

BAB III

ANALISIS KEBUTUHAN

3.1 Metode Analisis

Proses analisis dilakukan dengan menggunakan alat dan teknik yang disesuaikan dengan kebutuhan sistem. Pada sistem yang akan dikembangkan, analisis yang digunakan adalah analisis sistem berorientasi obyek karena aplikasi yang akan dibuat berbasis teknologi Java yang sifatnya juga berorientasi obyek. Tahap analisis sistem harus dilakukan dengan sebaik mungkin agar pengembangan perangkat lunak dapat berjalan dengan terstruktur dan jelas serta mampu terdokumentasi dengan baik ketika mengalami peningkatan kompleksitas yang signifikan. Pada tahap ini dijelaskan hal-hal apa saja yang terdapat pada sistem yang akan dibuat yaitu analisis aplikasi secara umum, analisis kode program, dan analisis antarmuka.

3.2 Hasil Analisis

3.2.1 Analisis Umum Aplikasi

Perangkat lunak yang akan dibangun nantinya diusahakan untuk menangani fitur-fitur standar yang sudah lazim ada dalam sebuah aplikasi IRC Client. Sebagai acuan aplikasi yang cukup berkelas dan sangat populer pemakaiannya di kalangan pengguna sistem operasi Windows, mIRC (<http://www.mirc.com>), menjadi referensi utama mengenai fitur-fitur aplikasi yang diimplementasikan.

Hal tersebut bukan berarti bahwa semua fitur yang dimiliki mIRC akan diadopsi seluruhnya, namun diupayakan agar aplikasi IRC Client untuk lingkungan J2ME mampu melakukan banyak hal yang telah dapat dilakukan oleh mIRC.

Setelah melalui proses analisis, ada beberapa hal utama yang harus dapat ditangani oleh sebuah aplikasi IRC Client, yaitu:

- a. Menghubungkan diri ke server.
- b. Melakukan registrasi user.
- c. Melakukan join ke channel.
- d. Menampilkan daftar user di channel.
- e. Mengirim dan menerima pesan di channel.
- f. Mengirim dan menerima pesan antar user.
- g. Keluar dari channel.
- h. Keluar dari jaringan.

3.2.2 Analisis Kode Program

Pembuatan kode-kode program nantinya akan selalu berkaitan dengan beberapa hal utama, yaitu yang berhubungan dengan jaringan TCP/IP, protokol IRC, dan langkah-langkah mengabstraksikannya dalam paradigma berorientasi obyek.

3.2.2.1 Pemrograman Socket

Dalam TCP/IP, untuk melakukan koneksi dan pertukaran data dilakukan dengan mengimplementasikan pemrograman socket ke dalam aplikasi. *Socket*

merupakan implementasi mendasar yang harus dilakukan untuk membentuk lapisan (*layer*) protokol di atasnya.

Pada dasarnya sebuah *socket* hanya melakukan dua hal utama, yaitu mengirim data dan menerima data. Sebuah *socket* melakukan koneksi TCP/IP berdasarkan nama komputer dan nomor port yang dituju. Nomor port menunjukkan layanan apa yang akan diakses, misalnya nomor port 6667 adalah standar Internet untuk layanan IRC.

Sebenarnya istilah *socket* digunakan untuk dua macam *socket*, yaitu *client socket* dan *server socket*. Aplikasi yang dibuat dalam penelitian ini adalah aplikasi client yang menggunakan *client socket*. Sedangkan *server socket* diimplementasikan oleh aplikasi server IRC.

3.2.2.2 Protokol IRC

Dalam jaringan komputer, protokol digunakan untuk melakukan komunikasi data dengan tujuan tertentu. Oleh karena di Internet terdapat banyak sekali layanan, maka harus ada suatu mekanisme untuk membedakannya, yaitu menggunakan protokol. Untuk protokol IRC, mekanisme dan tata cara komunikasinya diuraikan secara lengkap dalam RFC 1459.

3.2.2.3 Struktur Aplikasi

Untuk mempermudah pembuatan aplikasi, maka diperlukan adanya pemecahan atau pembagian struktur aplikasi berdasarkan komponen-komponen atau obyek-obyek yang ada dalam aplikasi.

Dalam pembuatan aplikasi J2ME IRC Client, tentunya akan lebih mudah apabila komponen-komponen pembentuknya diperinci untuk tiap peran dan fungsinya masing-masing. Setelah dilakukan analisis, terdapat beberapa komponen utama yang diperlukan, yaitu:

- a. Komponen untuk menangani koneksi TCP/IP.
- b. Komponen untuk menangani protokol IRC.
- c. Komponen untuk menangani antarmuka (*display*).
- d. Komponen untuk menangani lingkungan J2ME.
- e. Komponen utama untuk mengawasi dan mengendalikan seluruh komponen lain.

3.2.3 Analisis Antarmuka

Fungsi-fungsi API (*Application Programming Interface*) dalam kelas-kelas J2ME yang disediakan khusus untuk perancangan antarmuka terlihat sangat sederhana. Hal tersebut bukan berarti bahwa Sun Microsystems sebagai vendor Java tidak mampu menyediakan kelas-kelas perancangan antarmuka yang lebih baik, namun menunjukkan keseriusan dalam mempertimbangkan portabilitas aplikasi, agar cita-cita untuk mewujudkan sebuah aplikasi yang mampu berjalan dengan baik di semua piranti (*run anywhere*) dapat tercapai. Dengan pertimbangan tersebut, maka aplikasi yang dibuat akan menggunakan fungsi-fungsi API standar bawaan J2ME agar portabilitas antar piranti dapat terjaga dengan lebih baik.

Dalam J2ME, komponen-komponen untuk perancangan GUI (*Graphical User Interface* atau Antarmuka Pemakai Grafis) relatif lebih sedikit dan lebih sederhana dibanding GUI-GUI pada aplikasi berbasis PC. Untuk perancangan form, biasanya menggunakan High-Level API yang sudah tersedia, yang terdiri dari beberapa kelas, yaitu Display, Screen, Ticker, TextBox, Alert, dan List. Kemudian ada juga kelas Form yang dapat dimasuki komponen-komponen di dalam Form, yaitu ChoiceGroup, DateField, Gauge, StringItem, TextField, Image, dan ImageItem.

Dari sedikit uraian tersebut, dapat dilihat bahwa dalam J2ME tidak disediakan API untuk penggunaan kelas semacam Button (tombol tekan). Sepertinya Sun Microsystem menyadari bahwa lingkungan layar J2ME yang sempit memang kurang kondusif untuk penggunaan tombol di dalam rancangan form. Untuk menangani berbagai aksi di dalam J2ME, telah disediakan High-Level API untuk manajemen event, yaitu dengan mengimplementasikan interface CommandListener. Setiap aksi yang dilakukan didefinisikan sebagai instance dari kelas Command untuk kemudian diproses oleh interface CommandListener ketika aksi tersebut diaktifkan oleh pemakai aplikasi. Dari beberapa penjelasan tersebut, dapat dilihat bahwa seakan-akan manajemen event berada di luar ruang lingkup perancangan GUI padahal tetap memiliki keterkaitan erat dengan perancangan GUI karena setiap aksi yang didefinisikan akan memunculkan suatu item dalam menu di layar.