

**APLIKASI PEMBatasan WAKTU BACA E-BOOK  
DENGAN METODE COUNTDOWN TIMER**

**SKRIPSI**



Disusun oleh :

**MOH. SYARIF HIDAYATULLAH**  
**09.18.186**

**JURUSAN TEKNIK INFORMATIKA S-1**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**2013**

## SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Moh. Syarif Hidayatullah

NIM : 0918186

Jurusan : Teknik Informatika S-1 Institut Teknologi Nasional  
Malang

Alamat : Pendowolimo Karangbinangun Lamongan

Dengan ini menyatakan dengan sebenarnya dan sejujurnya bahwa skripsi yang berjudul "*Aplikasi Pembatasan Waktu Baca E-book dengan Metode Countdown Timer*" adalah hasil dari penelitian saya sendiri dan bukan merupakan hasil plagiasi karya orang lain.

Malang, 6 April 2013

Yang menyatakan,



Moh. Syarif Hidayatullah  
NIM. 0918186

## KATA PENGANTAR

Dengan memanjatkan syukur Alhamdulillah kepada Tuhan Yang Maha Esa karena dengan rahmat yang diberikan sehingga para penulis dapat menyelesaikan skripsi ini dapat terselesaikan dengan baik sebagai salah satu persyaratan untuk menyelesaikan pendidikan Strata 1 Jurusan Teknik Informatika Institut Teknologi Nasional Malang.

Dan tidak ketinggalan dalam kesempatan ini penulis juga mengucapkan terima kasih yang sebesar-besarnya kepada:

1. **Kedua Orang tua**, serta seluruh keluarga besar yang telah mendukung baik secara moral maupun materil dan juga telah mendoakan sehingga penulis dapat menyelesaikan skripsi ini dengan baik.
2. **Bapak Ir. Soeparno Djiwo, MT**, selaku Rektor Institut Teknologi Nasional Malang.
3. **Bapak Joseph Dedy Irawan, ST., MT**, selaku Ketua Jurusan Teknik Informatika Institut Teknologi Nasional Malang dan juga sebagai Dosen Pembimbing I yang telah bersabar dan ikhlas dalam membimbing penulis dalam penyusunan skripsi ini.
4. **Ibu Nurlaily Vendyansyah, ST.** selaku Dosen Pembimbing II dalam Membimbing Skripsi Teknik Informatika Institut Teknologi Nasional Malang.
5. Seluruh Staf Karyawan Institut Teknologi Nasional Malang yang terlibat secara langsung atau tidak langsung dalam membantu kelancaran pelaksanaan penyusunan Skripsi.
6. Serta semua pihak yang telah membantu secara langsung maupun tidak langsung sehingga penulis dapat menyelesaikan Skripsi ini.

Penulis menyadari bahwa skripsi ini jauh dari kata sempurna sehingga penulis menerima segala saran dan kritik terhadap skripsi untuk pengembangan penelitian kedepannya.

Akhir kata penulis mohon maaf yang sebesar-besarnya bila mana dalam penyusunan Skripsi ini terdapat kekurangan serta kesalahan. Semoga Skripsi ini bermanfaat bagi kita semua dan mendapat Ridho dari Allah SWT.

Malang, 5 Pebruari 2013

Moh. Syarif Hidayatullah  
0918186

## DAFTAR ISI

LEMBAR PERSETUJUAN .....	ii
LEMBAR PERNYATAAN .....	iii
LEMBAR PERSEMBAHAN .....	iv
ABSTRAK .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	x
DAFTAR TABEL.....	xi
BAB I .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	2
1.5 Metodologi Pemecahan Masalah .....	3
1.6 Sistematika Penulisan .....	3
BAB II TINJAUAN PUSTAKA .....	5
2.1 <i>E-book</i> .....	5
2.2 Countdown Timer .....	5
2.3 Java .....	7
2.4 JDK (Java Development Kit) .....	8
2.5 JPedal .....	12
2.6 Java GUI .....	12
2.7 PDF .....	14
2.8 SQLite .....	16
BAB III PERANCANGAN SISTEM .....	17
3.1 Analisa Sistem .....	17
3.2 Spesifikasi Aplikasi .....	17
3.3 Flowchart .....	18
3.4 Block Diagram Sistem .....	19
3.5 Perancangan Aplikasi .....	20

<b>BAB IV IMPLEMENTASI DAN PENGUJIAN</b> .....	25
4.1 Setup Aplikasi .....	25
4.2 Implementasi Sistem .....	25
4.3 Hasil dan Pengujian Aplikasi .....	26
4.3.1 Membuka Aplikasi .....	26
4.3.2 Fungsi Tombol .....	27
4.3.3 Proses Countdown Timer .....	28
<b>BAB V PENUTUP</b> .....	31
5.1 Kesimpulan .....	31
5.2 Saran .....	31
<b>DAFTAR PUSTAKA</b> .....	32
<b>LAMPIRAN</b> .....	33

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dalam zaman yang serba instan seperti sekarang ini mendorong kita untuk berpikir cepat dalam memenuhi kebutuhannya. Termasuk kebutuhan akan ilmu pengetahuan kita enggan untuk pergi ke suatu perpustakaan atau toko buku untuk sekedar mencari referensi atau pengetahuan yang kita cari. Dibantu dengan kemajuan teknologi yang memunculkan berbagai bentuk *digital library* atau perpustakaan online bahkan untuk membeli buku kita dapat memesan melalui *internet* dan menunggu buku yang kita pesan sampai kerumah dengan melalui jasa antar.

*E-book* adalah salah satu hasil kemajuan teknologi bidang buku yang sekarang banyak digunakan oleh sebagian penulis untuk meyebar luaskan hasil karyanya. *E-book* atau elektronik book selain mempunyai ukuran yang relative kecil dibandingkan dengan buku-buku konvensional yang biasanya kita beli di pasaran *e-book* juga mudah didapatkan. Pembaca dengan mudah membaca atau memiliki *e-book* hanya dengan cara membuka suatu situs web yang banyak tersedia di *internet* dan mencari *e-book* atau referensi yang dicari. Sehingga waktu yang dibutuhkan untuk mendapatkan suatu referensi semakin singkat. *E-book* juga dapat mengurangi penggunaan kertas untuk penulisannya. Karena bentuknya yang *digital* dan umumnya berupa *pdf, html, exe* dan banyak yang lainnya.

Kemunculan *e-book* juga memunculkan beberapa masalah terutama dalam hal hak kekayaan intelektual seorang penulis, dengan adanya buku yang berbentuk digital ini pembaca dengan mudah membaca dan mengambil referensi didalamnya tanpa memperhatikan hak yang harus didapatkan oleh seorang penulis. Lain halnya dengan penulis buku yang konvensional yang apabila seorang pembaca ingin membaca buku harus membayar sejumlah biaya yang nantinya biaya tersebut diberikan dengan penulis dan beberapa pihak yang turut mengedarkan buku tersebut.

Untuk mengurangi masalah di atas penulis bermaksud melakukan penelitian yang berjudul "**Pembatasan Waktu Baca *E-book* dengan Metode Count**

*Down Timer*”, yang diharapkan nantiya aplikasi desktop berbasis java ini dapat menjadi bentuk *e-book* yang baru. Sehingga para penulis tidak perlu khawatir akan hasil karyanya dibaca orang dengan waktu yang lama tanpa adanya imbalan yang diberikan kepada penulis. Penulis dapat menyeting waktu yang akan digunakan dalam membaca *e-book* dan apabila waktu telah habis maka *e-book* tidak dapat dibuka lagi. Apabila pembaca ingin membaca buku lagi pembaca harus mengunduh ulang buku tersebut yang tentunya dengan persyaratan yang diberikan oleh penulis.

### 1.2 Rumusan Masalah

Adapun rumusan masalah dalam aplikasi ini adalah sebagai berikut:

- a) Bagaimana membuat suatu pembatasan waktu baca bagi *e-book* dengan metode *countdown timer*.
- b) Bagaimana membangun suatu aplikasi pembatasan waktu baca waktu baca *e-book* dengan bahasa pemrograman java.

### 1.3 Tujuan

Pembuatan aplikasi ini bertujuan untuk mengimplementasikan metode *countdown timer* sebagai acuan dalam pembatasan waktu baca *e-book* dengan menggunakan bahasa pemrograman java.

### 1.4 Batasan Masalah

Dalam pembuatan aplikasi ini perlu adanya batasan masalah untuk memfokuskan pembahasan. Adapun batasan masalah tersebut adalah sebagai berikut:

- a) Buku yang dapat dibatasi waktu aksesnya terlebih berbentuk *e-book* dengan format pdf.
  - b) E-book hasil generate berformat .jar
  - c) E-book akan otomatis tidak dapat dibaca apabila waktu yang telah ditentukan telah habis.
  - d) Aplikasi dibangun dengan bahasa pemrograman java.
-



- e) Data base untuk menyimpan waktu bersifat *standalone* menjadi satu kesatuan dengan sistem aplikasi.
- f) Aplikasi dapat digunakan dalam Sistem Operasi yang telah terinstal JDK (*Java Development Kit*) versi 7.

### 1.5 Metodologi Pemecahan Masalah

Metode yang digunakan dalam pembahasan skripsi ini adalah:

- a) Studi Literatur  
Mencari referensi yang berhubungan dengan perencanaan dan pembuatan program yang akan dibuat.
- b) Perancangan Program  
Sebelum melaksanakan pembuatan program, dilakukan perancangan terhadap program yang meliputi merancang keseluruhan program.
- c) Pembuatan program  
Pada tahap ini realisasi program yang dibuat, dilakukan perakitan sistem terhadap seluruh hasil rancangan yang telah dibuat.
- d) Pengujian program  
Untuk mengetahui cara kerja program, maka dilakukan pengujian secara keseluruhan.

### 1.6 Sistematika Penulisan

Dalam penyusunan skripsi ini agar lebih mudah dipahami maka dibuatlah suatu sistematika penulisan sebagai berikut:

#### **BAB I : PENDAHULUAN**

Berisi latar belakang, perumusan masalah, batasan masalah, tujuan dan metodologi penelitian.

#### **BAB II : LANDASAN TEORI**

Berisi teori-teori yang menunjang dalam proses pembuatan skripsi ini.

#### **BAB III : PERANCANGAN SISTEM**

## BAB II LANDASAN TEORI

### 2.1 *E-Book*

*E-book* atau *elektronik book* adalah suatu buku elektronik atau digital yang dapat dibaca melalui komputer secara elektronis. *E-book* memiliki beberapa bentuk namun yang paling sering kita jumpai adalah *e-book* dalam bentuk pdf atau html. Selain berbentuk pdf dan html *e-book* juga banyak dijumpai dalam bentuk exe yang dalam pengoperasiannya pembaca harus menginstall installernya.

*E-book* banyak disukai banyak penulis dikarenakan karakteristik *e-book* yang memiliki ukuran relatif kecil dan cara pembuatannya sangat mudah. *E-book* memiliki beberapa manfaat atau kelebihan diantaranya adalah sebagai berikut:

- a. Memiliki ukuran yang kecil dan penyimpanannya dapat disimpan dalam harddisk maupun flashdisk dan media penyimpanan data digital yang lain.
- b. Dapat dibaca dimana-mana dan mudah dibawa dibandingkan buku cetak.
- c. Tidak mudah rusak karena pengaruh dari luar berbeda dengan buku cetak yang dapat rusak apabila terkena air atau dimakan usia.
- d. Mudah dalam pengoperasian kata atau frasa dapat dicari atau dilacak dengan cepat dan mudah.
- e. Dapat digandakan dan didistribusikan dengan mudah<sup>[1]</sup>.

### 2.2 *Countdown Timer*

*Countdown timer* terdiri dari tiga kata yaitu *count* yang berarti penghitung atau menghitung dan *down* yang berarti turun atau kebawah sedangkan *timer* berarti pewaktu atau waktu. Sehingga arti dari *countdown timer* adalah penghitung waktu mundur maksudnya adalah suatu proses atau dimana perhitungan waktu yang biasanya dilakukan secara bertambah terus keatas namun pada *countdown timer* perhitungan waktu dilakukan secara kebawah atau menurun. *Countdown timer* hampir sama dengan perhitungan atau timer pada umumnya yang apabila setiap 60 detik menit akan bertambah 1 menit dan setiap 60 menit jam akan bertambah 1 jam namun dalam *countdown timer* perhitungan dilakukan dengan melakukan pengurangan setiap 60 detik maka menit akan

dikurangi 1 menit dan setiap 60 menit jam akan dikurangi 1 jam. Perhitungan tersebut akan dilakukan sampai proses menemukan suatu kondisi yang mengharuskan berhenti. Dalam aplikasi ini *countdown timer* di set berhenti ketika semua kondisi waktu jam, menit dan detik berada pada angka 0. Sehingga proses akan keluar dari perulangan.

Dalam perhitungan *countdown timer* terlebih dahulu kita tentukan waktu awal, waktu akhir dan seberapa cepat waktu akan melakukan pengurangan. Misal waktu awal kita set 2 jam atau sama dengan 120 menit sama dengan 7200 detik dan 7200000 milidetik dan waktu terakhir adalah 0 dan percepatan hitungan mundur sebesar 1000 milidetik. Dengan kondisi seperti di atas maka perhitungan akan dimulai pada jam 02 : 00 : 00 ketika program dijalankan secara otomatis waktu akan secara konstan berkurang 1 point dimulai dari detik selama 1000 milidetik atau 1 detik. Sehingga waktu akan berubah menjadi 01 : 59 : 59 ketika detik telah mencapai 00 maka pengurangan akan dilakukan pada menit dan detik akan kembali ke angka 59 namun pada menit 58 sehingga hasil waktu yang ditunjukkan 01 : 58 : 59. Begitu pula apabila menit menjadi 00 maka pengurangan akan dilakukan pada jam sehingga menit akan kembali ke angka 59 tetapi dalam keadaan jam 00, hasilnya adalah 00: 59 : 59. Akan terjadi perulangan sampai menemukan suatu keadaan dimana angka pada jam, menit dan detik menunjukkan angka 00 : 00 : 00 maka dengan demikian waktu telah habis. Karena keadaan 00 : 00 : 00 sesuai dengan keadaan waktu akhir yang telah ditentukan pada awal program.

Dalam perhitungan *countdown timer* percepatan pengurangan waktu juga dapat disesuaikan dengan kebutuhan misalkan dalam percepatan pengurangan kita tentukan sebesar 100 milidetik sama dengan 0,1 detik sehingga pengurangan waktu akan lebih cepat 0,9 dari waktu normal yaitu 1000 milidetik atau 1 detik. Dapat pula percepatan pengurangan dapat lebih diperbesar misalkan sebesar 10000 milidetik atau 10 detik maka perhitungan akan lebih lambat sebesar 9 detik dari waktu normal. Selain dapat melakukan perhitungan selama jam, menit dan detik *countdown timer* dapat juga digunakan dalam perhitungan hari, bulan maupun tahun dengan menentukan angka dari masing-masing variable waktu.

Dan menambahkan kondisi pada setiap variable waktu yang lebih kecil berangka 0 maka variable yang lebih besar akan berkurang satu point.

### 2.3 Java

Sejarah singkat java berawal dari tahun 1991 para profesor SUN Microsystem yang dipimpin oleh Patrick Naughton dan James Gosling ingin merancang bahasa komputer untuk perangkat consumer seperti TV cable box. Karena perangkat tersebut tidak memiliki banyak memori, bahasanya harus berukuran kecil dan mengandung kode yang handal. Juga karena manufaktur-manufaktur berbeda memilih processor yang berbeda pula maka bahasa harus bebas dari manufaktur. Maka proyek tersebut diberi nama "GREEN".

Kebutuhan akan kode yang bebas dari platform apapun mengantar tim untuk mempelajari bahasa Pascal yang pernah dicoba Niklaus Wirth, pencipta bahasa Pascal telah merancang bahasa portabel yang menghasilkan intermediate code untuk mesin hipotesis. Yang kemudian sering disebut dengan mesin maya (*virtual machine*) kemudian kode ini dapat digunakan diberbagai mesin yang memiliki interpreter. Green memilih virtual mesin untuk menepis isu utama yang menghancurkan bahasa netral terhadap arsitektur mesin apapun. Karena para tim dalam proyek tersebut memiliki basic pada C++ dan bukan pada Pascal maka kebanyakan sintaks pada bahasa diambil dari bahasa C++ dan mengadopsi orientasi objek bukan bahasa pemrograman procedural.

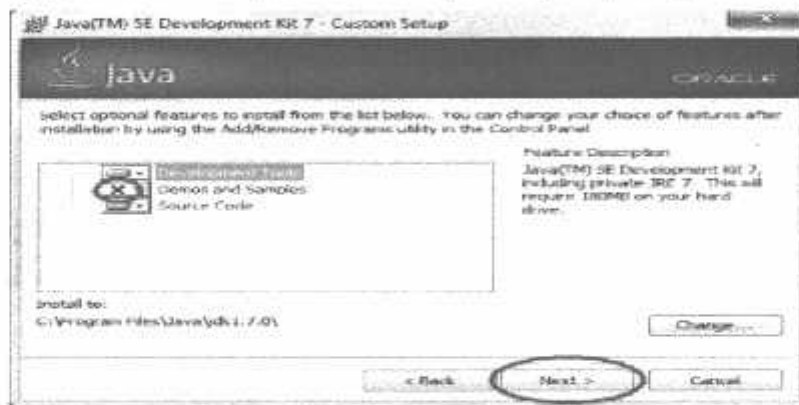
Java pada mulanya dikenal dengan bahasa pemrograman yang digunakan untuk memprogram internet namun dalam kenyataannya java digunakan untuk sebagai salah satu alternatif bahasa pemrograman sebelumnya yaitu bahasa C++. Yang pada awalnya bahasa C++ kurang diminati karena struktur dari bahasa C++ sendiri memiliki banyak kekurangan diantara source code yang tidak terstruktur. Dengan kemunculan bahasa pemrograman java yang bersifat aman, *portabel*, kokoh, *multiplatform*, interaktif dan tentunya berbasis orientasi objek atau yang biasa dikenal dengan OOP<sup>[3]</sup>.

Proses eksekusi bahasa pemrograman java dengan cara mengkompilasi (*compile*) yaitu menterjemahkan keseluruhan baris kode dan menterjemahkan (*interpret*) yaitu menterjemahkan kode program secara perbaris sehingga proses



*Gambar 2.1 Instalasi JDK*

- b. Klik next, pastikan Development tools terpilih seperti pada gambar 2.2



*Gambar 2.2 Instalasi JDK 2*

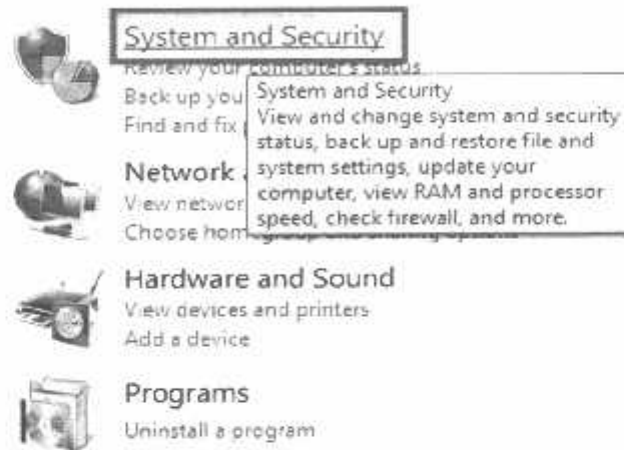
- c. Klik next, dan tunggu sampai proses instalasi selesai yang ditunjukkan oleh gambar 2.3



*Gambar 2.3 Proses Instalasi Selesai*

- d. Klik finish
- e. Apabila proses instalasi telah selesai maka langkah selanjutnya adalah setting JDK supaya dapat sesuai dengan olatform yang digunakan.
- f. Masuk ke **control panel** -> **system and security** seperti pada gambar 2.4

Adjust your computer's settings



ber:

*Gambar 2.4 Setting JDK Masuk ke System And Security*

- g. Pilih system seperti gambar 2.5



t

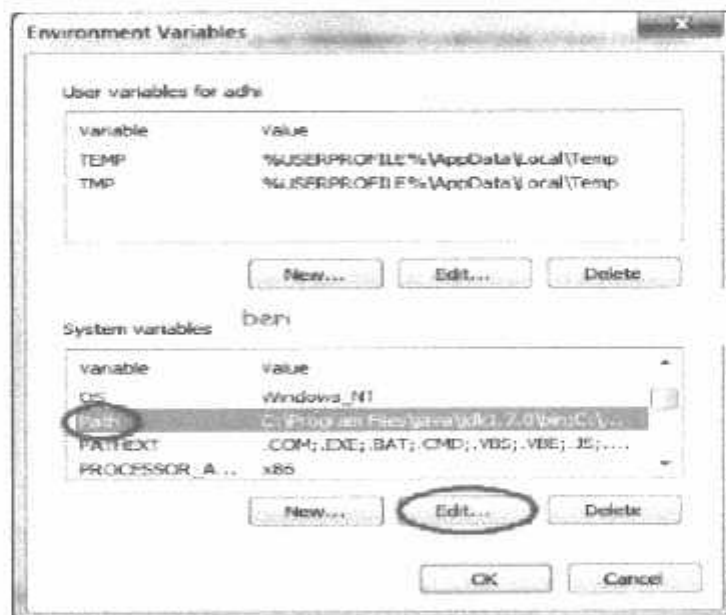
*Gambar 2.5 Advance System Settings*

- h. Pilih Environment variables->advanced seperti pada gambar 2.6



Gambar 2.6 Enviroment Variables

- i. Cari path->edit seperti ditunjukkan pada gambar 2.7



Gambar 2.7 Setting Path

- j. Tambahkan pada baris tulisan terakhir tanda (;) dan tambahkan **C:\Program Files\java\jdk1.7.0\bin;**
- k. Klik finish maka proses instalasi dan setting JDK telah selesai.

## 2.5 JPedal

JPedal adalah sebuah library yang ada dalam Java yang memungkinkan kita untuk memanipulasi data terkait file pdf. JPedal library didistribusikan dengan lisensi dari GPL. Dengan JPedal Library kita dapat membuat suatu aplikasi pdf viewer, pdf reader, print pdf dan beberapa manipulasi lain yang berhubungan dengan file pdf.

JPedal Library juga mendukung dengan berbagai bentuk font dan juga diaplikasikan dengan beberapa jenis warna seperti RGB, Gray, CMYK dan warna lainnya. JPedal Library juga dapat menampilkan berbagai jenis format baku seperti jpeg, tif, gif dan berbagai format baku lainnya. JPedal juga dilengkapi beberapa fitur seperti search halaman sehingga pengguna tidak perlu di pusingkan dengan halaman yang buku yang banyak. Pengguna hanya perlu memasukkan nomer halaman yang dibutuhkan dan mengetikan halaman tersebut kedalam text box yang tersedia. Tampilan pdf sendiri dapat ditampilkan menjadi seluruh halaman dalam satu tampilan dan satu persatu sesuai dengan keinginan pengguna<sup>[2]</sup>.

## 2.6 Java GUI

Dalm buku *How to Program* dijelaskan GUI adalah "*Graphical User Interface present a user friendly mekanisme for interactif with an application. GUI gives an applicationna distinctive look an Fell*" yang berarti GUI adalah suatu aplikasi yang menyajikan mekanisme *user friendly* yang interaktif dengan satu aplikasi. GUI memberikan suatu aplikasi yang berbeda yaitu *look and fell*<sup>[11]</sup>. Dengan sifat *user friendly* java dapat mengerti akan keutuhan dari user sehingga antara user dan aplikasi timbul suatu interaksi yang baik. GUI juga dapat dilihat dan dirasakan artinya aplikasi GUI dapat dilihat hasilnya dan dapat dirasakan manfaatnya yang berhubungan dengan sifat *user friendly*<sup>[11]</sup>.

Java GUI adalah pemrograman dengan bahasa Java yang dibuat menggunakan aplikasi yang berbasis GUI. Tujuannya adalah menambahkan beberapa komponen yang tidak bisa dibuat dalam basis text. Komponen-konponen tersebut bisa berupa tombol, gambar, dan lain-lain. Tujuannya adalah untuk memudahkan user menggunakan program yang dibuat tersebut. Kalau dilihat



pengertian tentang GUI secara umum adalah Interaksi yang dapat dilaksanakan oleh user melalui menu dan icon yang diperlihatkan dalam modus grafik. Contoh implementasi GUI-based shell ini adalah pada sistem operasi Microsoft Windows.

### 1. Jenis-Jenis Gui

Diantara jenis-jenis gui di java adalah:

- a. AWT (Abstract Window Toolkit) – SUN Microsystem -> Oracle Product  
 AWT (Abstract Window Toolkit) ini adalah GUI Toolkit pertama pada bahasa pemrograman Java, sayang-nya AWT ini sangat-sangat kekurangan komponen yang biasa digunakan untuk membangun sebuah aplikasi desktop secara lengkap (komponen tabel saja tidak ada ) Terlepas dari kurang-nya komponen GUI yang terdapat pada AWT (Abstract Window Toolkit), aplikasi yang dibangun menggunakan AWT (Abstract Window Toolkit) akan tampak seperti aplikasi native. Maksudnya yaitu, jika aplikasi yang dibangun menggunakan AWT (Abstract Window Toolkit) ini dijalankan pada Sistem Operasi Windows. Maka aplikasi ini akan terlihat seperti aplikasi Windows pada umum-nya, dan begitu juga jika dijalankan pada Sistem Operasi Mac ataupun GNU/Linux. Kenapa ini bisa terjadi, karena AWT (Abstract Window Toolkit) ini benar-benar memanggil native subrutin untuk menggambar setiap komponen-nya ke layar.
- b. SWT (Standart Widget Toolkit) – IBM Product -> Eclipse Foundation  
 SWT (Standart Widget Toolkit) ini adalah sebuah GUI Toolkit yang dikeluarkan oleh IBM sebagai alternatif dari AWT/Java Swing milik SUN Microsystem, yang membedakan antara SWT (Standart Widget Toolkit) dan AWT/Java Swing adalah SWT ini benar-benar mengakses native GUI library yang terdapat pada Sistem Operasi melalui JNI (Java Native Interface). Dengan model seperti ini, memungkinkan tampilan aplikasi yang dibangun menggunakan GUI Toolkit SWT menjadi sama persis dengan aplikasi native lain-nya. Kekurangan dari model pemanggilan native GUI library seperti ini adalah kita harus menyediakan library untuk tiap-tiap Sistem Operasi target aplikasi kita.

c. QtJambi – Trolltech -> Nokia Product -> Stopped and Taken By Community

Pernah menggunakan Desktop Environment KDE ? Ingin membuat aplikasi yang tampilan-nya mirip dengan KDE ? Kalau teman-teman ingin membangun aplikasi yang tampilan-nya tampak seperti aplikasi yang terdapat pada KDE tapi masih ingin menggunakan bahasa java sebagai dasar-nya, maka QtJambi adalah pilihan yang tepat untuk teman-teman. Karena QtJambi ini merupakan binding Qt framework dengan bahasa Java, tetapi sayang-nya proyek QtJambi sudah tidak disupport oleh Nokia dan secara resmi telah ditutup. Untung-nya, awal tahun ini ada beberapa developer yang peduli dengan kelangsungan proyek ini dan akhir-nya membuat sebuah komunitas untuk melanjutkan pengembangan proyek QtJambi.

d. JavaGNOME – Community Product

Kalau QtJambi diatas ditujukan untuk teman-teman yang sudah akrab dengan API (Application Programming Interface) Qt Framework, berbeda dengan JavaGNOME. Proyek ini lebih dikhususkan untuk teman-teman pecinta GTK atau yang paling banyak dikenal yaitu GNOME. Sama seperti GUI Toolkit SWT dan QTJambi, JavaGNOME ini juga mengakses native library tetapi API yang digunakan adalah API dari GTK. Untuk teman-teman yang sudah terbiasa membangun aplikasi menggunakan Glade, maka teman-teman bisa men-design form-nya menggunakan Glade dan memanggil-nya menggunakan bahasa java melalui JavaGNOME.

## 2.7 PDF

PDF (Portable Document Format) adalah sebuah format berkas yang dibuat oleh Adobe System pada tahun 1993 untuk keperluan pertukaran dokumen digital. Format PDF digunakan untuk merepresentasikan dokumen dua dimensi yang meliputi teks, huruf, citra dan grafik vektor dua dimensi. Pada Acrobat 3-D, kemampuan PDF juga meliputi pembacaan dokumen tiga dimensi. PDF telah menjadi standar ISO pada tanggal 1 Juli 2008 dengan kode ISO 32000-1:2008.

## BAB III

### ANALISA DAN PERANCANGAN SISTEM

#### 3.1 Analisa Sistem

Dalam sistem Aplikasi Pembatasan Waktu Baca *E-book* perlu diketahui berdasarkan apa suatu *e-book* harus dibatasi dalam waktu bacanya. Dalam aplikasi ini penulis mengemukakan suatu metode *countdown timer* yang dimana dalam sistem *countdown timer* waktu akan berkurang sebanyak 1 point selama rentan waktu 1000 m/s atau 1 detik waktu normal. Dengan metode *countdown timer* sistem pembatasan waktu baca *e-book* akan dengan mudah terlaksana, karena sistem akan mengacu pada perhitungan waktu oleh *countdown timer*. Apabila waktu pada *countdown timer* menunjukkan 0 maka sistem akan mengeksekusi sistem Pembatasan Waktu Baca *E-book*.

Apabila dalam tengah proses aplikasi berjalan aplikasi ditutup maka secara otomatis sistem akan menyimpan waktu terakhir kedalam database yang ada di database portable yaitu *sqlite*. Dan apabila aplikasi di buka atau dijalankan lagi di kemudian waktu maka waktu yang digunakan bukan waktu semula pada awal pertama kali aplikasi dibuka melainkan sistem akan mengacu waktu dimana waktu yang terakhir digunakan atau waktu terakhir sebelum aplikasi ditutup. Sistem akan mengacu kepada database yang menjadi tempat timer atau waktu tersimpan.

Selain penggunaan *countdown timer* yang perlu diperhatikan dalam aplikasi ini adalah penggunaan *Library eksternal* dari netbeans yaitu *Jpedal*. *Jpedal* digunakan untuk pembacaan *e-book pdf* yang dalam aplikasi ini sebagai objek utama. Dalam *Jpedal* sudah terdapat berbagai *header* yang digunakan untuk membaca *e-book pdf*.

#### 3.2 Spesifikasi Aplikasi

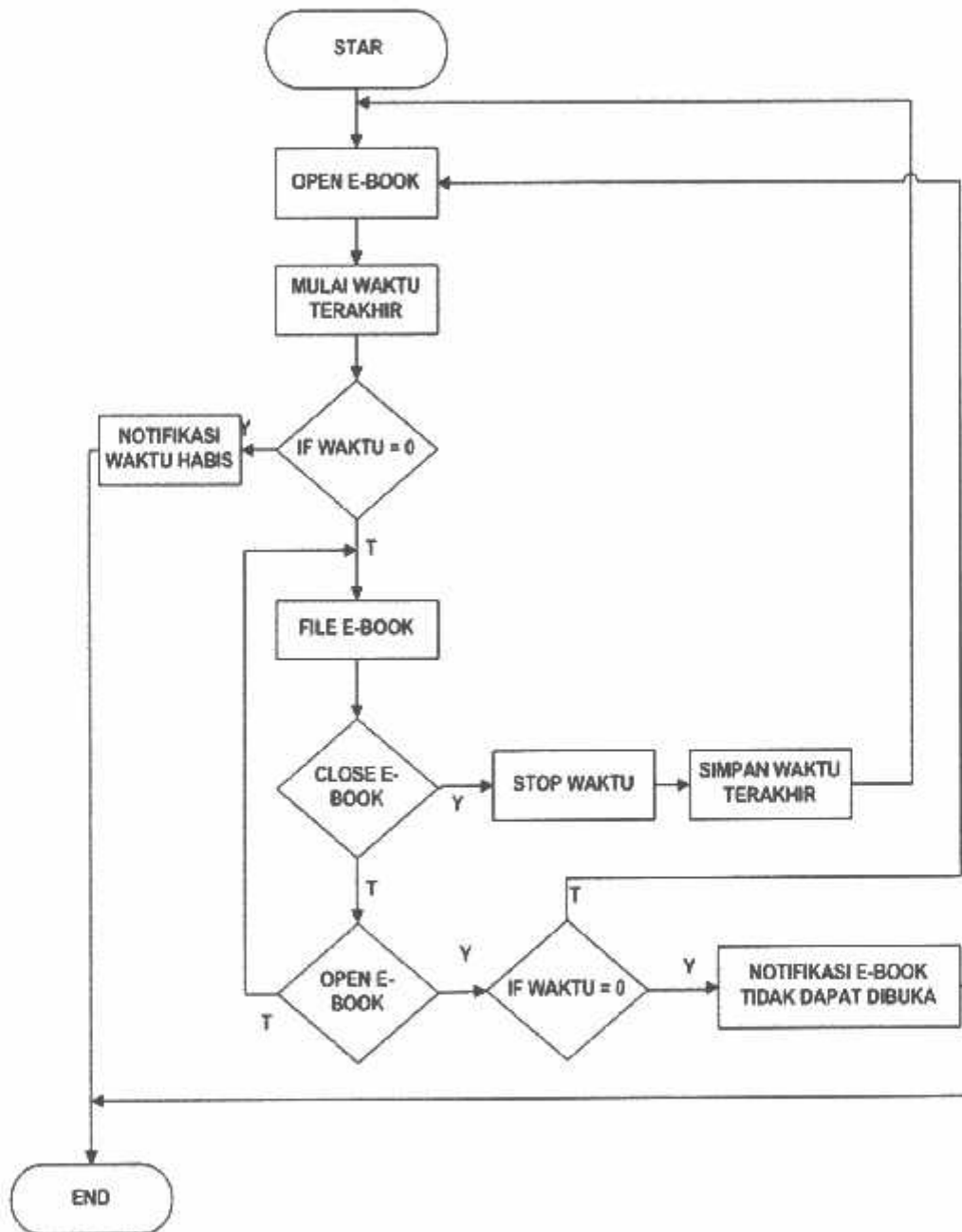
Untuk mempermudah perancangan dari aplikasi ini maka dibutuhkan beberapa *device* diantara adalah sebagai berikut:

1. Aplikasi dibangun dengan menggunakan bahasa pemrograman *Java*.
2. *Compiler* yang digunakan adalah *Netbeans Versi 7.2.1*.

3. *Library* eksternal yang digunakan adalah *Jpedal, sqlite-jdbc-3.7.2*.
4. Aplikasi *runtime* berextensi *jar*.
5. Aplikasi dapat dijalankan di berbagai *Operating Sistem* yang telah terinstal *JDK (Java Development Kit) versi 7*.

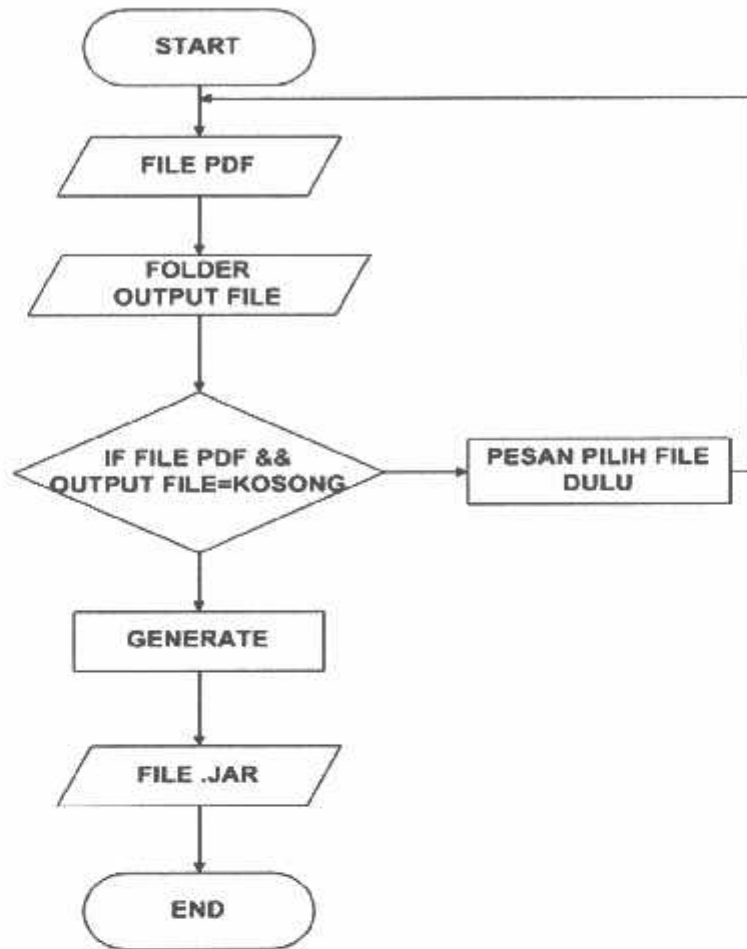
### 3.3 Flowchart

#### a. Flowchart Aplikasi User



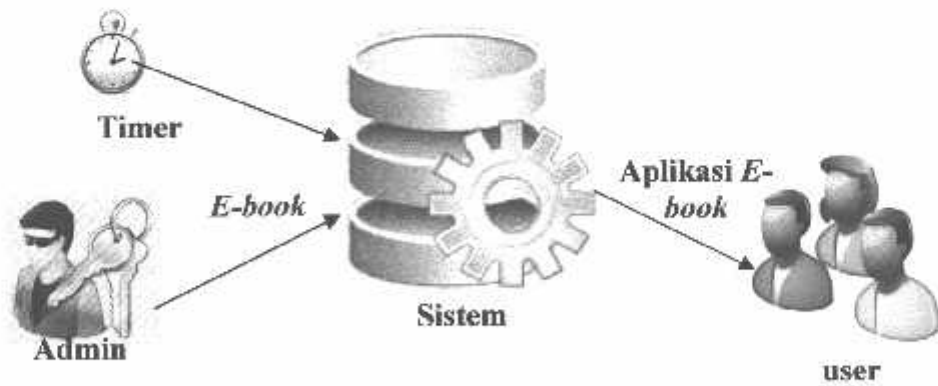
Gambar 3.1 Flow Chart Aplikasi User

### b. Flow Chart Aplikasi Admin



Gambar 3.2 Flow Chart Aplikasi Admin

### 3.4 Block Diagram Sistem

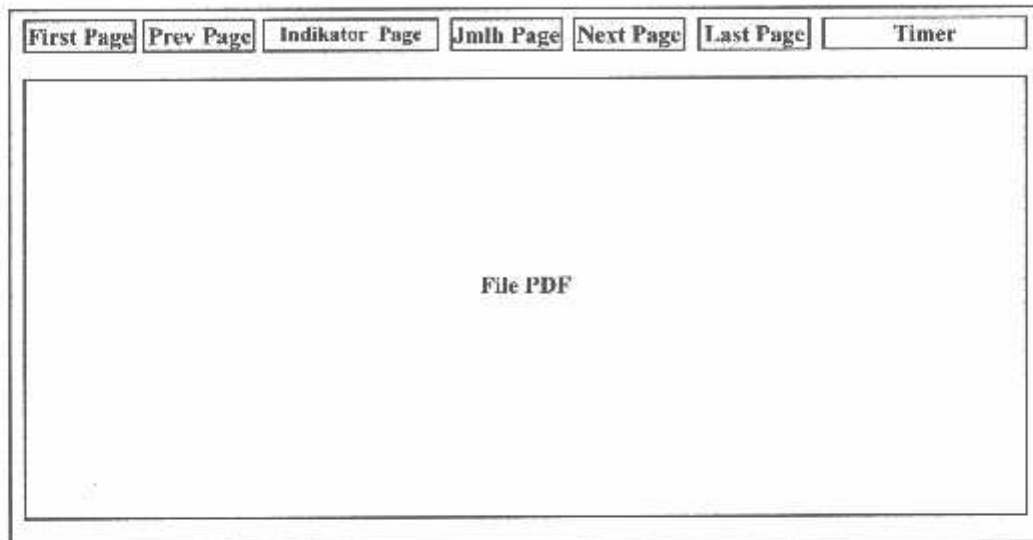


Gambar 3.3 Block Diagram Sistem

### 3.5 Perancangan Aplikasi

#### a. Desain GUI (*Grapical User Interface*) untuk *User*

Tampilan untuk pembaca perlu di rancang untuk lebih membantu pembaca dalam membaca file pdf. Rancangan dari desain tampilan untuk pembaca seperti dalam gambar 3.4 di bawah ini.



*Gambar 3.4 Desain GUI user*

Di dalam desain *GUI* memiliki beberapa tombol yang mempunyai fungsi sebagai berikut:

1. First page : digunakan untuk menuju ke halaman 1 atau halaman paling depan dari file.pdf.
2. Prev page : digunakan untuk menuju ke halaman sebelumnya dari file .pdf
3. Indikator page : berfungsi untuk menampilkan urutan halaman yang sedang dibaca kepada *user*.
4. Jmlh page : merupakan keterangan dari jumlah keseluruhan halaman file.
5. Next page : tombol yang berfungsi untuk menuju ke halaman berikutnya dari file .pdf.
6. Last page : digunakan untuk menuju ke halaman terakhir dari file .pdf.
7. Timer : merupakan indikator dari waktu yang tersedia untuk *user* dalam membaca file .pdf.

### b. Desain GUI (*Graphical User Interface*) untuk admin

Dalam posisi admin perlu adanya suatu form yang berfungsi untuk mempermudah admin dalam mengelola aplikasi tanpa harus membuka *source code* yang ada dalam aplikasi. Desain form untuk admin dapat dijelaskan dalam gambar 3.5 di bawah ini.

The diagram shows a rectangular window titled "JUDUL APLIKASI". Inside the window, there are two rows of input fields. The first row has a label "File PDF" followed by a text input box and a button labeled "Pilih File". The second row has a label "Output" followed by a text input box and a button labeled "Pilih Direktori". At the bottom left of the window, there is a button labeled "Generate".

Gambar 3.5 Desain GUI admin

Dalam desain *GUI* aplikasi untuk admin memiliki beberapa fitur diantaranya adalah sebagai berikut:

1. File PDF : sebagai indikator file.pdf yang akan digenerate.
2. Output : sebagai indikator tempat file.jar hasil generate.
3. Pilih File : tombol untuk memilih file.pdf dalam direktori.
4. Pilih Direktori : tombol untuk memilih tempat file.jar hasil generate.
5. Generate : yaitu tombol vital dalam aplikasi admin yang digunakan untuk mengenerate file.pdf yang dipilih menjadi file.jar dengan timer.

### c. Kebutuhan sistem

Untuk dapat membangun aplikasi PDF reader beberapa *software* yang harus disiapkan adalah sebagai berikut:

#### 1. SublimeText 2

SublimeText adalah salah satu aplikasi yang digunakan untuk mengetikkan *source code* yang pada setiap aplikasi.

## BAB IV IMPLEMENTASI DAN PENGUJIAN

Dalam bab IV ini akan dijelaskan tentang Implementasi Aplikasi dan Pengujian Aplikasi untuk mengetahui apakah aplikasi sudah sesuai dengan yang telah direncanakan.

### 4.1 Setup Aplikasi

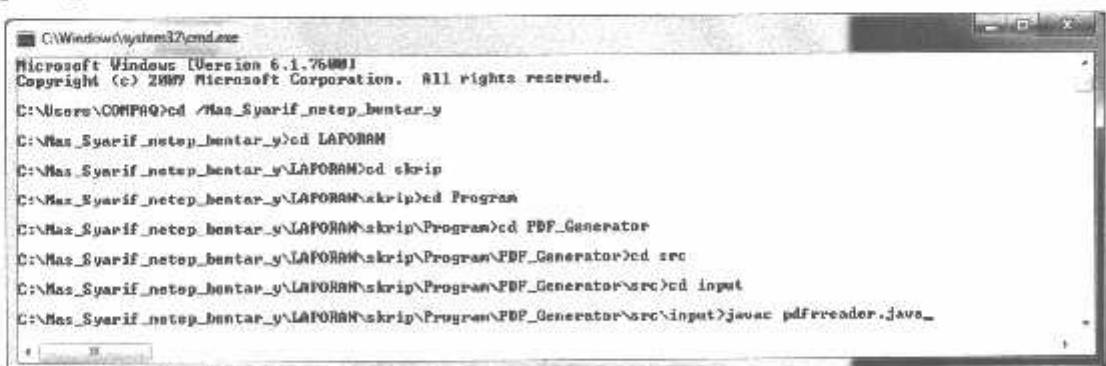
Dalam pengujian aplikasi Pembatasan Waktu Baca *E-book* dengan Metode *Countdown Timer* adalah sebagai berikut:

1. *Sublime Text 2*.
2. *JDK* versi 7.
3. *Windows 7*.
4. *Ubuntu 12.10*.

### 4.2 Implementasi Sistem

Dalam tahap implementasi ini dilakukan proses *build* aplikasi kedalam suatu *extensi* yang dapat langsung digunakan dalam sistem operasi yang dimiliki oleh user. User perlu juga menginstal *JDK (Java Development Kit)* versi 7. Proses implementasi dilakukan dengan cara *megepack* semua file yang ada di dalam project menjadi sehingga dapat langsung digunakan oleh user.

1. Masuk pada *command prompt* pada windows atau *terminal* pada ubuntu.
2. Masuk pada *direktory* tempat penyimpanan file *PDFReader.java*.
3. *Compile* file dengan mengetikkan perintah *javac PDFReader.java*. Seperti pada gambar 4.1 di bawah ini.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\COMPRO>cd /Mas_Syarif_netop_bentar_y
C:\Mas_Syarif_netop_bentar_y>cd LAPORAN
C:\Mas_Syarif_netop_bentar_y\LAPORAN>cd skrip
C:\Mas_Syarif_netop_bentar_y\LAPORAN\skrip>cd Program
C:\Mas_Syarif_netop_bentar_y\LAPORAN\skrip\Program>cd PDF_Generator
C:\Mas_Syarif_netop_bentar_y\LAPORAN\skrip\Program\PDF_Generator>cd src
C:\Mas_Syarif_netop_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src>cd input
C:\Mas_Syarif_netop_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src\input>javac pdfreader.java_
```

Gambar 4.1 *Compilasi File pdfReader.java dengan Command prompt*



4. Setelah itu masuk ke folder dimana file *frmGenerator.java* tersimpan.
5. Dan ketikkan perintah berikut untuk mengcompile *javac frmGenerator.java*. Seperti pada gambar 4.2 di bawah ini.

```

C:\Windows\system32\cmd.exe
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src\input>cd /Mas_Syarif_netep_bentar_y
C:\Mas_Syarif_netep_bentar_y>cd /Mas_Syarif_netep_bentar_y
C:\Mas_Syarif_netep_bentar_y>cd LAPORAN
C:\Mas_Syarif_netep_bentar_y\LAPORAN>cd skrip
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip>cd Program
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program>cd PDF_Generator
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator>cd src
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src>javac frmGenerator.java_

```

Gambar 4.2 Compilasi File *frmGenerator.java* dengan Command prompt

6. Setelah semua file telah dicompile dan tidak ada *source code* yang *error* maka *packge* keseluruhan file menjadi satu kesatuan dapat dilakukan.
7. Masih dalam folder yang dengan *frmGenerator.java* ketikkan perintah *jar cmf manifest.mf Generator.jar \*.class* pada *command prompt* seperti pada gambar 4.3 di bawah ini.

```

C:\Windows\system32\cmd.exe
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src\input>cd /Mas_Syarif_netep_bentar_y
C:\Mas_Syarif_netep_bentar_y>cd /Mas_Syarif_netep_bentar_y
C:\Mas_Syarif_netep_bentar_y>cd LAPORAN
C:\Mas_Syarif_netep_bentar_y\LAPORAN>cd skrip
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip>cd Program
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program>cd PDF_Generator
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator>cd src
C:\Mas_Syarif_netep_bentar_y\LAPORAN\skrip\Program\PDF_Generator\src>jar cmf manifest.mf Generator.jar *.class

```

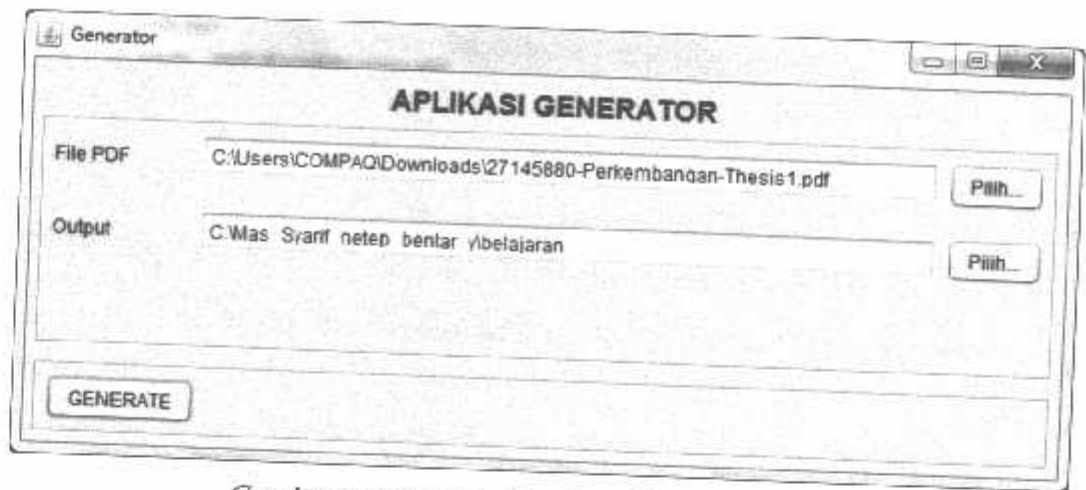
Gambar 4.3 Proses Packing seluruh File.java dan file .class

### 4.3 Hasil dan Pengujian Aplikasi

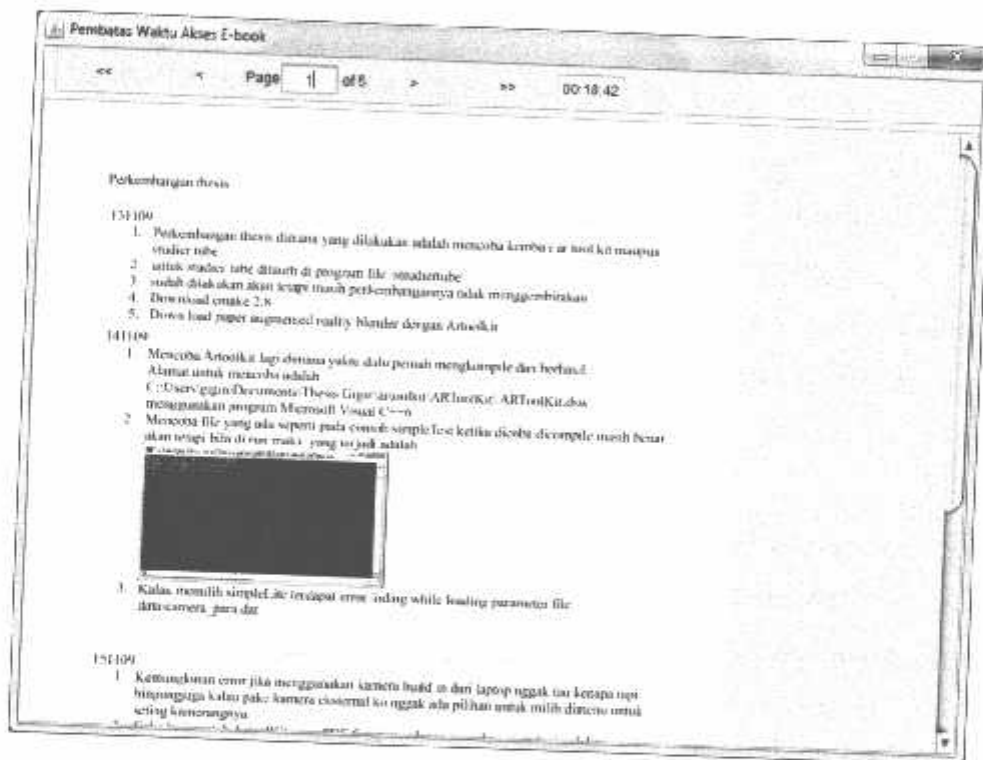
Dalam tahap ini aplikasi di uji menggunakan sistem operasi Windows 7 Ultimate dan JDK versi 7.

#### 4.3.1. Membuka Aplikasi

Untuk membuka hasil dari *generate* pada *direktory* pada saat memilih tempat *output* pada proses *generate* dengan cara *double click* maka akan muncul file yang tadi di pilih dengan timer.



Gambar 4.4 Hasil Aplikasi untuk Admin



Gambar 4.5 Hasil Form user

#### 4.3.2. Fungsi Tombol

Pengujian selanjutnya dilakukan untuk mengetahui apakah tombol yang ada pada aplikasi berjalan sesuai dengan fungsi yang diberikan.

Table 4.1 Hasil pengujian tombol aplikasi

Tombol	Hasil
Next	Sesuai dengan halaman yang dimaksud
Last	Sesuai dengan halaman yang dimaksud,

## BAB V PENUTUP

### 5.1. Kesimpulan

Dari proses penelitian aplikasi Pembatasan Waktu Baca *E-book* dengan Metode *Countdown Timer* penulis mendapat beberapa kesimpulan yaitu:

1. Dari hasil pengujian tombol semua tombol yang ada pada aplikasi PDF Reader berjalan sesuai dengan fungsinya. Sehingga aplikasi dapat mempermudah user dalam membaca file.
2. Dari hasil pengujian proses *countddown timer* waktu berjalan sesuai pengurangan 1 detik. Sehingga proses membaca akan berakhir pada waktu yang telah diberikan telah habis. Dan file tidak dapat dibuka lagi.
3. Dari hasil pengujian juga didapatkan bahwa bahasa pemrograman java merupakan bahasa pemrograman yang *multiplatform* yang dapat berjalan pada sistem operasi yang berbeda yaitu pada Windows 7 dan Ubuntu 12.10.
4. Dalam proses *compilasi* Ubuntu 12.10 lebih cepat 6 detik dibandingkan dengan Windows 7. Ini dikarenakan sifat dari ubuntu yang merupakan sistem operasi *open source*.

### 5.2. Saran

Aplikasi ini diharapkan dapat dikembangkan lebih lanjut sehingga lebih sempurna. Untuk itu penulis menyarankan kepada pembaca sebagai berikut:

1. Pengembangan aplikasi kedalam suatu aplikasi *mobile* untuk dapat membantu para user untuk membaca *e-book* didalam perangkat *mobile*.
2. Mengembangkan aplikasi yang dapat dijalankan tidak hanya pada JDK (*Java Development Kit*) versi 7 dan keatas melainkan dapat dijalankan pada sistem operasi yang belum terinstall JDK versi 7 maupun sistem operasi yang belum terinstall JDK sama sekali.
3. Pengembangan aplikasi kedalam suatu sistem yang dapat mendukung sistem pembatasan waktu akses seperti perpustakaan online.

## DAFTAR PUSTAKA

- [1]. Deitel, Paul. Deitel Harvey. 2010. *Java How to Program*. New Jersey: Pearson Education, Inc.
- [2]. Naughton, Patrick. 1996. *Java Hand Book "Konsep Dasar Pemrograman Java"*. Andi.
- [3]. Raharjo, Budi. Heriyanto, Imam. Haryono, Arif. 2007. *Mudah Belajar Java*. Bandung: Informatika.

# **LAMPIRAN**



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
PROGRAM STUDI TEKNIK INFORMATIKA S-1  
Jl. Raya Karanglo Km. 2 Malang

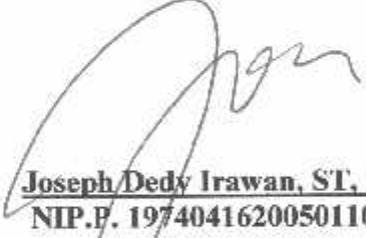
---

**BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Moh. Syarif Hidayatullah  
NIM : 0918186  
Jurusan : Teknik Informatika S-1  
Judul : APLIKASI PEMBatasan WAKTU BACA *E-BOOK* DENGAN  
METODE COUNTDOWN TIMER

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :  
Hari : Jum'at  
Tanggal : 15 Februari 2013  
Nilai : 90,13 (A)

Panitia Ujian Skripsi :  
Ketua Majelis Penguji

  
**Joseph Dedy Irawan, ST, MT**  
**NIP.P. 197404162005011002**  
Anggota Penguji :

Penguji Pertama



**Febriana Santi Wahyuni, Skom, MKom**  
**NIP.P. 1031000425**

Penguji Kedua



**Yosep Agus Pranoto, ST.**  
**NIP.P. 1031000432**

---



**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**PROGRAM STUDI TEKNIK INFORMATIKA S-1**  
**Jl. Raya Karanglo Km. 2 Malang**

**FORMULIR PERBAIKAN SKRIPSI**

Nama : Moh. Syarif Hidayatullah  
NIM : 0918186  
Jurusan : Teknik Informatika S-1  
Judul : APLIKASI PEMBATAAN WAKTU BACA *E-BOOK* DENGAN  
METODE COUNTDOWN TIMER

TANGGAL	PENGUJI	URAIAN
15 Februari 2013	I	- Tujuan penulisan - Tinjauan pustaka (Landasan Teori) - Tata bahasa penulisan - Daftar pustaka - Kesimpulan
15 Februari 2013	II	- Bahasa baku - Flowchart sistem - Penambahan form untuk admin

Anggota Penguji :

Penguji Pertama

**Febriana Santi Wahyuni, Skom, MKom**  
NIP.P. 1031000425

Penguji Kedua

**Yosep Agus Pranoto, ST.**  
NIP.P. 1031000432

Mengetahui

Dosen Pembimbing I

**Joseph Dedy Irawan, ST, MT**  
NIP.P. 197404162005011002

Dosen Pembimbing II

**Nurlaily Vendyansyah, ST.**



## FORMULIR BIMBINGAN SKRIPSI

Nama : MOH. SYARIF HIDAYATULLAH  
NIM : 09.18.186  
Masa Bimbingan :  
Judul Skripsi : APLIKASI PEMBATAAN WAKTU BACA E-BOOK  
DEMIKIAN METODE COUNTDOWN TIMER

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	3/10/2012	Sistem Aplikasi	
2	24/11/2012	BAB I & BAB II	
3	6/12/2012	BAB III	
4	23/12/2012	BAB IV & Hasil Pengujian	
5	13/1/2013	Makalah Seminar Hasil & BAB V	
6	28/1/2013	Muhammasi	
7	7/2/2013	Alli LAD.	
8	12/2/2013	Alli LUMBER.	
9			
10			

Malang, 12 Februari 2013

Dosen Pembimbing

Joseph Dedy Irawan, S.T. MT.  
NIP. : 197404162005011002



## Lampiran I

### Source code frmGenerator

```
import java.io.*;
import javax.swing.JFileChooser;
import javax.swing.JOptionPane;

/**
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author budi
 */
public class frmGenerator extends javax.swing.JFrame {

    File file_input = null, file_output = null;
    String jar_creator = File.separator+"Program
Files"+File.separator+"Java"+File.separator+"jdk1.7.0"+File.separator+"bin"+File.
separator+"jar";
    String separator = File.separator;

    /**
     * Creates new form frmGenerator
     */
    public frmGenerator() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        fc = new javax.swing.JFileChooser();
        jLabel1 = new javax.swing.JLabel();
        jPanel3 = new javax.swing.JPanel();
        btn_pilih_input = new javax.swing.JButton();

```

```

txt_output = new javax.swing.JTextField();
jLabel3 = new javax.swing.JLabel();
txt_input = new javax.swing.JTextField();
jLabel2 = new javax.swing.JLabel();
btn_pilih_output = new javax.swing.JButton();
jPanel2 = new javax.swing.JPanel();
btn_generate = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Generator");

jLabel1.setFont(new java.awt.Font("Dialog", 1, 18)); // NOI18N
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("APLIKASI GENERATOR");

jPanel3.setBorder(javax.swing.BorderFactory.createEtchedBorder());

btn_pilih_input.setText("Pilih...");
btn_pilih_input.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_pilih_inputActionPerformed(evt);
    }
});

jLabel3.setText("Output");

jLabel2.setText("File PDF");

btn_pilih_output.setText("Pilih...");
btn_pilih_output.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_pilih_outputActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addContainerGap()

```

```

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel2)
    .addComponent(jLabel3))
.addGap(45, 45, 45)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addComponent(txt_output,
javax.swing.GroupLayout.PREFERRED_SIZE, 461,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(btn_pilih_output,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addComponent(txt_input,
javax.swing.GroupLayout.PREFERRED_SIZE, 461,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(btn_pilih_input)))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);
jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel2)
    .addComponent(btn_pilih_input))
    .addComponent(txt_input,
javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addGap(18, 18, 18)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(txt_output,
javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btn_pilih_output))
        .addGap(57, 57, 57))
    };

jPanel2.setBorder(javax.swing.BorderFactory.createEtchedBorder());

btn_generate.setText("GENERATE");
btn_generate.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_generate_click(evt);
    }
});

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(btn_generate)
            .addGap(15, 15, 15))
        );
jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGap(15, 15, 15)
            .addComponent(btn_generate)
            .addGap(15, 15, 15))
        );

```



```

private void btn_pilih_inputActionPerformed(java.awt.event.ActionEvent evt)
{
    //GEN-FIRST:event_btn_pilih_inputActionPerformed
    // TODO add your handling code here:
    // set atribut hanya boleh memilih file saja, bukan folder
    fc.setSelectionMode(JFileChooser.FILES_ONLY);

    // tampilkan pilihan file
    int ret = fc.showOpenDialog(this);

    // jika dipilih
    if (ret == JFileChooser.APPROVE_OPTION) {
        file_input = fc.getSelectedFile();
        txt_input.setText(file_input.getAbsolutePath());
    }
}
//GEN-LAST:event_btn_pilih_inputActionPerformed

private void btn_pilih_outputActionPerformed(java.awt.event.ActionEvent
evt) {
    //GEN-FIRST:event_btn_pilih_outputActionPerformed
    // TODO add your handling code here:
    // set atribut hanya boleh memilih file saja, bukan folder
    fc.setSelectionMode(JFileChooser.DIRECTORIES_ONLY);

    // tampilkan pilihan file
    int ret = fc.showOpenDialog(this);

    // jika dipilih
    if (ret == JFileChooser.APPROVE_OPTION) {
        file_output = fc.getSelectedFile();
        txt_output.setText(file_output.getAbsolutePath());
    }
}
//GEN-LAST:event_btn_pilih_outputActionPerformed

private void btn_generate_click(java.awt.event.ActionEvent evt) {

    // cek input kosong?

    if (file_input == null || file_output == null) {
        JOptionPane.showMessageDialog(this, "Silakan pilih file input dan output
terlebih dahulu.");
    }
    else {
        File direktori_aplikasi = new
File(frmGenerator.class.getProtectionDomain().getCodeSource().getLocation().ge
tPath());
        String direktori = direktori_aplikasi.getParentFile().getPath();
    }
}
}

```

```

// JOptionPane.showMessageDialog(this, direktori);

// ambil nama file untuk disimpan di setting, dll
String file_output_name = file_input.getName().replace(" ",
"_").replace("(", "").replace(")", "");
byte[] output_name = file_output_name.getBytes();

// setting filename, taruh di folder input/filename.txt yang akan dijadikan
aplikasi JAR
try {
    FileOutputStream writer = new FileOutputStream(direktori + separator +
"input" + separator + "filename.txt");
    writer.write((new String()).getBytes());
    writer.write(output_name);
    writer.close();
} catch(IOException ex) {
    System.out.println(ex.getMessage());
}

// delete semua file di folder pdf
File[] files = new File(direktori + separator + "input"+ separator
+"pdf").listFiles();

for (int index = 0; index < files.length; index++) {
    files[index].delete();
}

// delete semua file di folder db
files = new File(direktori + separator + "input" + separator +
"db").listFiles();

for (int index = 0; index < files.length; index++) {
    files[index].delete();
}

// sebelum dicopy, update timer.db sesuai setting detik yg diinput oleh
user
// buat sendiri rif, masak gak bisa :p
// kalo gak bisa

// copy db
File output_db = new File(direktori + separator + "input" + separator +
"db" + separator + "timer.db");
try {

```

---

```

        InputStream templateStream = new FileInputStream(direktori +
separator + "input" + separator + "timer.db");
        OutputStream out = new FileOutputStream(output_db);

        byte[] buf = new byte[1024];
        int len;
        try {
            while ((len = templateStream.read(buf)) > 0) {
                out.write(buf, 0, len);
            }
            templateStream.close();
            out.close();
            System.out.println("File DB copied.");
            JOptionPane.showMessageDialog(this, "File DB copied.");

        } catch (IOException e) {
            System.out.println(e.getMessage());
        }
    } catch (FileNotFoundException e) {
        System.out.println(e.getMessage());
    }
}

// copy file pdf ke folder input/pdf yang akan di write ke aplikasi JAR
File output_pdf = new File(direktori + separator + "input" + separator +
"pdf" + separator + "file.pdf");
try {
    InputStream templateStream = new FileInputStream(file_input);
    OutputStream out = new FileOutputStream(output_pdf);

    byte[] buf = new byte[1024];
    int len;
    try {
        while ((len = templateStream.read(buf)) > 0) {
            out.write(buf, 0, len);
        }
        templateStream.close();
        out.close();
        System.out.println("File PDF copied.");
        JOptionPane.showMessageDialog(this, "File PDF copied.");

    } catch (IOException e) {
        System.out.println(e.getMessage());
    }
} catch (FileNotFoundException e) {
    System.out.println(e.getMessage());
}

```

---



```

    }

    // buat file JAR dengan menggunakan cli (command line interface)
    Runtime cli = Runtime.getRuntime();
    try {
        File wd = new File(direktori + separator + "input");
        //JOptionPane.showMessageDialog(this, wd.getAbsolutePath());

        String generate = jar_creator + " cmf manifest.txt " +
file_output.getAbsolutePath() + separator + file_output_name + ".jar
Db_timer.class pdfReader.class pdfReader$1.class pdfReader$2.class
pdfReader$3.class pdfReader$4.class pdfReader$5.class pdfReader$6.class
pdfReader$6$1.class db pdf com org native filename.txt";

        try {
            Process pr = cli.exec(generate, null, wd);
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(this, ex.getMessage());
        }

        JOptionPane.showMessageDialog(this, "File: " +
file_output.getAbsolutePath() + separator + file_output_name + ".jar created!");
    } catch (Exception ex) {
        System.out.println(ex.getMessage());
    }

}
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {

```

---

```

        javax.swing.UIManager.setLookAndFeel(info.getClassName());
        break;
    }
}
} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(frmGenerator.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(frmGenerator.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(frmGenerator.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(frmGenerator.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);
    }
}
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new frmGenerator().setVisible(true);
    }
});
}
// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton btn_generate;
private javax.swing.JButton btn_pilih_input;
private javax.swing.JButton btn_pilih_output;
private javax.swing.JFileChooser fc;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JTextField txt_input;
private javax.swing.JTextField txt_output;
// End of variables declaration//GEN-END:variables
}

```

---

## Lampiran II

### Source code pdfReader

```
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.io.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import org.jpedal.PdfDecoder;
import org.jpedal.exception.PdfException;
import org.jpedal.fonts.FontMappings;

/**
 *
 * @author budi
 */
public class pdfReader extends javax.swing.JFrame {

    private Thread timer = null;
    private int jam = 0, menit = 0, detik = 0;
    private static Db_timer db = null;
    int counter = 0;
    String waktu;
    static String filename = null;
    static String separator = "/";

    // static String file = "/pdf/file.pdf";
    static File APP = new File("PDFReader.jar");

    /**
     * PDF
     */

    /**the actual JPanel/decoder object*/
    private PdfDecoder pdfDecoder;

    /**name of current PDF file*/
    private String currentFile = null;

    /**current page number (first page is 1)*/
```

---

```

        //if file has a null password it will have been decoded and isFileViewable
will return true
        while(!pdfDecoder.isFileViewable()) {

            /** popup window if password needed */
            String password = JOptionPane.showInputDialog(this,"Please enter
password");

            /** try and reopen with new password */
            if (password != null) {
                try {
                    pdfDecoder.setEncryptionPassword(password);
                } catch (PdfException e) {
                    e.printStackTrace(); //To change body of catch statement use File |
Settings | File Templates.
                }
            }
        }
        return true;
    }
    return true;
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        viewer = new javax.swing.JScrollPane();
        toolbar_utama = new javax.swing.JToolBar();
        btn_first = new javax.swing.JButton();
        btn_prev = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        pageCounter2 = new javax.swing.JTextField();
        pageCounter3 = new javax.swing.JLabel();
        btn_next = new javax.swing.JButton();
        btn_last = new javax.swing.JButton();
        txt_timer = new javax.swing.JTextField();

```

---

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Pembatas Waktu Akses E-book");
setName("frmReader"); // NOI18N
setPreferredSize(new java.awt.Dimension(800, 600));
setResizable(false);

viewer.setBorder(javax.swing.BorderFactory.createEtchedBorder());

toolbar_utama.setRollover(true);

btn_first.setText("<<");
btn_first.setBorder(javax.swing.BorderFactory.createCompoundBorder());
btn_first.setBorderPainted(false);
btn_first.setFocusable(false);
btn_first.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
btn_first.setPreferredSize(new java.awt.Dimension(80, 27));
btn_first.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
btn_first.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_firstActionPerformed(evt);
    }
});
toolbar_utama.add(btn_first);

btn_prev.setText("<");
btn_prev.setFocusable(false);
btn_prev.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
btn_prev.setPreferredSize(new java.awt.Dimension(80, 27));
btn_prev.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
btn_prev.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_prevActionPerformed(evt);
    }
});
toolbar_utama.add(btn_prev);

jLabel1.setText("Page");
toolbar_utama.add(jLabel1);

pageCounter2.setHorizontalAlignment(javax.swing.JTextField.CENTER);
pageCounter2.setPreferredSize(new java.awt.Dimension(50, 27));
toolbar_utama.add(pageCounter2);

pageCounter3.setText("of");
toolbar_utama.add(pageCounter3);
```

---

```

btn_next.setText(">");
btn_next.setFocusable(false);
btn_next.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
btn_next.setPreferredSize(new java.awt.Dimension(80, 27));
btn_next.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
btn_next.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_nextActionPerformed(evt);
    }
});
toolbar_utama.add(btn_next);

btn_last.setText(">>");
btn_last.setFocusable(false);
btn_last.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
btn_last.setPreferredSize(new java.awt.Dimension(80, 27));
btn_last.setVerticalTextPosition(javax.swing.SwingConstants.BOTTOM);
btn_last.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_lastActionPerformed(evt);
    }
});
toolbar_utama.add(btn_last);

txt_timer.setEditable(false);
txt_timer.setText("00:00:00");
toolbar_utama.add(txt_timer);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(toolbar_utama, javax.swing.GroupLayout.DEFAULT_SIZE,
545, Short.MAX_VALUE)
        .addComponent(viewer)
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
            .addComponent(toolbar_utama,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

---

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(viewer, javax.swing.GroupLayout.DEFAULT_SIZE, 281,
Short.MAX_VALUE))
    );

    getAccessibleContext().setAccessibleDescription("");

    pack();
} // </editor-fold> //GEN-END:initComponents

private void btn_nextActionPerformed(java.awt.event.ActionEvent evt)
//GEN-FIRST:event_btn_nextActionPerformed
    // TODO add your handling code here:
    if(currentFile!=null && currentPage<pdfDecoder.getPageCount()){
        currentPage += 1;
        try {
            pdfDecoder.decodePage(currentPage);
            pdfDecoder.invalidate();
            repaint();
        } catch (Exception e1) {
            System.err.println("forward 1 page");
            e1.printStackTrace();
        }
    }

//    set page number display
    pageCounter2.setText(String.valueOf(currentPage));
}
} //GEN-LAST:event_btn_nextActionPerformed
private void btn_lastActionPerformed(java.awt.event.ActionEvent evt)
//GEN-FIRST:event_btn_nextActionPerformed
    // TODO add your handling code here:
    if(currentFile!=null && currentPage<pdfDecoder.getPageCount()){
        currentPage = pdfDecoder.getPageCount();
        try {
            pdfDecoder.decodePage(currentPage);
            pdfDecoder.invalidate();
            repaint();
        } catch (Exception e1) {
            System.err.println("forward to last page");
            e1.printStackTrace();
        }
    }

//    set page number display
    pageCounter2.setText(String.valueOf(currentPage));
}
}

```

---

```

        @Override
        public void run() {
            startTimer();
        }
    };

    timer = new Thread(r, "Timer Thread");
    timer.start();

} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
}

private void startTimer() {
    try {
        if (counter <= 0) {
            JOptionPane.showMessageDialog(this, "Waktu habis, terima kasih telah
membaca buku ini. Aplikasi akan dihapus secara otomatis.");
            viewer.setViewportView(null);
            timer.stop();
        }
        else {
            counter = counter - 1;

            // kalkulasi jam, menit, detik
            jam = (int)Math.floor( counter / 3600 );
            menit = counter % 3600;
            menit = (int)Math.floor( menit / 60);
            detik = counter % 60;

            waktu = (jam < 10) ? "0" + String.valueOf(jam) : String.valueOf(jam);
            waktu += ":";
            waktu += (menit < 10) ? "0" + String.valueOf(menit) :
String.valueOf(menit);
            waktu += ":";
            waktu += (detik < 10) ? "0" + String.valueOf(detik) :
String.valueOf(detik);

            // update label timer
            txt_timer.setText(waktu);

            System.out.println(counter);
            Thread.sleep(1000);
            startTimer();
        }
    }
}

```

---



```

    }

    } catch (Exception ex) {
        System.out.println(ex.getMessage());
    }
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(pdfReader.class.getName()).log(java.util.logging
g.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(pdfReader.class.getName()).log(java.util.logging
g.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(pdfReader.class.getName()).log(java.util.logging
g.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(pdfReader.class.getName()).log(java.util.logging
g.Level.SEVERE, null, ex);
    }
}
//</editor-fold>

```

---

```

// cek nama file db di dalam aplikasi JAR

String line = null;
try {
    InputStreamReader file_timer = new InputStreamReader
(pdfReader.class.getClass().getResourceAsStream("/filename.txt"));
    BufferedReader reader = new BufferedReader(file_timer);

    while ((line = reader.readLine()) != null) {
        filename = line;
    }
} catch (IOException ex) {
    System.out.println(ex.getMessage());
}

// cek file db di folder tmp
File tempDir = new File(System.getProperty("java.io.tmpdir") + separator +
filename + ".db");

// jika file sudah ada, baca file tsb
if (tempDir.exists()) {
    System.out.println("Timer DB ready.");
}
else {
    System.out.println("File: " + tempDir.getAbsolutePath() + " is not exist.");
    System.out.println("Copy Timer DB ke tmp folder");

    try {
        InputStream templateStream =
pdfReader.class.getClass().getResourceAsStream("/db/timer.db");
        OutputStream out = new FileOutputStream(tempDir);

        byte[] buf = new byte[1024];
        int len;
        try {
            while ((len = templateStream.read(buf)) > 0) {
                out.write(buf, 0, len);
            }
            templateStream.close();
            out.close();
            System.out.println("Timer DB copied.");
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }
    }
}

```

---

```

    }
    } catch (FileNotFoundException e) {
        System.out.println(e.getMessage());
    }
}

// koneksi db
db = new Db_timer(tempDir.getAbsolutePath());

try {
    db.konek();
    db.ambil_data_timer();
} catch (ClassNotFoundException ex) {
    Logger.getLogger(pdfReader.class.getName()).log(Level.SEVERE, null, ex);
}

// jika waktu belum habis
if (db.counter > 0) {

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        @Override
        public void run() {
            //
            System.out.println(pdfReader.class.getClass().getResource(file).toString());
            // System.exit(0);

            String absolutePath = null;
            try {

                File tempDir = new File(System.getProperty("java.io.tmpdir"));
                File temporaryFile = new File(tempDir, filename);

                InputStream templateStream =
pdfReader.class.getClass().getResourceAsStream("/pdf/file.pdf");
                OutputStream out = new FileOutputStream(temporaryFile);

                byte[] buf = new byte[1024];
                int len;
                try {
                    while ((len = templateStream.read(buf)) > 0) {
                        out.write(buf, 0, len);
                    }
                    templateStream.close();
                    out.close();
                }
            }
        }
    });
}
}

```

---