BAB I
PENDAHULUAN

1.1. Latar Belakang

Beragam fitur yang ditawarkan oleh bahasa pemrograman berorientasi objek antara lain lebih mudah digunakan dalam pembangunan sistem, waktu pengembangan dan penggunaan (*reuse*) kode program lebih tinggi, tidak adanya pemisah antara fase desain dan analisis, analis dan programer tidak dibatasi dengan batasan implementasi sistem, relasi antara objek dan entitas umumnya dapat dimapping dengan baik seperti kondisi pada dunia nyata, memungkinkan adanya perubahan dan kepercayaan diri yang tinggi terhadap kebenaran *software* yang membantu untuk mengurangi resiko pada pembangunan sistem yang komplek, *encapsulation* data dan *method* memungkinkan penggunaan kembali pada proyek lain, memungkinkan adanya standarisasi objek yang memudahkan memahami desain dan mengurangi resiko pelaksanaan proyek, serta dimungkinkan seorang analis untuk memecah masalah menjadi pecahan-pecahan masalah yang di-*manage* secara terpisah menjadi alasan dipilihnya bahasa pemrograman berorientasi objek diperusahaan-perusahaan yang bergerak dibidang IT (Supriyanto, S.Tp).

Tingginya tren penggunaan bahasa pemrograman berorientasi objek terutama bahasa Java (Tiobe Software, 2012) menuntut programer harus memiliki kemampuan untuk mampu membangun perangkat lunak berdasarkan konsep pemrograman berorientasi objek. Menurut hasil pengamatan terhadap beberapa buku referensi pemrograman berorientasi objek dan hasil kuisioner dari beberapa mahasiswa teknik informatika Universitas Komputer Indonesia yang telah mengambil mata kuliah pemrograman dapat disimpulkan bahwa mahasiswa mengalami kesulitan dalam memahami konsep pemrograman berorientasi objek dikarenakan perbedaan cara penulisan *source code* dalam buku referensi dengan cara penulisan yang digunakan dalam *Integrated Development Environtment* (IDE) yang ada seperti *Netbeans dan Eclipse.* Antar muka IDE yang terlalu rumit juga menyulitkan mahasiswa dalam proses belajar dikarenakan mahasiswa harus

mempelajari terlebih dulu antar muka yang komplek sebelum mempelajari konsep-konsep pemrograman berorientasi objek.

Atas dasar permasalahan tersebut perlu dikembangkan sebuah *e-learning* yang dapat memudahkan mahasiswa dalam proses belajar pemrograman berorientasi objek dengan mudah. Maksud mudah dalam penelitian ini adalah dalam proses pembelajaran pemrograman berorientasi objek materi disusun dengan mengedepankan unsur kemudahan dalam penerapan *source code* yang terdapat dalam materi sehingga mahasiswa dapat secara langsung mengolah *source code* yang terdapat pada materi dan melihat hasilnya. Selain itu mahasiswa dapat menciptakan sebuah *object* dari suatu *class* secara langsung untuk memudahkan pemahaman konsep pemrograman berorientasi objek.

1.2. Identifikasi Masalah

Berdasarkan identifikasi masalah yang mengacu pada permasalahan diatas maka dirumuskan masalah sehubungan dengan pembangunan aplikasi ini, yaitu: Bagaimana cara membuat *e-learning* untuk memudahkan mahasiswa mempelajari pemrograman berorientasi objek.

1.3. Maksud Dan Tujuan

Berdasarkan permasalahan yang diteliti, maka maksud dari penulisan TA ini adalah untuk membangun aplikasi *e-learning* untuk mempermudah proses pembelajaran bahasa pemrograman berorientasi objek.

Sedangkan tujuan yang akan dicapai dalam penelitian ini adalah mempermudah mahasiswa dalam belajar pemrograman berorientasi objek.

1.4. Batasan Masalah

Berdasarkan hasil penelitian, maka dibuatlah batasan-batasan masalah sebagai berikut:

1. Aplikasi *e-learning* untuk pemrograman berorientasi objek berupa aplikasi berbasis *website*.
2. Materi pembelajaran mengacu pada buku Pemrograman Dengan Bahasa C++ dan Java dari Nol menjadi Andal (M. Shalalahudin, Rosa A. S. 1994).

3. Bahasa pemrograman yang dipakai dalam materi pembelajaran adalah bahasa pemrograman Java.
4. Pendekatan analisis dan desain yang digunakan ialah dengan pendekatan *Object-oriented Analysis and Design* (OOAD).
5. Metode yang digunakan dalam pengembangan adalah metode *Waterfall.*
6. Aplikasi pembangun yang digunakan adalah *Netbeans* 7.2 dengan bahasa pemrograman yang digunakan adalah PHP, HTML, CSS dan JavaScript
7. *Live compiler* menggunakan *Application Programing Interface* (API) dari ideone.com

1.5. Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penulisan penelitian ini adalah sebagai berikut:

1.5.1. Tahap Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

   Pengumpulan data dengan cara mengumpulkan literatur, jurnal, *paper* dan bacaan-bacaan yang berkaitan dengan judul penelitian.
2. Observasi.

   Teknik pengumpulan data dengan mengadakan penelitian dan peninjauan langsung terhadap permasalahan yang diambil.
3. Kuisioner

   Kuisioner adalah teknik pengumpulan data dengan cara memberikan daftar pertanyaan tertulis yang ditujukan kepada responden. Jawaban responden atas semua pertanyaan dalam kuesioner kemudian dicatat atau direkam.

1.5.2. Tahap Pembuatan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan untuk membangun aplikasi *e-learning* untuk pemrograman berorientasi objek ini menggunakan paradigma model *waterfall* (Ian Sommerville, 2010), yang meliputi beberapa proses diantaranya:

1. *Requirements analysis and definition*

   Tahap ini mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Tahap ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap.

2. *System and software design*

   Tahap ini merupakan kegiatan mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh aplikasi yang akan dibangun. Tahap ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap.

3. *Implementation and unit testing*

   Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji baik secara unit.

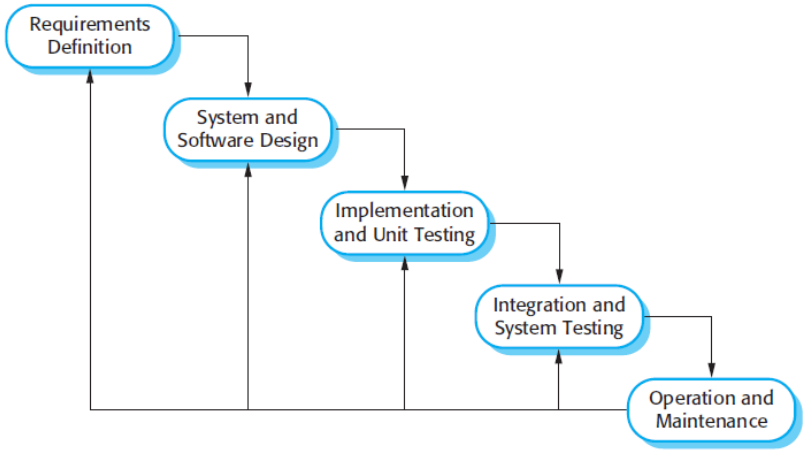
4. *Integration and system testing*

   Penyatuan unit-unit program kemudian diuji secara keseluruhan (*system testing*).

5. *Operation and maintenance*

   Mengoperasikan program dilingkungannya dan melakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi sebenarnya.

Secara umum tahapan pada model *waterfall* dapat dilihat pada gambar 1.1 dibawah ini:



Gambar 1.1. Model *Waterfall*

1.6. Sistematika Penulisan

Sistematika penulisan laporan penelitian ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut :

BAB I. PENDAHULUAN

Menguraikan tentang latar belakang permasalahan, mencoba merumuskan inti permasalahan yang dihadapi, menentukan tujuan dan kegunaan penelitian, yang kemudian diikuti dengan pembatasan masalah, asumsi, serta sistematika penulisan.

BAB II. LANDASAN TEORI

Membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang telah pernah dilakukan sebelumnya termasuk sintesisnya.

BAB III. ANALISIS DAN PERANCANGAN SISTEM

Menganalisis masalah dari model penelitian untuk memperlihatkan keterkaitan antar variabel yang diteliti serta model matematis untuk analisisnya.

BAB IV. IMPLEMENTASI DAN PENGUJIAN SISTEM

Merupakan tahapan yang dilakukan dalam penelitian secara garis besar sejak dari tahap persiapan sampai penarikan kesimpulan, metode dan kaidah yang diterapkan dalam penelitian. Termasuk menentukan variabel penelitian, identifikasi data yang diperlukan dan cara pengumpulannya, penentuan sampel penelitian dan teknik pengambilannya, serta metode/teknik analisis yang akan dipergunakan dan perangkat lunak yang akan dibangun jika ada.

BAB V. KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran yang sudah diperoleh dari hasil penulisan tugas akhir dan saran yang diperlukan untuk pengembangan aplikasi yang telah dibangun.