

**REFACTORING SIMULASI ROKOK FILTER BERBASIS MULTIMEDIA  
DENGAN PENDEKATAN OOP**

Skripsi



oleh

**LILIANA SARI SETYA WIDJAJA**  
**23070193**

PROGRAM STUDI SISTEM INFORMASI FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
TAHUN 2014

**REFACTORING SIMULASI ROKOK FILTER BERBASIS MULTIMEDIA  
DENGAN PENDEKATAN OOP**

Skripsi



Diajukan kepada Program Studi Sistem Informasi Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana  
Sebagai Salah Satu Syarat dalam Memperoleh Gelar  
Sarjana Komputer

Disusun oleh

**LILIANA SARI SETYA WIDJAJA**  
**23070193**

PROGRAM STUDI SISTEM INFORMASI FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
TAHUN 2014

## PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

### **Refactoring Simulasi Rokok Filter Berbasis Multimedia dengan Pendekatan OOP**

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi dan pencabutan gelar kesarjanaan saya.

Yogyakarta, 7 Oktober 2013



LIANA SARI SETYA WIDAJA  
23070193

**HALAMAN PERSETUJUAN**

Judul Skripsi : Refactoring Simulasi Rokok Filter Berbasis Multimedia  
dengan Pendekatan OOP  
Nama Mahasiswa : LILIANA SARI SETYA WIDJAJA  
N I M : 23070193  
Matakuliah : Skripsi  
Kode : SI4046  
Semester : Genap  
Tahun Akademik : 2012/2013



Telah diperiksa dan disetujui di Yogyakarta,  
pada tanggal 10 Oktober 2013

Dosen Pembimbing I

LILIANA SARI SETYA WIDJAJA, S.Kom., M.T.

Dosen Pembimbing II

ANTONIUS RACHMAT C. SKom., M.Cs

@UKDW

HALAMAN PENGESAHAN

REFACTORIZING SIMULASI ROKOK FILTER BERBASIS MULTIMEDIA  
DENGAN PENDEKATAN OOP


Oleh: LILIANA SARI SETYA WIDJAJA / 23070193

Dipertahankan di depan Dewan Penguji Skripsi  
Program Studi Sistem Informasi Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana - Yogyakarta  
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Komputer  
pada tanggal  
17 Mei 2013

Yogyakarta, 7 Oktober 2013  
Muharraman,

Dewan Penguji:

1. Drs. WIMMIE HANDIWIJODI, M.T.
2. KATON WIJANA, S.Kom., M.T.
3. ANTONIUS RACHMAD, S.Kom., M.Cs.
4. Ir. NJOO HARIANTO KRUMAD, S.T., M.T., M.Sc.

  
Dekan  
  
Drs. WIMMIE HANDIWIJODI, M.T.

Ketua Program Studi  
  
YETI LQSLAN, S.Kom., M.T.



## ABSTRAK

Kesulitan programmer dalam pengembangan program simulasi mesin rokok yang penulis gunakan sebagai bahan penelitian ini adalah kode program yang panjang dan berulang – ulang. Program simulator mesin rokok yang penulis gunakan sebagai bahan penelitian ini dibuat dengan bahasa pemrograman ActionScript 3.0. ActionScript 3.0 merupakan bahasa pemrograman yang digunakan dalam pembangunan game maupun animasi dengan Flash yang telah mendukung konsep *Object Oriented Programming (OOP)*. OOP merupakan sebuah paradigma pemrograman yang berdasarkan objek dan objek – objek tersebut saling berinteraksi untuk menyelesaikan suatu permasalahan. Pada penelitian ini penulis akan menerapkan konsep OOP pada program simulasi mesin rokok filter yang telah dibangun dengan metode *refactoring*, yaitu pengembangan sebuah perangkat lunak dengan menitik beratkan pada proses penulisan program.

Proses penelitian ini dimulai dengan mempelajari tentang konsep *Object Oriented Programming* dan *refactoring*, lalu menganalisa kode program yang buruk, memecah kemudian menentukan kelas dan metode menjadi lebih spesifik. Menambahkan dan menukar metode yang diperlukan untuk efisiensi. Kemudian mengetest program untuk mendapatkan hasil yang sama.

Implementasi proses penelitian ini pada program yang lama berupa pengelompokan gambar dan tombol yang saling berkaitan dalam kelas tersendiri sehingga jumlah *scene* dan *frame* berkurang menjadi 1(satu) *frame* di dalam 1(satu) *scene*. Penambahan kelas *checker* dilakukan untuk menghindari pengulangan baris kode.

Kata Kunci : *Refactoring*, OOP, AS3

## DAFTAR ISI

HALAMAN JUDUL .....	i
PERNYATAAN KEASLIAN SKRIPSI .....	ii
HALAMAN PERSETUJUAN .....	iii
HALAMAN PENGESAHAN.....	iv
UCAPAN TERIMA KASIH.....	v
ABSTRAK .....	vi
DAFTAR ISI .....	vii
DAFTAR GAMBAR .....	ix
<b>BAB 1 PENDAHULUAN</b>	
1.1. Latar Belakang Masalah .....	1
1.2. Rumusan Masalah .....	1
1.3. Batasan Masalah .....	1
1.4. Tujuan Penelitian .....	2
1.5. Metode Penelitian .....	2
1.6. Sistematika Penulisan.....	2
<b>BAB 2 TINJAUAN PUSTAKA</b>	
2.1. Tinjauan Pustaka .....	4
2.2. Landasan Teori .....	4
2.2.1. Object Oriented Programming (OOP).....	4
2.2.1.1. Kelas .....	5
2.2.1.2. Objek .....	6
2.2.1.3. Konstruktor .....	6
2.2.1.4. Enkapsulasi .....	7
2.2.2 Perbedaan Prosedural dan OO Programming .....	8
2.2.3 Refactoring .....	8
2.2.3.1. Merubah Desain Prosedural Menjadi Objek .....	9
2.2.1.2. Ekstrak Metode .....	10



2.2.4 Animasi frame-by-frame.....	10
<b>BAB 3 PERANCANGAN SISTEM</b>	
3.1. Spesifikasi Sistem .....	11
3.1.1 Hardware .....	11
3.1.2 Software .....	11
3.2 Analisis Data.....	11
3.3 Perancangan Sistem .....	12
3.3.1 Flowchart Sistem .....	12
3.3.2 State Diagram .....	16
3.3.3 Class Diagram .....	16
3.4 Perancangan Antarmuka .....	17
<b>BAB 4 PENERAPAN DAN ANALISIS SISTEM</b>	
4.1. Implementasi Perancangan Antarmuka .....	19
4.1.1 Halaman Menu Utama .....	19
4.1.2 Halaman Galeri Video .....	19
4.1.3 Halaman Permainan .....	20
4.2. Implementasi OOP pada Sistem .....	21
4.2.1 Pembuatan Kelas .....	21
4.2.2 Perubahan Kode Program .....	23
<b>BAB 5 KESIMPULAN DAN SARAN</b>	
5.1. Kesimpulan .....	25
5.2. Saran .....	25
DAFTAR PUSTAKA .....	26
<b>LAMPIRAN A : LISTING PROGRAM</b>	
<b>LAMPIRAN B : DOKUMEN – DOKUMEN TERKAIT</b>	

# BAB 1

## PENDAHULUAN

### 1.1. Latar Belakang

Kendala yang dihadapi dalam pengembangan program simulasi mesin rokok yang penulis gunakan sebagai bahan penelitian ini adalah kode program yang panjang dan berulang – ulang karena menggunakan desain prosedural. Program simulasi ini dibuat dengan bahasa pemrograman ActionScript 3.0. Pada dasarnya ActionScript 3.0 merupakan bahasa pemrograman yang digunakan dalam pembangunan game maupun animasi dengan Flash yang mendukung konsep OOP (*Object Oriented Programming*). Keuntungan dari model OOP yaitu selain objek – objeknya dapat digunakan ulang untuk program lain, program yang dibuat menjadi lebih ringkas dan lebih mudah untuk dikembangkan.

Untuk itu dalam penelitian ini, konsep OOP akan diterapkan pada program simulasi ini dengan harapan dapat memberikan kemudahan dalam pengembangan sistem dikemudian hari, mengingat dalam proses produksi rokok filter tidak hanya menggunakan satu jenis mesin. Agar tidak terjadi perbedaan penggunaan antara simulator yang sebelum dan yang sesudah menerapkan konsep OOP, maka digunakanlah metode *refactoring*. Dengan menggunakan metode *refactoring*, perubahan dilakukan pada kode programnya. Perubahan yang dilakukan dapat berupa pemecahan maupun penambahan kelas atau metode.

### 1.2. Rumusan Masalah

Dari latar belakang masalah di atas dapat dirumuskan masalah sebagai berikut :

Bagaimana cara mengimplementasikan konsep OOP pada program simulator mesin rokok ini dengan metode *refactoring*?

### **1.3. Batasan Masalah**

Batasan permasalahan dalam penelitian ini adalah :

1. Level dari simulator yang akan digunakan sebagai bahan penelitian adalah level “Normal” yaitu pengguna langsung pada tahap prosedur menghidupkan mesin.

### **1.4. Tujuan Penelitian**

Mengimplementasikan konsep OOP pada sistem yang dibangun agar mendapatkan manfaat – manfaat dari konsep OOP adalah tujuan dari penelitian ini.

### **1.5. Metode Penelitian**

Metode penelitian yang penulis gunakan untuk penyusunan penelitian ini adalah :

1. Studi pustaka dilakukan untuk mendukung penelitian ini. Penulis mengumpulkan bahan pembahasan mengenai desain dan implementasi OOP menggunakan An3 dan juga serta jurnal untuk teori pendukung yang lain.
2. Interview dan survey lapangan telah dilakukan oleh penulis pada saat kerja praktik tahun 2011 lalu sehingga data – data yang diperlukan untuk pembuatan penelitian ini sudah penulis dapatkan.
3. Pembuatan dilakukan dengan cara mengganti kode program dari program yang lama dengan konsep OOP. Penambahan kelas dan metode akan dilakukan sesuai dengan metode *refactoring*.

### **1.6. Sistematika Penulisan**

Penulisan skripsi ini terbagi dalam lima bagian, yang pertama adalah bab pendahuluan. Bab ini membahas tentang latar belakang penelitian, perumusan, batasan masalah, tujuan penelitian, serta metode dan sistematika penulisan skripsi ini. Kemudian teori pendukung yang digunakan beserta beberapa contohnya akan dibahas dalam bab tinjauan pustaka.

Dalam bab perancangan sistem akan dibahas mengenai gambaran proses dan perancangan antarmuka sistem yang akan dibangun. Sistem yang telah selesai dibangun dan diimplementasikan akan dibahas pada bab implementasi sistem yang kemudian akan ditarik kesimpulan tentang kelemahan dan kelebihan sistem, serta usulan untuk pengembangan sistem lebih lanjut yang akan dirangkum dalam bab kesimpulan dan saran.

@UKDW

## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Cara mengimplementasikan konsep OOP pada program simulator mesin rokok ini, yaitu :

1. Menganalisa kode program yang buruk, seperti duplikasi kode.
2. Memecah kelas yang lingkupnya terlalu umum.
3. Memecah metode yang panjang.
4. Menambahkan kelas maupun metode untuk efisiensi.
5. Mengetes untuk memastikan program menghasilkan hasil yang sama.

#### 5.2. Saran

1. Perlu dilakukan penyempurnaan baik algoritma maupun *library* yang dibuat oleh penulis berkaitan dengan pengetahuan penulis yang terbatas tentang *tools* pada Flash sehingga tidak bisa dimanfaatkan dengan maksimal.
2. Penyempurnaan tampilan aplikasi terutama tombol pada panel – panel agar lebih detail lagi.

## DAFTAR PUSTAKA

- Aditomo, R C. (2010). Penerapan Momentum Tumbukan Antar Objek Pada Game Animasi Billiard, *Skripsi*, Teknik Informatika, Universitas Kristen Duta Wacana, Yogyakarta.
- Billah, S. A., Hariadi, S. (2010). Rancang Bangun Aplikasi Sistem Sidang Skripsi Menggunakan Metode Object Oriented. *MATIC*, 4(1), 32 – 45.
- Braunstein, Roger. (2010). *ActionScript3.0 Bible*. John Wiley & Sons.
- Fowler, Martin. (2008). *Refactoring: Improving the Design of Existing Code*. Addison Wesley Longman, Inc.
- Kepple, Dru, (n.d). AS3 101: OOP Introduction – Basix. Retrieved from <http://active.tutsplus.com/tutorials/actionscript/as3-101-oop-introduction-basix/>.
- Lott, Joey. (2007). *Action Script 3.0 Cookbook*. California:O'Reilly Media,Inc.
- Mens, Tom. (2004). A Survey of Software Refactoring. *A formal foundation for software refactoring*. Project G.0452.03. FWO.
- Milbourne, P., Kaplan, C., & Oliver, M. (2009). *The Essential Guide To Flash With Action Script*. Apress.
- Nugroho, Adi. (2004). *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*. Bandung : Informatika.

Peters, Keith. (2007). *Foundation ActionScript 3.0 Animation: Making things move!*.  
Apress.

Rosenzweig, Gary. (2008). *ActionScript 3.0 Game Programming University*. Que  
Indianapolis USA.

Soewignjo, Santosa. (2005). *Let's Animate*, Jakarta: PT. TriExs Trimacindo.

@UKDWN