

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam melakukan pendekatan dengan OOP *developer* memerlukan UML (*Unified Modelling Language*) untuk menggambarkan model aplikasi. UML akan memudahkan *developer* ketika melanjutkan pada tahap penulisan program karena gambaran luas aplikasi sudah ada, sehingga *developer* dapat melanjutkan ke bagian detail.

Ketika *developer* menggunakan UML, *developer* memerlukan waktu yang lebih lama untuk melakukan pekerjaannya. Karena *developer* harus memodelkan gambaran aplikasi melalui UML lalu kemudian menerjemahkan UML ke dalam bentuk tulisan program (*coding*). Sebuah aplikasi “UML Java Code Generator” dapat membantu *developer* dalam proses penerjemahan UML ke dalam bentuk *coding*.

Tahap Perancangan pada aplikasi merupakan tahap penting yang menentukan kualitas suatu aplikasi. Untuk merancang aplikasi dengan baik dan terstruktur sehingga rancangan aplikasi berkualitas dan dapat mudah dimengerti, maka penulis akan menerapkan *design pattern*. *Design pattern* adalah solusi umum dalam menangani masalah perancangan aplikasi. *Design pattern* dapat meningkatkan skalabilitas dan maintainabilitas dari aplikasi. Dalam merancang aplikasi ini, penulis akan menerapkan *composite pattern* sebagai desain struktural dan *command pattern* sebagai desain behavioral.

Harapan dari penelitian ini dapat membuat aplikasi yang terstruktur dengan baik sehingga skalabilitas dan maintainabilitas dapat meningkat. Dan memberi kontribusi dan menggali potensi dalam pengembangan topik *design pattern*.

1.2. Perumusan Masalah

Berdasarkan latar belakang masalah yang telah dipaparkan sebelumnya, maka perumusan masalah pada penelitian ini sebagai berikut:

- a. Bagaimana memilih *design pattern* yang cocok untuk aplikasi?
- b. Bagaimana menerapkan *design pattern* pada aplikasi “UML Java Code Generator”?

1.3. Tujuan dan Manfaat

Berdasarkan perumusan masalah yang telah dipaparkan sebelumnya, maka dibuatlah tujuan dan manfaat dari penelitian ini.

1.3.1. Tujuan

Tujuan yang ingin dicapai dalam Tugas Akhir ini adalah sebagai berikut:

- a. Memilih *design pattern* yang cocok untuk aplikasi “UML Java Code Generator”.
- b. Merancang dan membangun aplikasi dengan menerapkan *design pattern*.

1.3.2. Manfaat

Adapun manfaat yang diharapkan dari Tugas Akhir ini adalah sebagai berikut:

- a. Aplikasi ini dapat dijadikan sebagai aplikasi *designer* UML untuk memfasilitasi *developer* dalam mendesain aplikasi.
- b. Aplikasi ini dapat dijadikan sebagai aplikasi generator untuk memudahkan user dalam melakukan tahap *coding*.
- c. Meningkatkan dan mengembangkan pengetahuan bagi penulis dan pembaca tentang *design pattern*.

1.4. Batasan Masalah

Dalam penelitian ini untuk membatasi permasalahan agar tetap terarah dan tidak menyimpang dari apa yang sudah direncanakan sebelumnya. Penulis membuat batasan-batasan masalah sebagai berikut:

- a. Pembangunan aplikasi “UML Java Code Generator” ini menggunakan bahasa pemrograman java.
- b. Aplikasi hanya dapat *men-generate* UML ke dalam pemrograman java.
- c. Fungsionalitas dalam aplikasi adalah *open, create, read, edit, delete, save, dan generate*.
- d. *Class* dan *method* yang *di-generate* hanya *header* tanpa disertai isi.
- e. *Design pattern* yang diterapkan adalah *command pattern* dan *composite pattern*.
- f. Relasi antara *class* dan *interface* adalah “has-a” dan “is-a”.

1.5. Metodologi Penelitian

Perencanaan awal dalam pengerjaan Tugas Akhir diperlukan agar dapat diselesaikan sesuai ketentuan jadwal kegiatan pada kegiatan proposal. Untuk itu perencanaan awal digambarkan pada Gambar 1.1.



Gambar 1.1 *Flowchart* Metodologi Penelitian

Adapun penjelasan dari Gambar 1.1 mengenai *flowchart* metodologi yang digunakan dalam penyusunan Tugas Akhir ini yaitu:

a. Studi Literatur

Pada tahap ini dilakukan studi literatur mengenai *design pattern* dan pembuatan aplikasi menggunakan bahasa pemrograman java. Studi literatur yang digunakan diantaranya jurnal ilmiah, artikel, buku, dan sumber lain baik dalam bentuk *hardcopy* maupun *online* yang terkait dengan judul skripsi.

b. Perancangan Sistem

Perancangan sistem dilakukan dengan membuat rancangan UI (*User Interface*), rancangan *usecase*, rancangan kelas, dan rancangan *design pattern* yang diterapkan. Rancangan UI yang dilakukan meliputi tampilan utama dimana diagram UML yang dibangun digambarkan pada tampilan ini, tampilan *Create Class* untuk membuat *class*, tampilan *Create Interface* untuk membuat *interface*, tampilan *Create Relation* untuk membuat relasi antar *class* atau *interface*, tampilan *Create*

Variable dan *Constants* untuk membuat *variable* dan *constant*, dan tampilan *Create Method* untuk membuat *method*. Rancangan *usecase* menggambarkan hubungan *user* dengan menu-menu pada aplikasi yang dibuat. Rancangan kelas berupa diagram UML kelas dari aplikasi. Rancangan *design pattern* berupa diagram UML kelas berdasarkan *design pattern* yang diterapkan.

c. Implementasi Sistem

Pada tahap ini dilakukan pembuatan aplikasi serta mengimplementasikan rancangan-rancangan yang sudah dibuat pada tahap sebelumnya.

d. Uji Coba dan Evaluasi

Pada tahap ini dilakukan pengujian untuk penyempurnaan aplikasi. Apabila terjadi kegagalan dan *error*, maka dilakukan perbaikan sampai tidak ada kegagalan dan *error*. Pada tahap ini juga dilakukan evaluasi dari hasil yang didapat dari tahap implementasi sistem.

e. Penyusunan Laporan

Tahap terakhir adalah penyusunan laporan yaitu menyusun laporan akhir yang memuat proses dokumentasi tahapan awal sampai dengan kesimpulan akhir pengerjaan aplikasi ini.

1.6. Sistematika Penulisan

Sistematika penulisan yang digunakan untuk menyelesaikan Tugas Akhir ini, antara lain:

BAB I PENDAHULUAN

Pada bab ini terdiri dari latar belakang yang menjadi alasan penelitian tugas akhir, perumusan masalah dari latar belakang tersebut, batasan masalah agar penyelesaian tugas akhir tetap terarah dan tidak menyimpang, tujuan dan manfaat dari tugas akhir, metodologi penelitian, serta sistematika penulisan.

BAB II STUDI PUSTAKA

Pada bab ini terdiri dari tinjauan pustaka dan landasan teori. Tinjauan pustaka juga memaparkan hasil penelitian sebelumnya sebagai penguat pengerjaan penelitian Tugas Akhir ini. Sumber pustaka dan teori yang diambil untuk penelitian ini maksimal lima tahun.

BAB III PERANCANGAN SISTEM

Pada bab ini menjelaskan rancangan sistem yang diusulkan untuk aplikasi “*UML Java Code Generator*” mulai dari rancangan kelas, rancangan penerapan *design pattern*, rancangan *usecase* hubungan antar *user* dengan aplikasi, dan rancangan UI yang representasikan dengan *layout*.

BAB IV IMPLEMENTASI DAN ANALISIS

Pada bab ini menjelaskan implementasi dari rancangan-rancangan yang dijelaskan pada bab sebelumnya. Pada bab ini juga menjelaskan serangkaian uji coba yang dilakukan. Mulai dari uji coba yang dilakukan penulis sendiri hingga uji coba yang dilakukan oleh user lain melalui uji verifikasi dan kuisisioner.

BAB V KESIMPULAN DAN SARAN

Pada bab ini, penulis menjelaskan kesimpulan dari hasil uji coba penelitian dan berisi saran yang dapat dijadikan pertimbangan untuk penelitian selanjutnya.

