

RANCANG BANGUN APLIKASI MENINGKATKAN TINGKAT RELEVANSI PENCARIAN TUGAS AKHIR (STUDI KASUS PADA PERPUSTAKAAN INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA)

Ikhsan Pratama Wahyudi¹⁾ Tutut Wuriyanto²⁾ Sulistiwati³⁾

Program Studi/Jurusan Sistem Informasi

STMIK STIKOM Surabaya

Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email : 1)ikhsan.p.wahyudi@gmail.com, 2)tutut@stikom.edu, 3)sulist@stikom.edu

Abstract: The library is a vital part of an educational institution like university, especially for the Institute of Business and Information Stikom Surabaya. The availability of the final task collection became the main reason users to use library facilities. But handling in the search collections not implement the concept of Information Retrieval yet. Information retrieval used to produce information or documents which relevant to keyword from source by the amount that are numerous and is not structured as a collection of final task literatures .To solve these problems, the necessary repairs final task search facility by implementing the concept of Information Retrieval. The application of Information Retrieval on an abstract document is expected to produce a collection of final task literatures which is relevant to the search. From the testing, the application capable of producing the collection of final tasks more than by search through query when given keywords with combination affix (prefix and suffix).

Keywords: Library, Information Retrieval, Cosine Similarity

Perpustakaan merupakan bagian vital dalam suatu institusi pendidikan, khususnya bagi Institut Bisnis dan Informatika Stikom Surabaya. Perpustakaan menyediakan berbagai macam bentuk dan jenis literatur yang dibutuhkan bagi pengguna perpustakaan. Banyaknya data dan informasi yang tersedia menjadi alasan utama pengguna untuk memanfaatkan fasilitas perpustakaan.

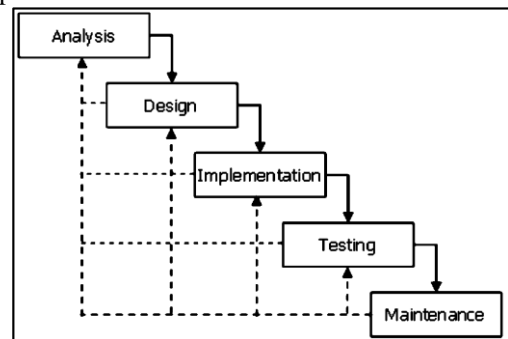
Tugas akhir mahasiswa merupakan salah satu bentuk koleksi yang terdapat dalam perpustakaan. Dengan jumlah literatur tugas akhir sebanyak 2.754 buah judul tugas akhir (sumber: <http://digilib.stikom.edu>, diakses 20 Maret 2015), perpustakaan membutuhkan fasilitas pencarian yang mampu menghasilkan pencarian sesuai dengan keinginan penggunanya. Semakin tinggi tingkat kecocokan atau relevansi hasil pencarian, semakin baik suatu aplikasi pencarian dokumen tugas akhir.

Namun dalam fasilitas pencarian koleksi tugas akhir saat ini, perpustakaan digital Stikom Surabaya belum menerapkan konsep *Information Retrieval* (IR). Dengan menerapkan langkah-langkah *Information Retrieval* seperti *indexing*, *pembobotan*, dan kemiripan kosinus memungkinkan menghasilkan pencarian yang relevan.

Pencarian dengan menerapkan konsep *Information Retrieval* menitikberatkan pada dokumen abstrak tugas akhir. Semakin banyak pembedaharaan kata dalam dokumen abstrak, semakin baik hasil dari proses *Information Retrieval* sehingga tingkat relevansi dapat meningkat jika dibandingkan pencarian melalui *query*.

METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian tugas akhir ini menerapkan konsep pengembangan sistem *System Development Life Cycle* (SDLC) dengan model *Waterfall* seperti pada Gambar 1.



Gambar 1. Model SDLC *Waterfall*

Information Retrieval

Konsep *Information Retrieval* merupakan konsep pencarian informasi dari sumber yang tidak terstruktur. Menurut Manning (2009), *Information Retrieval* adalah menemukan dokumen dari domain yang tidak terstruktur (bentuk teks) berupa kebutuhan informasi dari jumlah koleksi yang sangat banyak. Perkembangan jaringan dan selalu bertambahnya jumlah data yang tersedia, mengharuskan penerapan *Information Retrieval* dalam segala aspek. Salah satunya adalah perpustakaan. Komponen-komponen dalam *Information Retrieval* adalah *indexing*, pembobotan, dan perhitungan kemiripan kosinus.

Indexing

Indexing menurut Bawono (2012) merupakan pembentukan pola pengenalan dari proses pengumpulan kata-kata dalam dokumen digital. Tahap-tahap *indexing* yaitu:

1. *Tokenization*, yaitu pemecahan kalimat menjadi pengenalan perkata. Kalimat “saya belajar sistem informasi” dipecah menjadi “saya”, “belajar”, “sistem”, dan “informasi”.
2. *Stopword* merupakan tahap penghapusan kata-kata yang bukan bagian dari kata kunci, seperti kata hubung atau tanda baca untuk mengurangi waktu durasi *indexing* dan mengurangi *noise* dalam perhitungan kemiripan. Contoh *stopword* adalah “juga”, “yaitu”, “dan”, “kenapa”, “jika”, dan lain sebagainya.
3. *Stemming* adalah perubahan bentuk dari kata-kata pada suatu dokumen menjadi bentuk kata dasar (Agusta: 2009). Metode *stemming* yang digunakan dalam penelitian ini adalah metode *stemming* Nazief & Adriani dimana metode ini merupakan pengembangan metode *stemming* Algoritma Porter. Apabila metode Porter untuk bahasa Inggris, maka metode Nazief & Adriani untuk bahasa Indonesia.

Pembobotan

Pembobotan menurut Manning (2009) adalah pemberian nilai terhadap istilah (*term*) yang terdapat dalam dokumen. Semakin sering suatu *term* muncul dalam suatu dokumen, semakin tinggi bobot dokumen tersebut. Sebaliknya, semakin banyak dokumen yang mengandung sebuah *term*, semakin kecil nilai bobot *term* tersebut. Metode ini dinamakan

pembobotan *Term Frequency/Inverted Document Frequency* (TF/IDF). Rumus pembobotan adalah sebagai berikut:

$$TF_{d,t} = f_{d,t} \dots\dots\dots (1)$$

$$IDF_t = \log\left(\frac{N}{df_t}\right) \dots\dots\dots (2)$$

$$W_{d,t} = TF_{d,t} * IDF_t \dots\dots\dots (3)$$

Dimana:

- d = dokumen ke-d.
- t = *term* (istilah) ke-t dari dokumen.
- f(d,t) = jumlah kemunculan *term* atau istilah t pada dokumen d.
- N = jumlah seluruh dokumen di dalam *Database* ditambah kata kunci pencarian.
- df_t = jumlah dokumen yang memiliki istilah t.

Cosine Similarity

Perhitungan kemiripan kosinus (*cosine similarity*) menurut Bawono (2012) menggunakan konsep trigonometri, yaitu mengukur kemiripan antara dua objek berdasarkan sudut yang dibentuk oleh kedua objek tersebut. Semakin besar nilai kosinus (mendekati 1), maka semakin mirip objek yang dibandingkan. Dalam hal ini kata kunci dengan kata dalam dokumen yang tersedia.

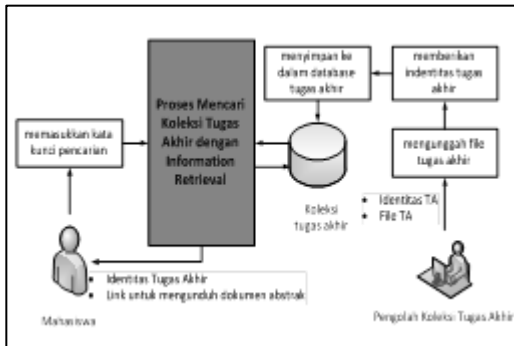
$$similarity(\vec{d}_j, \vec{q}) = \frac{\vec{d}_j * \vec{q}}{|\vec{d}_j| * |\vec{q}|} = \frac{\sum_{i=1}^t (w_{ij} * w_{iq})}{\sqrt{\sum_{i=1}^t (w_{ij})^2 * \sum_{i=1}^t (w_{iq})^2}} \dots\dots\dots (4)$$

Dimana:

- d = dokumen
- t = *term* (istilah)
- q = *query*
- W_{ij} = TF-IDF kata ke-i dari dokumen ke-j
- W_{iq} = TF-IDF kata ke-i dari *query*

Gambaran Umum Aplikasi

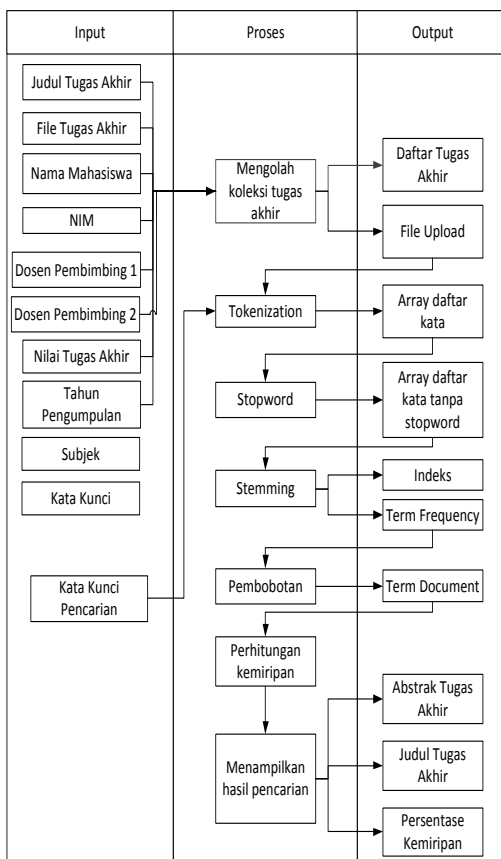
Untuk memahami proses yang akan dijalankan oleh aplikasi diperlukan sebuah gambaran umum aplikasi yang akan dibangun. Terdapat dua pengguna dengan proses inti yaitu pencarian koleksi tugas akhir dengan *information retrieval*. Gambaran umum aplikasi dapat dilihat pada Gambar 2.



Gambar 2. Gambaran Umum Aplikasi

Diagram Blok

Diagram blok pada Gambar 3 menggambarkan rancangan kebutuhan aplikasi dengan mengetahui masukan (*input*) dibutuhkan, proses-proses yang berjalan, dan keluaran (*output*) yang diinginkan oleh pengguna.



Gambar 3. Diagram Blok

Pada diagram blok, terdapat beberapa *input*, *proses*, dan *output* antara lain:

1. *Input*

- a. Judul Tugas Akhir.

- b. *File* Tugas Akhir, merupakan dokumen tugas akhir dengan format DOC, DOCX, atau format PDF yang tidak terproteksi.
- c. Nama Mahasiswa merupakan mahasiswa yang menulis tugas akhir.
- d. NIM, merupakan Nomor Induk Mahasiswa penulis tugas akhir.
- e. Dosen Pembimbing 1 dan 2, merupakan dosen yang menjadi pembimbing mahasiswa penulis tugas akhir.
- f. Nilai Tugas Akhir, merupakan nilai yang diperoleh mahasiswa penulis tugas akhir.
- g. Tahun Pengumpulan, menunjukkan tahun saat tugas akhir selesai dikerjakan.
- h. Kata kunci tugas akhir merupakan *keywords* yang disediakan berdasarkan abstrak.
- i. Kata Kunci Pencarian, merupakan daftar kata-kata yang dimasukkan mahasiswa untuk mencari judul tugas akhir yang diinginkan.

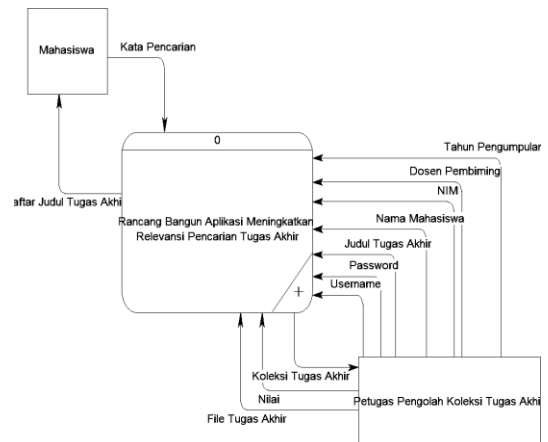
2. *Proses*

- a. Mengolah koleksi tugas akhir, merupakan proses mengunggah file tugas akhir dan data-data pendukung ke dalam basis data koleksi.
- b. Proses *tokenization*, merupakan proses mengubah rangkaian kalimat dalam abstrak tugas akhir menjadi potongan kata. Dalam proses ini terjadi perubahan menjadi huruf kecil dan membuang karakter-karakter yang bukan karakter *alphanumeric*.
- c. Proses *stopword*, merupakan proses membuang kata-kata dalam *array* yang bukan kata kunci atau membuang kata-kata pelengkap.
- d. Proses *stemming*, merupakan proses perubahan kata-kata dalam *array* menjadi kata dasar (tanpa imbuhan), atau menganggap suatu kata menjadi kata dasar apabila merupakan kata serapan atau istilah bahasa asing.
- e. Proses pembobotan, merupakan pemberian nilai berdasarkan frekuensi kemunculan menggunakan rumus TF-IDF.
- f. Proses perhitungan kemiripan, merupakan perhitungan kemiripan dokumen dengan kata kunci pencarian

- dengan menggunakan rumus *cosine similarity*.
- g. Proses menampilkan hasil pencarian, dengan mengurutkan judul tugas akhir dari tingkat kemiripan tertinggi yang ditampilkan dalam bentuk tautan untuk mengunduh dan melihat detail dari judul tugas akhir.
3. *Output*
- a. Daftar Tugas Akhir, merupakan daftar tugas akhir yang telah diunggah oleh petugas perpustakaan.
 - b. *File Upload*, merupakan daftar *file* dokumen yang telah diunggah. *File* yang digunakan sebagai masukan proses *information retrieval* adalah *file* dokumen abstrak.
 - c. *Array* daftar kata, merupakan hasil pemecahan kalimat menjadi kata yang disimpan dalam bentuk *array*.
 - d. *Array* daftar kata tanpa *stopword*, merupakan daftar kata yang sudah tidak mengandung kata-kata selain kata utama.
 - e. Indeks, merupakan daftar istilah yang terdapat dalam kesemua dokumen abstrak.
 - f. *Term frequency*, merupakan daftar istilah yang telah diberikan jumlah frekuensi kemunculannya.
 - g. *Term document*, merupakan identitas dari istilah berada pada dokumen tugas akhir yang terkait. Selain itu pada *term document* terdapat bobot dari masing-masing istilah
 - h. Abstrak Tugas Akhir, merupakan abstrak yang ditampilkan setelah melalui proses *information retrieval*.
 - i. Judul Tugas Akhir, merupakan judul tugas akhir yang sesuai dengan hasil *information retrieval*.
 - j. Persentase Kemiripan, merupakan hasil perhitungan kemiripan dari proses *information retrieval*.

Diagram Konteks

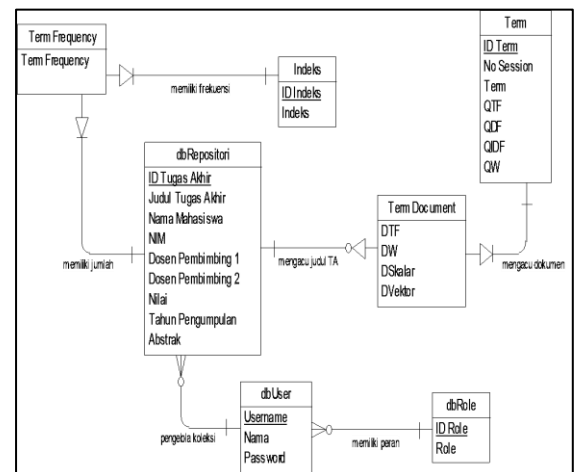
Diagram konteks atau *context diagram* pada Gambar 4 menggambarkan konsep basis data yang akan dibuat. Terdapat aliran data-data yang digunakan dalam pembuatan aplikasi ini.



Gambar 4. Diagram Konteks

Conceptual Data Model (CDM)

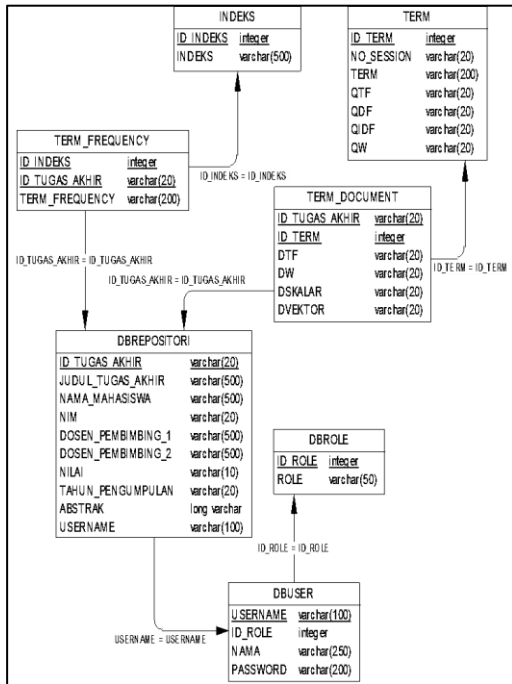
Conceptual Data Model (CDM) pada Gambar 5 merupakan gambaran struktur konsep tabel-tabel dalam basis data yang akan digunakan. Dalam CDM terdapat penamaan tabel, kolom, dan relasi-relasi yang terjadi di dalamnya.



Gambar 5. CDM

Physical Data Model (PDM)

Physical Data Model (PDM) pada Gambar 6 merupakan gambaran nyata mengenai tabel dan kolom apa saja yang pasti akan dibuat dan digunakan pada aplikasi ini.



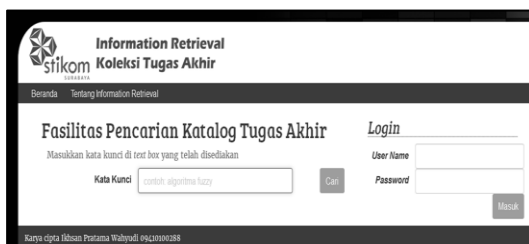
Gambar 6. PDM

HASIL DAN PEMBAHASAN

Aplikasi pencarian koleksi tugas akhir dengan penerapan *Information Retrieval* ini menggunakan basis *web*. Sehingga penggunaannya menggunakan peramban atau *browser* yang mendukung.

Halaman Utama

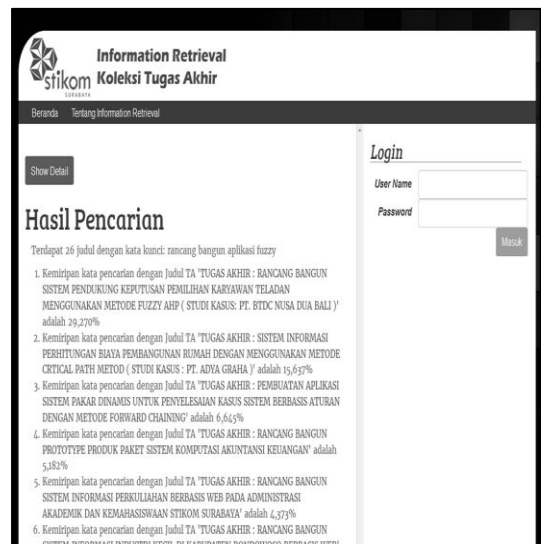
Halaman utama pada Gambar 7 berfungsi untuk memberikan informasi bahwa pengguna telah memasukkan alamat dengan tepat, berfungsi untuk memasukkan kata pencarian, dan untuk melakukan proses otentikasi pengguna. Pengguna dapat langsung melakukan pencarian dengan memasukkan kata kunci tanpa melalui otentikasi pengguna. Terdapat peringatan-peringatan yang membantu pengguna jika terjadi kesalahan. Salah satunya adalah warna *textbox* akan berwarna merah apabila pengguna tidak memasukkan karakter *keyboard*.



Gambar 7. Halaman Utama

Halaman Hasil Pencarian

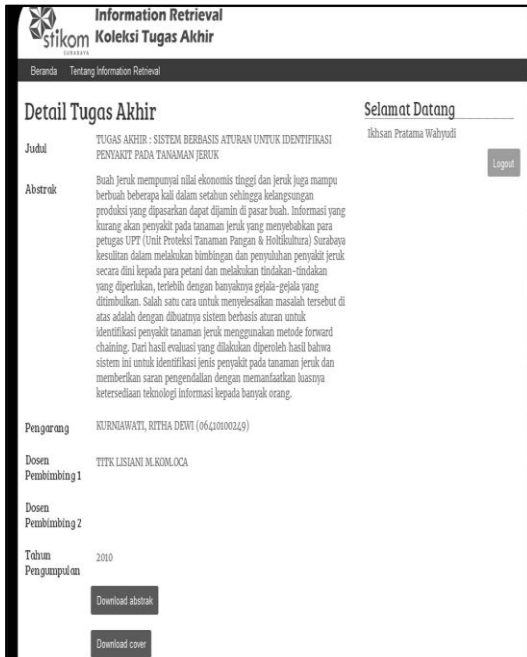
Halaman hasil pencarian pada Gambar 8 akan muncul jika pengguna memasukkan kata kunci dan menekan tombol “Cari” pada halaman utama. Halaman hasil pencarian akan menampilkan daftar judul koleksi tugas akhir yang diurutkan dari nilai kemiripan kosinus tertinggi, serta memberikan pilihan kepada pengguna apakah ingin melihat hasil perhitungan *Information Retrieval* atau tidak dengan menekan “Show Detail”. Setiap daftar judul tugas akhir diberikan *link* untuk menampilkan detail dari tugas akhir.



Gambar 8. Halaman Hasil Pencarian

Halaman Detail Tugas Akhir

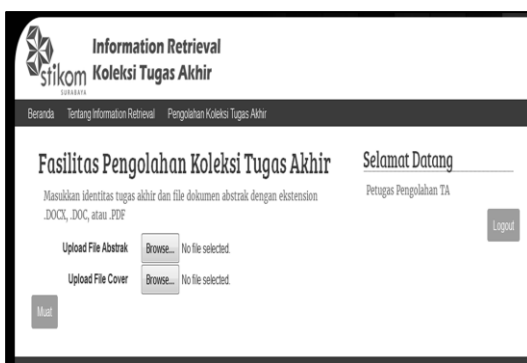
Halaman detail tugas akhir pada Gambar 9 akan muncul jika pengguna menekan salah satu judul tugas akhir pada halaman hasil pencarian. Halaman detail menampilkan judul, abstrak, nama pengarang, tahun pengumpulan, nama dosen pembimbing 1 dan pembimbing 2. Serta fasilitas untuk mengunduh *file* tugas akhir jika pengguna telah melakukan otentikasi sebelumnya.



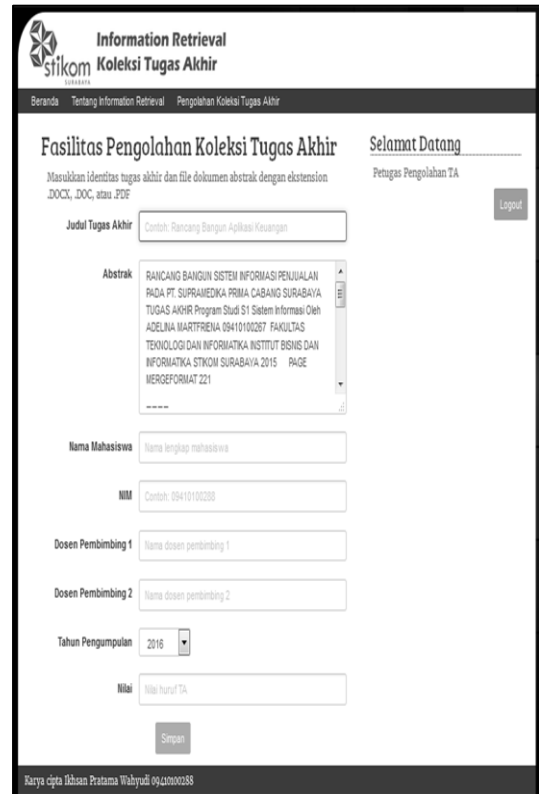
Gambar 9. Halaman Detail Tugas Akhir

Halaman Pengolahan Koleksi Tugas Akhir

Menu untuk mengakses halaman ini akan muncul pada halaman utama jika pengguna melakukan otentikasi sebagai petugas pengolahan koleksi tugas akhir. Pada halaman pengolahan sesuai pada Gambar 10 dan Gambar 11, terdapat fasilitas pengunggahan file tugas akhir seperti abstrak. Pengunggahan ini dimaksudkan agar pengisian *text area* abstrak terisi secara otomatis tanpa harus melakukan *copy-paste* sebelumnya. Disertai juga pengisian identitas-identitas tugas akhir lainnya.



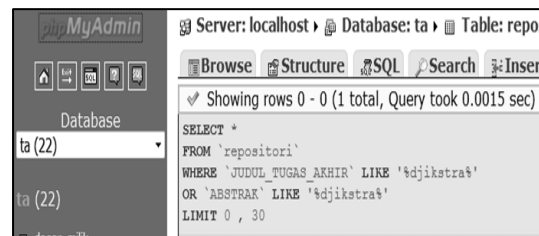
Gambar 10. Halaman Pengunggahan File Tugas Akhir



Gambar 11. Halaman Pengolahan Koleksi Tugas Akhir

Pengujian Peningkatan Relevansi

Untuk mengetahui peningkatan relevansi hasil pencarian, dilakukan perbandingan pencarian melalui *query* dan melalui aplikasi. Pencarian melalui *query* dilakukan pada *Database Management System* (DBMS) phpMyAdmin seperti pada Gambar 12 dimana penyimpanan basis data aplikasi menggunakan DBMS ini. *Query* yang diberikan merupakan *query like* "%...%" untuk menghasilkan pencarian yang diharapkan sesuai dengan potongan kata kunci pencarian. Hasil perbandingan pencarian dapat dilihat pada Tabel 1.



Gambar 12 Pencarian Melalui Query

Tabel 1. Hasil Perbandingan Pencarian

Kata Kunci	Hasil Pencarian Pada Query	Hasil Pencarian Pada Aplikasi
“djikstra”	1 judul	1 judul
“AHP”	2 judul	2 judul
“rute”	2 judul	2 judul
“forward”	5 judul	5 judul
“prototype”	1 judul	1 judul
“fuzzy”	1 judul	1 judul
“algoritma djikstra”	4 judul	4 judul
“stikom surabaya”	16 judul	16 judul
“rekam medis”	3 judul	3 judul
“memilih”	Tidak ada	6 judul
“terintegrasi”	5 judul	5 judul
“diuji”	Tidak ada	8 judul
“cadangan”	1 judul	1 judul
“betapapun”	1 judul	1 judul
“pemilihan”	5 judul	6 judul
“penentuan”	4 judul	12 judul
“meramalkan”	Tidak ada	1 judul
“meramalkan penentuan”	4 judul	13 judul
“penentuan meramalkan”	4 judul	13 judul
“penilaian perusahaan”	15 judul	15 judul
“perusahaan penilaian”	15 judul	15 judul
“pencatan keuangan”	8 judul	11 judul
“keuangan pencatatan”	8 judul	11 judul
“karyawan fuzzy AHP”	8 judul	8 judul
“penentuan rute terpendek”	5 judul	13 judul
“pemilihan pengadaan barang”	10 judul	11 judul
“aplikasi peramalan kebutuhan”	20 judul	30 judul
“metode untuk meramalkan pengadaan barang yang	45 judul	25 judul

Kata Kunci	Hasil Pencarian Pada Query	Hasil Pencarian Pada Aplikasi
tidak menentu”		
“fuzzy fuzzy”	1 judul	1 judul
“diuji diuji”	Tidak ada	8 judul

Berdasarkan hasil pengujian relevansi baik melalui *query* maupun melalui aplikasi, dapat dirangkum dalam menjadi sebuah tabel seperti pada Tabel 2 mengenai rangkuman perbandingan tingkat relevansi. Perbandingan yang diamati adalah jumlah hasil pencarian pada aplikasi terhadap pencarian melalui *query*. Setiap perubahan, baik meningkat maupun menurun, atau tidak ada perbedaan hasil dirangkum untuk mengetahui penyebab terjadinya hal tersebut. Tidak ada klasifikasi khusus mengenai peningkatan maupun penurunan tingkat relevansi. Hasil pencarian melalui aplikasi lebih banyak 1 (satu) judul akan dinilai sebagai peningkatan. Begitu pula dengan penurunan tingkat relevansi.

Tabel 2. Rangkuman Perbandingan Tingkat Relevansi Hasil Pencarian

No.	Jenis Kata Kunci Pencarian	Hasil
1.	1 kata kunci dengan bentuk kata dasar.	Tidak ada perbedaan hasil.
2.	1 kata kunci dengan bentuk kata dasar bahasa asing.	Tidak ada perbedaan hasil.
3.	2 kata kunci dengan bentuk kata dasar.	Tidak ada perbedaan hasil.
4.	1 kata kunci dengan bentuk imbuhan awalan.	Terjadi peningkatan hasil pencarian pada aplikasi.
5.	1 kata kunci dengan bentuk imbuhan akhiran.	Tidak ada perbedaan hasil.
6.	1 kata kunci dengan bentuk imbuhan kombinasi.	Terjadi peningkatan hasil pencarian pada aplikasi.
7.	2 kata kunci dengan bentuk	Terjadi peningkatan

No.	Jenis Kata Kunci Pencarian	Hasil
	imbunan kombinasi.	hasil pencarian pada aplikasi.
8.	3 kata kunci dengan bentuk kata dasar.	Tidak ada perbedaan hasil
9.	3 kata kunci dengan bentuk gabungan kata dasar dan kata berimbunan.	Terjadi peningkatan hasil pencarian pada aplikasi.
10.	Berupa kalimat umum yang terdapat <i>stopword</i> .	Terjadi penurunan hasil pencarian pada aplikasi.

Berdasarkan rangkuman pada Tabel 2, dapat dilihat bahwa:

1. Tidak terjadi perbedaan hasil antara pencarian melalui aplikasi maupun melalui *query* apabila kata kunci pencarian hanya berupa kata dasar atau hanya berupa kata dasar serapan bahasa asing. Baik kata kunci yang terdiri atas 1 (satu) kata dasar maupun lebih dari 1 (satu) kata dasar. Hal ini disebabkan aplikasi tidak menemukan bentuk kata berimbunan dari kata dasar yang dimaksud. Serta pencarian melalui *query* juga hanya menemukan kata yang tepat sesuai dengan kata dasar dasar.
2. Tidak terjadi perbedaan hasil antara pencarian melalui aplikasi maupun melalui *query* ketika diberikan kata kunci berupa kata dengan imbuhan akhiran. Hal ini disebabkan aplikasi tidak menemukan bentuk kata dasar atau bentuk kata imbuhan lain pada basis data repositori. Dan pencarian melalui *query* juga hanya menemukan bentuk kata yang tepat sesuai dengan kata kunci dengan imbuhan akhiran tersebut.
3. Terjadi peningkatan hasil pada pencarian melalui aplikasi setiap menggunakan kata kunci berupa kata berimbunan awalan atau imbuhan kombinasi (awalan dan akhiran). Hal ini dikarenakan aplikasi mampu mencari bentuk dasar dari kata kunci tersebut dan mencari bentuk imbuhan lainnya dengan kata dasar yang sama dan menampilkan ke dalam hasil pencarian. Sehingga hasil pencarian

menjadi lebih banyak jika dibandingkan melalui *query* yang hanya mencari bentuk kata yang tepat sesuai dengan kata kunci.

4. Terjadi peningkatan hasil pencarian dengan menggunakan aplikasi pencarian dengan mengkombinasikan kata kunci dengan bentuk kata dasar dan kata berimbunan kombinasi. Hal ini disebabkan dengan alasan yang sama dengan melakukan pencarian menggunakan kata kunci dengan imbuhan kombinasi.
5. Terjadi penurunan hasil pencarian dengan menggunakan aplikasi jika dibandingkan dengan menggunakan *query* jika kata kunci berupa kalimat umum yang mengandung kata *stopword*. Hal ini disebabkan aplikasi melakukan tahapan penghapusan kata *stopword* tersebut untuk mempersingkat proses *Information Retrieval* dan mengurangi *noise* perhitungan kemiripan kosinus. Sedangkan pencarian melalui *query* mencari kesemua rangkaian kata pada kata kunci. Sehingga hasil pencarian melalui *query* menjadi lebih banyak.

SIMPULAN

Berdasarkan uji coba, dapat ditarik kesimpulan bahwa:

1. Aplikasi mampu melakukan pencarian koleksi tugas akhir dengan menerapkan konsep *Information Retrieval*.
2. Aplikasi mampu meningkatkan hasil pencarian yang relevan dengan kata kunci yang berupa kata berimbunan kombinasi awalan dan akhiran.
3. Tidak terjadi perbedaan hasil antara pencarian melalui aplikasi dan melalui *query* pencarian jika kata kunci berupa kata dasar, kata dasar serapan bahasa asing, maupun kata dengan imbuhan akhiran.
4. Terjadi penurunan tingkat relevansi pencarian pada aplikasi dengan diberikan kata kunci berupa kalimat yang mengandung kata *stopword* jika dibandingkan pencarian melalui *query*.

RUJUKAN

- Agusta, Ledy. 2009. *Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani Untuk Stemming Dokumen Teks Bahasa Indonesia*.

- Makalah pada Konferensi Nasional Sistem dan Informatika, Bali.
- Arsip Nasional Republik Indonesia. 2010. *Prosedur Tetap Nomor 05 Tahun 2010 Tentang Layanan Perpustakaan*. Tersedia:
<http://anri.go.id/assets/download/22Protap%20No%2005%20Tahun%202010%20tentang%20Layanan%20Perpustakaan.pdf>. Diakses 8 April 2015.
- Bassil, Youssef. 2012. *A Simulation Model for the Waterfall Software Development Life Cycle*. *International Journal of Engineering & Technology (online)*, Vol. 2, No. 5.
<http://arxiv.org/ftp/arxiv/papers/1205/1205.6904.pdf>. Diakses 8 September 2015.
- Bawono, Sahirul Alim Tri. 2012. *Information Retrieval Meningkatkan Pencarian Data Yang Relevan*. Yogyakarta: Universitas Gadjah Mada.
- Elmasri, Ramez dan Navathe, Shamkant B. 2003. *Fundamentals of Database Systems 4th ed*. New York: Pearson.
- Herdiyeni, Yeni. 2008. *Search Engine – Text Retrieval dan Image Retrieval*. Tersedia:
http://cs.ipb.ac.id/~yeni/files/artikel_008%20-%20Search%20Engine.pdf. Diakses 29 September 2015.
- Kadir, Abdul. 2008. *Dasar Pemrograman Web Dinamis Menggunakan PHP Edisi Revisi*. Yogyakarta: ANDI.
- Manning, Christopher D., et.al. 2009. *An Introduction to Information Retrieval – Online Edition*. Cambridge: Cambridge University Press.
- Sardi, Irawan. 2004. *Manajemen, Desain, dan Pengembangan Situs Web Dengan Macromedia Dreamweaver Dan Adobe Photoshop 7.0*. Jakarta: Elex Media Komputindo.
- World Wide Web Consortium (W3C). *HTML, CSS, JavaScript, and PHP*. Tersedia:
<http://www.w3.org/standards>. Diakses: 19 Maret 2015.