

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

Komputer telah banyak membantu dan menunjang kehidupan manusia. Perangkat lunak atau aplikasi merupakan bagian yang tak terpisahkan dari sistem komputer. Berbagai aplikasi yang lebih besar dan kompleks telah banyak dikembangkan untuk memudahkan manusia dalam beraktifitas.

Untuk mengembangkan sebuah aplikasi dibutuhkan bahasa pemrograman. Salah satu bahasa pemrograman yang banyak digunakan adalah Java. Bahasa pemrograman ini memiliki banyak keunggulan dibanding bahasa pemrograman lain. Keunggulan Java diantaranya dapat digunakan untuk mengembangkan aplikasi desktop, web dan mobile. Selain itu, aplikasi yang dihasilkan dapat berjalan di semua sistem operasi atau biasa disebut *cross-platform*.

Dalam pengembangan sebuah aplikasi Java, sering kali terdapat kode yang harus ditulis secara berulang-ulang dan memiliki karakteristik yang sama. Misal pada saat penulisan kode POJO (*Plain Old Java Object*) atau *Java Bean* yang biasanya merupakan *mapping* dari struktur data. Penulisan kode yang dilakukan secara berulang-ulang tentu cukup memakan waktu dan kurang efektif. Untuk mengantisipasi hal tersebut biasanya programmer memanfaatkan generator kode untuk mempermudah proses pengkodeannya.

Generator kode yang penulis maksud adalah aplikasi yang mampu menulis kode program aplikasi dengan hanya memberikan data-data *logic* yang dibutuhkan kemudian aplikasi tersebut akan mengkodekan *logic-logic* tersebut

secara otomatis. Generator kode telah banyak dikembangkan untuk membantu pembuatan aplikasi baik yang berjalan pada *platform* web, desktop ataupun mobile . Tujuannya adalah untuk menghemat waktu dan biaya pengembangan.

Selain menggunakan generator kode, programmer juga menggunakan framework atau library untuk membantu proses pengkodean baik *front-end* maupun *back-end*. Framework atau library yang dapat membantu proses pengkodean adalah TiwulFX dan Hibernate.

TiwulFX adalah *library* yang memiliki komponen-komponen kustom dari JavaFX . JavaFX sendiri adalah sebuah *tool-kit* GUI (*Graphical User Interface*) terbaru dari Java untuk mengembangkan aplikasi yang kaya konten. JavaFX dirancang untuk menyediakan bagi para *developer* Java sebuah *platform* yang baru, ringan, dengan performa yang tinggi. Tujuannya adalah untuk menggantikan *swing* untuk membangun aplikasi desktop.

Salah satu *framework* yang dapat membantu dalam penulisan kode *back-end* adalah Hibernate ORM. ORM (*Object Relational Mapping*) merupakan sebuah teknik pemrograman yang menjembatani perbedaan tipe data pada konsep pemrograman yang berorientasi objek dengan konsep RDBMS. Hibernate telah memiliki generator kode, namun hanya menghasilkan kode *mapping* object dari tabel database. Kode tersebut belum mampu untuk menjalankan CRUD (*Create Retrieve Update Delete*). Jadi, programmer masih perlu menuliskan kode-logic untuk melakukan CRUD yang meliputi kode *front-end* dan *back-end* untuk masing-masing POJO secara manual. Pada dasarnya kode-logic CRUD memiliki karakteristik yang hampir sama. Jika dalam sebuah

aplikasi terdapat banyak object tabel, proses tersebut tentulah memakan waktu dan kurang efektif.

Berdasarkan latar belakang tersebut maka judul yang digunakan dalam penyusunan skripsi ini adalah, **“Generator Kode Program Java Berbasis Desktop Menggunakan Framework Hibernate dan TiwulFX”**.

### **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dikemukakan sebelumnya, maka dapat dirumuskan beberapa masalah yang akan diangkat pada penelitian ini, yaitu bagaimana membangun aplikasi generator kode Java yang menggunakan framework Hibernate dan TiwulFX?

### **1.3 Batasan Masalah**

Beberapa batasan masalah yang digunakan dalam skripsi ini adalah sebagai berikut :

1. Hasil dari generator kode yang dihasilkan adalah *class* POJO (*Plain Old Java Object*), UI (*User Interface*) dan file FXML dari setiap tabel yang telah ditentukan oleh pengguna aplikasi ini.
2. Kode yang dihasilkan hanya bisa CRUD untuk sebuah POJO saja. Jika terdapat relasi dalam POJO maka programmer harus mengkostumasi sendiri *logic-logic* yang diinginkan.
3. Antar muka yang digunakan menggunakan *command line* melalui *cmd* atau *terminal*.
4. Bahasa yang digunakan dalam antar muka menggunakan bahasa Inggris.

#### 1.4 Maksud dan Tujuan Penelitian

Maksud dan tujuan dari penelitian serta penyusunan skripsi ini adalah sebagai berikut :

1. Membangun aplikasi generator kode dengan menggunakan *framework* Hibernate dan TiwulFX yang dapat membantu programmer dalam mengembangkan aplikasi Java Desktop.
2. Menambah wawasan penulis tentang teknologi TiwulFX/JavaFX dan Hibernate ORM.

#### 1.5 Metode Penelitian

Metode yang digunakan dalam menyelesaikan tugas akhir ini adalah sebagai berikut :

##### 1.5.1 Pengumpulan Data

1. Wawancara

Yaitu metode dengan cara wawancara secara langsung orang yang berhubungan ataupun orang yang lebih ahli mengenai penelitian yang di buat ini. Dalam penelitian ini penulis mewancarai Amrullah. Beliau merupakan programmer Java senior dan juga *team-leader* sekaligus perancang TiwulFX.

2. Studi Pustaka

Dilaksanakan dengan cara mencari data atau informasi dari buku-buku atau literatur yang berkaitan dengan permasalahan yang dibahas serta melakukan pencarian data di Internet.

### 1.5.2 Pengembangan Aplikasi

#### 1. Analisa Kebutuhan

Analisa kebutuhan merupakan fase penentuan sarana dan prasarana yang dibutuhkan untuk membangun sistem berdasarkan tujuan yang telah ditentukan. Dalam tahap ini juga dikaji data apa saja yang diolah dan hasil keluaran kode logic-logic apa saja yang dihasilkan oleh aplikasi ini.

#### 2. Perancangan Sistem

Dalam tahap ini dilakukan pembuatan struktur sistem aplikasi dan pendukungnya. Perancangan menggunakan *flow-chart* dengan mempertimbangkan tujuan aplikasi ini.

#### 3. Implementasi

Dilakukan penerjemahan konsep ke dalam bentuk perangkat lunak dengan mempergunakan rancangan sistem, rancangan data dan rancangan proses yang telah dibuat serta mempertimbangkan hasil analisa kebutuhan.

#### 4. Uji coba

Dalam tahap ini dilakukan uji coba terhadap aplikasi yang telah dibuat guna memastikan hasil perancangan dan implementasi yang tepat untuk digunakan. Baik aplikasi generator kode maupun kode yang dihasilkan diuji menggunakan metode *white box* dan *black box testing*.

## 1.6 Sistematika Penulisan

Agar dapat tercapai penulisan yang sistematis mengenai permasalahan sebagai hasil penelitian, maka akan lebih baik dan lebih jelas serta terarah apabila terlebih dahulu diberi gambaran sistematika secara ringkas mengenai susunan skripsi ini, maupun tentang apa yang dikandung dalam skripsi ini, sehingga akan mempermudah dalam pemahaman dan pembatasannya. Secara garis besar sistematika penulisan laporan skripsi ini adalah sebagai berikut :

### **BAB I PENDAHULUAN**

Berisi tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian dan sistematika penulisan laporan penelitian.

### **BAB II LANDASAN TEORI**

Dalam bab ini membahas mengenai teori – teori yang menunjang perancangan dan pembuatan aplikasi, dan penjelasan mengenai *software – software* yang digunakan dalam perancangan aplikasi.

### **BAB III ANALISA DAN PERANCANGAN SISTEM**

Pada bab ini akan dijelaskan tentang tahap – tahap analisis dan perancangan aplikasi yang akan dibuat.

### **BAB IV IMPLEMENTASI SISTEM DAN PEMBAHASAN**

Dalam bab ini akan dijelaskan hasil dari perancangan sistem, pengujian, dan implementasi sistem yang telah dibuat.

**BAB V PENUTUP**

Bab ini merupakan bagian akhir yang berisikan kesimpulan dan saran dari seluruh isi laporan.

