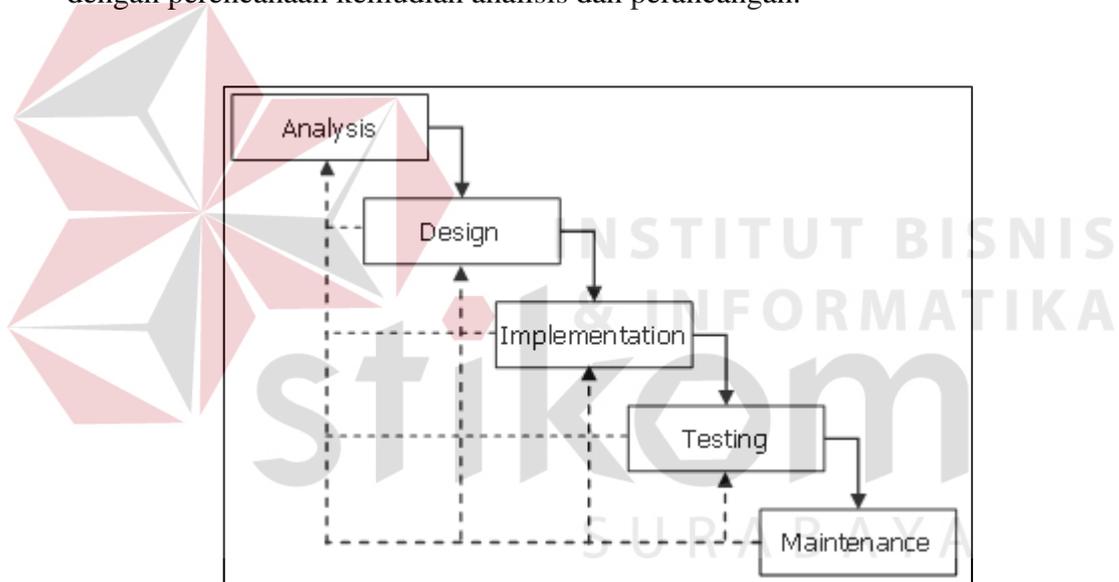


## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

Dalam pembuatan aplikasi haruslah menerapkan konsep *System Development Life Cycle* (SDLC) untuk memastikan bahwa aplikasi yang dibangun sesuai dengan langkah-langkah yang tepat dan menghasilkan aplikasi yang sempurna. Tahapan-tahapan dalam model SDLC *Waterfall* akan diterapkan dalam pembuatan aplikasi pencarian tugas akhir ini yang diawali terlebih dahulu dengan perencanaan kemudian analisis dan perancangan.



Gambar 3.1 Tahap SDLC Model *Waterfall*.

#### 3.1 Perencanaan

Sebelum melakukan analisis, perencanaan mengenai aplikasi yang akan dibangun merupakan tahapan awal sebelum melaksanakan SDLC. Perencanaan dilakukan untuk mengetahui fakta-fakta yang ada, pokok-pokok permasalahan yang dihadapi saat ini, proses-proses yang sedang dijalankan, siapa saja pengguna yang terlibat, dan hal-hal lain yang mendukung dalam perancangan aplikasi

pencarian ini nantinya. Untuk memperoleh fakta-fakta awal dan pengumpulan data dilakukan melalui wawancara, pengamatan atau observasi, dan studi literatur.

### 3.1.1 Wawancara

Wawancara dilakukan untuk mengetahui bagaimana sistem pencarian koleksi tugas akhir pada perpustakaan khususnya pada perpustakaan digital Stikom Surabaya berjalan saat ini. Selain itu wawancara juga dilakukan untuk mengetahui hal-hal yang berkaitan mengenai pengolahan koleksi tugas akhir.

Adapun hasil wawancara yang telah dilakukan adalah sebagai berikut:

1. Berdasarkan wawancara dengan Kepala Perpustakaan Stikom Surabaya Bapak Rudi Santoso menghasilkan fakta bahwa aplikasi Digilib Stikom Surabaya merupakan hasil kerja sama dengan pihak Institut Teknologi Sepuluh Nopember Surabaya (ITS) sehingga metode pencarian yang digunakan belum bisa diketahui.
2. Kemudian untuk mengetahui hal-hal yang berkaitan dengan pengolahan koleksi tugas akhir dilakukan wawancara dengan Petugas Pengolah Koleksi Tugas Akhir Saudara Annuh Liwan Nahar menghasilkan informasi-informasi sebagai berikut:
  - a) Daftar buku tugas akhir, buku cetak tugas akhir, dan CD (*Compact Disc*) tugas akhir mahasiswa berasal dari Pusat Pelayanan Tugas Akhir (PPTA) Stikom Surabaya yang diserahkan kepada perpustakaan untuk digunakan sebagai koleksi tugas akhir.
  - b) Daftar buku tugas akhir yang diberikan, kemudian dicocokkan dengan buku cetak dan CD tugas akhir sebelum disimpan identitas tugas akhir dan *file-file* tugas akhir ke dalam basis data repositori tugas akhir.

- c) Identitas tugas akhir yang disimpan dalam basis data repositori tugas akhir adalah judul tugas akhir, pengarang, tahun terbit, Nomor Induk Mahasiswa (NIM), abstrak, dosen pembimbing 1 dan pembimbing 2, dan kata kunci.
- d) Untuk memasukkan identitas abstrak, petugas membuka CD tugas akhir mahasiswa yang sesuai dengan judul tugas akhir dan membaca *file* dokumen abstrak dengan format doc, docx, atau pdf dan menyalin isinya untuk direkatkan (*copy-paste*) ke dalam formulir pengisian abstrak identitas tugas akhir.
- e) *File* tugas akhir yang disimpan dalam basis data repositori tugas akhir adalah *file cover*, abstrak, daftar isi, bab 1, bab 2, bab 3, bab 4, bab 5, dan daftar pustaka. *File-file* ini terdapat pada setiap CD tugas akhir mahasiswa yang disertakan dalam pengumpulan buku tugas akhir.

### 3.1.2 Pengamatan

Pengamatan dilakukan untuk mengetahui bagaimana sistem pencarian koleksi tugas akhir pada perpustakaan digital Stikom Surabaya berjalan. Pengamatan dilakukan dengan mengakses halaman *web* Digilib Stikom Surabaya yaitu <http://digilib.stikom.edu>. Pengamatan berikutnya adalah bagaimana proses pencarian dan menampilkan koleksi tugas akhir berjalan. Dimulai dengan memasukkan kata kunci pada kotak isian yang tersedia pada halaman utama *website*. Pencarian dilakukan dengan kata kunci yang berbeda dan memasukkan dua atau lebih kata pada kotak isian dan menekan tombol “cari” untuk melihat hasil yang ditampilkan. Tampilan hasil pencarian pada *web* Digilib Stikom Surabaya dapat dilihat pada Gambar 3.2.

The screenshot displays the STIKOM Digital Library search interface. At the top, there are navigation links for 'SITUS LAIN', 'PERPUSTAKAAN STIKOM', 'PERPUSTAKAAN WEBOPAC', 'E-RESOURCES CENTER', and 'STIKOM SURABAYA'. The main header reads 'STIKOM DIGITAL LIBRARY INSTITUTIONAL REPOSITORY'. A search bar contains the keyword 'fuzzy'. Below the search bar, there are options for 'Metode' (Free Text Search, Expanded Search) and a 'CARI' button. The search results show 'Total rekord : 111 koleksi'. The first result is 'PERANCANGAN DAN IMPLEMENTASI PENGENDALI MOTOR INDUKSI DENGAN METODE FUZZY ADAPTIVE INTERNAL MODEL CONTROL' by DEWA KADEK B. MAHAYANA (99410209061) - S1 Sistem Informasi. The page also features a 'LOGIN AREA' with fields for NIM/NID/NIK/USERNAME and PIN/PASSWORD, and a 'REGISTRASI MEMBER' link.

Gambar 3.2 Hasil Pencarian Koleksi Tugas Akhir Digilib Stikom Surabaya

Pada judul-judul koleksi tugas akhir yang ditampilkan merupakan tautan (*link*) untuk menampilkan lebih detail mengenai identitas tugas akhir. Pada salah satu judul tugas akhir yang dipilih, identitas yang ditampilkan adalah judul, pengarang, NIM, subyek, kata kunci, abstrak dalam bahasa Indonesia, abstrak dalam bahasa Inggris, kontributor, dan daftar *file* dokumen yang dapat diunduh dengan format pdf. Namun pengunduhan hanya bisa dilakukan setelah melalui proses pendaftaran menjadi anggota Digilib Stikom Surabaya.

Untuk mengetahui apakah metode pencarian telah menerapkan *Information Retrieval* atau tidak, diperlukan pengamatan dengan metode *reverse engineering* pada *file* Digilib Stikom Surabaya. *File* yang diamati adalah

“xsearch.php”, dimana *file* ini digunakan setelah pengguna menekan tombol “Cari”. Pemanggilan *file* ini dapat dilihat pada Gambar 3.3.



Gambar 3.3 Pemanggilan *File* “xsearch.php”

Berdasarkan pengamatan pada *file* “xsearch.php”, tidak ditemukan penerapan langkah-langkah pada konsep *Information Retrieval*. Langkah-langkah seperti *stemming*, penghapusan *stopword*, penghitungan pembobotan, dan penghitungan kemiripan kosinus tidak dilakukan

### 3.1.3 Studi Literatur

Selain menerapkan wawancara dan pengamatan untuk memperoleh informasi dan fakta-fakta awal, diperlukan juga proses studi literatur. Studi literatur merupakan suatu kegiatan untuk mengumpulkan dan mempelajari bahan-bahan literatur yang berhubungan dengan permasalahan yang dibahas pada penelitian tugas akhir ini. Sumber literatur dapat berupa buku teori, baik buku fisik maupun buku digital (*e-book*), jurnal-jurnal yang berkaitan dengan konsep *Information Retrieval* maupun yang berkaitan dengan pemrograman.

Studi literatur yang digunakan untuk penelitian ini adalah:

1. Studi literatur mengenai langkah-langkah pengembangan aplikasi.
2. Studi literatur mengenai konsep pencarian dan *information retrieval* melalui buku maupun jurnal.

3. Studi literatur mengenai pemrograman *web*, khususnya pemrograman PHP dan pemrograman basis data.
4. Studi literatur mengenai cara pembacaan dan mendapatkan isi dari suatu dokumen untuk dimasukkan ke dalam basis data.

## 3.2 Analisis

### 3.2.1 Analisis Permasalahan

Dari hasil wawancara dan pengamatan sebelumnya, dapat dianalisis permasalahan yang sedang dihadapi dalam pengolahan dan pencarian koleksi tugas akhir. Permasalahan yang dihadapi adalah sebagai berikut:

1. Metode pencarian koleksi saat ini masih belum diketahui.
2. Pengisian identitas abstrak tugas akhir ke dalam basis data repositori masih dengan teknik salin-rekat (*copy-paste*) dari *file* dokumen abstrak tugas akhir.

Berdasarkan permasalahan tersebut, disiapkan beberapa solusi yang diberikan melalui pembuatan aplikasi nantinya. Solusi-solusi tersebut adalah sebagai berikut:

1. Pencarian koleksi menerapkan konsep *information retrieval*, dikarenakan jumlah data yang cukup banyak dengan pengolahan utama berupa kata, kalimat, atau teks. Dalam *information retrieval* terdapat metode-metode yang digunakan dalam setiap tahapannya, seperti metode *stemming* Nazief & Adriani dan metode *cosine similarity*.
2. Pengisian abstrak dapat dilakukan secara otomatis dengan memberikan fitur aplikasi yang mampu menangkap dan membaca isi dari dokumen abstrak.

Sehingga diharapkan proses yang saat ini sedang dijalankan menjadi lebih cepat dan singkat.

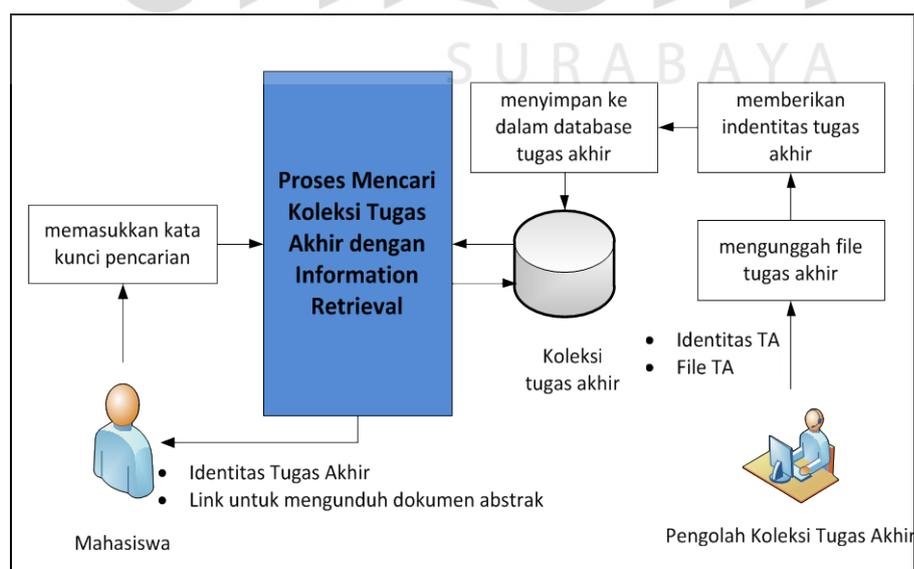
### 3.2.2 Analisis Sistem

Berdasarkan hasil wawancara dan pengamatan serta studi literatur, dapat dianalisis kebutuhan-kebutuhan dari aplikasi yang akan dibangun. Disebabkan banyaknya jumlah koleksi tugas akhir yang tersedia saat ini, maka perlu diterapkan *information retrieval* dalam pencarian koleksi tugas akhir. Analisis kebutuhan dari aplikasi pencarian koleksi tugas akhir adalah sebagai berikut:

1. Proses utama yang terjadi adalah proses mengolah koleksi tugas akhir, proses mencari koleksi tugas akhir, dan proses menampilkan hasil pencarian.
2. Pengguna yang terlibat dalam aplikasi ini adalah petugas pengolah koleksi tugas akhir dan mahasiswa sebagai pengguna dalam pencarian koleksi tugas akhir.
3. Hak akses yang diberikan kepada pengguna petugas pengolah koleksi tugas akhir berupa hak akses untuk memasukkan identitas dan *file* tugas akhir ke dalam basis data koleksi, hak akses untuk mengubah identitas tugas akhir, hak akses untuk melihat koleksi tugas akhir, hak akses untuk melakukan pencarian koleksi tugas akhir berdasarkan identitas tertentu, seperti nama, NIM, atau judul tugas akhir.
4. Hak akses yang diberikan kepada pengguna mahasiswa berupa hak akses untuk melakukan pencarian dengan kata kunci, hak akses untuk melihat hasil pencarian, dan hak akses untuk melihat detail identitas akhir dan mengunduh *file* tugas akhir.

5. Data yang diperlukan untuk pengolahan basis data repositori koleksi tugas akhir adalah identitas tugas akhir berupa judul, pengarang, tahun terbit, Nomor Induk Mahasiswa (NIM), abstrak, dosen pembimbing 1 dan pembimbing 2, subjek, kata kunci, dan *file-file* dokumen tugas akhir yang sesuai dengan identitas tugas akhir.
6. Data yang diperlukan untuk pencarian tugas akhir adalah kata kunci pencarian yang dimasukkan oleh pengguna mahasiswa dan daftar indeks kata.
7. Informasi yang dihasilkan dalam proses menampilkan hasil pencarian adalah judul tugas akhir yang telah diurutkan dari kemiripan tertinggi, dan tautan untuk membaca detil tugas akhir dan mengunduh *file* tugas akhir.

Untuk memahami proses yang akan dijalankan oleh aplikasi diperlukan sebuah gambaran umum aplikasi yang akan dibangun. Gambaran umum aplikasi dapat dilihat pada Gambar 3.4.



Gambar 3.4 Gambaran Umum Aplikasi.

Pada Gambar 3.4 menjelaskan gambaran umum dari proses-proses yang dilakukan oleh pengguna dan aplikasi. Pada gambaran umum aplikasi terdapat dua pengguna yaitu mahasiswa dan pengolah koleksi tugas akhir. Untuk pengguna petugas pengolah koleksi tugas akhir bertugas untuk memasukkan data-data tugas akhir ke dalam basis data koleksi. Pengguna mahasiswa berperan dalam proses pencarian tugas akhir. Proses inti yang dijalankan aplikasi adalah proses mencari koleksi tugas akhir dengan *information retrieval* yang berfungsi untuk menghasilkan daftar koleksi tugas akhir dengan tingkat relevansi yang sesuai dengan kata pencarian.

Dari gambaran umum tersebut juga dapat dianalisis proses-proses yang terjadi di aplikasi yang akan dibangun. Proses-proses tersebut adalah:

1. Proses mengolah koleksi tugas akhir

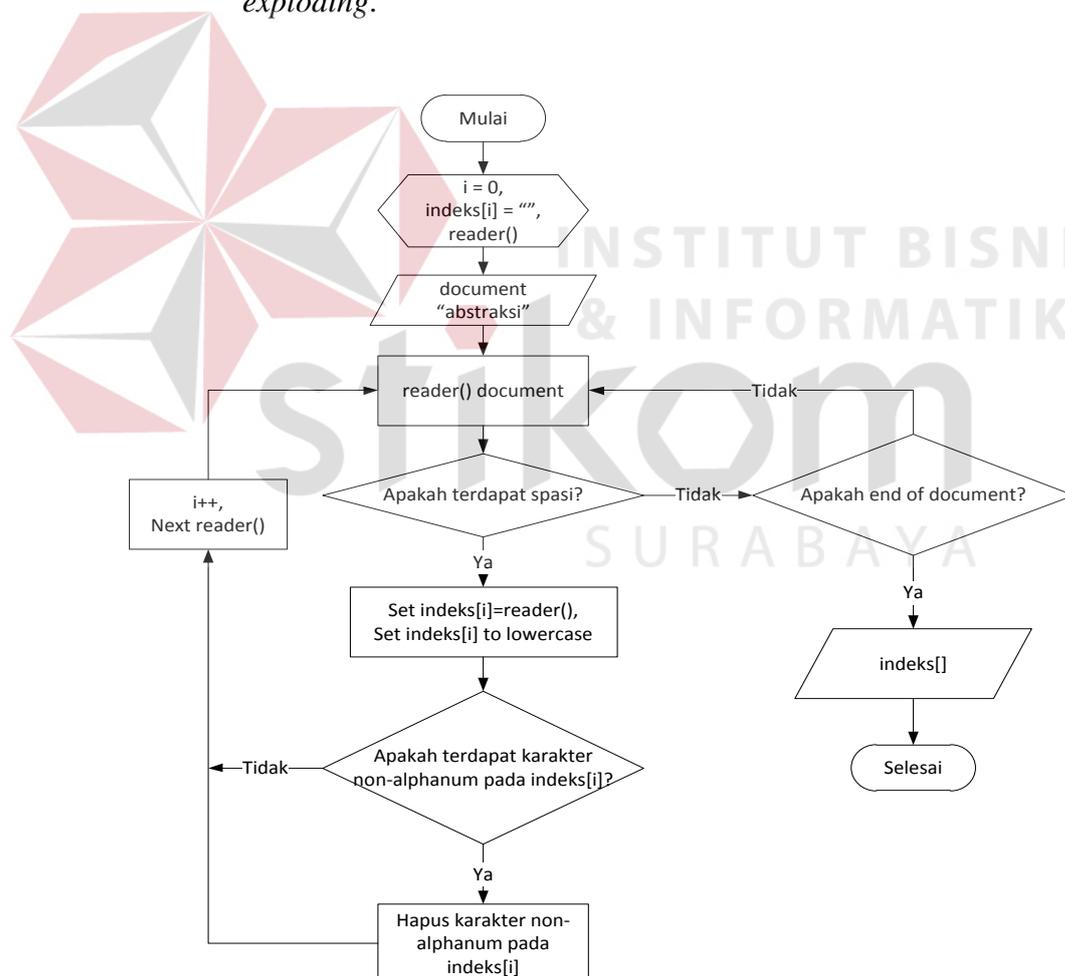
Pada proses ini pengguna yang terlibat adalah petugas pengolah koleksi tugas akhir. Pengguna memasukkan data identitas tugas akhir dan *file* dokumen dan menyimpannya ke dalam basis data koleksi tugas akhir.

2. Proses mencari dan menampilkan koleksi dengan *information retrieval*

Pada proses ini terjadi langkah-langkah dalam *information retrieval*, yaitu *indexing* (*tokenization*, *stopwords*, dan *stemming*), pembobotan, dan perhitungan kemiripan (*similarity*). Yang menjadi masukan pada proses ini adalah kata kunci pencarian yang diberikan oleh pengguna mahasiswa dan yang akan menjadi keluaran adalah hasil perhitungan *cosine similarity*.

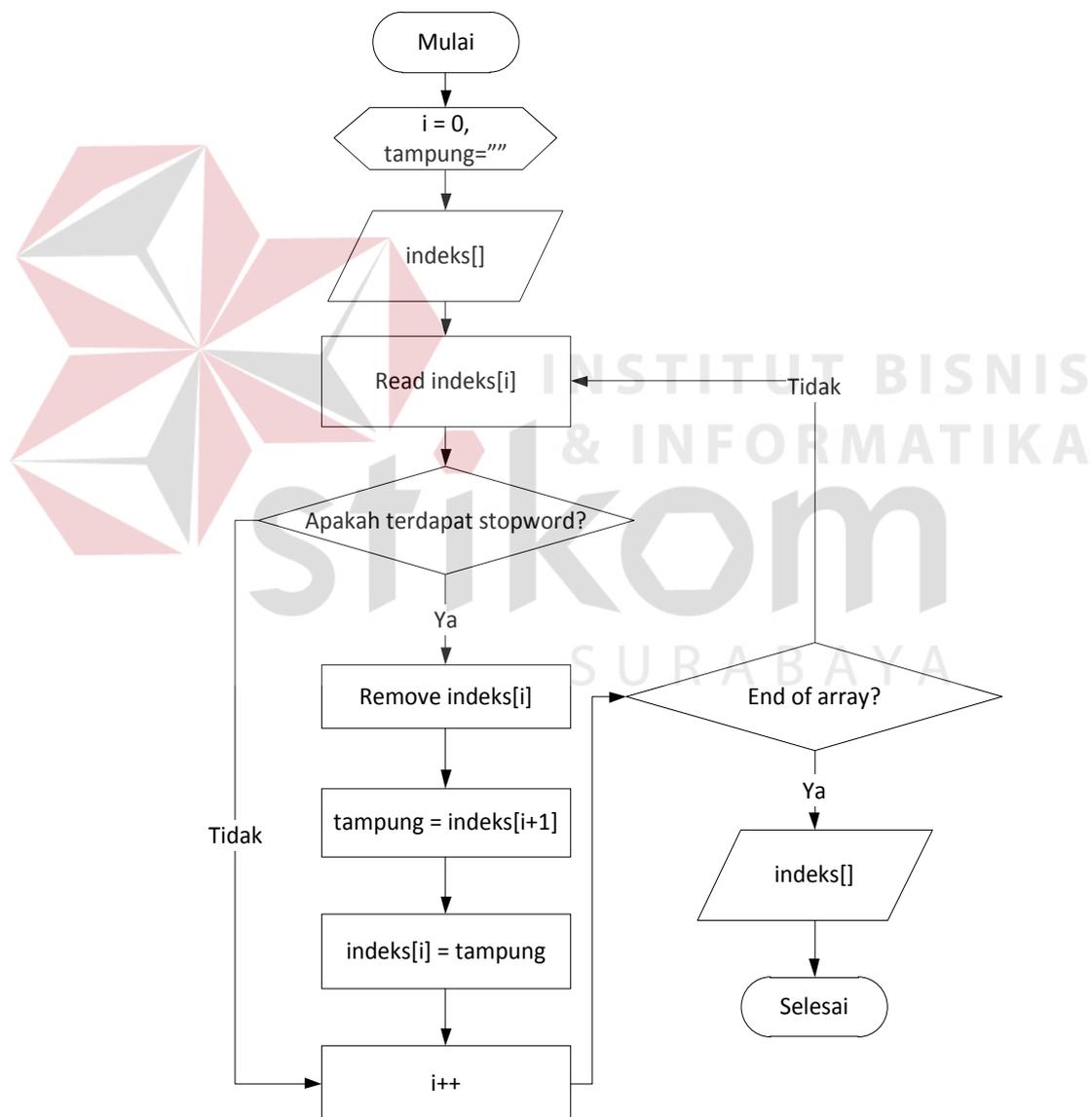
- a. Proses *tokenization* abstrak tugas akhir dan kata pencarian dilakukan dengan menggunakan fungsi dari PHP yaitu *exploding*. *Exploding* mengubah susunan kalimat menjadi susunan *array* kata. Proses

*tokenization* pada dokumen abstrak terjadi pada saat pengguna petugas pengolah koleksi tugas akhir melakukan penyimpanan data tugas akhir. Pada saat aplikasi menyimpan data tersebut, aplikasi secara otomatis membaca isi dari dokumen abstrak untuk dilakukan *tokenization* dengan fungsi *exploding* PHP yang disimpan dalam bentuk *array*. Proses yang sama juga dilakukan apa pengguna mahasiswa memasukkan deretan kata kunci pada fasilitas pencarian. Dapat dilihat pada Gambar 3.5 mengenai *flowchart* logika *exploding*.



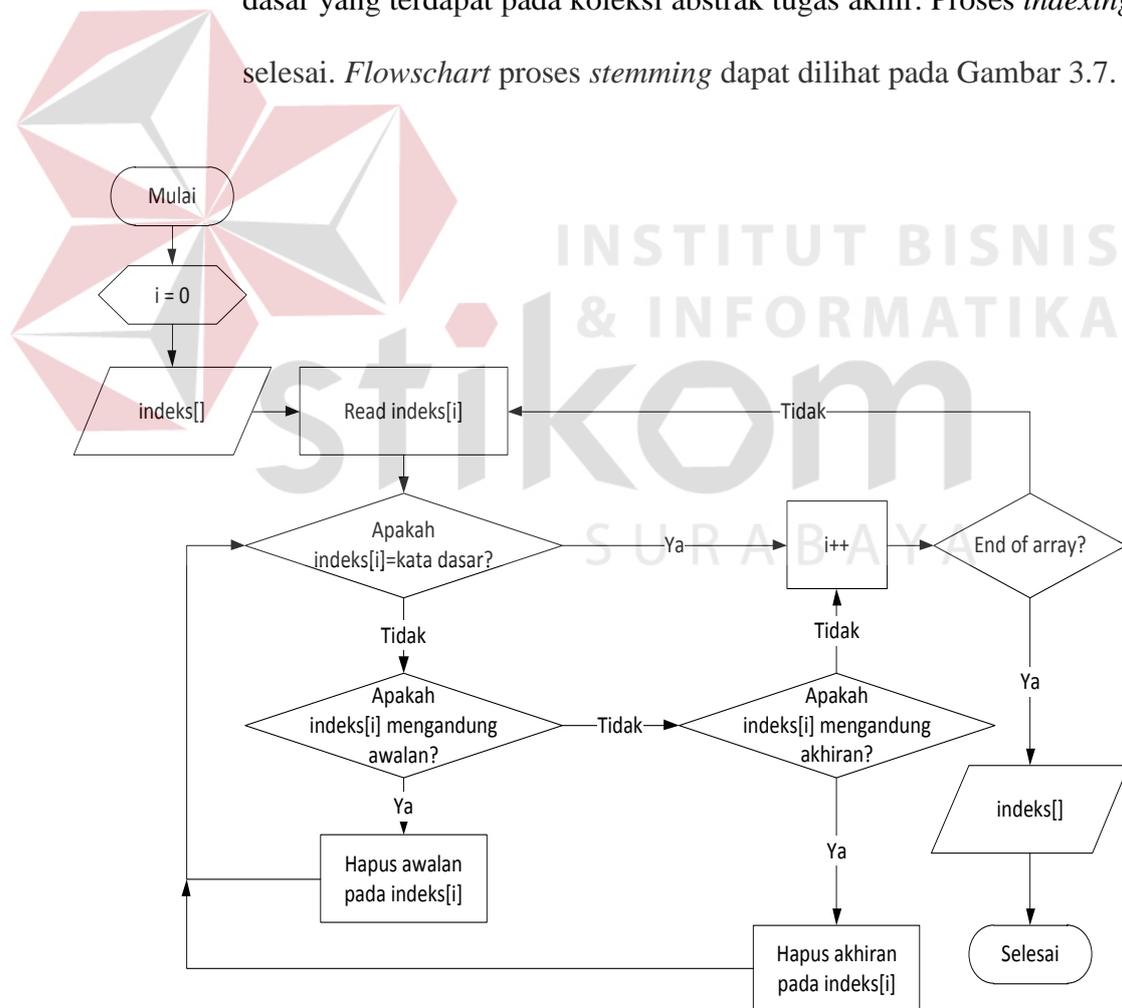
Gambar 3.5 *Flowchart Exploding*

- b. Setelah *array* menangkap daftar kata, berikutnya adalah memeriksa setiap isi *array* apakah mengandung kata yang terdapat pada basis data *stopword* atau tidak. Jika isi dari *array* mengandung kata dalam *stopword*, maka proses berikutnya adalah menghapus isi dari *array* tersebut. Untuk mengetahui lebih jelas proses penghapusan *stopword*, dapat melihat *flowchart* pada Gambar 3.6.



Gambar 3.6 *Flowchart Menghapus Stopword*

- c. Setelah isi *array* bersih dari *stopword*, langkah berikutnya adalah mengubah isi *array* yang ada menjadi kata dasar melalui proses *stemming*.. Dengan membaca karakter-karakter awalan dan akhiran pada isi *array*, dilakukan penghapusan imbuhan sesuai dengan metode *stemming* bahasa Indonesia. Untuk membantu proses *stemming* diperlukan basis data kata dasar bahasa Indonesia. Apabila isi dari *array* sudah berbentuk kata dasar, aplikasi akan menyimpan isi *array* tersebut pada basis data Indeks. Indeks berisikan daftar kata dasar yang terdapat pada koleksi abstrak tugas akhir. Proses *indexing* selesai. *Flowschart* proses *stemming* dapat dilihat pada Gambar 3.7.

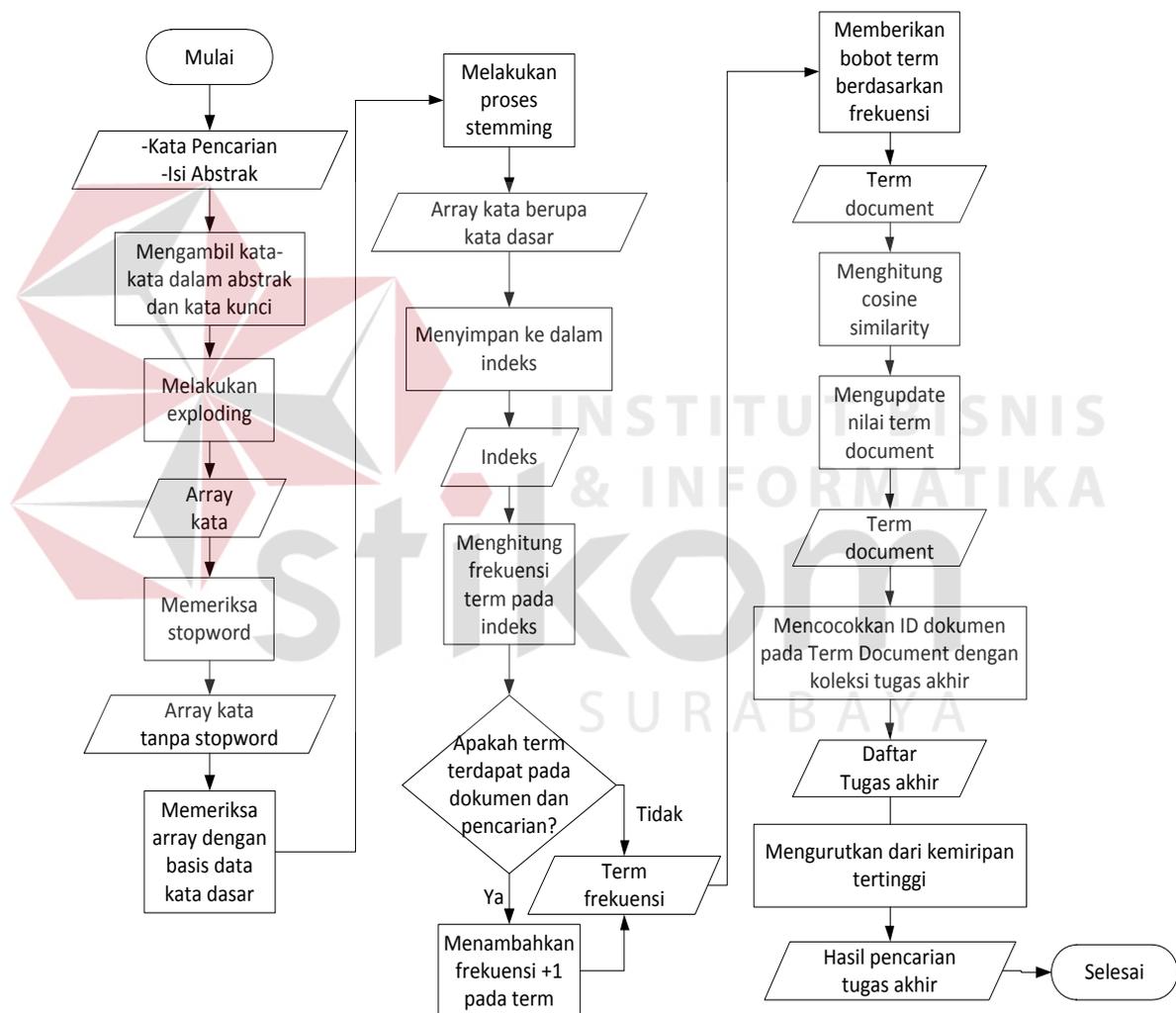


Gambar 3.7 *Flowchart Stemming*

- d. Melanjutkan langkah *information retrieval* berikutnya adalah proses pembobotan yang didahului dengan menghitung jumlah frekuensi kemunculan kata indeks pada abstrak tugas akhir dan kata kunci pencarian. Kemudian nilai dijadikan dasar sebagai pemberian bobot untuk masing-masing kata indeks atau *term*. Hasil perhitungan frekuensi *term* ini dimasukkan ke dalam basis data *term frequency* untuk dipergunakan dalam langkah berikutnya.
- e. Setelah mengetahui frekuensi dari masing-masing *term*, langkah berikutnya adalah menghitung bobot dari *term* dari masing-masing dokumen dan menghitung bobot dari kata pencarian. Hasil perhitungan bobot dimasukkan ke dalam basis data *term document*. Kemudian melakukan perhitungan panjang vektor dengan menggunakan hasil perhitungtan kuadrat bobot sebelumnya dari masing-masing *term* dan memperbarui basis data *term document* untuk kolom-kolom variabel perkalian vektor.
- f. Setelah mengetahui bobot masing-masing *term* per dokumen, langkah berikutnya adalah mencari total bobot untuk dilakukan kemiripan dokumen dengan kata pencarian.dan dilakukan *cosine similarity*. Hasil perhitungan *cosine similarity* akan dimasukkan ke dalam basis data yang kemudian akan diurutkan dari kemiripan tertinggi.
- g. Langkah terakhir yaitu menampilkan hasil pengurutan kemiripan yang telah dilakukan pada proses sebelumnya dan memberikan identitas tugas akhir yang sesuai dengan kata kunci pencarian. Serta

diberikan pula tautan (*link*) untuk mengunduh *file* abstrak. Pengurutan tampilan koleksi tugas akhir diurutkan dari persentase tingkat kemiripan tertinggi menuju yang terendah.

Untuk membantu memahami alur pencarian dan menampilkan hasil *information retrieval* dapat dilihat *flowchart* pada Gambar 3.8.



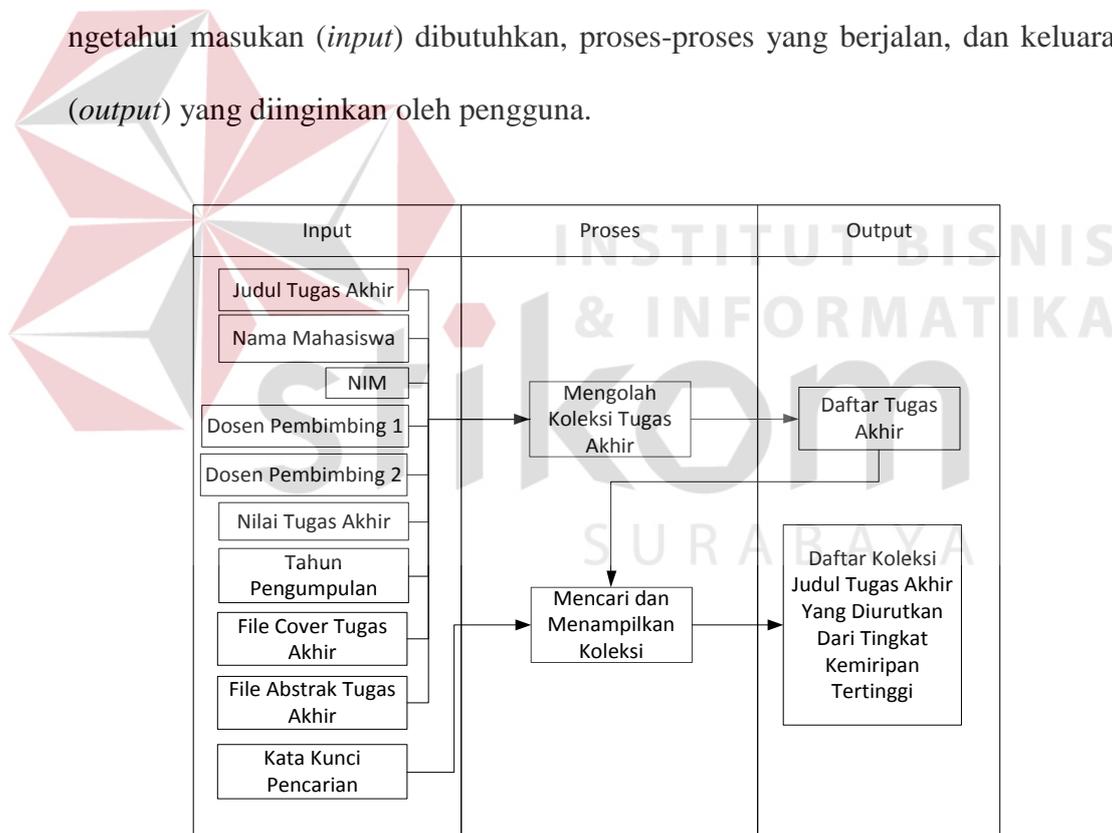
Gambar 3.8 *Flowchart Information Retrieval*

### 3.3 Perancangan Sistem

Untuk membantu dalam pembuatan aplikasi pencarian tugas akhir, diperlukan perancangan yang menggambarkan urutan dan langkah-langkah pengerjaan dalam pembuatan aplikasi. Dengan merancang sistem, aplikasi yang akan dibangun tidak akan melenceng dari rencana awal dan digunakan sebagai pedoman dalam pembuatan aplikasi.

#### 3.3.1 Diagram Blok

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



Gambar 3.9 Diagram Blok Aplikasi Pencarian

Dalam diagram blok sesuai pada Gambar 3.9, terdapat *input*, proses, dan *output* yang terlibat di dalamnya dimana penjelasannya adalah sebagai berikut:

## 1. *Input*

*Input* yang terdapat dalam diagram blok yaitu:

- a. Judul Tugas Akhir.
- b. Nama Mahasiswa merupakan mahasiswa tugas akhir.
- c. NIM, merupakan Nomor Induk Mahasiswa tugas akhir.
- d. Dosen Pembimbing 1, merupakan dosen yang menjadi pembimbing 1 mahasiswa penulis tugas akhir.
- e. Dosen Pembimbing 2, merupakan dosen yang menjadi pembimbing 2 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 Pencarian, merupakan daftar kata-kata yang dimasukkan mahasiswa untuk mencari judul tugas akhir yang diinginkan.

## 2. *Proses*

Proses yang terdapat dalam diagram blok yaitu:

- a. Mengolah koleksi tugas akhir, merupakan proses mengunggah file tugas akhir dan data-data pendukung ke dalam basis data koleksi.
- b. Mencari dan Menampilkan Koleksi, merupakan proses inti dari aplikasi pencarian koleksi tugas akhir. Dengan memberikan masukan berupa kata kunci pencarian, aplikasi akan melakukan proses-proses *Information Retrieval* dan menampilkan daftar judul tugas akhir yang sesuai dengan kata kunci pencarian.

### 3. *Output*

*Output* yang terdapat dalam diagram blok, yaitu:

- a. Daftar Tugas Akhir, merupakan daftar tugas akhir yang telah diunggah oleh petugas perpustakaan.
- b. Daftar Koleksi Judul Tugas Akhir Yang Diurutkan Dari Tingkat Kemiripan Tertinggi, merupakan daftar judul tugas akhir hasil pencarian yang dilakukan pengguna. Daftar judul tugas akhir yang ditampilkan berkaitan dengan kata kunci pencarian dan diurutkan dari kemiripan kosinus tertinggi.

#### 3.3.2 *System Flow*

Diagram alir sistem atau *system flow* menggambarkan urutan proses sistem dari aplikasi yang akan dibangun. Dengan *system flow* dapat diketahui kebutuhan-kebutuhan tabel basis data dan data yang mengalir. Penggambaran *system flow* terdiri dari Proses Mengolah Koleksi Tugas Akhir dan Proses Mencari dan Menampilkan Koleksi.

##### 1. Proses Mengolah Koleksi Tugas Akhir

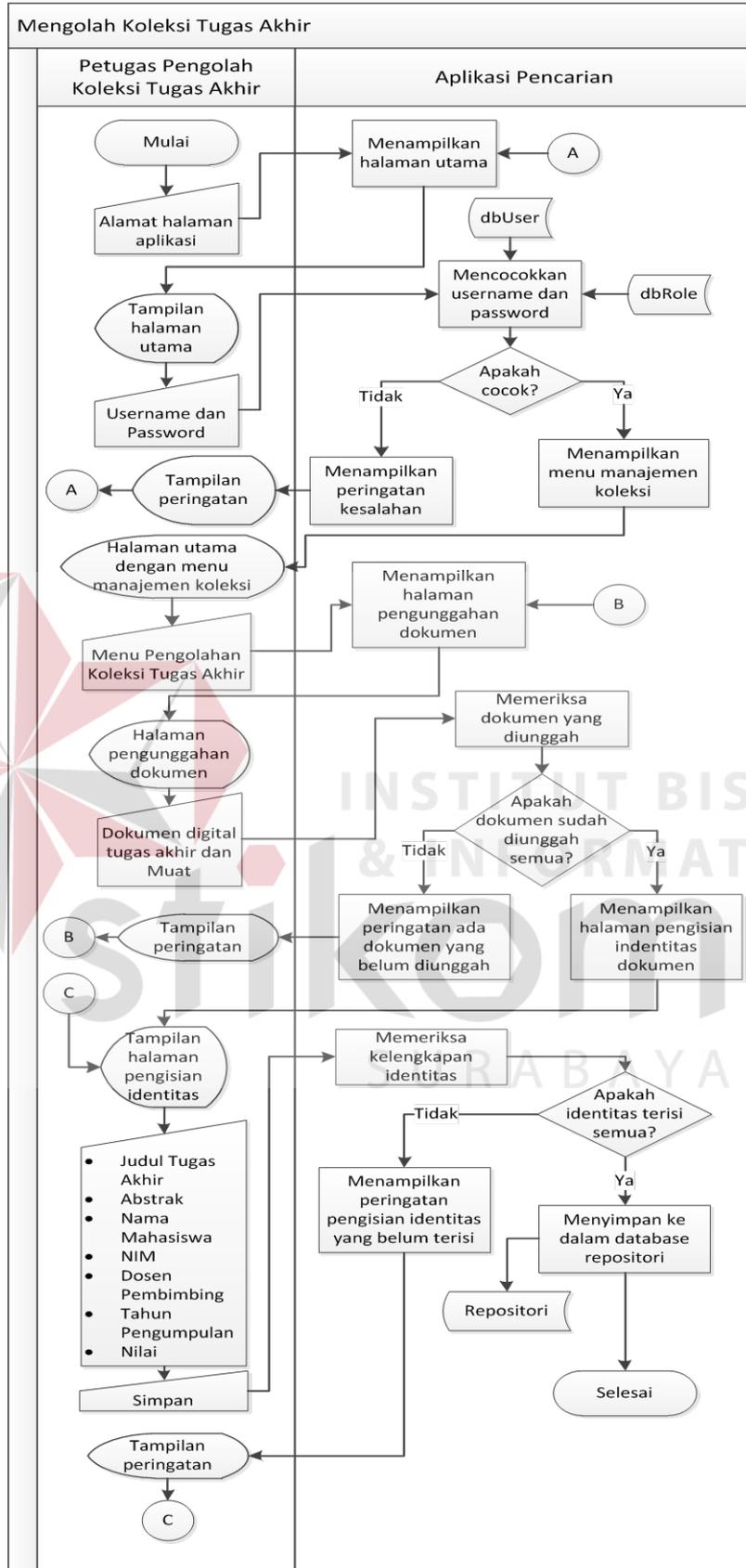
Pada Gambar 3.10 menggambarkan mengenai aliran sistem saat melakukan proses mengolah koleksi tugas akhir. Proses ini merupakan interaksi antara aplikasi dengan pengguna Petugas Pengolah Koleksi Tugas Akhir. Hal ini menyesuaikan dengan proses yang telah dijalankan selama ini. Proses ini dimulai dengan petugas membuka peramban *web* dan memasukkan alamat aplikasi dilanjutkan dengan mengisi *username* dan kata kunci pengguna. Aplikasi akan mencocokkan dengan data pengguna yang terdapat dalam basis data pengguna dan peran yang diberikan. Apabila data pengguna tidak ditemukan atau tidak

cocok antara *username* dan kata kunci, aplikasi akan memberikan peringatan dan kembali membuka halaman utama. Apabila data yang dimasukkan cocok dengan data pengguna di dalam basis data, maka aplikasi melanjutkan untuk membuka halaman utama dengan tambahan pilihan menu yang berfungsi untuk memasukkan data tugas akhir.

Dengan memilih menu Pengolahan Koleksi Tugas Akhir, pengguna diberikan tampilan halaman untuk mengunggah *file* dokumen tugas akhir. Hal ini dimaksudkan untuk memudahkan pengguna untuk memeriksa isi dari dokumen abstrak tugas akhir tanpa membuka dokumen tersebut. Setelah pengguna mengunggah keseluruhan *file* dokumen tugas akhir, aplikasi akan menampilkan halaman pengisian identitas tugas akhir, seperti Judul Tugas Akhir, Abstrak, Nama Mahasiswa, NIM, Dosen Pembimbing 1, Dosen Pembimbing 2, Tahun Pengumpulan, dan Nilai.

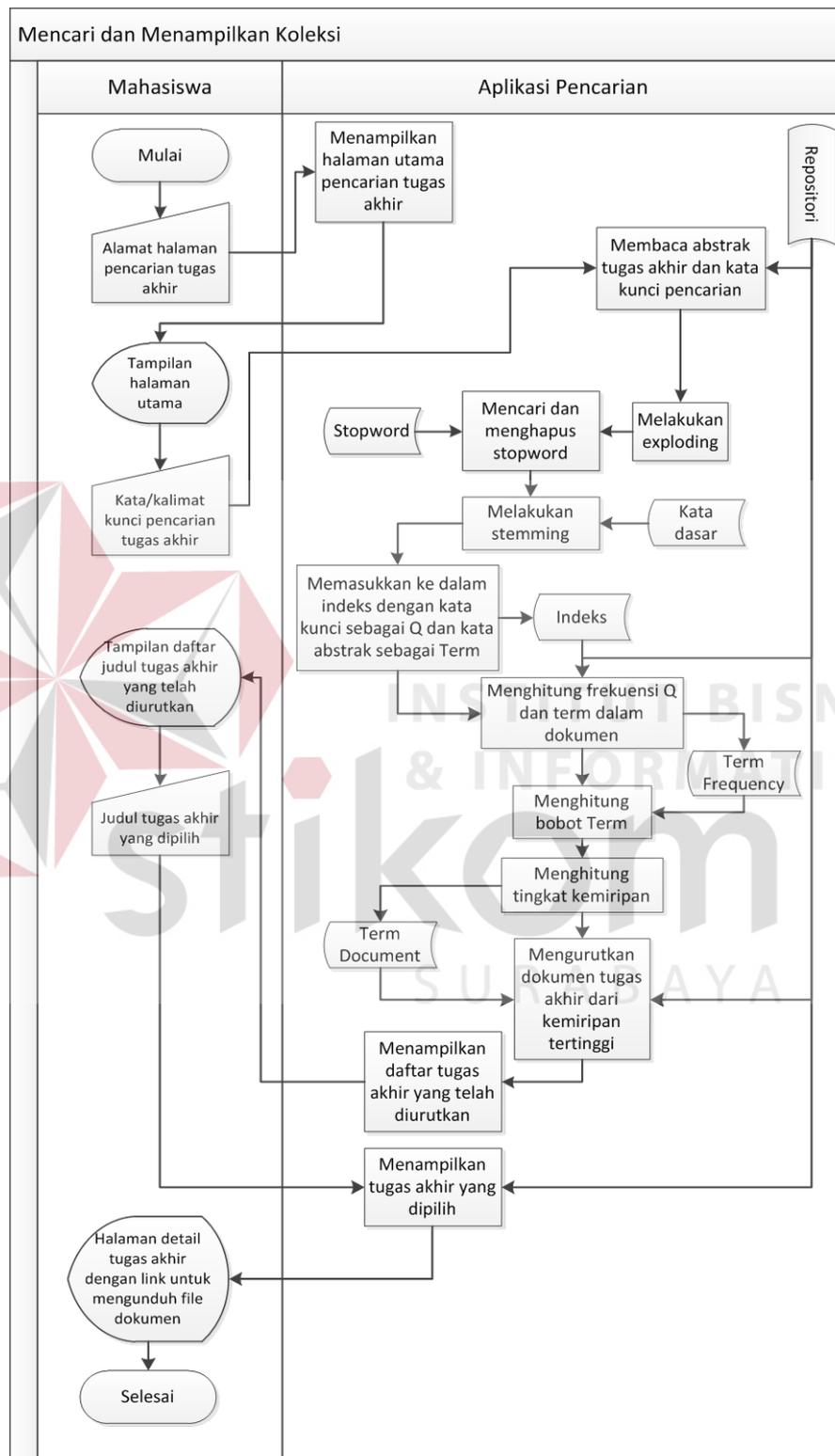
Pada halaman pengisian, isi kotak pengisian Abstrak sudah terisi secara otomatis berdasarkan pembacaan dokumen abstrak yang diunggah sebelumnya. Pengguna bisa mengubah isi abstrak tersebut apabila terdapat kesalahan pengetikan.

Setelah pengguna menyelesaikan pengisian identitas tugas akhir, pengguna melakukan penyimpanan dengan menekan tombol “Simpan”. Kemudian aplikasi menyimpan identitas dan dokumen tugas akhir tersebut ke dalam basis data.



Gambar 3.10 System Flow Proses Mengolah Koleksi Tugas Akhir

## 2. Proses Mencari dan Menampilkan Koleksi



Gambar 3.11 Proses Mencari dan Menampilkan Koleksi

Proses mencari dan menampilkan koleksi seperti pada Gambar 3.11 merupakan proses inti aplikasi pencarian tugas akhir. Proses ini menerapkan konsep *Information Retrieval*, seperti proses pengolahan kata/kalimat pencarian, perhitungan frekuensi *term* dalam dokumen, pembobotan, dan proses perhitungan kemiripan. Pada proses ini pengguna yang terlibat adalah pengguna mahasiswa atau pengguna lainnya yang ingin mencari judul tugas akhir. Untuk penelitian tugas akhir ini lebih mengutamakan kepada mahasiswa Stikom Surabaya.

Pada proses ini dimulai dengan mahasiswa memasukkan alamat *website* pencarian judul tugas akhir dan aplikasi akan memunculkan halaman utama. Dalam halaman utama terdapat *field* yang diisi untuk memasukkan kata kunci pencarian dan tombol untuk melakukan pencarian. Saat pengguna menekan tombol pencarian, aplikasi akan melakukan proses awal yang sama seperti pada proses memasukkan data tugas akhir, yaitu *exploding*, menghapus *stopword*, *stemming*, dan memasukkan ke dalam basis data index. Kata pada *field* pencarian akan di-*exploding* untuk memecah menjadi daftar kata *array*. Fungsi *exploding* ini akan membantu jika kata pencarian mengandung dua atau lebih kata. Proses berikutnya adalah menghapus *stopword* pada kata pencarian dengan mengacu pada basis data *stopword*. Hal ini untuk membantu jika pengguna memasukkan kalimat yang cukup panjang dan mengandung kata yang seharusnya bukan kata kunci.

Berikutnya adalah melakukan *stemming* pada kata pencarian dengan mengacu pada basis data kata dasar. Setelah kata pencarian menjadi kata dasar, langkah selanjutnya adalah memasukkan ke dalam basis data indeks sebagai Q atau *query* pencarian. Dalam basis data indeks ini mengandung istilah atau *term*

yang terdapat pada koleksi dokumen (D) abstrak tugas akhir yang akan dihitung perbandingan frekuensinya dengan kata pencarian (Q). Setelah diketahui nilai frekuensi kemunculan *term* pada Q dan D, langkah berikutnya adalah memberikan bobot dengan mengalikan frekuensi *term* sesuai rumus pembobotan pada bagian landasan teori. Setelah semua *term* pada Q dan D diberikan bobot, maka langkah berikutnya adalah menghitung tingkat kemiripannya.

Menghitung tingkat kemiripan D dan Q dimulai dengan menghitung jumlah hasil perkalian skalar antara Q dengan dokumen-dokumen yang ada. Kemudian menghitung panjang vektor setiap dokumen yang dilanjutkan dengan menghitung tingkat kemiripan dengan rumus *cosine similarity*. Keseluruhan proses perhitungan kemiripan menggunakan Persamaan 2.4.

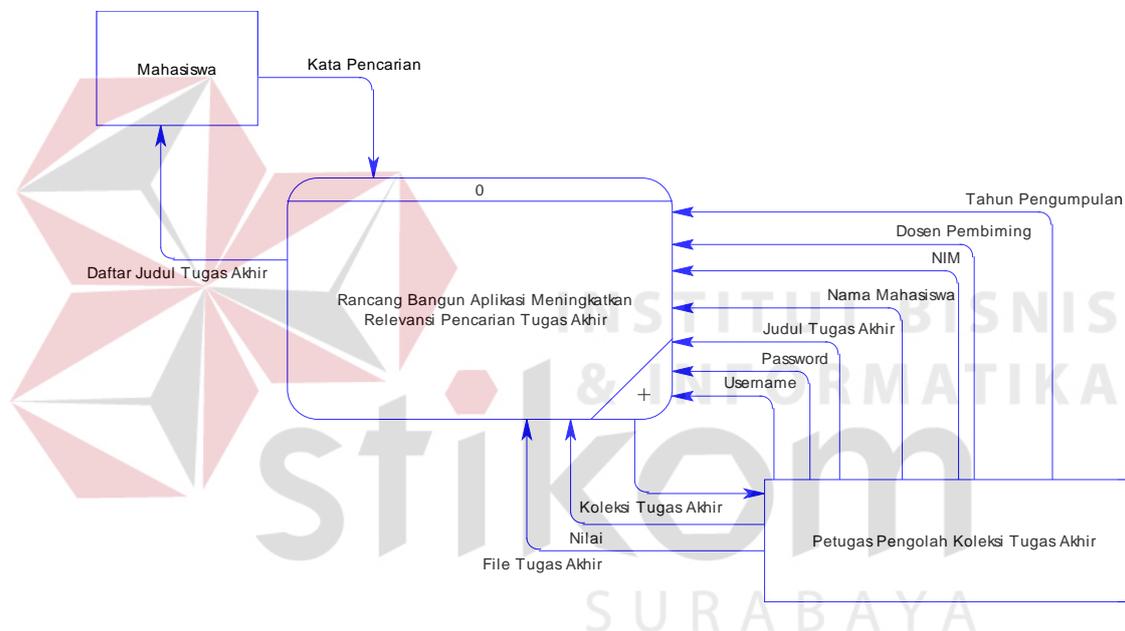
Hasil dari proses perhitungan *cosine similarity* setiap dokumen dengan Q, diurutkan dari tingkat persentase kemiripan yang tertinggi ke tingkat kemiripan terendah dari dokumen-dokumen yang terkait. Daftar judul tugas akhir yang telah diurutkan hasil pencariannya ditampilkan ke pengguna sehingga pengguna dapat memilih judul dokumen tugas akhir apa yang diinginkan. Dengan menekan judul tugas akhir yang dipilih, aplikasi menampilkan identitas tugas akhir yang lebih detail disertai dengan tautan untuk mengunduh dokumen tugas akhir tersebut.

### 3.3.3 Data Flow Diagram (DFD)

Diagram aliran data atau DFD menggambarkan proses dalam analisis dan perancangan perangkat lunak, khususnya dengan pendekatan terstruktur. Pada DFD akan dijelaskan mengenai aliran data yang terdapat dalam aplikasi.

### 1. Diagram Konteks (*Context Diagram*)

Diagram konteks merupakan diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. Pada aplikasi pencarian judul tugas akhir ini, terdapat dua entitas atau pengguna aplikasi, yaitu Mahasiswa dan Petugas Pengolah Koleksi Tugas Akhir. Penggambaran diagram konteks dapat dilihat pada Gambar 3.12.



Gambar 3.12 Diagram Konteks Aplikasi Pencarian Tugas Akhir

Pada Gambar 3.12 terdapat aliran-aliran data yang mengalir di dalam aplikasi. Baik yang mengalir ke dalam aplikasi maupun yang diterima entitas. Aliran data yang tersedia pada diagram konteks adalah:

1. Kata Pencarian
2. Daftar Judul Tugas Akhir
3. *Username*
4. *Password*

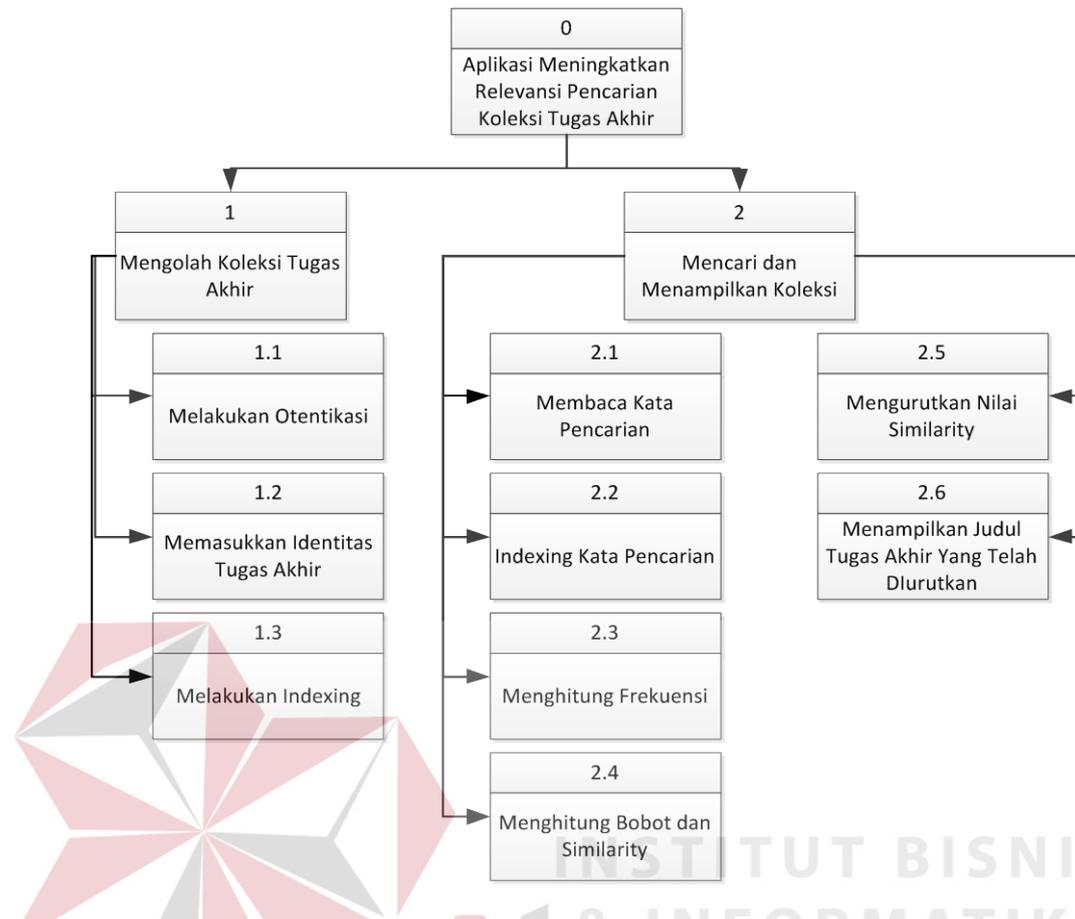
5. Judul Tugas Akhir
6. Nama Mahasiswa
7. NIM
8. Dosen Pembimbing
9. Tahun Pengumpulan
10. *File* Tugas Akhir
11. Nilai
12. Koleksi Tugas Akhir

Entitas dan aliran data pada diagram konteks akan selalu digunakan untuk penggambaran proses yang lebih dalam (*breakdown*). *Breakdown* dari diagram konteks akan dijabarkan pada DFD Level 0 dan Level 1.

## 2. Diagram Jenjang

Penggambaran diagram jenjang dimaksudkan untuk membantu dalam menentukan proses-proses yang terjadi dalam DFD. Diagram jenjang aplikasi pencarian koleksi tugas akhir dapat dilihat pada Gambar 3.13.

Pada Gambar 3.13, dapat dilihat pemberian nomor-nomor pada setiap proses yang terjadi merupakan urutan dari proses yang berjalan pada aplikasi. Selain itu juga menunjukkan tingkat DFD yang akan dibangun nantinya. Penomoran dan penamaan akan selalu digunakan dalam pembuatan DFD.

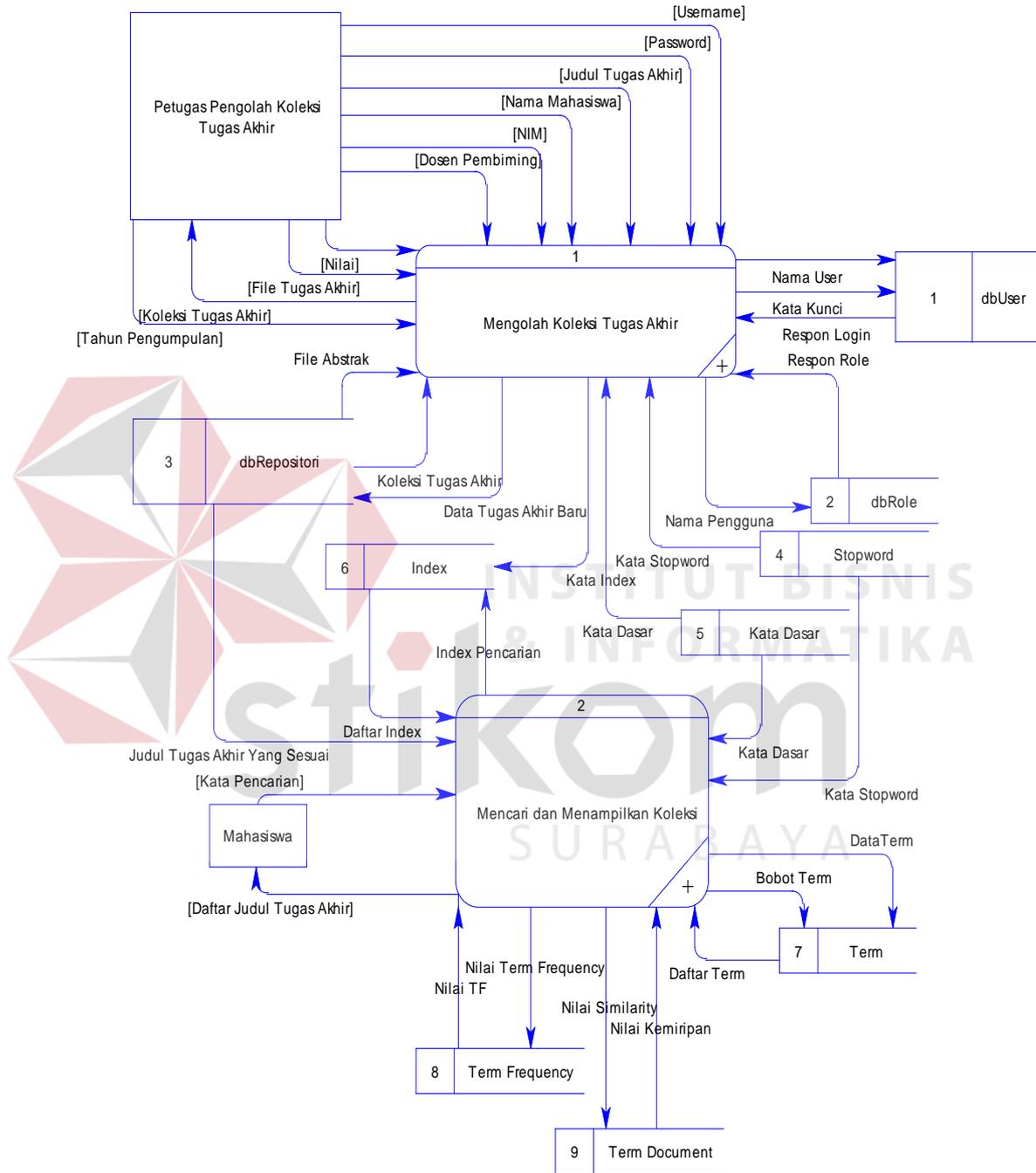


Gambar 3.13 Diagram Jenjang Aplikasi Pencarian Koleksi Tugas Akhir

### 3. DFD Level 0

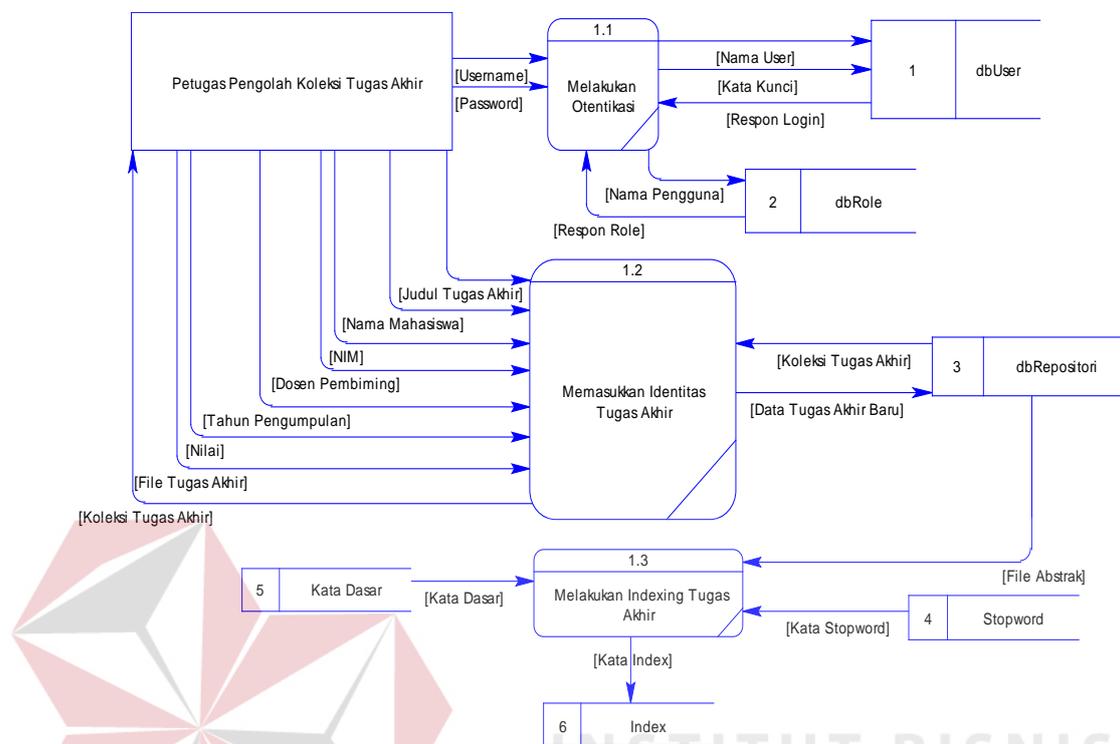
Pada Gambar 3.14 menggambarkan rangkaian aliran data pada DFD Level 0. DFD Level 0 merupakan hasil *breakdown* dari diagram konteks dengan proses-proses yang terjadi berdasarkan pada aliran sistem (*system flow*). Proses-proses utama yang terjadi dalam DFD Level 0 adalah Mengolah Koleksi Tugas Akhir dan Mencari dan Menampilkan Koleksi. Dalam DFD Level 0 terdapat penambahan jumlah aliran data yang merupakan pengembangan dari diagram konteks. Penambahan notasi *data store* juga dilakukan sebagai gambaran mengenai tabel-tabel dalam basis data yang akan digunakan dalam pembuatan aplikasi ini. Masing-masing proses yang terdapat pada DFD Level 0 akan di-

breakdown menjadi DFD Level 1. Hal ini dimaksudkan untuk memperjelas proses-proses yang terjadi di dalamnya.



Gambar 3.14 DFD Level 0

#### 4. DFD Level 1 Proses Mengolah Koleksi Tugas Akhir



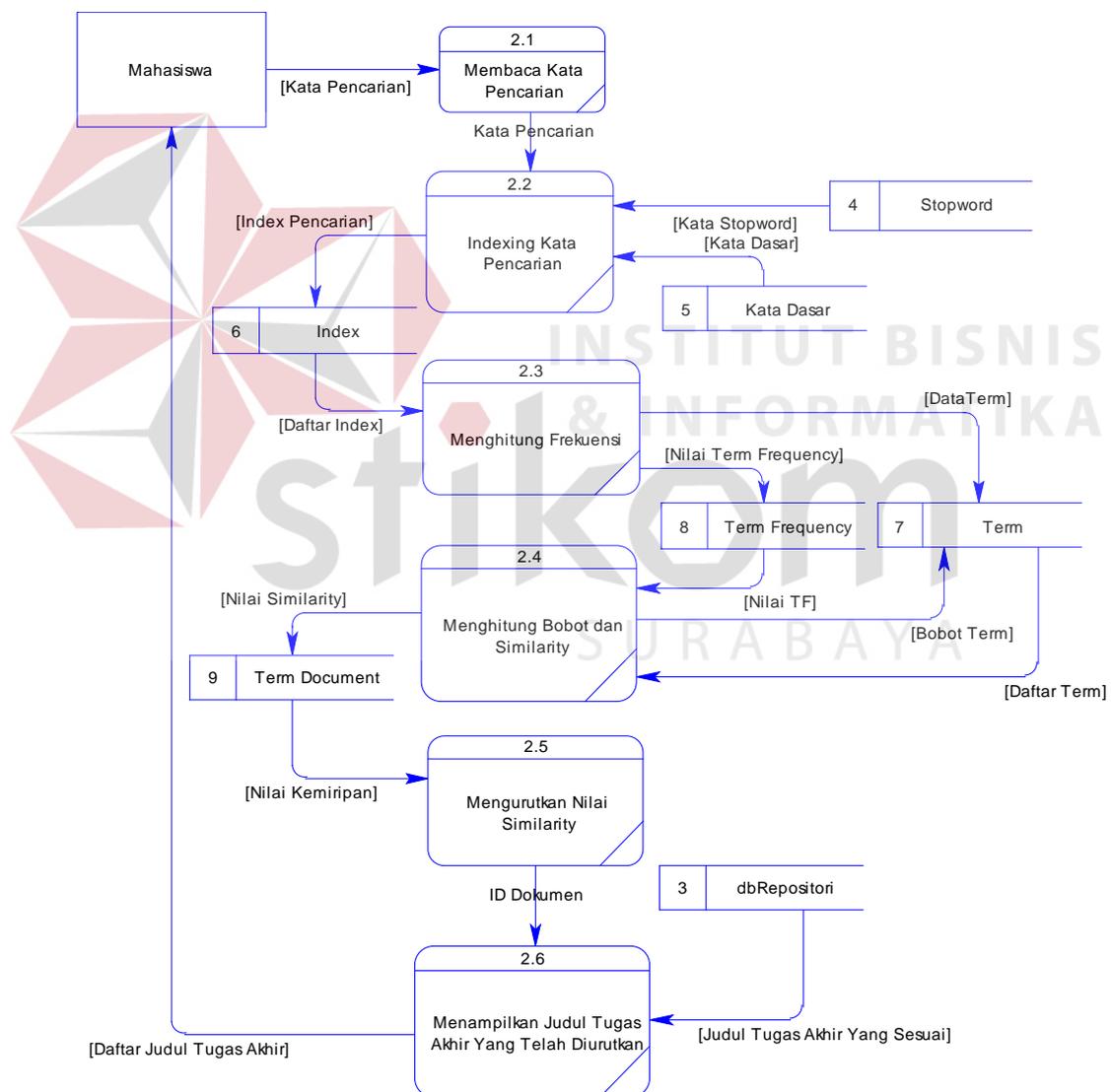
Gambar 3.15 DFD Level 1 Proses Mengolah Koleksi Tugas Akhir

Pada Gambar 3.15, memperjelas jalannya proses memasukkan data tugas akhir yang diterangkan sebelumnya pada DFD Level 0. Pada DFD Level 1 ini, terdapat tiga proses yang terjadi, yaitu Melakukan Otentikasi, Memasukkan Data Tugas Akhir, dan Melakukan *Indexing* Tugas Akhir. Pada DFD Level 1 ini, langkah awal dari konsep *information retrieval* yaitu *indexing* telah dilakukan. Hal ini disebabkan aplikasi menjalankan secara otomatis apabila pengguna selesai menyimpan data tugas akhir.

#### 5. DFD Level 1 Mencari dan Menampilkan Koleksi

Pada Gambar 3.16 merupakan penggambaran lebih detail dari proses *information retrieval*. Pada DFD Level 1 ini terdapat enam proses yang berjalan

di dalamnya, yaitu Membaca Kata Pencarian, *Indexing* Kata Pencarian, Menghitung Frekuensi, Menghitung Bobot dan *Similarity*, Mengurutkan Nilai *Similarity*, dan Menampilkan Judul Tugas Akhir Yang Telah Diurutkan. Pada DFD Level 1 ini, konsep *Information Retrieval* berjalan hingga perhitungan kemiripan (*similarity*). Apabila tidak ada proses pencarian maka *information retrieval* tidak berjalan.



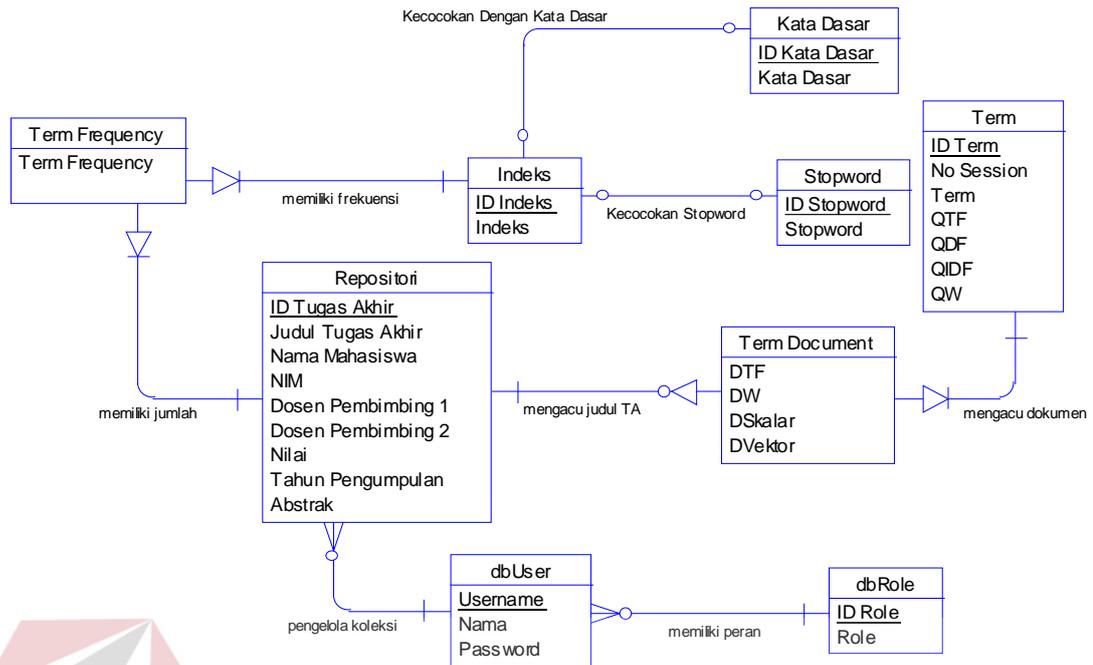
Gambar 3.16 DFD Level 1 Mencari dan Menampilkan Koleksi

### 3.3.4 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) menggambarkan basis data yang ada pada aplikasi pencarian. ERD dalam pengelolaan ini akan dibagi menjadi 2, yakni *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM).

#### 1. *Conceptual Data Model* (CDM)

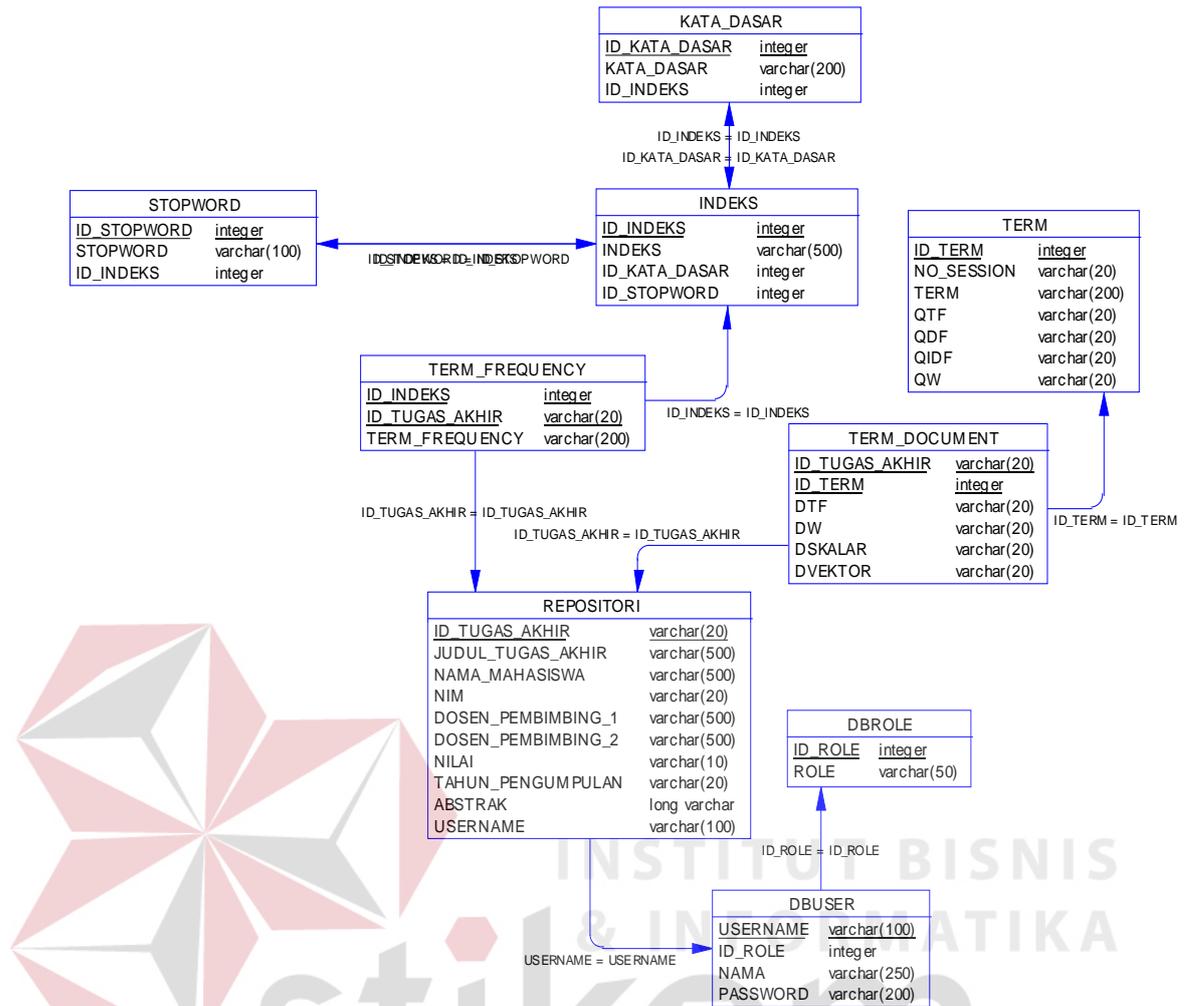
Gambar 3.17 menunjukkan struktur basis data dari aplikasi yang akan dibangun. Dengan mempersiapkan tabel-tabel dan kolom-kolom untuk kebutuhan penyimpanan data, dapat mempermudah dan menjadi acuan pembuatan struktur basis data. Pada aplikasi ini akan disiapkan sembilan tabel dengan masing-masing tabel terdapat sejumlah kolom. Setiap tabel memiliki kolom sebagai *primary key* sebagai pembeda dari setiap baris pada tabel yang sama. Selain terdapat tabel-tabel dan kolom-kolom, terdapat juga hubungan antar tabel atau *relationship* dengan jenis yang berbeda-beda. Terdapat *relationship one-to-many* dan *dependent*. Salah satu contoh *relationship one-to-many* adalah antara tabel “dbUser” dengan “dbRole”. Dalam hal ini penjelasan menurut notasi *relationship one-to-many* adalah satu data pada tabel “dbUser” hanya dapat memiliki satu data pada tabel “dbRole” dan satu data pada tabel “dbRole” dapat memiliki lebih dari satu data pada tabel “dbUser”. Sedangkan untuk *relationship dependent* merupakan hubungan ketergantungan. Isi tabel yang satu bergantung pada data tabel yang lain. Contohnya adalah tabel “TermFrequency” bergantung pada tabel “Indeks”. Isi dari tabel “TermFrequency” akan ada jika isi tabel “Indeks” tersedia atau sebaliknya.



Gambar 3.17 CDM Aplikasi Pencarian

## 2. *Physical Data Model (PDM)*

Gambar 3.18 *Physical Data Model (PDM)* merupakan hasil *generate* dari CDM dimana bentuk konsep dari struktur basis data aplikasi dikembangkan menjadi bentuk yang lebih jelas. Aliran dari *primary key* dan jenis data dari tiap kolom tergambar dengan jelas dan memudahkan pembuat basis data untuk membangunnya. PDM inilah yang akan menjadi landasan utama dalam pembuatan basis data aplikasi yang akan dibangun.



Gambar 3.18 PDM Aplikasi Pencarian

### 3.3.5 Struktur Tabel

Struktur tabel merupakan penjabaran dan penjelasan dari basis data. Dalam struktur tabel dijelaskan fungsi dari masing-masing tabel hingga fungsi masing-masing kolom yang ada di dalam tabel. Selain itu juga terdapat tipe data dari masing-masing *field* beserta konstrainnya.

#### 1. Tabel DBUSER

*Primary Key* : USERNAME

*Foreign Key* : ID\_ROLE

Fungsi : Menyimpan daftar pengguna aplikasi

Tabel 3.1 DBUSER

No	Field	Type Data	Length	Constraint
1.	USERNAME	Varchar	100	PK
2.	ID_ROLE	Integer	-	FK
3.	NAMA	Varchar	250	-
4.	PASSWORD	Varchar	200	-

## 2. Tabel DBROLE

*Primary Key* : ID\_ROLE

*Foreign Key* : -

Fungsi : Menyimpan daftar *role* atau batasan akses pengguna

Tabel 3.2 DBROLE

No	Field	Type Data	Length	Constraint
1.	ID_ROLE	Integer	-	PK
2.	ROLE	Varchar	50	-

## 3. Tabel INDEKS

*Primary Key* : ID\_INDEKS

*Foreign Key* : ID\_KATA\_DASAR, ID\_STOPWORD

Fungsi : Menyimpan daftar kata indeks dari abstrak dan pencarian

Tabel 3.3 INDEKS

No	Field	Type Data	Length	Constraint
1.	ID_INDEKS	Integer	-	PK
2.	INDEKS	Varchar	500	-
3.	ID_KATA_DASAR	Integer	-	FK
4.	ID_STOPWORD	Integer	-	FK

## 4. Tabel TERM\_FREQUENCY

*Primary Key* : ID\_TERM\_FREQUENCY

*Foreign Key* : ID\_INDEKS, ID\_TUGAS\_AKHIR

Fungsi : Menyimpan frekuensi dari istilah (*term*)

Tabel 3.4 TERM\_FREQUENCY

No	Field	Tipe Data	Length	Constraint
1.	ID_TERM_FREQUENCY	Integer	-	PK
2.	ID_INDEKS	Integer	-	FK
3.	ID_TUGAS_AKHIR	Varchar	20	FK
4.	TERM_FREQUENCY	Varchar	200	-

#### 5. Tabel TERM

*Primary Key* : ID\_TERM

*Foreign Key* : -

Fungsi : Menyimpan daftar *term* dan perhitungan frekuensi dengan *query* (Q) pencarian.

Tabel 3.5 TERM

No	Field	Tipe Data	Length	Constraint
1.	ID_TERM	Integer	-	PK
2.	NO_SESSION	Varchar	20	-
3.	TERM	Varchar	200	-
4.	QTF	Varchar	20	-
5.	QDF	Varchar	20	-
6.	QIDF	Varchar	20	-
7.	QW	Varchar	20	-

#### 6. Tabel REPOSITORY

*Primary Key* : ID\_TUGAS\_AKHIR

*Foreign Key* : USERNAME

Fungsi : Menyimpan repository tugas akhir

Tabel 3.6 REPOSITORI

No	Field	Type Data	Length	Constraint
1.	ID_TUGAS_AKHIR	Integer	-	PK
2.	JUDUL_TUGAS_AKHIR	Varchar	500	-
3.	NAMA_MAHASISWA	Varchar	500	-
4.	NIM	Varchar	20	-
5.	DOSEN PEMBIMBING_1	Varchar	500	-
6.	DOSEN PEMBIMBING_2	Varchar	500	-
7.	NILAI	Varchar	10	-
8.	TAHUN PENGUMPULAN	Varchar	20	-
9.	ABSTRAK	Long Varchar	-	-
10.	USERNAME	Varchar	100	-

## 7. Tabel TERM\_DOCUMENT

*Primary Key* : -

*Foreign Key* : ID\_TUGAS\_AKHIR, ID\_TERM

*Fungsi* : Menyimpan daftar *term* dan perhitungan frekuensi dengan dokumen tugas akhir (D)

Tabel 3.7 TERM\_DOCUMENT

No	Field	Type Data	Length	Constraint
1.	ID_TUGAS_AKHIR	Varchar	20	FK
2.	ID_TERM	Integer	-	FK
3.	DTF	Varchar	20	-
4.	DW	Varchar	20	-
5.	DSKALAR	Varchar	20	-
6.	DVEKTOR	Varchar	20	-

## 8. Tabel KATA\_DASAR

*Primary Key* : ID\_KATA\_DASAR

*Foreign Key* : ID\_INDEKS

*Fungsi* : Menyimpan daftar kata dasar Bahasa Indonesia

Tabel 3.8 KATA\_DASAR

No	Field	Type Data	Length	Constraint
1.	ID_KATA_DASAR	Integer	-	PK
2.	KATA_DASAR	Varchar	200	
3.	ID_INDEKS	Integer	-	FK

## 9. Tabel STOPWORD

*Primary Key* : ID\_STOPWORD

*Foreign Key* : ID\_INDEKS

Fungsi : Menyimpan daftar *stopword* Bahasa Indonesia

Tabel 3.9 STOPWORD

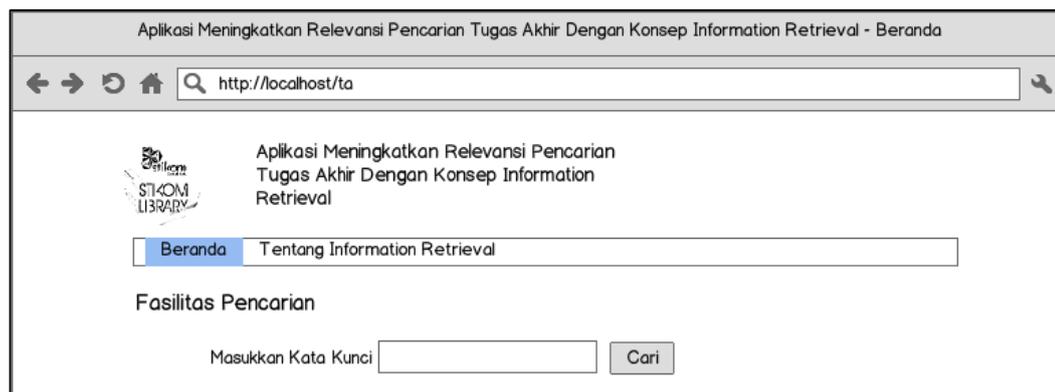
No	Field	Type Data	Length	Constraint
1.	ID_STOPWORD	Integer	-	PK
2.	STOPWORD	Varchar	100	
3.	ID_INDEKS	Integer	-	FK

### 3.3.6 Desain *Interface* dan *Input/Output* (I/O)

Penggambaran desain tatap muka (*interface*) dan *input/output* (I/O) sangat membantu pengembang aplikasi untuk menentukan posisi komponen interaktif dengan penggunanya. Dengan menggambarkan desain ini, akan lebih jelas penggunaan aplikasi yang akan dibangun.

#### 1. Desain tampilan halaman utama pencarian

Desain tampilan halaman utama pencarian dibuat sesederhana mungkin sehingga pengguna lebih memahami fungsi dari halaman yang dimaksud. Desain tampilan halaman utama pencarian dapat dilihat pada Gambar 3.19.



Gambar 3.19 Rancangan Tampilan Halaman Utama Pencarian

## 2. Desain *form login* pengguna

*Form login* pengguna berfungsi untuk memberikan otentikasi terhadap pengguna yang berinteraksi langsung dengan aplikasi berdasarkan peran/*role* yang diberikan. Jika *role* sebagai mahasiswa, maka pengguna hanya dapat melihat halaman hasil pencarian dan dapat mengunduh *file* tugas akhir hasil pencarian. Jika *role* sebagai petugas pengolah koleksi tugas akhir, maka akan muncul menu baru berupa halaman pengolahan koleksi tugas akhir. Jika pengguna tidak melakukan otentikasi atau *login* hanya dapat melakukan pencarian tanpa bisa mengunduh *file* tugas akhir. Tampilan dapat dilihat pada Gambar 3.20.

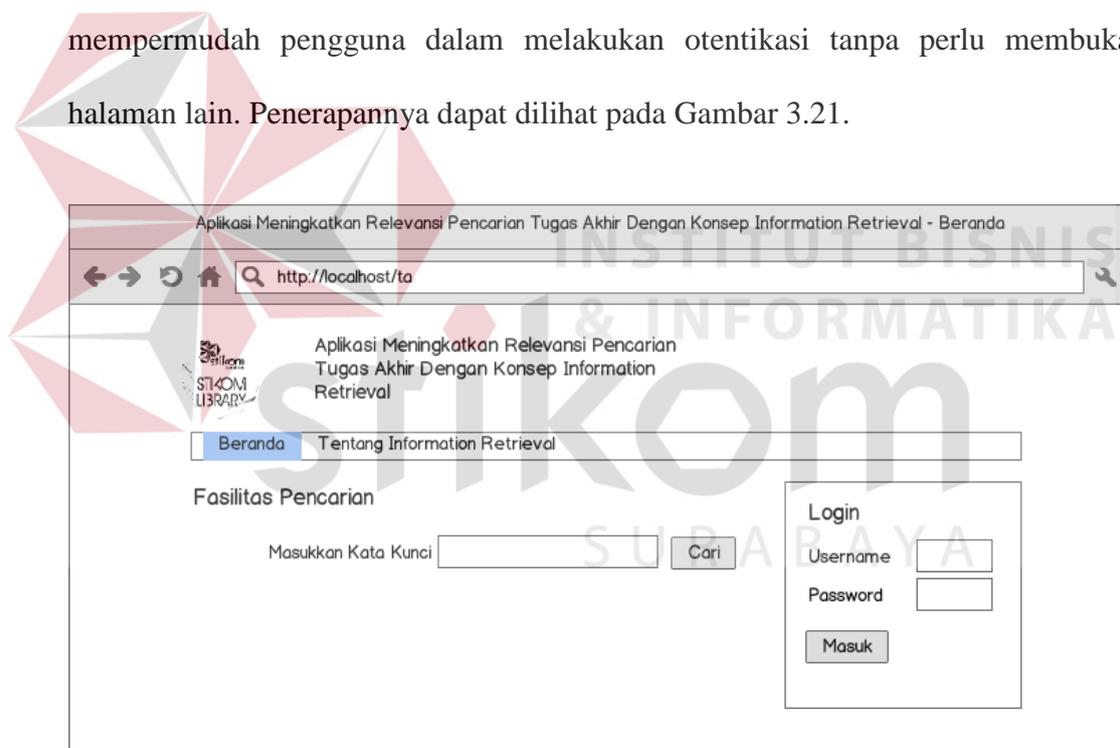
Pada Gambar 3.20 dapat dilihat bahwa pengguna wajib mengisi *username* dan *password* yang sesuai dengan *database* pengguna. Apabila pengguna salah memasukkan maka akan muncul tampilan terdapat kesalahan *login*. Jika benar maka akan muncul data pengguna pada bagian kanan layar menggantikan tampilan *form login*.



A wireframe of a login form titled "Login". It contains two input fields: "Username" and "Password". Below the input fields is a button labeled "Masuk".

Gambar 3.20 Rancangan Tampilan *Form Login* Pengguna

Dalam penerapan pada aplikasi nantinya, tampilan *form login* akan disatukan dengan halaman utama pencarian. Hal ini dimaksudkan untuk mempermudah pengguna dalam melakukan otentikasi tanpa perlu membuka halaman lain. Penerapannya dapat dilihat pada Gambar 3.21.



Gambar 3.21 Rancangan Penerapan *Form Login* Pada Halaman Utama

### 3. Desain tampilan pengolahan koleksi tugas akhir

Tampilan *form* pengolahan koleksi tugas akhir berfungsi untuk pengguna petugas pengolah koleksi untuk menambah atau mengubah daftar koleksi tugas akhir yang ada. Selain itu berfungsi untuk melihat daftar koleksi tugas akhir yang

telah disimpan selama ini dengan mencari berdasarkan judul, nama atau NIM.

Tampilan *form* pengolahan koleksi tugas akhir dapat dilihat pada Gambar 3.22.

The screenshot shows a web browser window with the URL `http://localhost/ta/pengolahan/`. The page title is "Aplikasi Meningkatkan Relevansi Pencarian Tugas Akhir Dengan Konsep Information Retrieval - Pengolahan Koleksi TA". The navigation menu includes "Beranda", "Pengolahan Koleksi", "Tentang Information Retrieval", and "Logout".

The main section is titled "Koleksi Tugas Akhir". It features a search area with a dropdown menu labeled "Cari Tugas Akhir" and a "Cari" button. The dropdown menu options are "Berdasarkan", "Judul", "Nama", "NIM", "Tahun", and "Dosen".

Below the search area is a table with the following columns: "Judul TA", "Pengarang", "Tahun", "NIM", "Abstrak", "Pembimbing 1", "Pembimbing 2", and "Edit". The table contains four rows of placeholder data (xxxx).

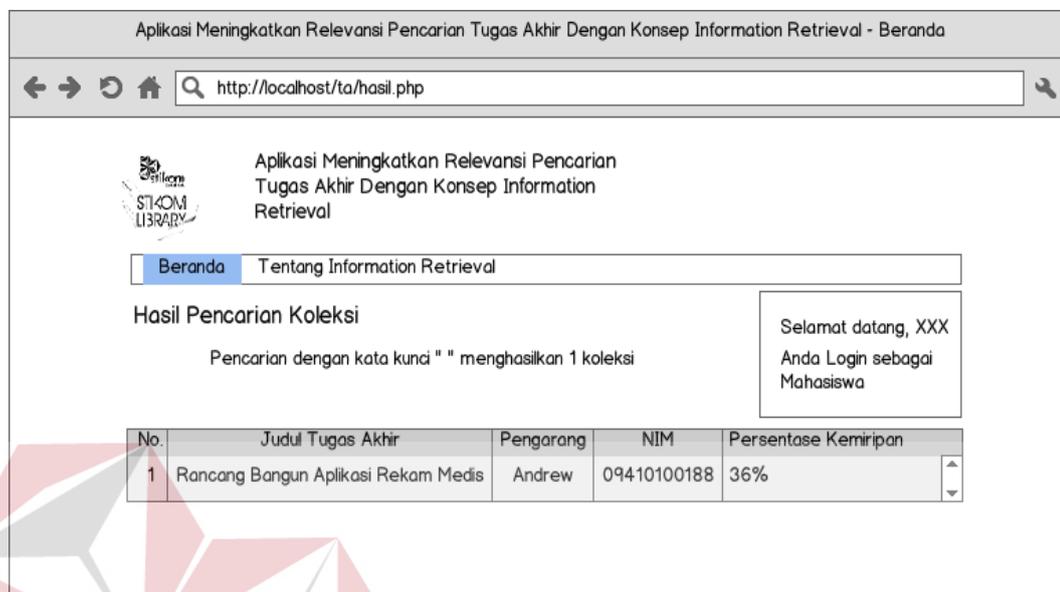
Below the table is a form titled "Tambah / Edit Koleksi Tugas Akhir". The form fields include: "Judul Tugas Akhir", "Pengarang", "NIM", "Tahun Terbit", "Abstrak" (with a text area), "File Abstrak" (with an "Upload" button), "Dosen Pembimbing 1", and "Dosen Pembimbing 2". A "Simpan / Ubah" button is located at the bottom of the form.

Gambar 3.22 Rancangan Tampilan *Form* Pengolahan Koleksi Tugas Akhir

#### 4. Desain tampilan hasil pencarian

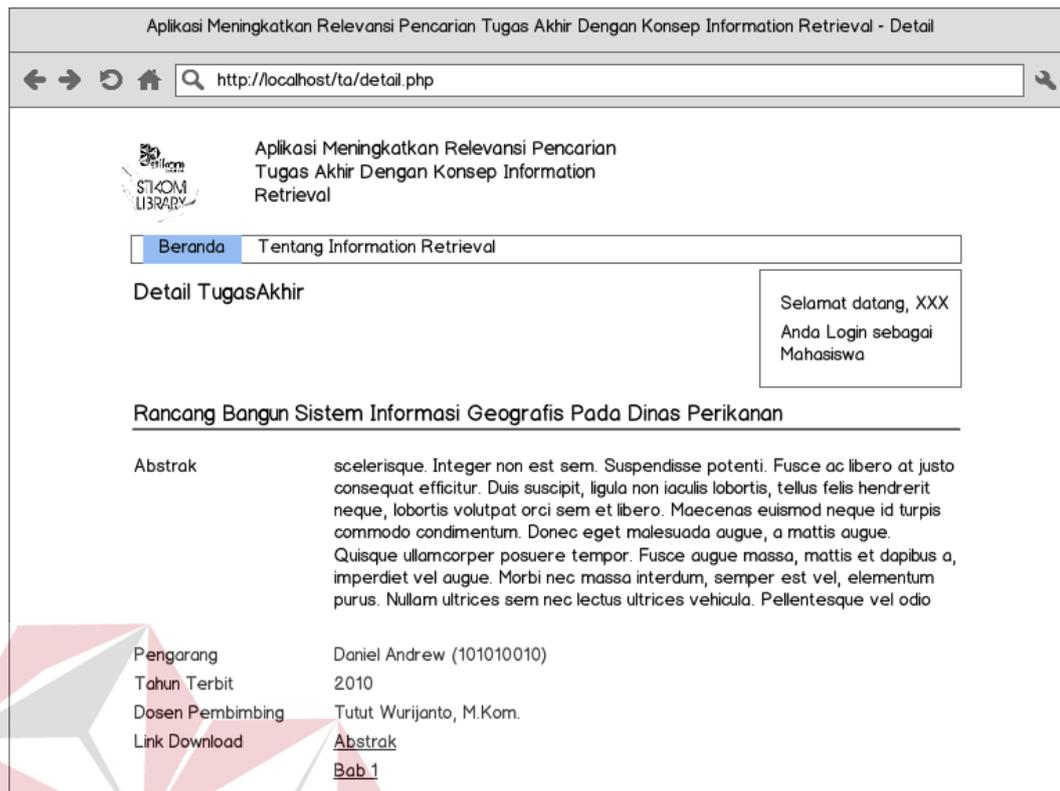
Tampilan ini berfungsi untuk menampilkan hasil pencarian yang telah dilakukan pengguna. Hasil pencarian berupa daftar judul tugas akhir beserta nama pengarang dan NIM. Setiap judul merupakan *link* tautan ke dalam halaman detail

tugas akhir dan halaman untuk mengunduh *file* tugas akhir jika pengguna telah melakukan *login*. Tampilan hasil pencarian dapat dilihat pada Gambar 3.23.



Gambar 3.23 Rancangan Tampilan Hasil Pencarian

Pengguna dapat memilih salah satu judul dan aplikasi akan membuka halaman baru yang berisikan detail dari judul tugas akhir yang dipilih dan *link* untuk mengunduh *file* tugas akhir. Apabila pengguna tidak melakukan *login* sebelumnya, maka *link* unduhan tidak akan aktif atau tidak tersedia. Tampilan detail judul tugas akhir dan *link* unduhan dapat dilihat pada Gambar 3.24.



Gambar 3.24 Rancangan Halaman Detail Tugas Akhir dan *Link* Unduhan

### 3.3.7 Perancangan Uji Coba

Perancangan pengujian digunakan sebagai pedoman untuk menguji aplikasi dan memastikan aplikasi berjalan sesuai dengan rancangan dan memenuhi keinginan pengguna serta apakah aplikasi telah mencapai tujuan awal dari penelitian yaitu tingkat relevansi yang meningkat. Metode pengujian menggunakan metode *Black Box* dengan lingkup pengujian mencakup pengujian fungsional dan pengujian terhadap pencapaian tujuan awal penelitian yaitu meningkatkan relevansi pencarian. Tabel 3.10 dan Tabel 3.11 menjelaskan rancangan pengujian fungsional terhadap aplikasi pencarian tugas akhir.

Untuk mewakili jumlah data tugas akhir yang ada pada perpustakaan sebanyak 2.754 judul tugas akhir, diperlukan sebuah sampel. Dengan

menggunakan rumus sampel Slovin dan menetapkan tingkat kesalahan 5%,

didapatkan nilai  $n = \frac{2754}{1+2754(0,05)^2} = 349,3 \approx 350$  sampel judul tugas akhir.

Pengujian terhadap peningkatan relevansi hasil pencarian dilakukan perbandingan hasil pencarian dengan menggunakan aplikasi dan pencarian menggunakan *query*. Untuk menentukan banyaknya pengujian kata kunci digunakan rumus sampel Slovin dengan menentukan populasi sebanyak 350 dan tingkat kesalahan 5% ditemukan jumlah sampel percobaan yang ditemukan sebanyak 190 percobaan dengan kata kunci yang berbeda-beda. Perancangan pengujian tingkat relevansi dapat dilihat pada Tabel 3.12.

Tabel 3.10 Rancangan Pengujian Pengolahan Koleksi Tugas Akhir

Objek Pengujian		Halaman Utama dan Halaman Pengolahan Koleksi Tugas Akhir		
Keterangan		Mengetahui apakah fungsi dalam pengolahan koleksi tugas akhir sudah berjalan dengan rencana.		
No	Tujuan Pengujian	Masukan	Keluaran	Hasil Pengujian
1.	Menampilkan menu pengolahan koleksi tugas akhir.	<i>Username</i> dan <i>password</i> khusus untuk petugas pengolah koleksi tugas akhir.	Muncul menu Manajemen Koleksi disertai dengan sub-menu Pengolahan Koleksi.	
2.	Menampilkan halaman fasilitas pengunggahan <i>file</i> tugas akhir.	Pengguna memilih sub-menu Pengolahan Koleksi.	Muncul halaman pengunggahan <i>file</i> tugas akhir.	
3.	Mengunggah <i>file</i> tugas akhir.	Pengguna memilih <i>file</i> dokumen tugas akhir dengan menekan tombol	Nama <i>file</i> dokumen tugas akhir muncul di samping tombol “Browse”.	

Objek Pengujian	Halaman Utama dan Halaman Pengolahan Koleksi Tugas Akhir			
Keterangan	Mengetahui apakah fungsi dalam pengolahan koleksi tugas akhir sudah berjalan dengan rencana.			
No	Tujuan Pengujian	Masukan	Keluaran	Hasil Pengujian
		“Browse”		
4.	Memastikan kotak pengisian Abstrak terisi sesuai dengan isi dokumen abstrak tugas akhir.	Pengguna menekan tombol “Muat” pada halaman pengunggahan <i>file</i> dokumen tugas akhir.	Aplikasi menampilkan halaman pengisian identitas tugas akhir dengan kotak isian Abstrak sudah terisi sesuai dengan isi dokumen abstrak secara otomatis	
5.	Menyimpan identitas tugas akhir.	Pengguna menekan tombol Simpan dengan kondisi <i>text box</i> tidak terisi.	Aplikasi memberikan peringatan pada <i>text box</i> yang wajib diisi.	
		Pengguna menekan tombol Simpan dengan kondisi <i>text box</i> terisi sempurna.	Aplikasi menyimpan koleksi tugas akhir pada basis data Repositori dan aplikasi menampilkan halaman pengunggahan dokumen tugas akhir	

Tabel 3.11 Rancangan Pengujian Pencarian dan Menampilkan Koleksi

Objek Pengujian	Halaman utama, halaman hasil pencarian, dan halaman detail tugas akhir.			
Keterangan	Mengetahui tampilan dan fungsi yang terdapat dalam halaman hasil pencarian berjalan dan menghasilkan keluaran yang diharapkan.			
No	Tujuan Pengujian	Masukan	Keluaran	Hasil Pengujian
1.	Menguji pemeriksaan karakter pencarian.	Tidak memasukkan karakter <i>keyboard</i> apa saja dan menekan tombol “Cari”.	Muncul peringatan wajib mengisi <i>textbox</i> pencarian.	
		Memasukkan karakter <i>keyboard</i> dan menekan tombol “Cari”.	Muncul halaman hasil pencarian judul koleksi tugas akhir disertai dengan daftar judul jika tersedia.	
2.	Menguji hasil pencarian.	Klik tombol “Cari”	Muncul daftar judul tugas akhir dengan indentitas disertai dengan persentase <i>similarity</i> yang diurutkan dari <i>similarity</i> tertinggi.	
		Klik tombol “Show Detail”	Muncul hasil proses perhitungan <i>information retrieval</i> .	

Tabel 3.12 Rancangan Pengujian Peningkatan Relevansi Pencarian

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
1.	sistem informasi industri kecil			
2.	aturan tindak pidana korupsi			
3.	sistem informasi nilai akademik perguruan tinggi			
4.	penilaian kinerja karyawan			
5.	penjadwalan cleaning service			
6.	penilaian kinerja pegawai			
7.	sistem informasi non akademik			
8.	sistem informasi perawatan pasien			
9.	sistem pemilihan rumah			
10.	sistem pendukung keputusan pemilihan jurusan perguruan tinggi			
11.	prototype produk komputasi akuntansi keuangan			
12.	aplikasi pendukung keputusan penentuan tindak pidana			
13.	monitoring perkembangan pendidikan anak sekolah			
14.	sistem pendukung keputusan pemilihan karyawan teladan			
15.	perencanaan kebutuhan material			
16.	sistem informasi penjualan			
17.	penentuan rute pengiriman barang			
18.	sistem pendukung keputusan pemilihan supplier material			
19.	sistem informasi program kerja pengadaan barang			
20.	sosialisasi kebijakan perusahaan teknologi informasi			
21.	identifikasi penyakit tanaman jeruk			
22.	perhitungan biaya pembangunan			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
	rumah			
23.	menentukan makanan sehat			
24.	penentuan kebijakan persediaan awal			
25.	peramalan pengadaan alat tulis kantor			
26.	pelacakan posisi			
27.	pemantauan kualitas sinyal GSM			
28.	sistem informasi perpustakaan			
29.	sistem pakar penyelesaian kasus			
30.	sistem informasi penilaian kinerja karyawan			
31.	pengiriman telur			
32.	manajemen pengolahan data obat			
33.	aplikasi pengiriman barang			
34.	quality operation			
35.	monitoring sapi perah			
36.	sistem informasi pembelian dan penjualan restoran			
37.	mining association rules			
38.	aplikasi teleradiologi			
39.	sistem informasi perkuliahan			
40.	peramalan penjualan			
41.	sistem pendukung keputusan open position forex			
42.	sistem informasi eksekutif akademik			
43.	aplikasi rekam medis			
44.	sistem informasi dana bos			
45.	analisa tingkat kepuasan pelanggan			
46.	menentukan program diet			
47.	sistem penilaian kinerja			
48.	penentuan calon pelamar kerja			
49.	sistem pendistribusian barang			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
50.	sistem informasi jadwal kapal sandar			
51.	mendeteksi penyakit lupus			
52.	aplikasi supply chain management			
53.	sistem pembelajaran interaktif			
54.	sistem pendukung keputusan perbaikan kualitas produksi			
55.	sistem informasi keamanan ruangan			
56.	sistem pendukung keputusan kelayakan kredit modal			
57.	sistem pendukung keputusan penilaian kompetensi			
58.	sistem informasi perbankan			
59.	sistem informasi travelling			
60.	mendiagnosa penyakit hati liver			
61.	klasifikasi sms suara pelanggan			
62.	monitoring tugas akhir			
63.	pendeteksi penyusup jaringan			
64.	sistem nilai mahasiswa			
65.	customer relationship management promosi film baru			
66.	pengelompokan pelanggan potensial			
67.	monitor kinerja salesman			
68.	penanggulangan bencana			
69.	penentuan rencana anggaran biaya			
70.	mengidentifikasi kerusakan excavator			
71.	prediksi resiko kredit			
72.	peramalan penjualan tekstil			
73.	search engine abstraksi tugas akhir			
74.	penentuan portofolio optimal			
75.	pengukuran kinerja perusahaan			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
76.	restorasi citra			
77.	alat bantu generate kelas			
78.	analisa unjuk kerja jaringan			
79.	pemilihan alat kontrasepsi			
80.	seleksi pemain sepak bola			
81.	pembelajaran multimedia interaktif			
82.	kerangka dokumen outsourcing			
83.	pemberian rekomendasi pemilihan lokasi bisnis			
84.	order bahan baku			
85.	data mining penjualan			
86.	optimalisasi kinerja pelatihan			
87.	aplikasi general ledger			
88.	penjadwalan pilot penerbangan			
89.	pengelompokan member potential			
90.	penentuan prioritas reparasi kapal			
91.	monitoring perangkat keras komputer			
92.	aplikasi transaksi penjualan katalog			
93.	aplikasi menentukan profitabilitas			
94.	aplikasi pemetaan kompetensi			
95.	sistem penerimaan karyawan			
96.	sistem informasi akuntansi			
97.	sistem informasi persediaan bahan baku			
98.	pemilihan rute loper koran			
99.	sistem pendukung keputusan kelayakan polisi memegang senjata api			
100.	pelayanan pelanggan pasang baru			
101.	proses penentuan jabatan karyawan			
102.	penentuan tender angkutan			
103.	penentuan skala prioritas penagihan piutang			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
104.	sistem informasi rekam medik pasien			
105.	aplikasi sistem absensi			
106.	evaluasi belajar online			
107.	sistem informasi posyandu			
108.	perhitungan gizi balita			
109.	sistem informasi pemesanan barang			
110.	promosi katalog buku baru			
111.	penentuan daerah potensial operasi taksi			
112.	penentuan pemilihan hotel			
113.	sistem penjadwalan produksi			
114.	penentuan siswa unggulan			
115.	peramalan sebaran organisme pengganggu tumbuhan			
116.	aplikasi komunikasi			
117.	menilai kelayakan investasi			
118.	konversi huruf latin jawa			
119.	terapi medis pasien			
120.	mendiagnosa penyakit imunisasi			
121.	perencanaan pernikahan			
122.	sistem informasi diet golongan darah			
123.	sistem informasi administrasi paroki			
124.	sistem informasi persewaan peralatan pesta			
125.	penilaian portofolio sertifikasi guru			
126.	perhitungan nilai manfaat pensiun			
127.	aplikasi penyedia data akreditasi perguruan tinggi			
128.	aplikasi perancangan campuran beton			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
129.	aplikasi pembacaan tuna netra			
130.	pemilihan kamera digital			
131.	aplikasi tagihan telepon rumah			
132.	sistem pakar pemilihan produk			
133.	aplikasi pelayanan member			
134.	optimalisasi sistem parkir			
135.	aplikasi simulasi antrian pelayanan			
136.	pemilihan operator telekomunikasi			
137.	sistem informasi prediksi realisasi			
138.	pendukung keputusan prioritas pemberangkatan			
139.	aplikasi monitoring gangguan listrik			
140.	aplikasi perhitungan biaya listrik			
141.	aplikasi komik			
142.	pemilihan tanaman hias			
143.	diagnosa penyakit jantung			
144.	sistem informasi layanan publik			
145.	pendukung keputusan pemilihan wisata			
146.	pendukung keputusan pembelian buku baru			
147.	pendukung keputusan pemilihan pelatihan			
148.	aplikasi pengukuran produktivitas			
149.	perencanaan pengadaan rumah sakit			
150.	aplikasi kamus dinamis			
151.	sistem perwalian			
152.	aplikasi pemilihan transportasi umum			
153.	penentuan harga pokok produksi			
154.	aplikasi diagnosis kerusakan komputer			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
155.	sistem informasi penjadwalan produksi			
156.	pemantauan peternakan ayam			
157.	optimalisasi perencanaan pengadaan komputer			
158.	monitoring keberadaan orang			
159.	sistem informasi pengiriman air minum			
160.	aplikasi penunjuk waktu lokasi tuna netra			
161.	sistem informasi pelelangan			
162.	aplikasi panduan informasi lokasi perguruan tinggi			
163.	aplikasi penentuan penyakit tanaman			
164.	sistem administrasi pasien			
165.	penjadwalan pengangkutan sampah			
166.	sistem informasi permintaan pembelian			
167.	diagnosis penyakit mata			
168.	aplikasi simulasi tes toefl			
169.	sistem informasi pendataan jemaat gereja			
170.	sistem informasi monitoring evaluasi demam berdarah			
171.	aplikasi manajemen surat			
172.	sistem informasi manajemen peternakan ayam			
173.	aplikasi mikrolet di malang			
174.	sistem informasi kepegawaian			
175.	sistem informasi hotel			
176.	aplikasi pembelajaran sandi pramuka			
177.	analisis simulasi pengaruh nilai			

Objek Pengujian	Hasil pencarian melalui perintah <i>query</i> pada DBMS dan hasil pencarian melalui aplikasi..			
Keterangan	Membandingkan jumlah hasil pencarian melalui perintah <i>query</i> dengan jumlah hasil pencarian oleh aplikasi.			
No.	Kata Kunci	Hasil Pencarian Melalui <i>Query</i>	Hasil Pencarian Melalui Aplikasi	Persentase
	akademik			
178.	monitoring evaluasi kinerja mesin			
179.	sistem informasi pemeliharaan kereta api			
180.	perancangan proses bisnis			
181.	aplikasi penilaian posisi karate			
182.	sistem informasi pengelolaan aset			
183.	sistem informasi penimbangan kendaraan			
184.	sistem pembelian barang			
185.	sistem informasi penggajian			
186.	sistem informasi pengarsipan berkas			
187.	pengendalian berkas			
188.	aplikasi pelayanan jasa perawatan mobil			
189.	pengendalian persediaan obat			
190.	aplikasi penentuan harga jual			