

## PENCARIAN DOKUMEN BERBASIS KONTEN PADA PERPUSTAKAAN DIGITAL UNTUK KOLEKSI TEKS DAN MULTIMEDIA

Umi Laili Yuhana<sup>1</sup>, Shintami Chusnul<sup>1</sup>, Supriadi<sup>2</sup>,  
<sup>1</sup>Jurusan Teknik Informatika ITS, <sup>2</sup>Alumni Jurusan Teknik Informatika ITS  
Kampus ITS Keputih, Sukolilo, Surabaya  
Email: [yuhana@if.its.ac.id](mailto:yuhana@if.its.ac.id), [shintami@its-sby.edu](mailto:shintami@its-sby.edu), [xsupri05@yahoo.co.id](mailto:xsupri05@yahoo.co.id)

### Abstrak

Perpustakaan digital dapat membantu pengguna mengelola dan menemukan kembali informasi atau bahan belajar yang berbasis teks dan multimedia. Perpustakaan digital di Jurusan Teknik Informatika telah dikembangkan untuk mengelola koleksi digital berbasis teks dan multimedia termasuk video. Merupakan suatu permasalahan tersendiri bagaimana menemukan kembali koleksi video yang berukuran besar yang hanya memiliki sedikit informasi tekstual. Selain kurangnya fitur tekstual, kecepatan internet yang rendah dan ukuran file yang besar merupakan tantangan dalam penanganan dokumen berbasis multimedia. Makalah ini memaparkan sebuah solusi pencarian dokumen berbasis konten dalam perpustakaan digital. Proses pencarian ini memanfaatkan informasi teks yang tersimpan dalam dokumen digital. Pada dokumen teks, informasi langsung didapatkan dari konten dokumen yang dimaksud, sementara pada dokumen multimedia, informasi didapatkan dari informasi tekstual yang terdapat di dalamnya. Informasi tekstual tersebut merupakan hasil dari konversi sumber data ke dalam bentuk teks. Titik tekan makalah ini adalah penerapan indexing yang baik untuk mempercepat proses pencarian dokumen, dan sistem dilengkapi dengan fitur search engine. Selain itu, isu penting lainnya adalah proteksi dokumen dimana user eksternal hanya bisa mengakses abstrak sedangkan user internal bisa mengakses dan mendownload dokumen. Dokumen dalam perpustakaan digital ini juga dilengkapi enkripsi sehingga tidak mudah untuk dilakukan copy-paste. Untuk mendukung kelengkapan koleksi, maka sistem perpustakaan digital ini juga mendukung koleksi multimedia.

Kata kunci: pencarian berbasis konten, perpustakaan digital, indexing.

### 1. Pendahuluan

Perkembangan dunia perpustakaan, dari segi data dan dokumen yang disimpan, dimulai dari perpustakaan tradisional yang hanya terdiri dari kumpulan koleksi buku tanpa katalog, kemudian muncul perpustakaan semi modern yang menggunakan katalog (*index*). Perkembangan mutakhir adalah munculnya perpustakaan digital (*digital library*) yang memiliki keunggulan dalam kecepatan pengaksesan karena berorientasi ke data digital. Di sisi lain, dari segi manajemen (teknik pengelolaan), dengan semakin kompleksnya koleksi perpustakaan, saat ini muncul kebutuhan akan penggunaan teknologi informasi untuk otomatisasi *business process* di perpustakaan. Sistem yang dikembangkan kemudian terkenal dengan sebutan sistem otomasi perpustakaan (*library automation system*).

Perpustakaan digital erat kaitannya dengan manajemen dokumen secara digital. Dokumen-dokumen yang ditangani tersebut termasuk koleksi multimedia. Manajemen koleksi multimedia memerlukan metode yang

berbeda dengan dokumen biasa, sehingga muncullah istilah manajemen data multimedia yang menangani koleksi multimedia mulai dari sumber (*source*), penyimpanan (*storage*) dan penyampaiannya (*presentation*). Karena koleksi perpustakaan digital diharapkan selengkap mungkin, maka kebutuhan akan pengelolaan koleksi multimedia ini perlu diakomodasi.

Makalah ini membahas tentang pencarian berbasis konten pada perpustakaan digital di Ruang Baca Jurusan Teknik Informatika. Titik tekan aplikasi ini adalah pada penerapan *indexing* yang baik untuk mempercepat proses pencarian dokumen. Untuk memenuhi kebutuhan tersebut maka sistem perpustakaan digital ini akan dilengkapi dengan fitur *search engine*. Sistem juga akan mengakomodasi kebutuhan proteksi dokumen. User eksternal yang belum teregister hanya akan bisa mengakses abstrak dokumen saja. Sedangkan user internal akan bisa mengakses dan *download* dokumen. Dokumen yang ada dalam perpustakaan digital ini juga akan di enkripsi sehingga tidak mudah dilakukan

*copy-paste* pada isi dokumennya. Dan untuk mendukung kelengkapan koleksi, perpustakaan digital ini juga menyediakan dukungan terhadap koleksi multimedia.

Makalah ini disusun dalam susunan sebagai berikut. Bagian pertama menjelaskan tentang latar belakang, permasalahan dan tujuan penulisan makalah. Bagian kedua berisi paparan tentang perpustakaan digital berbasis teks dan multimedia. Bagian ketiga mengulas tentang teknik indexing. Bagian keempat berisi pembahasan detail tentang tahapan proses manipulasi dokumen dan proses pencarian berbasis konten pada dokumen teks dan multimedia. Bagian kelima menjelaskan tentang uji coba dan analisis hasil. Bagian terakhir berisi kesimpulan dan penelitian lanjutan yang dapat dikembangkan.

## 2. Perpustakaan Digital

Perpustakaan digital adalah pengelolaan koleksi yang berupa obyek digital. Obyek digital tersebut dibuat atau dikumpulkan berdasarkan prinsip pengembangan koleksi dan diatur sedemikian rupa sehingga memungkinkan user untuk mengakses dan mengeksplorasi sumber tersebut seperti material perpustakaan pada umumnya. Obyek digital diperlakukan sebagai sumber yang tetap stabil dalam jangka waktu yang sangat lama dan proses yang sesuai diaplikasikan untuk memastikan kualitas dan survivabilitasnya.

### 2.1. Perpustakaan Digital Berbasis Teks

Jumlah dokumen pada perpustakaan digital biasanya berkembang dengan cepat seiring dengan berjalannya waktu. Bagaimana cara menyimpan (*store*), mengelola (*manage*) dan mencari (*search*) dokumen dalam perpustakaan digital merupakan masalah yang menantang. Dokumen pada perpustakaan digital disimpan sebagai data yang semi terstruktur (*semi-structured data*), sementara dalam database relasional tradisional, dokumen tersebut disimpan sebagai data terstruktur (*structured-data*). Sistem manajemen database relasional tidak bisa menangani data semi-terstruktur secara efisien dan tidak bisa memenuhi kebutuhan temu kembali teks berbasis konten (*content-based text retrieval*).

Telah banyak penelitian yang dilakukan pada data semi-terstruktur seperti *data*

*modeling*, *query language* untuk *text retrieval*, *index methods*, *algoritma text retrieval* dan *algoritma similarity search*. Hasil dari penelitian-penelitian tersebut telah banyak digunakan dalam sistem perpustakaan digital. SSREADER Digital Library, National Digital Library dan WanFang Database adalah perpustakaan digital yang populer di China. Semua perpustakaan digital mengelompokkan dokumen ke beberapa kelas dan mendukung *query* di dalam kelas tersebut. *Metadata search* dan *full text search* menggunakan *single keyword* atau *expression* juga didukung pada perpustakaan digital ini.

Contoh perpustakaan digital lain adalah *Greenstone digital library*, *UC Berkeley Digital library*, *Tufts Digital library*, *ACM digital library*, *NCSTRL* dan lain-lain. Fungsi-fungsi yang serupa juga didukung di perpustakaan digital tersebut seperti *metadata searching*, *full-text searching*, *documents classification* dan *browsing*.

### 2.2. Perpustakaan Digital Berbasis Video

Video digital merupakan tantangan penting dalam perpustakaan digital. Tantangan tersebut muncul karena ukuran dan kurangnya fitur yang tekstual, sehingga menambah kesulitan pada proses *retrieval*.

Sampai saat ini, usaha yang paling komprehensif dan menjanjikan untuk membangun perpustakaan digital yang menangani video digital adalah Infomedia Project. Infomedia menggunakan banyak variasi fitur visual (*color*, *faces*, *text superimpositions*) dan juga fitur tekstual (*speech to text transcripts*) untuk memungkinkan video digital menjadi *retrievable*.

*Project* penting lainnya termasuk IBM's CueVideo, yang telah mengintegrasikan variasi teknik dari *segmentation*, *indexing* dan *user interface* yang dikembangkan Almaden and Watson labs. Ada juga Digital Video Multimedia Group di Columbia yang mengerjakan proyek yang berkaitan dengan pembuatan *video summaries* secara otomatis. Kemudian ada Multimedia Information Retrieval Group di Dublin City University mengembangkan Fischlar Project, yang menyediakan video *broadcast* untuk komunitas universitas. Kelompok ini juga mengembangkan inovasi *user interface* untuk *video repository*. Sementara European

Union's ECHO Project mengembangkan penelitian di bidang *metadata scheme*.

Salah satu contoh sistem perpustakaan digital yang mampu menangani koleksi digital berbasis multimedia (*video digital*) adalah Open Video Digital Library atau OVDL[5]. OVDL menggunakan database MySQL dan modul *interface* Agile View yang digunakan sebagai *middleware* PHP.

OVDL mempunyai *interface browse* yang memungkinkan akses berdasarkan *genre* (*documentaries, educational, lecture, ephemerals, historical*), *duration* (*less than a minute, 1-2 minutes, 2-5 minutes, 5-10 minutes* dan *more than 10 minutes*), *color* (*color or black and white*), *sound* (*with sound or silent*) dan *contribution organization* (*CMU, Internet Archive, etc*). Untuk setiap kategori, diberikan jumlah dokumen yang ada pada kategori tersebut.

Sedangkan untuk fasilitas *search*, didukung 2 jenis proses *search*. *Quick search* dengan hanya memasukkan *keyword* dan *detailed search* yang melibatkan *metadata*.

Setelah *user* menentukan kategori yang diinginkan dengan menggunakan *interface browse* atau *search* maka informasi yang detail ditampilkan dan *preview* dari beberapa *segment* yang ditampilkan. Hal ini penting ketika menangani file video dengan ukuran yang besar. Karena akan butuh waktu untuk mengunduh atau melihatnya, dan akan sedikit sia-sia jika video yang dibuka atau diunduh tidak sesuai keinginan. Untuk mengurangi kemungkinan munculnya kejadian seperti itu, informasi detail tentang video dan *preview* dengan menampilkan *keyframe* bisa dijadikan solusi.

### 3. Indexing

Pada dasarnya ada banyak cara untuk membuat performa server komputer menjadi lebih baik saat dilakukan pengaksesan data pada database dengan skala VLDB (*Very Large Database*) atau OLDB (*Online Large Database*) [2]. Penggunaan *index* merupakan salah satu cara untuk mencapainya. *Index* pada kolom-kolom tabel database mempunyai fungsi seperti indeks kamus atau indeks buku. Dengan adanya *index* maka pencarian data akan lebih cepat dan tidak banyak menghabiskan sumber daya komputer. Pada dasarnya, *index* dibagi menjadi 2 macam, yaitu *clustered index* dan *nonclustered index*.

#### 1. Clustered index

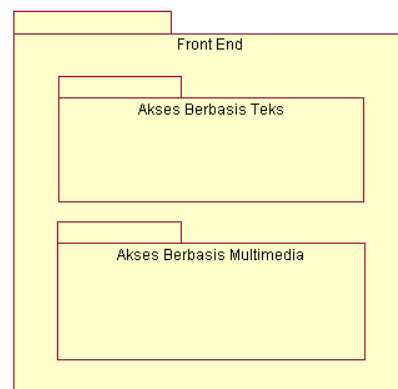
*Clustered index* dapat diumpamakan seperti index huruf pada sebuah kamus. *Clustered index* hanya bisa diterapkan sebanyak satu kali pada satu tabel. Secara otomatis, sebuah *primary key* akan menjadi *clustered index* pada tabel tersebut. *Clustered index* sebaiknya diterapkan pada kolom tabel yang paling sering digunakan pada saat pencarian data.

#### 2. Nonclustered index

Jika kita mengumpamakan *clustered index* seperti index huruf pada sebuah kamus, maka *nonclustered index* dapat diumpamakan seperti sebuah daftar indeks pada sebuah buku. *Non-clustered index* dapat diimplementasikan sebanyak 249 buah pada sebuah tabel. *Nonclustered index* berisi pointer-pointer yang menunjukkan lokasi sesungguhnya dari data yang kita cari saat dilakukan pencarian data. Cara ini sedikit lebih membutuhkan waktu pencarian dibanding dengan metode *clustered index*, namun pada database dengan skala OLDB atau VLDB, metode ini sangat membantu bila dibandingkan dengan penggunaan metode *table-scan*.

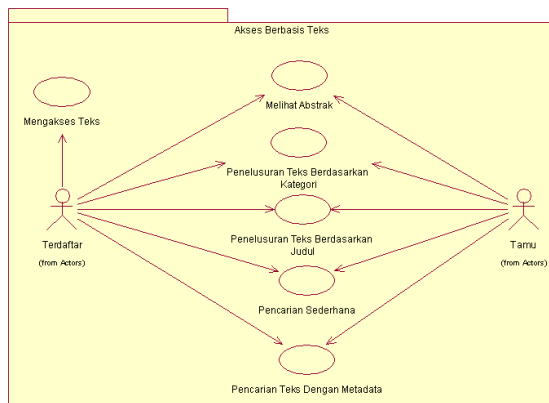
### 4. Pencarian Dokumen Teks dan Multimedia

Gambar 1 menunjukkan gambaran front end dari sistem perpustakaan digital untuk proses pengaksesan dan pencarian untuk dokumen teks dan multimedia. Sistem ini dikembangkan dengan ASP.Net[4] dan database SQL Server[2]. Use case diagram untuk akses dokumen teks digambarkan pada Gambar 2. Sementara use case diagram untuk akses dokumen multimedia digambarkan pada Gambar 3.



Gambar 1. Front End Perpustakaan Digital

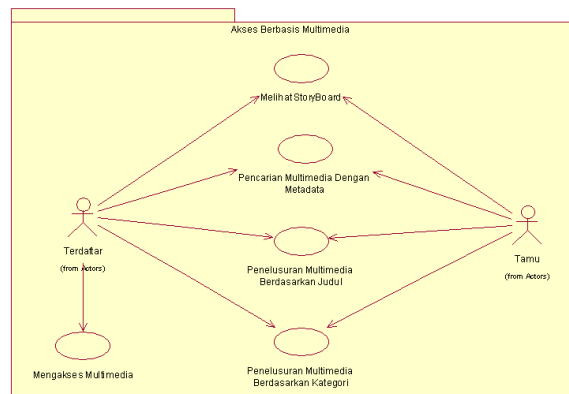
Pengguna yang dapat mengakses *front end* adalah pengguna terdaftar dan pengguna tamu. Pengguna terdaftar adalah pengguna yang harus terautentikasi terlebih dahulu. Pengguna tamu adalah pengguna yang tidak perlu diautentikasi atau dengan kata lain tidak melakukan login ke sistem.



Gambar 2. Use case diagram akses berbasis teks

Pada Gambar 2 dapat dilihat kalau pengguna terdaftar dan pengguna tamu dapat melihat abstrak dokumen teks, melakukan penelusuran teks berdasarkan kategori, melakukan penelusuran teks berdasarkan judul, melakukan pencarian sederhana dan pencarian teks dengan metadata. Hasil dari proses pencarian ini adalah daftar dokumen teks yang dicari. Untuk melihat isi dokumen secara keseluruhan, hanya dapat dilakukan oleh pengguna terdaftar melalui use case mengakses teks.

Pengaksesan dokumen multimedia juga dapat dilakukan oleh pengguna terdaftar dan pengguna tamu. Gambar 3 menunjukkan bahwa pengguna terdaftar dan pengguna tamu dapat melihat storyboard, pencarian multimedia dengan metadata, penelusuran multimedia berdasarkan judul dan penelusuran multimedia berdasarkan kategori. Isi dari dokumen multimedia hanya dapat dilihat secara keseluruhan oleh pengguna terdaftar.



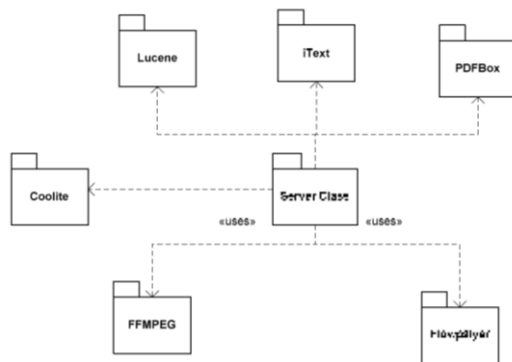
Gambar 3. Use case diagram akses berbasis multimedia

Untuk mendukung proses bisnis pencarian dokumen berbasis teks dan multimedia, digunakan beberapa pustaka sebagai berikut:

1. Lucene[1]. Lucene merupakan pustaka untuk *information retrieval* yang digunakan untuk mengimplementasikan mesin pencari dokumen berbasis teks. Lucene bisa melakukan indexing dan memungkinkan proses searching tanpa mempedulikan sumber data, format, ataupun bahasa pemrograman yang digunakan selama kita dapat mengkonversikannya dalam bentuk teks.
2. iText[6]. iText adalah pustaka untuk membuat dan memanipulasi file pdf. Dalam aplikasi ini, iText digunakan untuk memperoleh metadata dokumen teks dan memberi proteksi agar file pdf tidak bisa di *copy-paste*.
3. PDFBox[7]. Sama seperti iText, PDFBox merupakan pustaka untuk memanipulasi file PDF. Dalam aplikasi ini, PDFBox digunakan untuk mengekstrak teks dari file PDF.
4. FFMPEG[8]. FFMPEG adalah pustaka untuk konversi video dan audio. Dalam aplikasi ini, FFMPEG dimanfaatkan untuk membuat storyboard dengan cara membuat snapshot dari video pada frame tertentu.

5. Flowplayer[9]. Flowplayer adalah flash player yang digunakan untuk menampilkan dokumen multimedia
6. Coolite Toolkit[10]. Coolite adalah web control ASP.NET yang dilengkapi dengan dukungan AJAX. Coolite dibangun dengan menggunakan pustaka cross-browser JavaScript EXTJS. Penggunaan Coolite mempermudah pengembangan aplikasi website berbasis AJAX.

Hubungan logis antara aplikasi Perpustakaan Digital dengan pustaka-pustaka diatas tampak pada gambar 4.



Gambar 4. Diagram paket logis Perpustakaan Digital

Berikut adalah tahapan proses yang dilakukan di dalam sistem untuk memanipulasi dan menemukan koleksi digital yang ada didalam sistem.

#### a. Ekstraksi file PDF ke dalam bentuk teks

Karena sistem menggunakan search engine berbasis teks, maka file PDF harus diekstrak terlebih dahulu menjadi bentuk teks. Untuk melakukan hal tersebut, digunakan *library* untuk manipulasi PDF yaitu PDFBox[7].

#### b. Proses indexing dokumen berbasis teks

Proses *indexing* dan *searching* dalam sistem akan ditangani oleh *library* Lucene[1]. Dengan Lucene, kita bisa mendesain sendiri *index* yang akan kita buat sesuai dengan kebutuhan.

Sebagai penerapan dari konsep indexing, dibuat beberapa non-clustered index sebagai berikut:

```

CREATE NONCLUSTERED INDEX
nci_judul_teks ON
METADATA_TEKS (JUDUL_TEKS);

CREATE NONCLUSTERED INDEX
nci_kata_kunci_teks ON
METADATA_TEKS (KATA_KUNCI_TEKS);

CREATE NONCLUSTERED INDEX
nci_judul_multimedia ON
METADATA_MULTIMEDIA (JUDUL_MULTIMEDIA);

CREATE NONCLUSTERED INDEX
nci_kata_kunci_multimedia ON
METADATA_MULTIMEDIA (KATA_KUNCI_MULTIMEDIA);
  
```

Indeks dibuat untuk mendukung proses pencarian pada sejumlah besar teks dengan cepat.

#### c. Enkripsi dokumen berbasis teks

Tujuan melakukan enkripsi pada sistem ini adalah membuat konten file pdf tidak mudah di-copy-paste. Untuk melakukannya digunakan *library* manipulasi PDF lain yaitu iText[6]. Dalam aplikasi ini, iText digunakan untuk memperoleh metadata dokumen teks dan memberi proteksi agar konten file pdf tidak bisa di-copy-paste.

#### d. Proses pencarian berbasis konten untuk dokumen berbasis teks

Seperti yang telah disebutkan sebelumnya proses *searching* menggunakan *library* Lucene. Selain sangat cepat, Lucene juga bisa memberikan relevansi yang cukup tinggi antara *query* dan hasil pencarian.

Penelitian ini memanfaatkan indeks dalam melakukan pencarian dokumen. Pencarian tersebut dilakukan berdasarkan kata kunci yang dimasukkan oleh pengguna. Mesin pencari melakukan proses pencarian kata dalam *index* sehingga dapat ditemukan pada dokumen mana kata tersebut muncul. Selanjutnya, dokumen hasil pencarian yang telah diurutkan berdasarkan skor ditampilkan pada layar. Semakin besar skor suatu dokumen, semakin besar pula relevansi dokumen tersebut dengan kata kunci yang dimasukkan.

Selain itu, juga diperlukan dukungan pada *query single* dan *multiterm*, *query frase*, *wildcard*, *ranking* dan *sorting* hasil *searching* serta *syntax* yang mudah digunakan untuk memasukkan *query*.

**e. Proses ekstraksi frame sebagai Storyboard dokumen berbasis Video**

Untuk menangani dokumen berbasis video dilakukan proses ekstraksi frame. Untuk melakukan ekstraksi frame digunakan library FFmpeg[8]. Dengan FFmpeg kita bisa mengambil *frame* berdasarkan waktu. Misal, kita bisa mengambil *frame* pada menit ke-1, detik ke-200 dan lain sebagainya.

**f. Penanganan keamanan dengan hak akses**

Penanganan keamanan dilakukan dengan memberikan hak akses yang berbeda terhadap pengguna yang terdaftar dan yang tidak. Pengguna yang tidak terdaftar pada sistem hanya memiliki akses yang terbatas yaitu hanya bisa mengakses abstrak untuk dokumen teks dan *storyboard* untuk dokumen multimedia. Sementara pengguna terdaftar bisa mengakses dokumen secara penuh dan melakukan proses pengunduhan dokumen..

**5. Uji Coba dan Pembahasan Hasil**

Jenis uji coba dibedakan menjadi dua macam, yaitu uji coba fungsionalitas dan uji coba nonfungsionalitas. Uji coba fungsionalitas dilakukan untuk melihat apakah fungsi-fungsi dasar aplikasi berjalan sebagaimana mestinya. Sedangkan uji coba nonfungsionalitas dilakukan untuk melihat apakah performa aplikasi sesuai dengan yang diharapkan.

**5.1. Data Uji Coba**

Uji coba pencarian dokumen berbasis konten pada sistem ini dilakukan dengan memasukkan berbagai kata kunci sebagai kriteria pencarian. Dari proses tersebut didapatkan kondisi apakah pencarian dapat menghasilkan dokumen yang sesuai dengan kata kunci atau tidak. Dokumen hasil pencarian merupakan dokumen yang telah tersimpan dalam *server* sistem.

**5.2. Skenario dan Hasil Uji Coba**

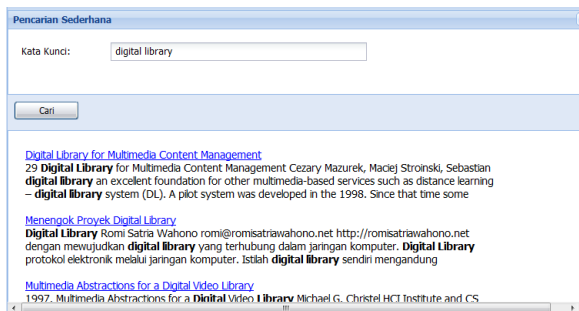
Uji coba dilakukan dengan 5 usecase. Uji coba pertama melakukan proses pencarian sederhana. Uji coba kedua melakukan proses pencarian dengan metadata. Uji coba ketiga untuk pengujian keamanan, uji coba keempat untuk pencarian dengan dan tanpa index. Uji coba kelima untuk melakukan uji coba relevansi kata kunci dengan hasil pencarian.

1. Proses pencarian sederhana dengan mesin pencari

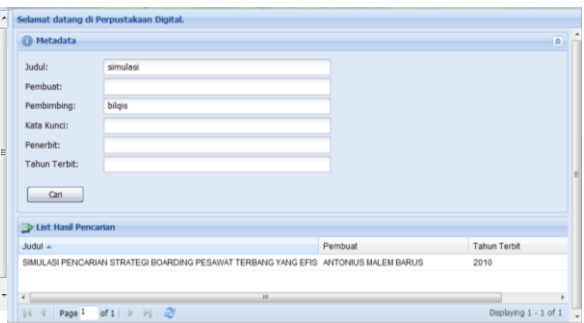
Tabel 1 Skenario proses pencarian sederhana

Tujuan	Menguji fungsi pencarian, melihat abstrak dan mengakses dokumen.
Pre-condition	Terkoneksi ke server Perpustakaan Digital
<i>Skenario 1</i>	<i>User login, melakukan pencarian, melihat abstrak dan mengakses dokumen.</i>
Input	Kata kunci
Output	User yang login dan telah terdaftar bisa melihat abstrak dan mengakses dokumen teks yang dicari.
Post-condition	- Pencarian bisa menghasilkan dokumen yang sesuai dengan kata kunci - Detail dokumen bisa muncul sesuai dokumen yang diakses - Dokumen bisa ditampilkan di web browser
<i>Skenario 2</i>	<i>User tidak login, melakukan pencarian, melihat abstrak dan mengakses dokumen.</i>
Input	Kata kunci
Output	Aplikasi memberikan pesan bahwa untuk mengakses dokumen, user harus login
Post-condition	Muncul pesan bahwa user harus login untuk mengakses dokumen.

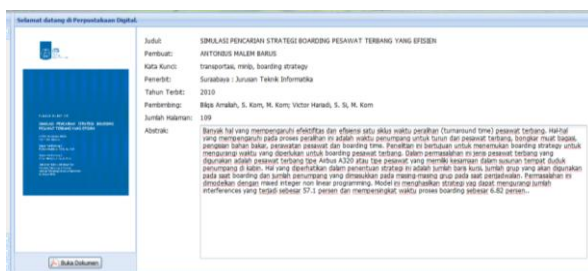
Gambar 5 menunjukkan hasil uji coba proses pencarian sederhana. Dengan memasukkan kata kunci yang dimaksud, sistem kemudian menampilkan daftar dokumen yang sesuai. Jika dokumen diklik maka akan muncul tampilan seperti gambar 6. Jika pengguna tamu atau pengguna yang tidak terdaftar menekan tombol Buka Dokumen maka akan muncul peringatan seperti pada Gambar 7. Namun jika yang membuka adalah pengguna terdaftar maka dokumen yang dimaksud akan terbuka dan dapat didownload. Ketika terbuka isi dokumen tidak dapat di copy paste ke dokumen lain.



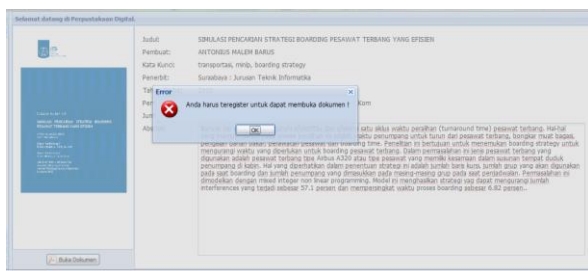
Gambar 5. Antarmuka setelah user memasukkan kata kunci dan menekan tombol cari



Gambar 8. Antarmuka daftar dokumen dengan hasil pencarian metadata



Gambar 6. Tampilan detail dokumen dan abstrak disertai gambar sampul dokumen



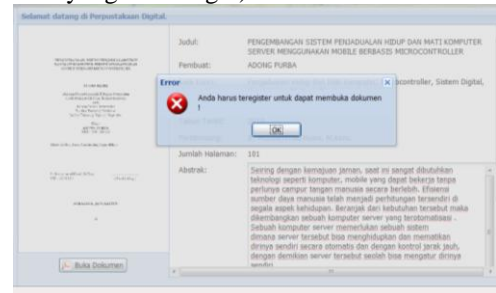
Gambar 7. Peringatan jika user yang belum login ingin mengakses dokumen

Gambar 8 merupakan tampilan untuk pencarian dokumen dengan meta data. Setelah pengguna memasukkan beberapa data metadata, seperti judul maka muncul daftar dokumen hasil pencarian.

3. Uji coba keamanan dengan pengaturan hak akses

Hak akses digunakan untuk menentukan apakah pengguna bisa mengakses halaman atau fitur tertentu. Parameter keberhasilan sistem menangani hak akses bisa diukur dari 2 skenario berikut ini:

a. Skenario 1 (pengaksesan dokumen oleh user yang tidak login)



Gambar 9. Tampilan pengaksesan dokumen oleh user yang tidak login

2. Proses pencarian dengan metadata  
Tabel 2 Skenario proses pencarian dengan metadata

Tujuan	Menguji fungsi pencarian pencarian dengan metadata.
Pre-condition	Terkoneksi ke server Perpustakaan Digital
Skenario	<b>User login, melakukan pencarian.</b>
Input	Metadata dokumen
Output	Muncul dokumen hasil pencarian yang sesuai.
Post-condition	- Pencarian bisa menghasilkan dokumen yang sesuai dengan metadata yang dimasukkan

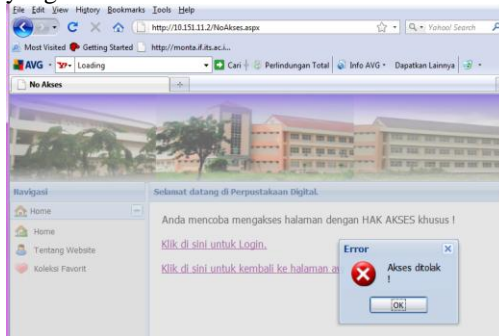
Kebutuhan keamanan menyatakan bahwa pengguna yang tidak login atau tidak terdaftar tidak boleh mengakses dokumen pada sistem. Pada gambar 9 di atas terlihat bahwa sistem berhasil membatasi akses user yang tidak login. Oleh sebab itu, diambil kesimpulan bahwa sistem berhasil memberikan perlindungan dokumen dengan menggunakan hak akses.

b. Skenario 2 (pengaksesan halaman admin oleh pengguna dengan cara mengetik URL)

Untuk menangani jika ada pengguna yang hendak mengakses halaman admin



dengan cara menyetik URL, karena dimungkinkan pengguna tersebut mengetahui URL dari halaman admin, maka sistem akan mengecek *session user*. Jika ternyata pengguna tidak mempunyai *session* atau tidak login, maka sistem akan memberikan peringatan bahwa pengguna tidak memiliki akses seperti yang tampak pada gambar 10. Hal ini membuktikan bahwa sistem mampu menangani keamanan halaman web dari pengguna yang tidak memiliki hak akses.



Gambar 10. Akses ke halaman admin ditolak

#### 4. Perbandingan pencarian dengan dan tanpa index

Untuk membandingkan kecepatan proses pencarian dengan menggunakan proses pencarian dengan menggunakan *indexing* dan tanpa dilakukan *indexing*, dibuatlah sebuah program dengan algoritma yang mirip dengan algoritma mesin pencari yang digunakan sistem.

Program ini akan menerima string input sebagai *query* yang akan dibandingkan dengan isi dokumen. Program akan menghitung frekuensi kemunculan *string* input pada setiap dokumen. Program tidak melakukan proses *sorting* berdasarkan *query* tetapi hanya menampilkan hasil penghitungan saja. Dokumen yang diakses program merupakan dokumen berekstensi txt. Agar proses perbandingan adil maka *running time* ketika program dan sistem telah stabil, bukan saat pertama kali program dan sistem di-load. Tabel 3 menunjukkan hasil perbandingan waktu yang diperlukan setelah melakukan uji coba.

Dengan asumsi *trend* yang ada pada tabel 3 terus bertahan seiring dengan bertambahnya jumlah dokumen, maka dapat diambil kesimpulan bahwa penerapan konsep *indexing* mampu mempercepat proses

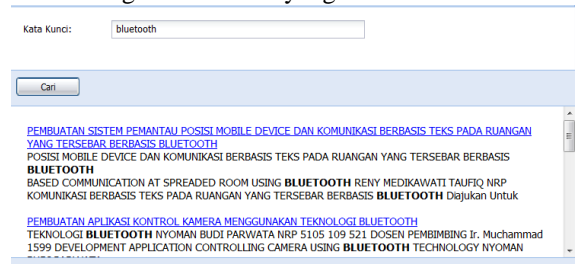
pencarian karena hanya membutuhkan ½ dari waktu yang dibutuhkan pencarian normal. Semakin besar jumlah dokumen yang dicari, maka semakin cepat pula proses pencarian bila dibandingkan dengan pencarian normal.

Tabel 3. Perbandingan lama running terhadap jumlah dokumen yang terdapat pada server

Jumlah dokumen	Running program (detik)	Running sistem (detik)
10	0,2656250	0,125
20	0,5781250	0,22

#### 5. Relevansi kata kunci dan dokumen hasil pencarian

Untuk menguji relevansi antara kata kunci yang dimasukkan dan dokumen hasil pencarian maka dilakukan uji coba dengan suatu kata kunci, misalnya *Bluetooth*. Diharapkan dokumen hasil pencarian akan sesuai dengan kata kunci yang dimasukkan.



Gambar 11 Uji coba dengan kata kunci *bluetooth*

Hasil uji coba dengan kata kunci *bluetooth* dapat menampilkan seluruh dokumen yang berhubungan dengan *bluetooth* seperti yang tampak pada gambar 11. Selain itu, dokumen hasil pencarian juga telah diurutkan berdasarkan skor. Semakin besar skor suatu dokumen, semakin besar pula relevansi dokumen tersebut dengan kata kunci yang dimasukkan.

Dari hasil uji coba di atas, dapat diambil kesimpulan bahwa pencarian dokumen dengan memanfaatkan mesin pencari bisa mempermudah untuk mengetahui relevansi antara dokumen hasil pencarian dan kata kunci. Hal ini disebabkan karena dokumen hasil pencarian telah diurutkan berdasarkan skor. Semakin besar skor suatu dokumen, semakin besar pula relevansi dokumen tersebut dengan kata kunci yang dimasukkan.

#### 6. Kesimpulan dan Saran



Dari hasil pengembangan dan pengujian didapatkan beberapa kesimpulan sebagai berikut:

1. Proteksi dokumen teks dapat diimplementasikan dengan menggunakan hak akses dan enkripsi.
2. Pencarian dokumen dilakukan menggunakan konsep *indexing*.
3. Pemanfaatan mesin pencari dengan pengindekan dapat mempercepat proses pencarian karena waktu yang dibutuhkan mesin pencari hanya ½ dari waktu yang dibutuhkan pencarian normal.
4. Pada pemanfaatan mesin pencari, semakin besar jumlah dokumen yang dicari, maka semakin cepat pula proses pencarian bila dibandingkan dengan pencarian normal.
5. Pencarian dokumen teks berbasis konten dengan memanfaatkan mesin pencari dapat memudahkan untuk mengetahui relevansi antara dokumen hasil pencarian dan kata kunci. Hal ini disebabkan karena dokumen hasil pencarian telah diurutkan berdasarkan skor yang menyatakan relevansi dokumen tersebut dengan kata kunci.

Agar sistem dapat berfungsi secara lebih optimal, sistem ini perlu dilakukan peningkatan tingkat keamanan dengan cara menerapkan proteksi yang lebih baik.

Misalkan, dokumen multimedia memerlukan proteksi khusus karena pada dasarnya file yang diakses telah masuk dalam penyimpanan *temporary* komputer *client* sehingga file bisa di-unduh walaupun tanpa login ke sistem.

## VI. Daftar Pustaka

- [1] Gospodnetic, O. dan Hatcher, E. 2004. **Lucene in Action**. Manning.
- [2] Jorden, Joseph. 2007. *SQL Server 2005 DBA Street Smarts*. Indiana. Wiley Publishing.
- [3] Lowagie, Bruno. 2007. **iText in Action**. Manning.
- [4] MacDonald, M., dan Szpuszta, M. 2007. **Pro ASP.NET 3.5 in C# 2008**. Apress.
- [5] Marchionini, G., dan Geisler, G. 2002. "The Open Video Digital Library". **D-Lib Magazine** Vol 8, Desember, 12.
- [6]. iText, <http://itextpdf.com>, diakses pada tanggal 2 Oktober 2009.
- [7]. PDFBox, [http://www.pdfbox.org/userguide/text\\_extraction.html](http://www.pdfbox.org/userguide/text_extraction.html) diakses pada tanggal 3 September 2009.
- [8]. FFMPEG. <http://www.ffmpeg.org/>, diakses pada 9 September 2009.
- [9]. Flowplayer, <http://flowplayer.org/>, diakses pada tanggal 2 Oktober 2010.
- [10]. Coolite Toolkit. <http://www.coolite.com/>, diakses pada tanggal 9 September 2010.