 Objek Multimedia

Menurut Sutopo, objek multimedia terbagi menjadi 6 kategori, yaitu :

1. **Teks**

Menurut Sutopo, hampir semua orang yang biasa menggunakan komputer sudah terbiasa dengan teks. Teks merupakan dasar dari pengolahan kata dan informasi berbasis multimedia. Dalam kenyataannya multimedia menyajikan informasi kepada audiens dengan cepat, karena tidak diperlukan membaca secara rinci dan teliti. Menurut Hofstetter kebanyakan system multimedia dirancang dengan menggunakan teks karena teks merupakan sarana yang efektif untuk mengemukakan ide-ide dan menyediakan instruksi-instruksi kepada *user* (pengguna).

2. **Image**

Menurut Sutopo, secara umum *image* atau grafik berarti *still image* seperti foto dan gambar. Manusia sangat berorientasi pada visual dan gambar merupakan sarana yang sangat baik untuk menyajikan informasi.

3. Animasi

Menurut Sutopo, animasi adalah pembentukan gerakan dari berbagai media atau objek yang divariasikan dengan gerakan transisi, efek-efek, juga suara yang selaras dengan gerakan animasi tersebut atau animasi merupakan penayangan *frame-frame* gambar secara cepat untuk menghasilkan kesan gerakan.

4. Audio

Menurut Sutopo (2003 : 13), penyajian audio atau suara merupakan cara lain untuk lebih memperjelas pengertian suatu informasi. Contohnya, narasi merupakan kelengkapan dari penjelasan yang dilihat melalui video. Suara dapat lebih menjelaskan karakteristik suatu gambar, misalnya musik dan suara efek *(sound effect)*. Salah satu bentuk bunyi yang bisa digunakan dalam produksi multimedia adalah *Waveform Audio* yang merupakan format *file* audio yang berbentuk digital. Kualitas produknya bergantung pada *sampling rate* (banyaknya sampel per detik). *Waveform* (wav) merupakan standar untuk Windows PC.

5. Video

Menurut Suyanto (2003: 279), video merupakan elemen multimedia paling kompleks karena penyampaian informasi yang lebih komunikatif dibandingkan gambar biasa. Walaupun terdiri dari elemen-elemen yang sama seperti grafik, suara dan teks, namun bentuk video berbeda dengan animasi. Perbedaan terletak pada penyajiannya. Dalam video, informasi disajikan dalam kesatuan utuh dari

objek yang dimodifikasi sehingga terlihat saling mendukung penggambaran yang seakan terlihat hidup.

6. Interactive Link

Menurut Sutopo (2002: 220), sebagian dari multimedia adalah interaktif, yang pengguna dapat menekan *mouse* atau objek pada *screen* seperti *button* atau teks dan menyebabkan program melakukan perintah tertentu. *Interactive link* dengan informasi yang dihubungkannya sering kali dihubungkan secara keseluruhan sebagai *hypermedia.* Secara spesifik, dalam hal ini termasuk *hypertext (hotword)*, *hypergraphics, hypersound,* dan *Interactive link* diperlukan bila pengguna menunjuk pada suatu objek atau *button* agar dapat mengakses program tertentu. *Interactive link* diperlukan untuk menggabungkan beberapa elemen multimedia sehingga menjadi informasi yang terpadu. Cara pengaksesan informasi pada multimedia terdapat dua macam, yaitu *linier* dan *non linier*. Informasi *linier* adalah informasi yang ditampilkan secara *sekuansial*, yaitu dari atas ke bawah atau halaman demi halaman, sedangkan pada informasi *non linier* dapat ditampilkan langsung sesuai dengan kehendak pengguna.

2.3 Format *Audio*

Berikut ini adalah beberapa bentuk format audio beserta dengan penjelasannya :

1. MP3

Singkatan dari MPEG, *Audio* Layer 3, MP3 menjadi format paling populer dalam musik digital. Kepopuleran MP3 disebabkan karena ukuran *file* nya yang kecil dengan kualitas yang tidak kalah dengan CD *audio*. Format ini dikembangkan dan dipatenkan oleh Fraunhofer Institute. Dengan *bitrate* 128 kbps, *file* MP3 sudah berkualitas baik. Namun MP3 Pro-format penerus MP3 menawarkan kualitas yang sama dengan *bitrate* setengah dari MP3. MP3 Pro kompatibel dengan MP3. Pemutar MP3 dapat memainkan *file* MP3 Pro namun kualitas suaranya tidak sebagus piranti yang mendukung MP3 Pro.

2. WAV

Standar suara *de-facto* di Windows. Hasil *ripping* dari CD pada awalnya direkam dalam format ini sebelum dikonversi ke format lain. Namun sekarang tahap ini sering dilewati. *File* dalam format ini biasanya tidak dikompresi dan karenanya berukuran besar.

3. AAC

AAC adalah singkatan dari *Advanced Audio Coding*. Format ini merupakan bagian standar Motion Picture Experts Group (MPEG), sejak standar MPEG-2 diberlakukan pada tahun 1997. *Sample rate* yang ditawarkan sampai 96 KHz-dua kali MP3. Format ini digunakan Apple pada toko musik online-nya, iTunes. Kualitas musik dalam format ini cukup baik bahkan pada *bitrate* rendah. iPod pemutar musik digital portabel dari Apple, adalah piranti terkemuka yang mendukung format ini.

4. WMA

WMA adalah salah satu format berkas suara yang ditawarkan Microsoft, Windows Media *Audio* (WMA), disukai para vendor musik online adalah dukungannya terhadap *Digital Rights Management* (DRM). DRM adalah fitur untuk mencegah pembajakan musik, hal yang sangat ditakuti oleh studio musik saat ini. Kelebihan WMA tidak hanya itu saja. Kualitas musik yang ditawarkan format WMA lebih baik daripada MP3. Tes yang dilakukan situs Extremetech.com menunjukkan format ini juga kualitasnya lebih baik daripada AAC. Format ini cukup populer. Piranti lunak dan piranti keras terbaru umumnya mendukung format ini. Namun dukungan belum seluas MP3, kendati hal ini bisa berubah dalam waktu-waktu mendatang. (http://id.wikipedia.org/wiki/Audio_digital)

5. Ogg Vorbis

Ogg Vorbis merupakan satu-satunya format *file* yang terbuka dan gratis. Format lain yang disebutkan di atas umumnya dipatenkan dan pengembang piranti lunak atau pembuat piranti keras harus membayar lisensi untuk produk yang dapat memainkan *file* dengan format terkait. Dari segi kualitas, kelebihan Ogg Vorbis adalah kualitas yang tinggi pada bitrate rendah dibandingkan format lain. Piranti lunak populer, Winamp dan pelopor pemutar MP3 portabel Rio sudah mendukung format ini dalam model terbarunya. Walaupun dukungan piranti keras terhadap format ini masih jarang.

6. Real*Audio*

Salah satu format yang biasa ditemukan pada *bitrate* rendah. Format dari RealNetworks ini umumnya digunakan dalam layanan *streaming audio*. Pada bitrate 128 kbps ke atas *RealAudio* menggunakan standar AAC MPEG-4.

7. MIDI

Format *audio* satu ini lebih cocok untuk suara yang dihasilkan oleh *synthesizer* atau piranti elektronik lainnya, tetapi tidak cocok untuk hasil konversi dari suara analog karena tidak terlalu akurat. *File* dengan format

ini berukuran kecil dan akhir-akhir ini sering digunakan dalam ponsel sebagai *ringtone.*

2.4 Gambas (Gambas Almost Means BASic)

GAMBAS (Gambas Almost Means BASic) dialek bahasa pemrograman BASIC yang berorientasi objek, disertai dengan lingkungan pengembangan terintegrasi (*Integrated Development Environment*). Gambas berjalan di sistem operasi GNU Linux dan sistem operasi *Unix-like* lainnya. Gambas ditujukan bagi pengguna Visual Basic yang bermigrasi ke GNU Linux.

Gambas juga berarti 'udang' dalam bahasa Spanyol, sesuai dengan logo yang digunakan Gambas, yakni seekor udang. Gambas yang dikembangkan di Paris oleh Benoît Minisini sejak 1999, merupakan perangkat lunak bebas yang dirilis di bawah lisensi GNU *General Public License*.

Terdapat fitur yang dimiliki Gambas ,yaitu:

1. Menggunakan basis data seperti MyQL dan PostgreSQL
2. Membangun aplikasi KDE dengan DCOP
3. Menerjemahkan program Visual Basic ke Gambas, dan menjalankannya di GNU Linux
4. Membangun solusi jaringan

Gambas dirancang untuk membangun aplikasi dengan antarmuka pengguna berbasis grafis (GUI) menggunakan Qt *toolkit*. IDE Gambas ditulis dengan menggunakan Gambas sendiri. Gambas memiliki fasilitas *GUI designer* untuk membantu membuat antarmuka.

2.5 GNU Linux

GNU Linux adalah nama yang diberikan kepada sistem operasi komputer bertipe Unix. GNU Linux merupakan salah satu contoh hasil pengembangan perangkat lunak bebas dan sumber terbuka utama. Seperti perangkat lunak bebas dan sumber terbuka lainnya pada umumnya, kode sumber GNU Linux dapat dimodifikasi, digunakan dan didistribusikan kembali secara bebas oleh siapa saja.

Nama "GNU Linux" berasal dari nama kernelnya (kernel GNU Linux), yang diperkenalkan tahun 1991 oleh Linus Torvalds. Sistemnya, peralatan sistem dan pustakanya umumnya berasal dari sistem operasi GNU, yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU adalah dasar dari munculnya nama alternatif GNU/GNU Linux.

2.5.1 Perbedaan GNU Linux dan Sistem Operasi Lainnya

Perbedaan utama antara GNU Linux dan sistem operasi populer lainnya terletak pada kernel GNU Linux dan komponen-komponennya yang bebas dan terbuka. GNU Linux bukan satu-satunya sistem operasi dalam kategori tersebut, walaupun demikian GNU Linux adalah contoh terbaik dan terbanyak digunakan. Beberapa lisensi perangkat lunak bebas dan sumber terbuka berdasarkan prinsip-prinsip *copyleft*, sebuah konsep yang menganut prinsip karya yang dihasilkan dari bagian *copyleft* harus juga merupakan *copyleft*. Lisensi perangkat lunak bebas yang paling umum, GNU GPL, adalah sebuah bentuk *copyleft*, dan digunakan oleh kernel GNU Linux dan komponen-komponen dari proyek GNU.

Proyek-proyek perangkat lunak bebas, walaupun dikembangkan dalam bentuk kolaborasi, sering dirilis secara terpisah. Dikarenakan lisensi-lisensi perangkat lunak bebas secara eksplisit mengijinkan distribusi ulang, terdapat proyek-proyek yang bertujuan untuk mengumpulkan perangkat lunak-perangkat lunak tersebut dan menjadikannya tersedia dalam waktu bersamaan dalam suatu bentuk yang dinamakan distribusi GNU Linux.

Sebuah distribusi GNU Linux, yang umum disebut dengan "*distro*", adalah sebuah proyek yang bertujuan untuk mengatur sebuah kumpulan perangkat lunak berbasis GNU Linux dan memfasilitasi instalasi dari sebuah sistem operasi GNU Linux. Distribusi-distribusi GNU Linux ditangani oleh individu, tim, organisasi

sukarelawan dan entitas komersial. Distribusi GNU Linux memiliki perangkat lunak sistem dan aplikasi dalam bentuk paket-paket dan perangkat lunak yang spesifik dirancang untuk instalasi dan konfigurasi sistem. Perangkat lunak tersebut juga bertanggung jawab dalam pemutakhiran paket. Sebuah Distribusi GNU Linux bertanggung jawab atas konfigurasi bawaan, sistem keamanan dan integrasi secara umum dari paket-paket perangkat lunak sistem GNU Linux. (http://maulbaikyah.blogspot.com/2010/06/sejarah-linux.html)

2.5.2 Hak cipta dan Merek Dagang

GNU Linux kernel dan sebagian besar perangkat lunak GNU menggunakan GNU General Public License (GPL) sebagai basis lisensinya. GPL mengharuskan siapapun yang mendistribusikan kernel GNU Linux harus membuat kode sumber tersedia bagi pengguna dengan kriteria yang sama. Tahun 1997, Linus Torvald menyatakan, “Menjadikan GNU Linux berbasis GPL sungguh merupakan hal terbaik yang pernah saya lakukan. Komponen penting lain dalam sistem GNU Linux diijinkan menggunakan lisensi selain dari GPL, banyak pustaka menggunakan GNU Lesser General Public License (LGPL), varian GPL yang lebih moderat, dan sistem X Window System menggunakan MIT License.

Linus Torvald telah menyatakan ke khalayak umum bahwa ia takkan memindahkan lisensi kernel GNU Linux yang saat ini menggunakan GPL versi 2 ke GPL versi 3, yang dikembangkan pada pertengahan tahun 2007, dengan alasan

beberapa ketentuan yang terdapat pada lisensi baru tersebut melarang penggunaan perangkat lunak dalam manajemen hak digital *(Digital rights management*).

Penelitian yang dilakukan pada tahun 2001 terhadap *Red Hat* GNU Linux 7.1 menemukan bahwa *distro* perangkat lunak tersebut terdiri atas 30 juta baris kode. Dengan menggunakan algoritma model biaya konstruktif (Inggris:*Constructive Cost Model*), penelitian itu memperkirakan bahwa *distro* tersebut membutuhkan waktu kira-kira delapan ribu tahun kerja untuk mengembangkannya. Menurut penelitian tersebut, jika semua elemen dari perangkat lunak tersebut dikembangkan dengan cara konvensional dalam arti sebagai perangkat lunak tertutup, pengembangan *distro* tersebut menelan biaya sebesar 1,08 miliar dolar (basis nilai tukar dolar tahun 2000) untuk dikembangkan di Amerika Serikat.

Sebagian besar kode (71%) ditulis dengan menggunakan bahasa pemrograman C, namun banyak bahasa lainnya juga ikut terlibat seperti C++, Assembly, Perl, Python, Fortran, dan berbagai bahasa skrip lain. Lebih dari separuh dari seluruh kode dilisensikan di bawah naungan GPL. Kernel GNU Linux itu sendiri terdiri atas 2,4 juta baris kode, atau sekitar 8% dari total keseluruhan.

Penelitian lainnya menghasilkan analisis yang sama terhadap *distro Debian* GNU/GNU Linux versi 4.0. *Distro* tersebut terdiri atas lebih dari 283 juta baris kode, dan penelitian tersebut memperkirakan biaya pengembangan yang

dibutuhkan sebesar 5,4 miliar euro jika dikembangkan sebagai perangkat lunak tertutup.

Di Amerika Serikat, GNU Linux merupakan merek dagang (SN: 1916230) yang dimiliki oleh Linus Torvalds. GNU Linux terdaftar sebagai "Program sistem operasi komputer bagi penggunaan komputer dan operasi". Merek dagang ini didaftarkan setelah ada suatu kejadian di mana seorang pemalsu bernama William R Della Croce Jr mulai mengirim surat kepada para distributor GNU Linux dan megklaim trademark GNU Linux adalah hak miliknya serta meminta royalti sebanyak 10% dari mereka. Para distributor GNU Linux mulai mendorong agar trademark yang asli diberikan kepada Linus Torvalds. Pemberian lisensi trademark GNU Linux sekarang di bawah pengawasan GNU Linux Mark Institute.

2.5.3 Macam – macam Distribusi GNU Linux

Terdapat banyak distribusi GNU Linux (lebih dikenali sebagai *distro*) yang dibuat oleh individu, grup, atau lembaga lain. Masing-masing disertakan dengan program sistem dan program aplikasi tambahan, di samping menyertakan suatu program yang memasang keseluruhan sistem di komputer (*installer program*).

Inti di setiap distribusi GNU Linux adalah kernel, koleksi program dari proyek GNU (atau proyek lain), cangkang (*shell*), dan atur cara *utilitas* seperti pustaka (*libraries*), kompilator, dan penyunting (*editor*). Kebanyakan sistem juga

menyertakan aturcara dan *utilitas* yang bukan GNU. Bagaimanapun, *utilitas* tersebut dapat dipisahkan dan sistem ala UNIX masih tersedia. Beberapa contoh adalah atur cara dan *utiliti* dari BSD dan sistem grafik-X (*X-Window System*). X menyediakan antarmuka grafis (GUI) yang umum untuk GNU Linux.

Contoh-contoh distribusi GNU Linux , yaitu :

1. Ubuntu dan derivatifnya : *Sabily* (Ubuntu Muslim Edition), KUbuntu, XUbuntu, EdUbuntu, *GoBuntu*, *Gnewsense*, UbuntuCE
2. *SuSE*
3. *Fedora*
4. *Mandriva*
5. *Slackware*
6. *Debian*
7. *Knoppix*
8. *Xandros*
9. *Sabayon*
10. *CentOS*
11. *Red Hat*

2.5.4 Distribusi GNU Linux Ubuntu

Karena yang digunakan dalam membuat program ini adalah menggunakan GNU Linux Ubuntu, maka disini dibahas sedikit tentang GNU Linux Ubuntu. Ubuntu

merupakan salah satu distribusi GNU Linux yang berbasiskan *Debian*. Proyek Ubuntu resmi disponsori oleh Canonical Ltd yang merupakan perusahaan milik seorang kosmonot asal Afrika Selatan Mark Shuttleworth. Nama Ubuntu diambil dari nama sebuah konsep ideologi di Afrika Selatan, "Ubuntu" berasal dari bahasa kuno Afrika, yang berarti "rasa peri kemanusian terhadap sesama manusia". Tujuan dari distribusi GNU Linux Ubuntu adalah membawa semangat yang terkandung di dalam Filosofi Ubuntu ke dalam dunia perangkat lunak. Ubuntu adalah sistem operasi lengkap berbasis GNU Linux, tersedia secara bebas dan mempunyai dukungan baik yang berasal dari komunitas maupun tenaga ahli profesional.

Ubuntu mempunyai filosofi sebagai berikut :

- Bahwa perangkat lunak harus tersedia dengan bebas biaya
- Bahwa aplikasi perangkat lunak tersebut harus dapat digunakan dalam bahasa lokal masing-masing dan untuk orang-orang yang mempunyai keterbatasan fisik.
- Bahwa pengguna harus mempunyai kebebasan untuk mengubah perangkat lunak sesuai dengan apa yang dibutuhkan.

Perihal kebebasan inilah yang membuat Ubuntu berbeda dari perangkat lunak berpemilik (proprietary), bukan hanya peralatan yang dibutuhkan tersedia secara

bebas biaya, tetapi juga mempunyai hak untuk memodifikasi perangkat lunak sampai perangkat lunak tersebut bekerja sesuai dengan yang inginkan.

Setiap rilis mempunyai nama kode dan nomor versi. Nomor versi berdasarkan tahun dan bulan dari rilis. Sebagai contoh, rilis Ubuntu yang pertama, 4.10, dirilis tanggal 20 Oktober 2004. Rilis Ubuntu keluar setiap 6 bulan sekali tiap bulan April dan Oktober. Rilis Ubuntu biasanya terdiri dari berbagai edisi, yaitu edisi *Desktop, Server*, dan *Netbook*. Perbedaan mendasar pada ketiganya adalah pada versi *desktop* terdapat tampilan desktop *manager* sedangkan pada edisi *server* tidak ada, hanya *command prompt* saja.

Daftar rilis Ubuntu yang telah dirilis dan yang direncanakan untuk dirilis. Perhatikan Tabel 1.

Versi	Tanggal Rilis	Nama kode	Sub Versi	Didukung sampai
4.10	20 Oktober 2004	Warty Warthog	Sounder	30 April 2006
5.04	8 April 2005	Hoary Hedgehog	Array	31 Oktober 2006
5.10	13 Oktober 2005	Breezy Badger	Colony	13 April 2007
6.06 LTS (*Long Term Support*)	1 Juni 2006	Dapper Drake	Flight	Juni 2009 (desktop)
				Juni 2011 (server)
6.10	26 Oktober 2006	Edgy Eft	Knot	April 2008
7.04	19 April 2007	Feisty Fawn	Herd	Oktober 2008
7.10	18 Oktober 2007	Gutsy Gibbon	Tribe	April 2009
8.04 LTS (*Long Term Support*)	21 April 2008	Hardy Heron	Siege	April 2011 (desktop)
				April 2013 (server)
8.10	30 Oktober 2008	Intrepid Ibex		
9.04	23 April 2009[3]	Jaunty Jackalope		
9.10	29 Oktober 2009[4]	Karmic Koala[5]		
10.04 LTS	29 April 2010[6]	Lucid Lynx[7]		

Tabel 1. Daftar Rilis Ubuntu

2.6 Visual Basic

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang menawarkan *Integrated Development Environment* (IDE) visual untuk membuat program perangkat lunak berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman (COM), Visual Basic merupakan turunan bahasa pemrograman BASIC dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat, Beberapa bahasa skrip seperti *Visual Basic for Applications* (VBA) dan *Visual Basic Scripting Edition* (VBScript), mirip seperti halnya Visual Basic, tetapi cara

kerjanya yang berbeda. Para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang disediakan oleh Microsoft Visual Basic Program-program yang ditulis dengan Visual Basic juga dapat menggunakan Windows API, tapi membutuhkan deklarasi fungsi luar tambahan. Dalam pemrograman untuk bisnis, Visual Basic memiliki pangsa pasar yang sangat luas. Dalam sebuah survey yang dilakukan pada tahun 2005, 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk Visual Basic, yang diikuti oleh C++, JavaScript, C#, dan Java.

2.6.1 Perkembangan Visual Basic

VB 1.0 dikenalkan pada tahun 1991, pendekatan yang dilakukan untuk menghubungkan bahasa pemrograman dengan GUI berasal dari *prototype* yang dikembang oleh "Alan Cooper" yang di sebut TRIPOD, Kemudian Microsoft mengontrak *copper* dan asosiasinya untuk mengembangkan *tripod* agar dapat digunakan di Windows 3.0 di bawah nama kode Ruby. Berikut Perjalanan Visual Basic (VB 1.0 Sampai VB 10) : Perjalanan dari Visual Basic (VB1 to VB 10).

1. Proyek "Thunder" dirintis
2. Visual Basic 1.0 (May 1991) di rilis untuk windows pada COMDEX/Windows Wordltrade yang dipertunjukan di Atlanta , Georgia
3. Visual Basic 1.0 untuk DOS dirilis pada bulan September 1992. Bahasa ini tidak kompatibel dengan Visual Basic For Windows. VB 1.0 for DOS

ini pada kenyataaanya merupakan versi kelanjutan dari compiler BASIC, QuickBasic dan BASIC Professional Development System.

4. Visual Basic 2.0 dirilis pada November 1992, cakupan pemrogramannya cukup mudah untuk digunakan dan kecepatannya juga telah dimodifikasi. Khususnya pada Form yang menjadikan *object* dapat dibuat secara seketika, serta konsep dasar dari Class modul yang berikutnya di implementasikan pada VB 4
5. Visual Basic 3.0 , dirilis pada musim panas 1993 dan dibagi menjadi versi strd dan professional. VB 3 memasukkan Versi 1.1 dari Microsoft Jet Database Engine yang dapat membaca serta menulis database Jet (atau access) 1.x
6. Visual Basic 4.0 (Agustus 1995) merupakan versi pertama yang dapat membuat windows program 32 bit sebaik versi 16 bit nya. VB 4 juga memperkenalkan kemampuan untuk menulis non-GUI class pada Visual Basic
7. Visual Basic 5.0 (February 1997), Microsoft merilis secara eksklusif Visual basic untuk versi windows 32 bit. Programmer yang menulis programnya pada versi 16 bit dapat dengan mudah melakukan import porgramnya dari VB4 ke VB5. dan juga sebaliknya, program VB5 dapat diimport menjadi VB4. VB 5 memperkenalkan kemampuan untuk membuat *User Control*.

8. Visual Basic 6.0 (pertengahan 1998) memperbaiki beberapa cakupan, temasuk kemapuannya untuk membuat Aplikasi *Web-based*. Visual Basic 6 di jadwalkan memasuki Microsoft "fasa non Supported" dimulai pada maret 2008
9. Visual Basic.NET (VB 7), dirilis pada tahun 2002, Beberapa yang mencoba pada versi pertama.NET ini mengemukakan bahwa bahasa ini sangat powerful tapi bahasa yang digunakan sangat berbeda dengan bahasa sebelumnya, dengan kekurangan diberbagai area, termasuk *runtime*-nya yang 10 kali lebih besar dari paket *runtime* VB6 serta peningkatan penggunan *memory*.
10. Visual Basic .NET 2003 (VB 7.1), dirilis dengan menggunakan NET framework versi 1.1.
11. Visual Basic 2005 (VB 8.0), merupakan itegrasi selanjutnya dari Visual Basic .NET. dan Microsoft memutuskan untuk menghilangkan kata-kata .NET pada judulnya. Pada rilis ini, Microsoft memasukkan beberapa fitur baru, diantaranya : *Edit* and *Continue*, mungkin inilah kekurangan fitur terbesar dari VB .NET. pada VB 2005 ini program diperbolehkan melakukan perubahan kode pada saat program sedang dijalankan Perbaikan pada Konversi dari VB ke VB NET12 Visual Basic.NET 2003 (VB 7.1), dirilis dengan menggunakan NET Kerangka kerja versi 1.1. IsNot Patent, merupakan salah satu fitur dari Visual Basic 2005 merupakan konversi If=Not X Is Y menjadi If X IsNot Y.

12. Visual Basic 2005 Express, merupakan bagian dari Product Visual Studio. Microsoft membuat Microsoft Visual Studio 2005 Express edition untuk pemula dan yang gemar dengan VB, salah satu produknya adalah Visual Basic 2005 Express yang merupakan produk gratis dari Microsoft.
13. Visual Basic "Orcas" (VB 9.0) , dijadwalkan yang dirilis pada tahun 2007 dan dibangun diatas .NET 3.5. Pada rilis ini, Microsoft menambahkan beberapa fitur, diantaranya : True Tenary operator, yaitu fungsi If(boolean,value, value) yang digunakan untuk menggantikan fungsi *IIF*, *LINQ Support, Ekspresi Lambda, XML Literals, Nullable types, Type Inference.*
14. Visual Basic 'VBx' (VB 10.0), Visual Basic 10, yang juga dkenal dengan nama VBx yang menawarkan dukungan untuk *Dynamic Language Runtime.*

Karena pemrograman Visual Basic sangat digemari oleh para programmer maka para pecinta *open source* membangun bahasa pemrograman Gambas sebagai salah satu tiruan Visual Basic yang berjalan pada sistem operasi GNU Linux. Gambas sendiri merupakan bahsa pemrograman kelanjutan dari Visual Basic karena Gambas juga menggunakan bahasa Basic.

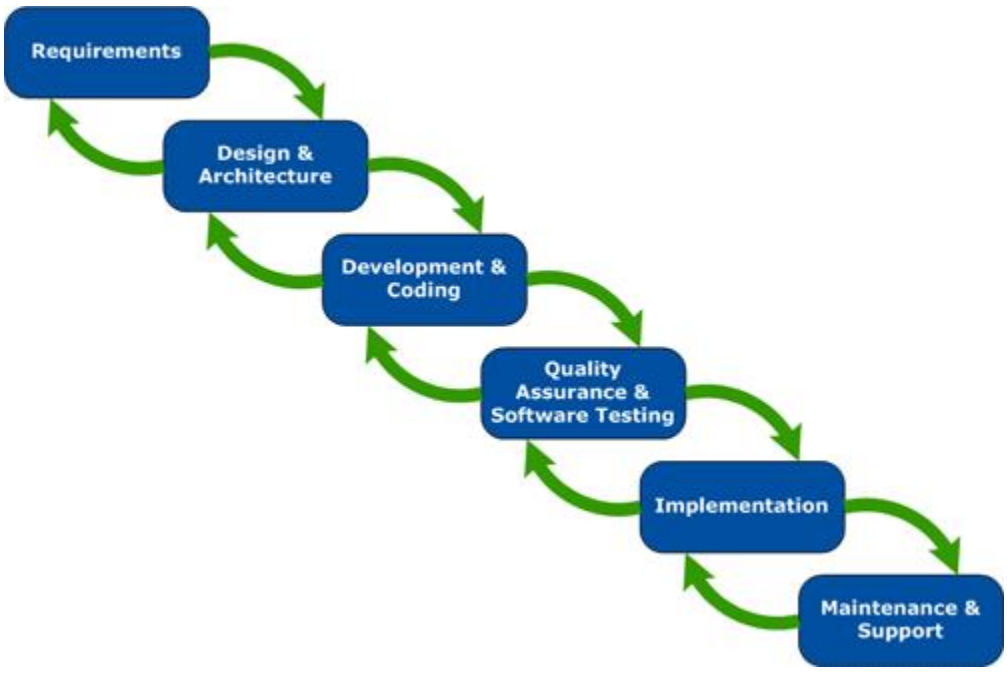
2.7 Metode *Waterfall*

Nama model ini sebenarnya adalah "*Linear Sequential Model*". Model ini sering disebut dengan "*classic life cycle*" atau model *Waterfall.* Model ini adalah model yang muncul pertama kali yaitu sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam *Software Engineering* (SE). Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing / verification,* dan *maintenance*. Disebut dengan *Waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap *requirement.*

BAB III

METODOLOGI PENELITIAN

Melihat perkembangan sistem operasi Linux yang semakin pesat, maka pada penelitian ini dibuat pemutar musik *file* berformat mp3 yang dibangun menggunakan pemrograman Gambas, hal ini dikarenakan pemrograman gambas adalah salah satu bahasa pemrograman BASIC yang berorientasi objek, disertai dengan lingkungan pengembangan terintegrasi (*Integrated Development Environment*). Gambas berjalan di sistem operasi Linux dan sistem operasi *Unix-like* lainnya. Gambas ditujukan bagi para pengguna Visual Basic yang ingin bermigrasi ke Linux. Dengan dibuatnya program ini diharapkan dapat membuat para *user* lebih cenderung menggunakan system operasi yang berbasis *open source* sehingga dengan mudah *user* dapat membuat dan mengembangkan program-program agar menjadi semakin lebih baik. Alur yang digunakan dalam membangun sistem tersebut adalah metode *Waterfall* , Metode *Waterfall* dijelaskan pada gambar 1.



Gambar 1. *Metode Waterfall*

Di dalam membuat program ini, menggunakan metode *Waterfall* disaat pengerjaannya. Berikut tahap–tahap yang dilakukan didalam membangun aplikasi MP3 Player :

1. *Requirements Analysis*

   Di tahap ini dicari dan menentukan tentang berbagai masalah yang dihadapi didalam membuat program, meliputi batasan masalah, rumusan masalah hingga tentang *studi literatur*. Di dalam rumusan masalah adalah membahas bagaimana cara *file* musik dapat dimainkan di Linux, bagaimana membangun sebuah player musik menggunaka bahasa

pemrograman *Gambas2.0* dan bagaimana cara memainkan *file* musik, menghentikan *file* musik, dan menyimpan *playlist*. Batasan masalah yang telah ditetapkan yaitu meliputi program hanya bisa diperuntukan untuk memutar *file* berformat MP3, bahasa pemrograman yang digunakan adalah *Gambas 2.0*, sistem operasi yang digunakan adalah Linux Ubuntu *9.10 Karmic Koala* dan program hanya memiliki fitur–fitur standar pemutar musik. Di dalam studi literatur penelitian ini mencari refrensi data dan informasi dari internet, buku, jurnal serta bertanya kepada teman yang mengerti tentang pemrograman *Visual Basic*.

2. *Sistem and Software Design*

   Dalam desain membuat program pemutar musik ini banyak yang dikerjakan, beberapa hal yang dikerjakan dalam membangun program musik player ini meliputi ,

   1. Menentukan Tools

      Dalam membuat program ini ditentukan sendiri *tools* yang digunakan meliputi perangkat keras maupun perangkat lunak.

   2. Menjalankan *File* Musik

      Di dalam program ini terdapat tombol *Play* yang berfungsi sebagai button untuk menjalankan *file* musik, tombol *Stop* yang berfungsi

untuk menghentikan proses membaca *file* musik, tombol *Next* disini berfungsi memindahkan lagu yang diputar dengan bergerak maju sesuai dengan isi *playlist*, sedangkan tombol *Prev* berfungsi untuk memindahkan lagu yang diputar dengan bergerak ke belakang. Menu–menu yang tersedia di program ini diantaranya membuka *file* (*Open File*) berfungsi memasukkan lagu ke dalam *playlist*, menyimpan playlist (*Save Playlist*) yang berfungsi untuk menyimpan daftar *playlist* yang telah dibuat, *Scroll bar* yang berfungsi untuk menunjukkan jalannya *file* musik saat dimainkan, *Timer* yang digunakan untuk mengetahui lama berjalannya *file* musik yang sedang dimainkan.

3. *User Interface*

   Di dalam membuat tampilan program yang sangat familiar dengan *user* dengan mendapatkan refrensi dari program–program tentang player musik yang sudah ada. *Interface* yang dapat membuat *user* merasa nyaman dan dengan mudah menggunakan program tersebut.

4. *Implementation*

   Di dalam tahap implementasi dilakukan penerapan design pada aplikasi menggunakan gambas dibuat di dalam satu *form* yang disitu berisi tentang

komponen–komponen yang digunakan di dalam program dan satu lembar modul fungsi yang digunakan untuk menjalankan setiap fungsi komponen yang telah dibuat dimulai dari memainkan lagu, menghentikan lagu hingga menyimpan *playlist*. Di dalam menjalankan implementasi program ini juga merujuk dari berbagai buku dan jurnal tentang multimedia.

5. *Integration and Testing*

   Di dalam tahap ini dilangsungkan penggabungan semua fungsi yang terdapat pada program dimulai dari memainkan lagu, menghentikan lagu, menjalankan *timer*, menjalankan *scroll bar*, hingga menyimpan *playlist*. *Testing* program dijalankan mengecek jika terdapat error yang bisa menghentikan program secara tiba–tiba ataupun terdapat fungsi–fungsi yang tidak berjalan sesuai dengan apa yang telah ditentukan.

6. *Operation and Maintenance*

   Pada tahap ini program yang telah selesai dilakukan *small release* ke *user*, agar program menjadi lebih bermanfaat serta program sangat mungkin untuk dikembangkan sesuai dengan kebutuhan *user*, jika di dalam program terdapat error–error kecil maka *user* dapat dengan mudah mengembangkan program ini karena program ini berbasis *open source* dan *legal*.

Tools yang digunakan :

- *PC intel core 2 duo* 1,8 Ghz
- *Ram* 1,5 Gb
- *Hardisk* 120 Gb
- *VGA* 256 Mb
- *SoundCard*
- *Keyboard*
- *Mouse*
- Sistem operasi Linux Ubuntu 9.10 *Karmic Koala*
- *Software* pemrograman *Gambas* 2.0

Waktu pelaksanaan :

- Pembuatan program ini dilaksanakan 1 April 2010 – 30 September 2010

BAB IV

HASIL DAN PEMBAHASAN

Aplikasi musik *player* yang dibangun merupakan *aplikasi* yang hanya dapat berjalan di system operasi GNU Linux. *Aplikasi* ini dibangun dengan menggunakan metode *Waterfall*. Pada pembuatan *program* melalui beberapa fase seperti *requirement and definition*, *system and software design*, *implementation and testing unit*, *integration and system testing* dan *operation and maintenance*.

4.1 Perancangan

4.1.1 *User Requirement dan Definition*

4.1.1.1 *User Requirement*

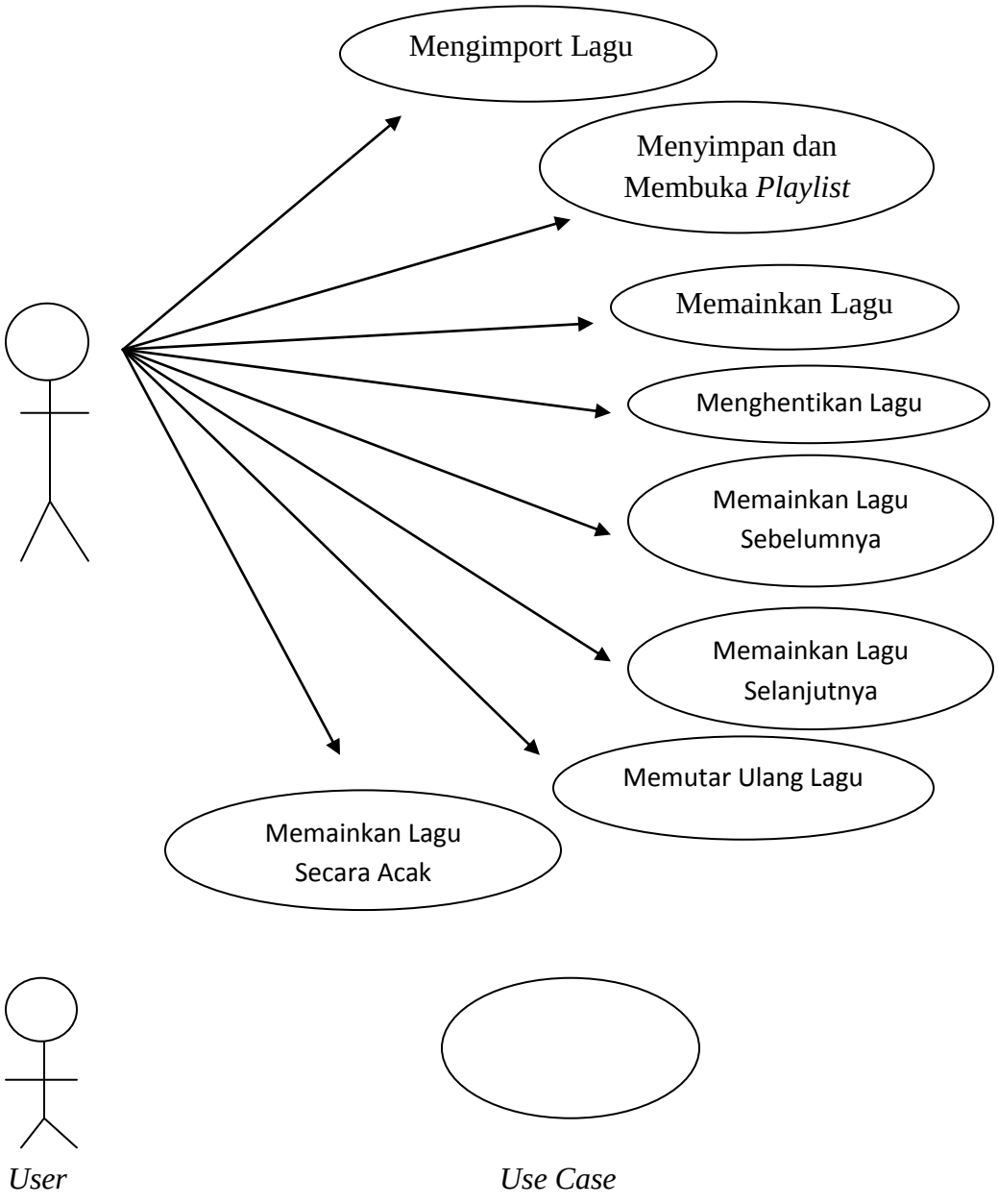
- Kebutuhan *user* adalah layanan yang diinginkan oleh *user* terhadap *aplikasi* yang dibangun beserta batasan-batasan operasionalnya. *Aplikasi* MP3 *player* ini menyediakan beberapa fungsi dalam memainkan atau menyimpan *playlist* dalam sebuah perangkat lunak.

- Merancang pembuatan *aplikasi* MP3 *player* yang dapat berjalan di sistem operasi GNU Linux Ubuntu.
- Membuat *playlist* sebagai sarana media penyimpanan daftar lagu yang diputar.

4.1.1.2 *Use Case*

Di dalam *usecase* dijelaskan bahwa *user* melihat fungsi-fungsi *aplikasi* MP3 *player* yang digunakan di dalam mengoperasikan *program* tersebut. Fungsi-fungsi yang terdapat dalam *program* tersebut meliputi, melihat mengimport *file*, menyimpan atau membuka *file playlist*, tombol *play*, melihat tombol *stop*, melihat tombol *previous*, melihat tombol *next* dan melihat *playlist* lagu yang diputar.

Aplikasi daftar kebutuhan *user* sebagai berikut Adapun beberapa fungsi yang diberikan oleh *aplikasi* ditunjukan dalam *usecase.* Berikut ini adalah *usecase* yang digunakan dalam membangun program :


Mengimport Lagu
Menyimpan dan Membuka *Playlist*
Memainkan Lagu
Menghentikan Lagu
Memainkan Lagu Sebelumnya
Memainkan Lagu Selanjutnya
Memutar Ulang Lagu
Memainkan Lagu Secara Acak
User
Use Case

Use Case

4.1.2 *System dan Sofware Design*

4.1.2.1 Membuat Tampilan *User Interface*

Interface merupakan tampilan antar muka yang diperlihatkan kepada *user*. Pada *aplikasi* MP3 *player* ini *interface* dibuat dengan menggunakan *program* Gambas 2.0 dan gambar yang digunakan dalam *aplikasi* ini dibuat dengan *software* Adobe Photoshop CS.

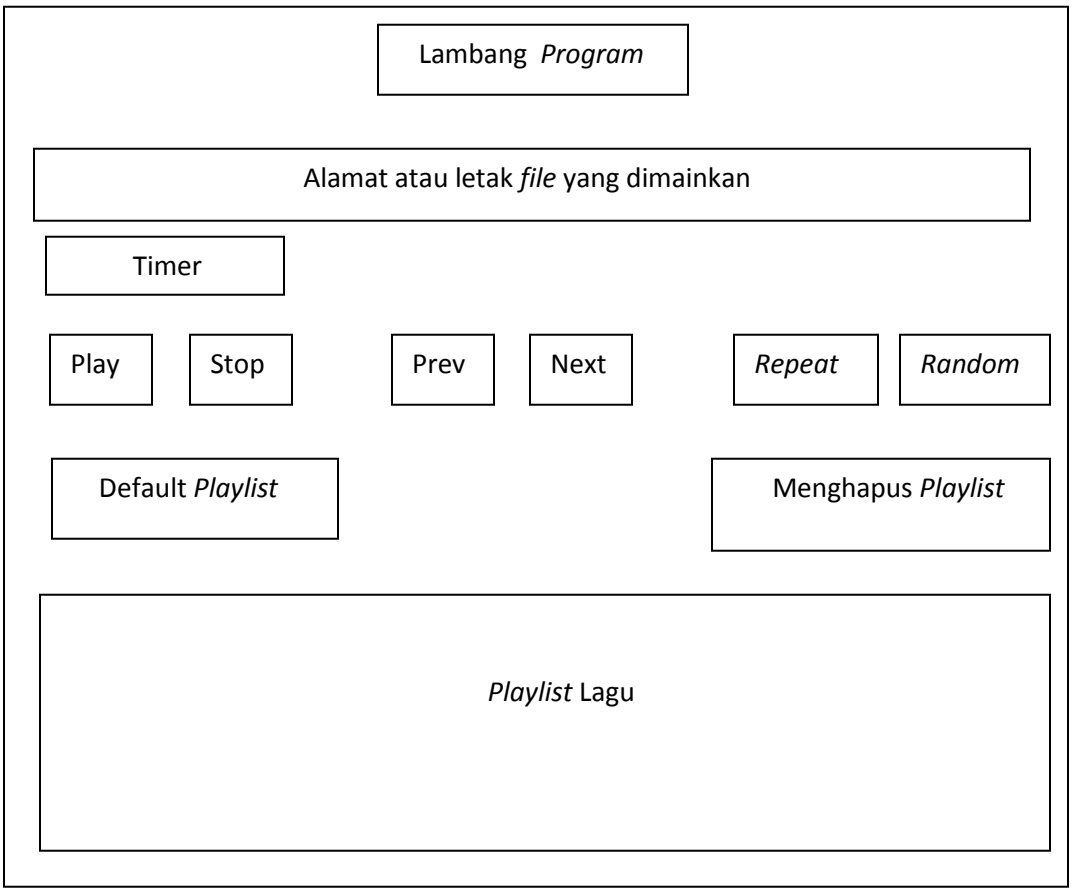
Perangkat Lunak *aplikasi* MP3 *player* ini memiliki beberapa form, seperti :

1. *Fmain.form*
2. *About.form*

1. Fmain.form

Fmain.form adalah form yang pertama kali dijalankan pada saat *aplikasi* dijalankan. Form ini adalah sebagai tampilan induk *program* yang didalam nya terdapat fitur–fitur untuk menjalankan *aplikasi* MP3 *player* ini. Form ini bisa langsung dapat digunakan oleh *user* tanpa harus melakukan registrasi atau login terlebih dahulu.

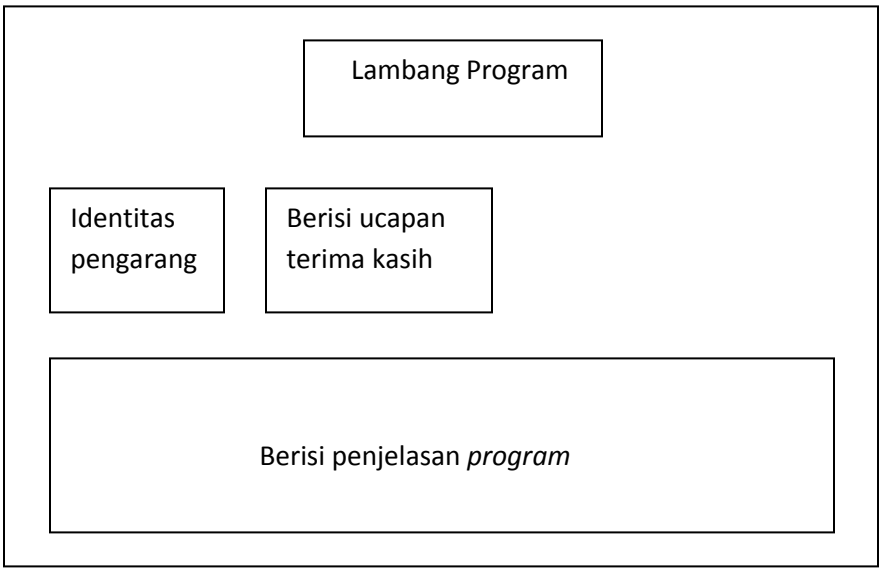
Di dalam form ini terdapat beberapa tombol yang digunakan seperti *Play*, *Stop*, *Next*, *Previous*, dan *Random*. Tampilan menu untuk mengimport lagu dan menyimpan *playlist* juga terdapat di dalam form ini. Penjelasan tentang Fmain.form diperlihatkan oleh gambar 2.



Gambar 2. *User Interface*

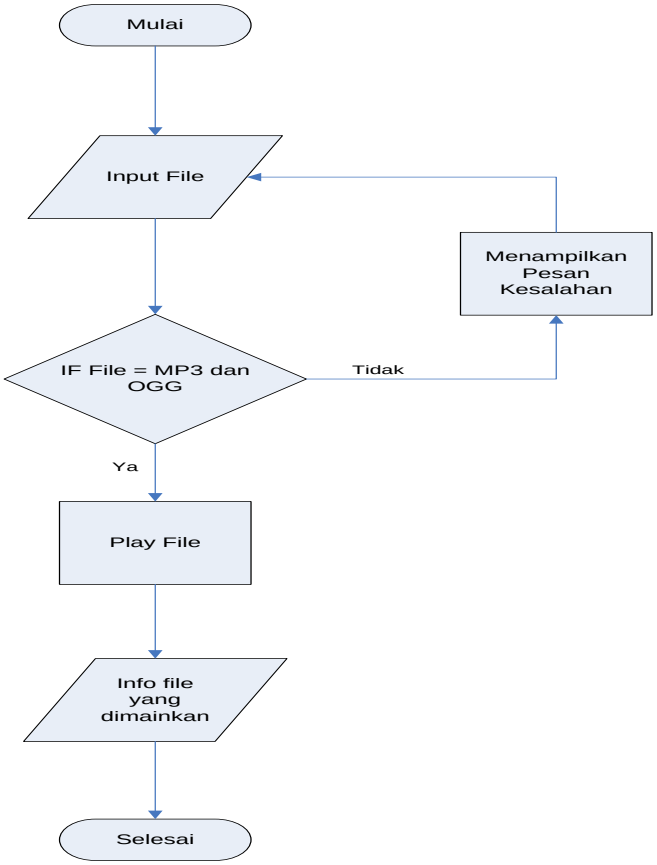
2. About.form

About.form merupakan form yang berisi tentang identitas *program* sebagai penjelasan waktu *program* dibangun hingga selesai, ucapan terima kasih kepada yang telah membantu di dalam mengerjakan skripsi dan juga berisi nama pembuat *program*. Penjelasan tentang About.form diperlihatkan oleh gambar 3.



Gambar 3. *Informasi Interface*

4.1.2.3 ***Flowchart***



Gambar 4. *Flow Chart*

4.1.3 *Implementation and Unit Testing*

Pada tahap ini desain yang telah dirancang diimplementasikan dengan menterjemahkan ke dalam kode-kode *program* menggunakan sebuah bahasa pemrograman, sekaligus melakukan pengujian terhadap unit-unit *program* yang telah dibuat. Pada tahap ini hal-hal yang dilakukan adalah :

4.1.3.1 Membuat kode-kode *program*

Berikut ini adalah pemotongan kode *program* dari unit-unit yang ada pada *aplikasi* MP3 *player* :

4.1.3.2 Mengimport *file* ke dalam *playlist*

Sebelum menjalankan *file* musik, *user* harus memasukkan *file* lagu terlebih dahulu ke dalam *playlist* yang bertujuan agar *program* dapat membaca *file* secara berurutan. Sedikit potongan coding di dalam memasukkan *file* musik ke *playlist*.

```
PUBLIC SUB Menu1_Click()
  'import file ke playlist
  Dialog.Title = "pilih file"
  Dialog.Path = User.Home
  Dialog.Filter = ["*.mp3;*.ogg", "Audio files"]
  IF Dialog.OpenFile() THEN RETURN
  'user menambahkan file ke dalam playlist
  IF timMusik.Enabled = TRUE THEN
    TRY ListBox1.Add(Dialog.Path, ListBox1.Count)
```

```
    IF ERROR
      Message.Info("sudah tersedia dalam playlis elemen")
    ELSE
      function.save_file(playlist_open_dir)
    ENDIF
```

File musik yang dapat diputar atau dimainkan musik *player* ini hanya *file* ber format MP3 dan OGG. Berikut ini adalah potongan *source code* yang digunakan dalam men verifikasi *file* yang hanya berformat MP3 dan OGG.

```
'memverifikasi file hanya ogg dan mp3
  FOR i = 0 TO (listfile.Count - 1)
    extension = Right(listfile[i], 3)
    IF extension = "ogg" OR IF extension = "mp3" OR IF extension = "MP3"
OR IF extension = "OGG" THEN
      ListBox1.Add(folder & "/" & listfile[i], ListBox1.Index = i)

    ENDIF
```

4.1.3.3 Memainkan Lagu

Didalam menjalankan lagu digunakan tombol *play* dalam memberikan perintah, sedikit potongan *source code* yang berfungsi dalam menjalankan lagu,

```
PUBLIC SUB Play_Click()
  'mulai memainkan lagu bila lagu pertama sudah diimport ke playlist
  IF ListBox1.List.Count = 0 THEN
```

```
    Menu1_Click
  'dapat mulai memainkan lagu dengan bersamaan timer berjalan
  ELSE
    'jika dalam playlist kosong maka user dapat mengimport lagu pertama
    IF ListBox1.Index = -1 THEN
      ListBox1.Index = 0
    ENDIF
    'kecuali jika dalam indeks terdapat lagu untuk mengindari perubahan
    index = ListBox1.Index
    function.check_song(ListBox1.Text)
  ENDIF
END
```

4.1.3.4 Menghentikan Lagu

Saat menghentikan proses memainkan lagu diperlukan tombol *Stop* untuk menghentikannya. Berikut potongan *source code* nya :

```
PUBLIC SUB Stop_Click()
  'menghentikan pemutaran lagu, timer serta mereset tampilan counter
  Musik.Stop
  timMusik.Stop
  lblPos.Text = "00:00:00"
  sldPos.Value = 0
END
```

4.1.3.5 Memainkan Lagu Selanjutnya

Saat *user* ingin memindahkan atau memutar *file* musik yang selanjutnya, maka tombol *next* digunakan yang mempunyai fungsi memindahkan *file* musik yang terdapat pada *playlist* yang sudah dibuat. Berikut potongan *source code* nya :

```
PUBLIC SUB next_Click()
  IF ListBox1.Index <> ListBox1.List.Count - 1 THEN
    INC ListBox1.Index
    timMusik.Stop
    function.check_song(ListBox1.Text)
  ELSE
    Stop_Click
    back.Enabled = FALSE
    next.Enabled = FALSE
    ListBox1.Index = 0
  ENDIF
END
```

4.1.3.6 Memainkan Lagu Sebelumnya

Saat *user* menginginkan lagu yang sebelumnya diputar kembali, maka digunakan tombol *Previous* yang berfungsi untuk mulai memainkan *file* musik yang sebelumnya. Berikut potongan *source code* nya :

```
PUBLIC SUB back_Click()
  IF ListBox1.Index <> 0 THEN
    DEC ListBox1.Index
    timMusik.Stop
    function.check_song(ListBox1.Text)
  ELSE
    Stop_Click
    back.Enabled = FALSE
    next.Enabled = FALSE
    ListBox1.Index = 0
  ENDIF
END
```

4.1.3.7 Memutar Lagu Secara Berulang

Jika *user* menginginkan *file* musik yang dimainkan berulang-ulang kali, maka dibutuhkan tombol *repeat*. Berikut potongan *source code* nya :

```
PUBLIC SUB shuffle_Click()
  IF REPEATS = FALSE THEN
    'jika tombol ditekan secara random
    IF random.Value = TRUE THEN
      random.Value = FALSE
    ENDIF
    REPEATS = TRUE
```

```
  ELSE
    REPEATS = FALSE
  ENDIF
END
```

4.1.3.8 Memutar Lagu Secara Acak

Di dalam memainkan lagu yang secara acak di dalam *playlist* yang telah dibuat maka menggunakan tombol *random*. Berikut potongan *source code* nya :

```
PUBLIC SUB random_Click()
  'jika tombol ditekan maka shuffle off
  IF shuffle.Value = TRUE THEN
    shuffle.Value = FALSE
  ENDIF
END
```

4.1.4 Integration dan *System Testing*

Setelah dibuatkan kode *program* dari setiap unit, selanjutnya unit-unit *program* tersebut diitegrasikan menjadi satu untuk dilakukan pengujian aplkasi perangkat lunak secara keseluruhan. Hal ini dilakukan untuk memenuhi persyaratan *aplikasi* yang telah dibuat. Pengujian yang dilakukan untuk menguji *aplikasi* secara keseluruhan umumnya menggunakan *White box testing.* Pada *aplikasi* perangkat

lunak Mp3 *player* dilakukan pengujian secara keseluruhan terhadap *aplikasi* Mp3 *player* ini. Berikut ini adalah hasil pengujian dari *Aplikasi* Mp3 *player* yang terintegrasi :

Tanggal pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

Pengujian : Memasukkan *File* Musik ke dalam *Aplikasi*

- Pengujian import *file* ke dalam *aplikasi* dilakukan langsung terhadap *aplikasi* Mp3 *player* yang sedang berjalan. Pengujian dilakukan dengan cara memasukkan *file* yang berformat MP3 dan OGG ke dalam *playlist*, selain *file* berformat tersebut tidak terbaca oleh browser.

Tanggal Pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

Pengujian : Menyimpan atau Membuka *Playlist*

- Pengujian yang dilakukan kali ini yaitu berfungsi menyimpan daftar lagu yang telah dibuat oleh *user*. *File* yang *playlist* yang disimpan berformat txt. Dan sebaliknya jika ingin membuka daftar lagu yang tersedia *user* hanya perlu membukanya lagi dengan cara meng klik dua kali yang terdapat pada browser load *playlist*.

Tanggal Pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

Pengujian : Memainkan *file* musik yang selanjutnya

- Pengujian memainkan musik yang selanjutnya atau sering juga disebut *next* yang dapat dilakukan dengan cara mengklik tombol *next* yang terdapat pada *aplikasi* MP3 *player*, dengan begitu lagu yang dimainkan berubah sesuai dengan urutan *playlist*.

Tanggal Pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

Pengujian : Memainkan *file* musik yang sebelumnya

- Pengujian yang dilakukan tahap ini berfungsi memainkan lagu yang sebelumnya atau sering juga disebut *previous*. Fungsi ini dapat dilakukan dengan cara meng klik tombol *previous* yang terdapa pada *aplikasi* MP3 *player*, *file* musik yang dimainkan sebelumnya sesuai dengan urutan yang terdapat pada *playlist* lagu.

Tanggal Pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

Pengujian : Memutar *File* Musik Berulang Kali Secara Otomatis

- Pada pengujian kali ini yaitu bagaimana *aplikasi* MP3 *player* dapat memutar lagu secara otomatis secara berulang kali. Fungsi tersebut dapat dilakukan dengan cara meng klik tombol *random* yang terdapat pada *aplikasi* MP3 *player*. *File* yang dimainkan secara berulang sesuai dengan *playlist*.

Tanggal Pengujian : 20 September 2010

Nama Penguji : Heri Setiawan Basuki

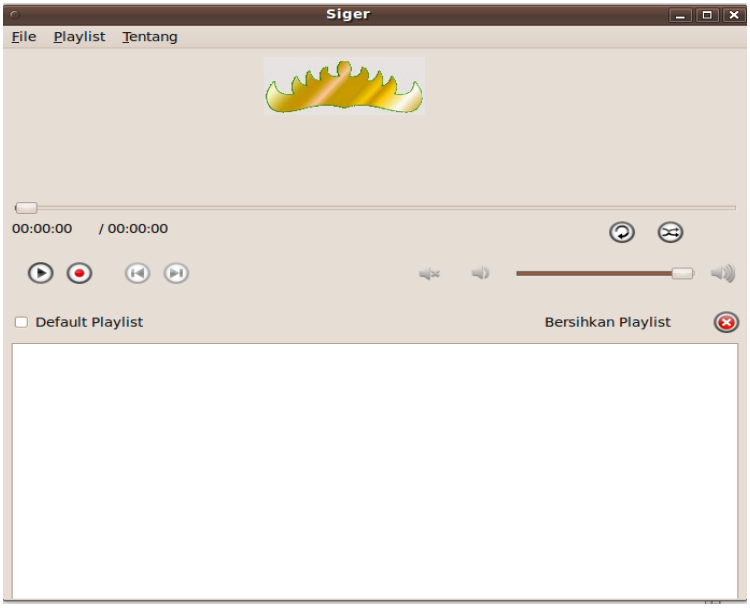
Pengujian : Memainkan *File* Musik Secara Acak

- Pengujian Memainkan *file* musik secara acak dilakukan dengan cara meng klik tombol *random* yang terdapat pada *aplikasi* MP3 *player*. Fungsi yang dilakukan yaitu *aplikasi* memainkan lagu secara *random* atau acak sesuai yang terdapat pada *playlist* dengan kata lain memainkan *file* musik secara tidak berurutan.

4.1.5 *Operation and Maintenance*

4.1.5.1 **Menjalankan Aplikasi**

Program MP3 *player* langsung dapat dijalankan setelah di install, program hanya mempunyai fungsi-fungsi dasar dalam mengoperasikan sebuah *player* pemutar musik. Berikut ini adalah tampilan awal program pada saat dijalankan. Dapat dilihat pada Gambar 5.

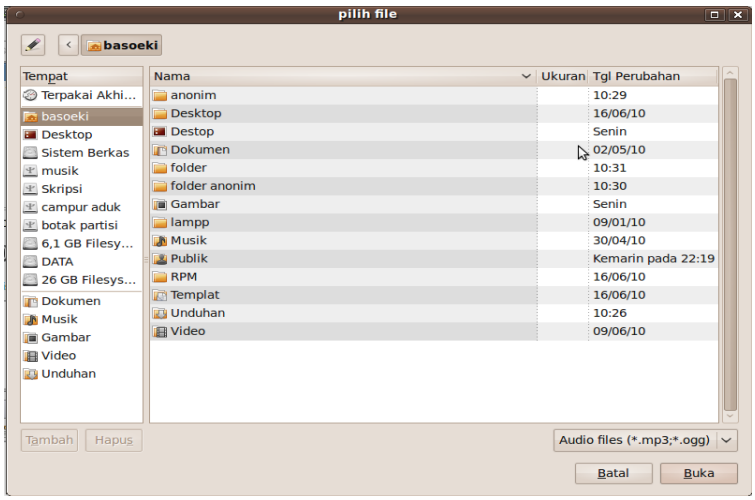


Gambar 5. *Tampilan Awal Program*

Tampilan awal program hanya memiliki beberapa tombol yaitu tombol play, stop, next, previous, random dan repeat. Pada tampilan awal juga disediakan tiga menu yaitu :

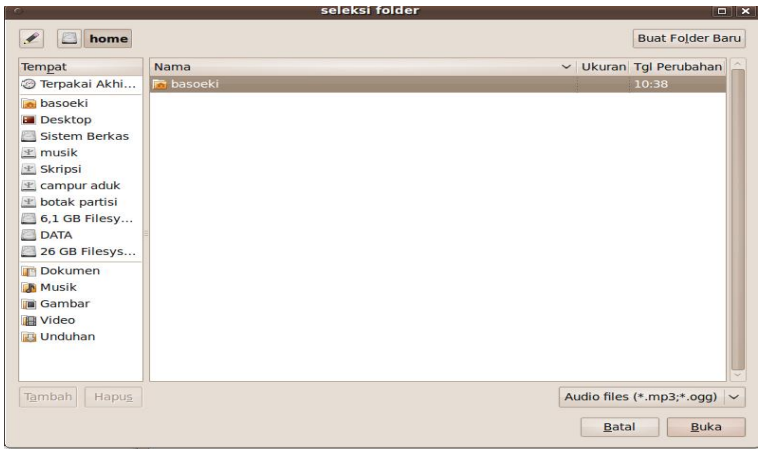
- *File*

1. Import *File* yang berfungsi memasukkan *file* musik ke dalam *playlist* berdasarkan *file* musik yang memiliki format MP3 dan OGG. Berikut ini adalah tampilan jendela pada saat proses pemilihan lagu yang dimainkan. Dapat dilihat pada Gambar 6.



Gambar 6. *Tampilan Import File*

2. Import Folder yang berfungsi memasukkan *file* secara sekaligus yang terdapat di dalam *folder* yang berisi *file* musik. Berikut ini adalah tampilan jendela pada saat memillih folder yang berisi *file* musik yang dimainkan. Dapat dilihat pada Gambar 7.

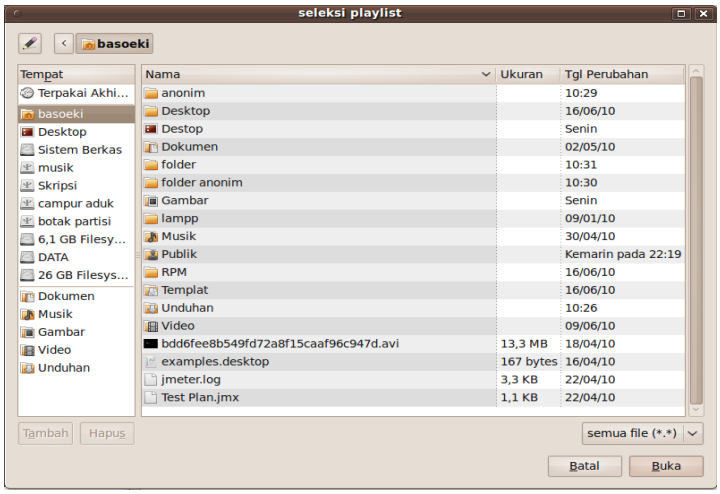


Gambar 7

Tampilan Import File per Folder

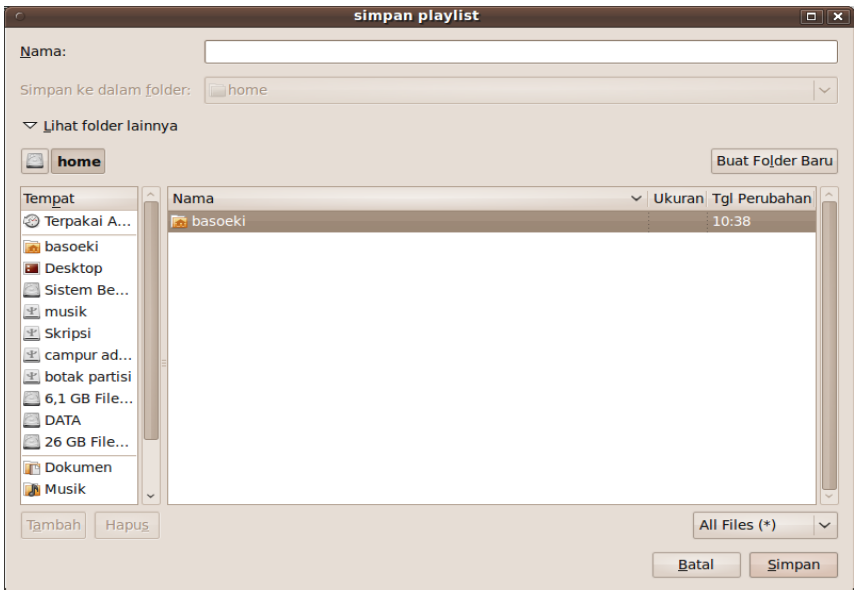
- *Playlist*

1. Membuka *playlist* yang berfungsi untuk membuka *playlist* lagu yang sudah pernah di buat. Berikut ini adalah tampilan jendela pada saat memilih *file playlist* yang di buka. Dapat dilhat pada Gambar 8.



Gambar 8. *Tampilan Open Playlist*

2. Menyimpan *playlist* yang berfungsi untuk menyimpan *playlist* lagu yang telah dipilih oleh *user*. *File playlist* disimpan dalam bentuk *txt.* Berikut ini adalah tampilan dari jendela menyimpan *playlist*. Dapat dilihat pada Gambar 9.



Gambar 9. *Tampilan Simpan Playlist*

➢ Tentang

1. Informasi, berisi tentang informasi tentang *software* yang telah dibuat.

2. Situs *web*, berfungsi untuk masuk ke dalam situs *web* pribadi peneliti. Dikarenakan peniliti belum mempunyai situs pribadi maka alamat yang digunakan sementara adalah http://www.google.com .

Berikut ini adalah gambar tampilan yang disajikan dibagian informasi. Dapat dilihat pada Gambar 10.

Gambar 10. *Tampilan Information*

4.2 Pembahasan

Pada program *aplikasi* MP3 *player* di GNU Linux dapat di ilustrasikan sebagai berikut, Terdapat *file* musik yang berformat MP3 dan OGG yang di mainkan oleh sebuah *player* pemutar musik. Di dalam men seleksi format lagu program hanya mendeteksi *file* musik yang berformat MP3 dan OGG.

Setelah pada masa tahap penyeleksian format musik, *player* memasukkan informasi tentang alamat *file* di dalam *playlist* yang telah tersedia, apabila *file* musik dijalankan *user* mengklik tombol *play* untuk memulai proses dalam

memainkan lagu, apabila *file* musik mulai dijalankan secara otomatis informasi tentang *file* musik ditampilkan oleh program dibagian atas program.

Pada saat program sedang memproses *file* musik yang telah di pilih, *user* mulai mendengarkan sebuah lagu yang dikeluarkan oleh sebuah *PC* atau *notebook* . Di dalam program ini *user* dapat memasukkan *file* musik lebih dari satu, bisa sekaligus ataupun memilih satu persatu. Jika *user* ingin sekaligus dalam memilih *file* musik, maka *user* menggunakan menu seleksi *folder* yang berfungsi memasukkan *file* musik sekaligus langsung dalam satu folder.

Jika *user* ingin memainkan lagu selanjutnya yang sudah terdaftar pada *playlist*, *user* dapat melakukan dengan secara manual yaitu dengan menekan tombol *next* untuk memulai proses memainkan *file* musik yang selanjutnya. Jika program dapat memainkan *file* musik yang selanjutnya berdasarkan *playlist* secara manual, maka *user* juga dapat memainkan *file* musik yang sebelumnya berdasarkan *playlist* yang telah dibuat dengan cara menekan tombol *previous* yang berfungsiuntuk memainkan lagu yang sebelumnya yang tersedia di dalam *playlist* yang telah dibuat oleh *user*.

Jika *user* ingin menghapus *playlist* yang sudah ada, *user* dapat menghapusnya dengan tombol bersihkan *playlist* yang berfungsi membersihkan kotak *playlist* secara keseluruhan. Jika *user* ingin memainkan lagu secara berulang-ulang maka *user* harus menekan tombol *repeat* yang berfungsi untuk memainkan lagu secara berulang. Di dalam memainkan lagu secara acak atau tidak sesuai daftar *playlist* yang tersedia maka *user* harus menekan tombol

random yang berfungsi memainkan *file* musik yang terdapat dalam *playlist* secara acak.

Perintah terakhir di dalam program pemutar musik ini adalah cara menghentikan proses memainkan lagu yaitu dengan cara menekan tombol *stop* yang berfungsi menghentikan semua proses program yang sedang berjalan

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah menyelesaikan *Pengembangan Aplikasi MP3 Player Pada System Operasi* GNU Linux, maka dapat diambil beberapa kesimpulan yaitu :

1. Pengembangan *aplikasi* MP3 *player* ini hanya memiliki fitur-fitur dasar di dalam sebuah *player* pemutar musik.
2. *Aplikasi* MP3 *player* ini hanya bisa memainkan dua bentuk format yaitu format MP3 dan OGG.

5.2 Saran

Penulis ingin memberikan beberapa saran yang mungkin dapat membantu dalam pengembangan *aplikasi* MP3 *player* ini yaitu :

1. *Aplikasi* ini dapat ditambahkan *equalizer* agar dapat mengeluarkan beberapa jenis suara.

2. *Aplikasi* dapat ditambahkan tombol pause supaya dapat menghentikan lagu dan melanjutkannya lagi tanpa harus memulai dari awal.
3. *Aplikasi* ini hanya dapat menampilkan letak direktori *file* yang sedang dimainkan, disarankan agar dapat menambahkan membuat *file* menampilkan judul lagu yang sedang di proses.
4. Dalam Pembuatan *aplikasi* ini masih terdapat kekurangan dan kelemahan, sehingga dibutuhkan masukan-masukan yang mendukung bagi kemajuan dan perkembangan *Aplikasi* MP3 player ini.

DAFTAR PUSTAKA

Abdul Kadir. 2006. *Pengenalan GNU Linux dan Unix*. Penerbit Andi. Yogyakarta 570 Halaman.

D'za. 2010. Mengenal Aneka Format *File* Musik Digital. Diakses 02 Mei 2010, 19.15 WIB.http://www.doktertomi.com/2006/02/25/mengenal-aneka-format-*file*-musik-digital/

D'za. 2010. Pengenalan Gambas. Diakses 03 Mei 2010, 19.20 WIB. http://osc.ugm.ac.id/ossdata/index.php?title=Gambas

D'za. 2010. Pengenalan Data Flow Diagram (DFD). Diakses 03 Mei 2010, 19.45 WIB. http://komputer-one.blogspot.com/2009/01/data-flow-diagram-dfd.html

Dian Rakyat. 2007. *Menguasai Gambas.* Penerbit Dian Rakyat. Yogyakarta 368 Halaman.

Juhaeri. 2009. multimedia_bagian_1.pdf http://ilmukomputer.org/wp-content/uploads/2009/07/juhaeri .Diakses 06 September 2010.19.46 WIB

Mahdi Ridho. 2006. *Panduan Aplikatif: Pemrograman Gambas - Pemrograman ala Visual Basic di Linux*. Penerbit Andi. Yogyakarta 168 Halaman.

UUS Rusmawan. 2009. *Koleksi Program Visual Basic 6.0: Konsep ADO Untuk Tugas Akhir & Skripsi (Edisi Baru).* Penerbit Elex Media Komputindo. Bandung 248 Halaman.