Bab I : PENDAHULUAN

Pada bagian ini akan dijelaskan mengenai gambaran umum mengenai pengembangan aplikasi Java Search Tool meliputi latar belakang, tujuan pembuatan aplikasi serta fitur dasar dan batasan pengembangan aplikasi.

I.1. Latar Belakang

Saat ini perkembangan teknologi media penyimpanan komputer telah berkembang dengan pesat, sehingga memungkinkan penggunanya untuk menyimpan dokumen dalam jumlah besar. Sementara itu, dengan banyaknya dokumen yang tersimpan, secara tidak langsung pengguna akan mengalami kesulitan dalam mencari dokumen yang dibutuhkan pada saat membutuhkannya. Di samping itu, perkembangan teknologi *search engine* memungkinkan untuk melakukan proses *indexing* dan pencarian data yang ada dalam media penyimpanan ataupun melalui jaringan internet.

Oleh karena itu penulis memilih untuk mengembangkan aplikasi *search engine* yang memungkinkan penggunanya untuk mencari dokumen yang terdapat pada komputer ataupun melalui jaringan internet.

I.2. Tujuan

Tujuan dari proyek tugas akhir ini adalah mengembangkan aplikasi *search engine* menggunakan metode *inverted file indexing* dan metode pencarian ruang *vector* yang juga mendukung pencarian secara *native* melalui *indexing service* pada *host operating system* dan mendukung pencarian melalui jaringan internet Google. Aplikasi dikembangkan dengan menggunakan bahasa pemrograman berorientasi objek *Java*.

I.3. Gambaran Umum

Pada saat pertama kali digunakan, aplikasi akan meminta pengguna untuk melakukan konfigurasi aplikasi umum seperti apakah pencarian akan dilakukan dengan menggunakan *index* dari *indexing service* sistem operasi

atau dengan membuat *index* baru. Apabila pengguna memilih untuk membuat *index* baru, maka pengguna akan diminta untuk memulai proses pembuatan *index*.

Setelah *index* dibuat, pengguna dapat memulai menggunakan fitur pencarian. Aplikasi akan membaca masukan dari pengguna berupa kata kunci dari dokumen yang akan dicari dan lokasi pencarian (komputer atau melalui Google). Setelah proses pencarian selesai dilakukan, hasil dari pencarian akan ditampilkan dalam bentuk tabel berisi keterangan nama dokumen, tanggal modifikasi terakhir dan lokasi dimana dokumen disimpan. Pengguna juga dapat memasukkan *bookmark* hasil pencarian ataupun mengatur *bookmark* yang telah ada. Fitur *bookmark* ini memungkinkan pengguna untuk melihat ulang hasil pencarian yang disimpan dengan lebih cepat.

Aplikasi ini juga memiliki fitur-fitur yang memungkinkan pengguna untuk membuka direktori tempat dokumen atau membuka dokumen dengan aplikasi lain yang terdapat pada sistem operasi yang digunakan. Untuk pengguna Mac OS X versi 10.5 atau lebih, tersedia fitur untuk melakukan *preview* dokumen menggunakan QuickLook.

I.4. Batasan Masalah

Batasan-batasan dalam pengembangan aplikasi ini adalah:

- Integrasi pencarian menggunakan *native indexing service* hanya dapat dilakukan pada sistem operasi berbasis UNIX yang mendukung perintah *locate* atau *mdfind*,
- Pencarian melalui Google hanya mendukung pencarian terhadap direktori GoogleBase dan *YouTube*,
- Pencarian terhadap *metadata* dan isi dokumen hanya mendukung dokumen dengan format teks, PDF, Microsoft Office 2000/XP, dan MP3 dengan *ID3Tag* versi 1.0 dan 2.3,

- *Update index* dilakukan secara manual oleh pengguna,
- *Stop word list* menggunakan bahasa Inggris dari Paice/Husk *stemmer*.

I.5. Detail Teknis

Berikut adalah beberapa detail teknis dari aplikasi ini:

- Pemrosesan dokumen tipe Microsoft Office 2000/XP dilakukan dengan mengunakan *library* Apache POI (Poor Obfuscation Implementation) versi 3.0.2 yang memiliki lisensi Apache *License*,
- Pemrosesan dokumen tipe PDF menggunakan *library* PDFBox versi 0.7.3 yang memiliki lisensi PDFBox *License*,
- Pemrosesan metadata untuk dokumen gambar menggunakan *package javax.imageio* dari JDK,
- Pemrosesan *ID3Tag* untuk dokumen MP3 dilakukan menggunakan *library* JID3 versi 0.4.6 yang memiliki lisensi GNU GPL,
- Pengenalan *mime type* dilakukan dengan menggunakan *library* MimeUtil yang memiliki lisensi GNU GPL,
- Pencarian terhadap GoogleBase dilakukan dengan menggunakan *parser* XML dari JDK,
- Metode *indexing* menggunakan *inverted file indexing*,
- Metode pencarian menggunakan pemodelan ruang vector,
- *Runtime* untuk pencarian menggunakan *index native* sistem operasi menggunakan Spotlight pada Mac OS X versi 10.4 atau lebih, untuk sistem operasi UNIX lainnya menggunakan *locate*,
- *Runtime* untuk melakukan *preview* dokumen pada sistem operasi Mac OS X versi 10.5 atau lebih menggunakan AppleScript dan QuickLook *manager*,

- Dapat berjalan dengan baik pada semua sistem operasi yang didukung oleh Java SE atau JRE minimal versi 1.6.0.

I.6. Definisi dan Singkatan

Bagian ini menjelaskan istilah-istilah teknis dan singkatan-singkatan yang digunakan pada laporan ini. Untuk istilah-istilah teknis yang terdapat pada laporan ini ditulis dengan cetak miring. Untuk istilah atau singkatan yang merujuk pada suatu merek dagang atau nama fitur dari suatu merek dagang akan ditulis dengan huruf pertama kapital tanpa cetak miring.

I.6.1. JRE(Java Runtime Environment) atau JVM

Aplikasi dasar yang memungkinkan aplikasi yang dibuat dalam bahasa pemrograman *Java* dalam berjalan pada setiap sistem operasi dan berbagai jenis arsitektur perangkat keras. Aplikasi Java Search Tool dikembangkan menggunakan teknologi Java SE(Standard Edition) yang merupakan *Java Runtime* untuk aplikasi *desktop*.

I.6.2. JDK(Java Development Kit)

Kumpulan dari *Compiler* dan *library* yang dikembangkan oleh SUN Microsystem untuk mengembangkan aplikasi *desktop* berbasis bahasa pemrograman *Java*. Aplikasi Java Search tool dikembangkan menggunakan *JDK* versi 1.6.

I.6.3. Search Engine

Aplikasi yang memungkinkan penggunanya untuk dapat melakukan proses pencarian informasi berdasarkan kata kunci yang diberikan. Pengembangan aplikasi Java Search Tool merupakan salah satu *search engine*.

I.6.4. Inverted File Indexing

Salah satu metode untuk melakukan proses *indexing* yang memungkinkan pembuatan dokumen *index*. Metode ini membagi dokumen *index* menjadi tiga bagian yaitu *inverted file*, *inverted list* dan *lexicon*. *Inverted file* berisi

keterangan mengenai jumlah dokumen dan nomor dokumen yang memiliki kata yang terdapat dalam *lexicon*. *Inverted list* berisi keterangan spesifik per dokumen terhadap kata yang terdapat dalam *lexicon*. Sedangkan *lexicon* sendiri berisi keseluruhan kata yang ditemukan dengan alamat dokumen yang memiliki kata tersebut.

I.6.5. *Vector Space Model* atau Pemodelan Ruang Vektor

Salah satu metode pencarian dokumen dengan menggunakan pendekatan aljabar. Metode ini akan memproses kata kunci dan dokumen *index* yang ada dengan pemodelan ruang vektor multi dimensi. Proses penilaian dokumen dilakukan berdasarkan nilai kemiripan yang didapat dari penghitungan bobot dokumen terhadap kata kunci. Metode ini merupakan metode yang sangat umum digunakan di banyak *search engine* termasuk Java Search Tool.

I.6.6. Index

Dokumen ataupun *database* yang berisi informasi mengenai dokumen-dokumen yang diperoleh melalui proses *indexing* dengan menggunakan metode tertentu. Java Search Tool dapat membuat dokumen *index* ini dengan menggunakan metode *inverted file indexing*.

I.6.7. Google dan GoogleBase

Aplikasi berbasis *web* yang berfungsi sebagai *web search engine* dikembangkan oleh Google, inc. Sementara itu, GoogleBase merupakan suatu fitur yang dikembangkan oleh Google untuk menyediakan informasi berupa produk, pekerjaan, jasa dan lain sebagainya. Informasi yang disediakan oleh GoogleBase merupakan data dari klien ataupun *business partner* dari Google sebagai sarana promosi bagi perusahaan tersebut.

I.6.8. Mime Type

Informasi tipe dokumen yang disimpan pada *header* suatu dokumen dengan panjang *byte* berbeda-beda untuk setiap tipe dokumen. Aplikasi

Java Search Tool memungkinkan pengenalan *mime type* menggunakan *library* MimeUtil.

I.6.9. ID3Tag, ID3Tag versi 1.0 dan ID3Tag versi 2.3

Standar informasi tambahan yang dapat dimasukkan ke dalam dokumen audio dan video. ID3Tag memungkinkan untuk memasukkan informasi spesifik seperti *title*, *artist*, *publisher, DRM* dan bahkan dokumen gambar pada dokumen audio dan video. Untuk ID3Tag versi 1.0, informasi yang bisa disimpan hanya informasi lagu sederhana seperti *singer*, *composer*, *title*, *album* dan tahun produksi. Sementara untuk ID3Tag versi 2.3, informasi seperti Digital Rights Management, dokumen gambar, lirik lagu dan informasi pembelian dapat disimpan dalam ID3Tag. Aplikasi Java Search Tool memungkinkan pencarian ID3Tag terhadap dokumen audio dengan format MP3 menggunakan *library* JID3 versi 0.4.6.

I.6.10. Runtime atau service

Aplikasi dasar yang pada umumnya dijalankan secara otomatis pada saat sistem operasi dijalankan. Cara akses untuk setiap *runtime* atau *service* ini diatur oleh masing-masing sistem operasi dan memiliki perlakuan yang berbeda-beda. Aplikasi Java Search Tool menggunakan *runtime* atau *service* tertentu untuk metode pencarian menggunakan *index native* sistem operasi.

I.6.11. Native dan native indexing service

Native dalam dokumen ini merujuk pada kemampuan aplikasi untuk melakukan akses terhadap proses atau *runtime* yang terdapat pada sistem operasi dimana aplikasi dijalankan. Sementara itu, *native indexing service* merujuk pada *runtime* atau *service* yang bertanggung jawab untuk melakukan proses *indexing* ataupun pencarian dokumen yang ditangani oleh sistem operasi.

I.6.12. Native Library

Merupakan *library* ataupun kelas-kelas yang berasal dari sistem operasi untuk mengangani proses-proses tertentu. Pada umumnya *native library* ditulis dalam bahasa pemrograman sesuai dengan bahasa pemrograman yang digunakan untuk membuat sistem operasi tersebut. Pada laporan ini, semua *native library* merujuk pada JNILIB atau *library* yang ditulis menggunakan bahasa pemrograman C++ yang dapat dipanggil melalui bahasa pemrograman Java.

I.6.13. AppleScript

Suatu bahasa pemrograman sederhana yang dikembangkan oleh Apple, inc untuk melakukan otomatisasi tugas-tugas sederhana dan eksekusi perintah. AppleScript merupakan fitur utama pada sistem operasi Mac OS X dan digunakan oleh Java Search Tool untuk melakukan eksekusi perintah konsol agar dapat melakukan *preview* dokumen secara *native* oleh *QuickLook*.

I.6.14. QuickLook

Runtime yang memungkinkan pengguna untuk dapat melakukan *preview* dokumen tanpa harus membuka aplikasi yang dihubungkan dengan tipe dokumen tersebut. *QuickLook* merupakan salah satu fitur utama yang terdapat pada sistem operasi Mac OS X versi 10.5 dengan kode produksi *Leopard*. *QuickLook* mendukung sistem *plugin* yang membuat *runtime* ini dapat mendukung hampir semua tipe data. Aplikasi Java Search Tool mendukung penggunaan *runtime* ini untuk memudahkan pengguna dalam melakukan *preview* dokumen hasil pencarian.

I.6.15. Spotlight

Aplikasi *search engine* yang mulai dikenalkan mulai dari sistem operasi Mac OS X versi 10.4 yang memungkinkan penggunanya untuk mencari dokumen dengan mudah dan cepat. *Spotlight* merupakan aplikasi yang bersifat *runtime* sehingga memungkinkan Java Search Tool untuk dapat

mengakses *runtime* ini untuk melakukan pencarian dengan menggunakan *index native* dari *runtime Spotlight*.

I.6.16. Locate

Aplikasi yang bersifat *runtime* yang memungkinkan penggunanya melakukan pencarian dokumen. Aplikasi ini umumnya secara otomatis terdapat pada sistem operasi berbasis UNIX seperti BSD, Linux, OpenSolaris dan Mac OS X. Java Search Tool menggunakan *runtime* ini untuk melakukan pencarian secara *native* pada sistem operasi berbasis UNIX selain Mac OS X(menggunakan *Spotlight*).

I.6.17. YouTube

Merupakan aplikasi berbasis *web* yang menyediakan jasa untuk berbagi dokumen video yang dapat diakses oleh penggunanya melalui *web streaming*. Aplikasi Java Search Tool memungkinkan pengguna untuk melakukan pencarian dokumen video yang terdapat pada *YouTube*.

I.6.18. DTrace

Merupakan bahasa pemrograman D yang dikembangkan oleh SUN Microsystems, inc untuk melakukan *tracing* atau *monitoring* sistem operasi keseluruhan atau suatu aplikasi yang sedang berjalan secara *low level* atau berinteraksi langsung dengan *kernel* sistem operasi tersebut. Pertama kali diimplementasikan pada sistem operasi Solaris 10. Saat ini beberapa sistem operasi modern berbasis *kernel* BSD (Berkeley System Distribution) dengan Mach *kernel* 3.0 telah mengimplementasikan DTrace menjadi bagian dari *kernel* sistem operasi tersebut.

DTrace dapat digunakan secara *code level* pada bahasa pemrograman umum seperti C++, Objective-C dan Java dengan metode *mixed mode*. Selain itu dapat juga dijalankan sebagai *script* terpisah yang ditujukan terhadap aplikasi yang akan diuji.

I.6.19. Apple Instruments

Merupakan aplikasi *development* yang terdapat pada sistem operasi Mac OS X versi 10.5 ke atas, yang memungkinkan para pengembang untuk merekam suatu aksi dan melakukan pengawasan terhadap penggunaan *resources* pada sistem. Aplikasi ini melakukan pengawasan *resources* dengan menggunakan DTrace yang telah disediakan maupun *script* DTrace yang didefinisikan oleh pengguna.

I.6.20. I/O dan I/O Usage

Merupakan suatu kondisi yang menggambarkan penggunaan *input* dan *output* berupa transfer data dari suatu aplikasi terhadap alat *input* dan *output*. Pada umumnya persentase I/O Usage ditujukan terhadap aktivitas media penyimpanan *non volatile* atau *disk drive*. Persentase nilai dipengaruhi oleh kegiatan pembacaan, penulisan dan proses baca-tulis secara bersamaan.

I.6.21. CPU dan CPU Usage

CPU atau Central Processing Unit merupakan suatu alat utama dalam sistem komputer yang berfungsi sebagai alat hitung dan *interpreter* untuk memproses setiap proses atau *thread* yang sedang dijalankan oleh sistem operasi ataupun suatu aplikasi. CPU Usage menggambarkan utilisasi penggunaan CPU saat menjalankan suatu proses dalam nilai persentase. Pada umumnya, persentase nilai CPU Usage digambarkan dengan jumlah CPU dikalikan 100%. Nilai CPU usage maksimal untuk mesin dengan satu CPU adalah 100%, mesin dengan dua CPU adalah 200% dan seterusnya.

I.6.22. Object Allocation

Merupakan suatu proses dimana sebuah aplikasi melakukan proses penciptaan objek dan mengalokasikan penggunaan memori pada sistem untuk menyimpan objek tersebut.

I.6.23. Memory Leak

Suatu kondisi dimana suatu aplikasi tidak melepas alokasi data di memori saat suatu objek sudah tidak digunakan atau bahkan aplikasi tersebut sudah tidak berjalan. Hal ini menyebabkan *resource* di memori tidak dapat dipakai oleh proses atau aplikasi lainnya yang berakibat memperlambat kinerja sistem dan dapat menyebabkan kelumpuhan sistem.

I.6.24. Garbage Collector

Merupakan sebuah *bot* ataupun *runtime* yang dijalankan oleh sistem atau *runtime* bahasa pemrograman untuk melakukan pembersihan data di memori dan melepas *resource* yang sudah tidak dipakai. Hal ini untuk menghindari suatu aplikasi atau proses menyebabkan *memory leak*. Biasanya ditulis dalam bahasa pemrograman *assembly* dan terdapat pada sistem operasi modern ataupun pada *runtime* bahasa pemrograman.

I.7. Struktur Laporan

Laporan tugas akhir ini terdiri dari enam bab dengan penjelasan sebagai berikut:

- Bab I. Pendahuluan

  Gambaran umum tentang aplikasi yang dikembangkan pada tugas akhir ini. Menjelaskan latar belakang pemilihan topik, istilah - istilah khusus serta pembatasan cakupan masalah yang akan diimplementasikan pada aplikasi.

- Bab II. Spesifikasi Produk

  Gambaran teknis dari aplikasi, berisi gambaran keseluruhan aplikasi dan spesifikasi produk.

- Bab III. Desain Perangkat Lunak

  Berisi pemodelan rancangan aplikasi dalam bentuk diagram-diagram beserta penjelasannya.

- Bab IV. Implementasi Desain

  Berisi implementasi pengembangan aplikasi dalam bentuk bahasa pemrograman dengan mengacu pada desain perangkat lunak yang ada pada bab 3.

- Bab V. Pengujian

  Membahas metode pengujian aplikasi untuk kemudian mengambil kesimpulan mengenai fungsionalitas produk akhir dan hasil yang dicapai serta perbandingan dengan gambaran yang tercantum pada bab 1 dan bab 2.

- Bab VI. Kesimpulan dan Saran

  Berisi kesimpulan dari pengembangan aplikasi dan saran-saran untuk pengembangan lebih lanjut di masa yang akan datang.