

SKRIPSI

PERANGKAT LUNAK KONVERSI KOMENTAR JAVA KE
 \LaTeX



Adli Fariz Bonaputra

NPM: 2012730082

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2018

UNDERGRADUATE THESIS

JAVA COMMENT TO L^AT_EX SOFTWARE CONVERSION



Adli Fariz Bonaputra

NPM: 2012730082

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2018

LEMBAR PENGESAHAN



PERANGKAT LUNAK KONVERSI KOMENTAR JAVA KE L^AT_EX

Adli Fariz Bonaputra

NPM: 2012730082

Bandung, 18 Desember 2018

Menyetujui,

Pembimbing

Dr. Veronica Sri Moertini

Ketua Tim Penguji

Dott. Thomas Anung Basuki

Anggota Tim Penguji

Elisati Hulu, M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng



PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PERANGKAT LUNAK KONVERSI KOMENTAR JAVA KE \LaTeX

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 18 Desember 2018



Adli Fariz Bonaputra
NPM: 2012730082

ABSTRAK

Tujuan skripsi mahasiswa Program Studi Teknik Informatika Fakultas Teknologi Informasi dan Sains (FTIS) Universitas Katolik Parahyangan (Unpar) adalah membuat perangkat lunak. Perangkat lunak yang dibuat umumnya menggunakan bahasa pemrograman *Java*. Bahasa pemrograman *Java* memiliki komentar yang berguna sebagai informasi dari kelas, *interface*, *method* dan juga *custom tag* yang dibuat, sehingga informasi tersebut dapat digunakan sebagai penjelasan perangkat lunak pada dokumentasi perangkat lunak. Untuk mendokumentasikan perangkat lunak yang dibuat, seluruh mahasiswa diwajibkan untuk menggunakan \LaTeX dalam pembuatan sebuah dokumentasi skripsi. \LaTeX merupakan bahasa *markup* untuk menyusun sebuah dokumentasi. Bahasa *markup* adalah sebuah bahasa yang menganotasikan dokumen dengan cara menambahkan sintaks tertentu agar dapat dibedakan. \LaTeX dapat menghasilkan dokumen yang terstruktur sesuai dengan yang ditulis oleh penulis. Umumnya bentuk akhir dari dokumen yang dibuat oleh \LaTeX biasanya berupa sebuah *file PDF*.

Pada salah satu bab dokumentasi skripsi, terdapat penjelasan dari setiap kelas pada perangkat lunak yang dibuat. Penjelasan tersebut sebenarnya dapat diambil dari *Javadoc* yang telah dibuat pada kelas *Java*, namun saat ini berdasarkan pengamatan tersebut masih diketik secara manual dari *Javadoc* ke dalam format \LaTeX , sehingga membutuhkan lebih banyak waktu untuk mendokumentasikan setiap kelas pada perangkat lunak yang dibuat.

Oleh karena itu, perlu dikembangkan sebuah perangkat lunak yang dapat mengekstraksi informasi pada *Javadoc* ke format \LaTeX secara otomatis. Perangkat lunak ini mengimplementasikan sebuah *Application Programming Interface* (API) yang digunakan untuk mengambil informasi berupa nama kelas, *interface*, *method* dan juga *custom tag* yang terdapat pada sebuah *file Java*.

Kata-kata kunci: Skripsi, *Javadoc*, *Doclet*, \LaTeX

ABSTRACT

The purpose of the undergraduate thesis is the Information Technology and Science Faculty (FTIS) Informatics Engineering Study Program at Parahyangan Catholic University (Unpar) is to make software. Software made generally uses the programming language *Java*. The programming language it Java has useful comments as information from the class, *interface*, *method* and also *custom tag* created, so that the information can be used as a software explanation in the device documentation soft. To document the software made, all students are required to use \LaTeX in making a thesis documentation. \LaTeX is the language of *markup* to compile a documentation. Language *markup* is a language that annotates documents by adding certain syntax so that they can be distinguished. \LaTeX can produce structured documents as written by the author. Generally the final form of a document created by \LaTeX is usually a *file* PDF.

In one of the thesis documentation chapters, there is an explanation of each class in the software created. This explanation can actually be taken from *Javadoc* which was created in the *Java* class, but currently based on these observations it is still typed manually from *Javadoc* into the format \LaTeX , so that it takes more time to document each class in the software created.

Therefore, it is necessary to develop a software that can extract information on *Javadoc* to \LaTeX format automatically. This software implements a *Application Programming Interface* (API) which is used to retrieve information in the form of a class name, *interface*, *method* and also *custom tag* contained in a *Java file*.

Keywords: Thesis, Javadoc, Doclet, \LaTeX

*Dipersembahkan untuk Keluarga, Karmila Puspitasari dan Orang
terdekat*

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa atas seluruh berkat yang diberikan kepada penulis sehingga dapat menyelesaikan tugas akhir dengan judul Perangkat Lunak Konversi Komentar Java ke \LaTeX dengan baik dan tepat waktu. Penulis juga berterima kasih kepada pihak-pihak yang telah memberikan dukungan dan bantuan kepada penulis dalam menyelesaikan tugas akhir ini, yaitu:

1. Orang tua dan keluarga yang selalu memberikan dukungan kepada penulis.
2. Bapak Pascal Alfadian sebagai dosen pembimbing yang telah membimbing penulis hingga dapat menyelesaikan tugas akhir ini.
3. Bapak Thomas Anung Basuki dan Bapak Elisati Hulu sebagai dosen penguji yang telah membantu dalam menguji tugas akhir ini.
4. Karmila Puspitasari yang tidak pernah berhenti untuk selalu menemani, memberikan dukungan, memberikan semangat kepada penulis hingga dapat menyelesaikan tugas akhir dan lulus.
5. Andre dan Momon yang sampai penulis tidak tahu harus berkata-kata apa. Terima kasih banyak.
6. Teman-teman Zero Hour Coffee dan Bahagia Kopi yang selalu memberikan semangat dan dukungan.
7. Teman-teman kontak 7B yang sudah memberikan tempatnya sebagai tempat penulis mencari inspirasi.
8. Teman-teman Teknik Informatika UNPAR angkatan 2012 yang telah berbagi ilmu dan selalu memberikan semangat kepada penulis.
9. Pihak-pihak lain yang belum disebutkan, yang berperan dalam penyelesaian tugas akhir ini.

Akhir kata, penulis berharap agar tugas akhir ini dapat bermanfaat bagi pembaca yang hendak melakukan penelitian dan pengembangan yang terkait dengan tugas akhir ini.

Bandung, Desember 2018

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	2
2 LANDASAN TEORI	5
2.1 Javadoc	5
2.1.1 <i>Processing of source files</i>	5
2.1.2 Terminologi	6
2.1.3 <i>Source Files</i>	6
2.1.4 <i>Generated Files</i>	9
2.2 Doclet	11
2.2.1 <i>Interface-interface</i> pada Doclet	11
2.2.2 Penggunaan Doclet	12
2.3 L ^A T _E X	13
2.4 <i>Syntax Java</i>	14
3 ANALISIS	17
3.1 Analisis Kebutuhan Perangkat Lunak	17
3.2 Analisis Program Sejenis TeXDoclet	20
4 PERANCANGAN	21
4.1 Rancangan Kelas Lengkap	21
4.2 Sequence Diagram	24
4.3 Rancangan Antarmuka	25
5 IMPLEMENTASI DAN PENGUJIAN	27
5.1 Implementasi Perangkat Lunak	27
5.1.1 Lingkungan Implementasi	28
5.2 Pengujian Perangkat Lunak	28
5.2.1 Pengujian Fungsional	28
5.2.2 Pengujian Eksperimental	29
6 KESIMPULAN DAN SARAN	31
6.1 Kesimpulan	31

6.2	Saran	31
DAFTAR REFERENSI		33
A	PENGUJIAN TERHADAP KODE PROGRAM SEDERHANA	35
A.1	Kode Program	35
A.2	Hasil Latex	36
A.3	Hasil PDF	38
B	PENGUJIAN TERHADAP KODE PROGRAM PERANGKAT LUNAK	41
B.1	Kode Program	41
B.2	Hasil Latex	45
B.3	Hasil PDF	47
C	PENGUJIAN TERHADAP KODE PROGRAM YANG MEMILIKI KOMENTAR KURANG LENGKAP	53
C.1	Kode Program	53
C.2	Hasil Latex	53
C.3	Hasil PDF	53
D	PENGUJIAN TERHADAP KODE PROGRAM SIAMODELS	55
D.1	Kode Program	55
D.2	Hasil Latex	94
D.3	Hasil PDF	121
E	HASIL PDF TEXDOCLET	163

DAFTAR GAMBAR

2.1	Hasil dokumentasi pada kelas atau <i>interface</i>	6
2.2	Hasil dokumentasi pada dari <i>package</i>	8
2.3	Hasil dokumentasi pada dari <i>overview</i>	9
2.4	Frame pada hasil dokumentasi	10
2.5	Contoh deklarasi kelas pada <i>Java</i>	15
2.6	Contoh deklarasi atribut pada <i>Java</i>	15
2.7	Contoh deklarasi <i>method</i> pada <i>Java</i>	16
4.1	Kelas Diagram	21
A.1	operasimatematika.pdf	40
B.1	javadoctolatex.pdf	51
D.1	javadoctolatex.pdf	162
E.1	TeXDoclet_report.pdf	188

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dalam sebuah penelitian, membuat dokumentasi perlu dilakukan. Dokumentasi yang dibuat bisa dalam bentuk *hardcopy* atau *softcopy*, tergantung kebutuhannya. Dokumentasi adalah kegiatan untuk mencatat suatu peristiwa atau aktivitas yang dianggap berharga atau penting. Dokumentasi yang sudah dibuat dapat menjadi referensi untuk memandu dalam melakukan sebuah aktivitas.

Dalam bidang teknologi informasi, dokumentasi kode program *Java* umumnya ditulis dalam komentar disetiap *file java*. Komentar tersebut berisikan informasi dari sebuah kelas, *interface*, *method* dan *custom tag*. *Javadoc* adalah sebuah *tools* yang dimiliki oleh *Java* yang berguna untuk mengambil informasi tersebut dan menjadikannya sebagai dokumentasi. Umumnya mendokumentasikan sebuah kelas, *interface*, *method* dan *custom tag*. Oleh karena itu, *Javadoc* sangatlah penting karena dapat memuat berbagai informasi dari sebuah *file Java*. Informasi tersebut dapat menjelaskan sebuah kelas yang dibuat dalam sebuah dokumentasi perangkat lunak.

Tujuan skripsi mahasiswa Program Studi Teknik Informatika Fakultas Teknologi Informasi dan Sains (FTIS) Universitas Katolik Parahyangan (Unpar) adalah membuat perangkat lunak. Perangkat lunak yang dibuat umumnya menggunakan bahasa pemrograman *Java*. Bahasa pemrograman *Java* memiliki komentar yang berguna sebagai informasi dari kelas, *interface*, *method* dan juga *custom tag* yang dibuat, sehingga informasi tersebut dapat digunakan sebagai penjelasan perangkat lunak pada dokumentasi perangkat lunak. Untuk mendokumentasikan perangkat lunak yang dibuat, seluruh mahasiswa diwajibkan untuk menggunakan \LaTeX dalam pembuatan sebuah dokumentasi skripsi. \LaTeX merupakan bahasa *markup* untuk menyusun sebuah dokumentasi. Bahasa *markup* adalah sebuah bahasa yang menganotasikan dokumen dengan cara menambahkan sintaks tertentu agar dapat dibedakan. \LaTeX dapat menghasilkan dokumen yang terstruktur sesuai dengan yang ditulis oleh penulis. Umumnya bentuk akhir dari dokumen yang dibuat oleh \LaTeX biasanya berupa sebuah *file PDF*.

Pada salah satu bab dokumentasi skripsi, terdapat penjelasan dari setiap kelas pada perangkat lunak yang dibuat. Penjelasan tersebut sebenarnya dapat diambil dari *Javadoc* yang telah dibuat pada kelas *Java*, namun saat ini berdasarkan pengamatan tersebut masih diketik secara manual dari *Javadoc* ke dalam format \LaTeX , sehingga membutuhkan lebih banyak waktu untuk mendokumentasikan setiap kelas pada perangkat lunak yang dibuat.

Oleh karena itu, perlu dikembangkan sebuah perangkat lunak yang dapat mengambil informasi pada *Javadoc* ke format \LaTeX secara otomatis. Perangkat lunak ini mengimplementasikan sebuah *Application Programming Interface* (API) yang digunakan untuk mengambil informasi berupa nama kelas, *interface*, *method* dan juga *custom tag* yang terdapat pada sebuah *file Java*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disebutkan di atas, maka dihasilkan sebuah poin yang menjadi rumusan masalah dari masalah ini. Rumusan masalah yang akan dibangun antara lain sebagai berikut:

- Bagaimana masukan yang sesuai dengan perangkat lunak?
- Bagaimana keluaran yang dihasilkan oleh perangkat lunak?
- Bagaimana membuat perangkat lunak yang dapat mengambil informasi dari sekumpulan *file Java* yang terdapat pada sebuah *package* dan mengonversikan format *Javadoc* ke dalam format \LaTeX secara otomatis?

1.3 Tujuan

Adapun tujuan yang ingin dicapai dari penelitian ini adalah menjawab rumusan masalah di atas, yaitu:

- Menganalisa struktur kode program *java* yang baik
- Menganalisa struktur \LaTeX yang akan sebagai keluaran perangkat lunak
- Membangun perangkat lunak yang dapat mengambil informasi dari sekumpulan *file Java* yang terdapat pada sebuah *package* dan mengonversikan format *Javadoc* ke dalam format \LaTeX secara otomatis.

1.4 Batasan Masalah

Agar pembahasan masalah tidak terlalu luas, masalah yang akan dikaji di dalam penelitian ini memiliki batasan, yaitu:

1. Perangkat lunak memanfaatkan *library Doclet* yang sudah disediakan.
2. Perangkat lunak menghasilkan *file \LaTeX* tanpa mengonversikan menjadi *file PDF*.

1.5 Metodologi

Untuk menyelesaikan penelitian ini disusunlah tahap-tahap tugas yang perlu dilakukan. Tahap-tahap yang dimaksud adalah sebagai berikut:

1. Melakukan studi literatur untuk mengetahui *syntax* yang terdapat pada \LaTeX dan mengetahui apa saja isi dari dokumentasi *Javadoc Doclet API*.
2. Melakukan survei terhadap format penulisan pada suatu bab pada skripsi yang berisi tentang dokumentasi perangkat lunak yang dibuat. Membutuhkan minimal 3 dokumen skripsi sebagai panduan format penulisan.
3. Mengimplementasikan langkah-langkah untuk mengonversi *Javadoc* ke format \LaTeX .
4. Melakukan pengujian terhadap perangkat lunak yang telah diimplementasi.
5. Menarik kesimpulan berdasarkan hasil pengujian.

1.6 Sistematika Pembahasan

1. Bab 1 Pendahuluan
Bab ini akan membahas mengenai latar belakang, rumusan masalah, tujuan, batas masalah, metodologi penelitian dan sistematika penulisan.

2. Bab 2 Landasan Teori

Bab ini akan membahas mengenai pengertian *Javadoc*, Doclet dan \LaTeX .

3. Bab 3 Analisis

Bab ini akan membahas mengenai analisis struktur \LaTeX dan analisis program sejenis TeXDoclet.

4. Bab 4 Perancangan

Bab ini akan membahas mengenai tahap-tahap perancangan dan penjelasan perangkat lunak.

5. Bab 5 Implementasi dan Pengujian

Bab ini akan membahas mengenai implementasi kode program dan pengujian perangkat lunak.

6. Bab 6 Kesimpulan dan Saran

Bab ini akan membahas mengenai kesimpulan dari penelitian yang telah dilakukan dan saran-saran untuk pengembangan lebih lanjut dari penelitian ini.