

**SISTEM PENCARIAN AYAT AL-QUR'AN  
BERDASARKAN TERJEMAHAN  
BAHASA INDONESIA  
DENGAN PEMODELAN RUANG VEKTOR**

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat  
Untuk Memperoleh Gelar Sarjana Teknik Pada  
Jurusan Teknik Informatika

Oleh :

**IMELDA SUKMA WULANDARI**  
**10751000284**



**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU  
PEKANBARU  
2013**

**SISTEM PENCARIAN AYAT AL-QUR'AN  
BERDASARKAN TERJEMAHAN BAHASA INDONESIA  
DENGAN PEMODELAN RUANG VEKTOR**

**IMELDA SUKMA WULANDARI**

**10751000284**

Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sultan Syarif Kasim Riau

**ABSTRAK**

Media pembelajaran agama pada saat ini sebenarnya banyak tersedia dalam bentuk pengajian yang terorganisir yang mengkaji kandungan ayat-ayat Al-Qur'an. Pada penelitian ini, dibuat sistem pencarian ayat Al-qur'an berdasarkan terjemahan bahasa Indonesia menggunakan pemodelan ruang vektor. Hasil pencarian menampilkan rangking dokumen-dokumen subtopik surah, berisi beberapa ayat dan terjemahan bahasa Indonesia yang relevan berdasarkan perhitungan nilai kesamaan antara kata dalam koleksi dokumen (korpus) dan masukan teks *query* oleh pengguna. Digunakan korpus terjemahan bahasa Indonesia dan korpus *image* ayat Al-qur'an dengan format ekstensi *.png* sebagai bahan pemrosesan temu kembali ayat. Pemrosesan teks pada tahapan preprosesing mengimplentasikan *stemming* dengan algoritma Nazief dan Adriani. Pengindeksian dilakukan dengan metode *term frequency-inverse document frequency* dengan menghitung frekuensi kemunculan kata pada tiap dokumen subtopik yang ditentukan sebelumnya sejumlah 800 buah, membentuk suatu vektor multidimensi dimana *term* yang terdapat pada dokumen dan kata kunci dilengkapi dengan nilai bobot masing-masing *term*. Sistem temu kembali informasi ini diimplementasikan menggunakan bahasa pemrograman Php dengan DBMS MySQL. Pengujian performansi dilakukan menggunakan *precision* dan *recall*. Dari hasil pengujian diketahui keakuratan dan kerelevanan pengembalian dokumen dari sistem yang telah dibangun dengan persentase rata-rata *precision* adalah 74% dan *recall* 100%, yang jumlah koleksi dokumen, *query* dan panjang *query* yang dimasukan pengguna sangat berpengaruh pada hasil pencarian.

Kata kunci: Pemodelan Ruang Vektor, *Recall* dan *Precision*, Sistem Temu kembali Informasi.

## DAFTAR ISI

HALAMAN JUDUL .....	i
LEMBAR PERSETUJUAN .....	ii
LEMBAR PENGESAHAN .....	iii
LEMBAR HAK ATAS KELAYAKAN INTELEKTUAL .....	iv
LEMBAR PERNYATAAN .....	v
LEMBAR PERSEMBAHAN .....	vi
ABSTRAK .....	vii
<i>ABSTRACT</i> .....	viii
KATA PENGANTAR .....	ix
DAFTAR ISI .....	xi
DAFTAR GAMBAR .....	xiv
DAFTAR TABEL .....	xvi
DAFTAR RUMUS .....	xviii
DAFTAR LAMPIRAN .....	xix
BAB I PENDAHULUAN .....	I-1
1.1. Latar Belakang .....	I-1
1.2. Rumusan Masalah .....	I-3
1.3. Batasan Masalah.....	I-4
1.4. Tujuan Pembahasan .....	I-4
1.5. Sistematika Penulisan .....	I-5
BAB II LANDASAN TEORI .....	II-1
2.1. Korpus .....	II-1
2.2. <i>Information Retrieval System</i> .....	II-1
2.3. <i>Arsitektur Information Retrieval</i> .....	II-4

2.4. Model dalam <i>Information Retrieval</i> .....	II-10
2.4.1. Model <i>Boolean</i> .....	II-10
2.4.2. Model Probabilistik .....	II-10
2.4.3. Model Ruang Vektor .....	II-11
2.5. Kualitas <i>Text Retrieval</i> .....	II-15
2.6. Pengujian Untuk Menilai Kemampuan Sistem .....	II-18
<b>BAB III METODOLOGI PENELITIAN</b> .....	<b>III-1</b>
3.1. Identifikasi Masalah .....	III-2
3.2. Perumusan Masalah.....	III-2
3.3. Studi Literatur .....	III-2
3.4. Analisa Sistem .....	III-2
3.5. Pengembangan Perangkat Lunak .....	III-4
3.5.1. Perancangan Sistem .....	III-4
3.6. Implementasi Sistem .....	III-4
3.7. Pengujian Sistem .....	III-5
3.8. Kesimpulan dan Saran.....	III-5
<b>BAB IV ANALISA DAN PERANCANGAN</b> .....	<b>IV-1</b>
4.1. Analisa Permasalahan .....	IV-1
4.2. Analisa Sistem .....	IV-4
4.2.1. Analisa Masukan ( <i>Input</i> ) .....	IV-4
4.2.2. Analisa Proses .....	IV-5
4.2.3. Analisa Keluaran ( <i>Output</i> ) .....	IV-5
4.3. Analisa Tahapan <i>Information Retrieval System</i> .....	IV-6
4.4. Perancangan Sistem .....	IV-17
4.4.1. <i>Context Diagram</i> .....	IV-17
4.4.2. Diagram Alir Data ( <i>Data Flow Diagram</i> ) .....	IV-18
4.5. Perancangan <i>Database</i> .....	IV-27
4.6. Perancangan Tampilan Menu Sistem .....	IV-31
<b>BAB V IMPLEMENTASI DAN PENGUJIAN</b> .....	<b>V-1</b>
5.1. Implementasi .....	V-1
5.1.1. Kebutuhan Perangkat Keras.....	V-1

5.1.2. Kebutuhan Perangkat Lunak .....	V-2
5.1.3. Hasil Implementasi <i>Interface</i> Sistem .....	V-2
5.2. Pengujian Kemampuan Sistem.....	V-12
5.2.1. Rencana Pengujian .....	V-12
5.2.2. Hasil Pengujian .....	V-13
5.2.3. Kesimpulan Pengujian Unjuk Kerja Sistem .....	V-28
BAB VI PENUTUP .....	VI-1
6.1. Kesimpulan.....	VI-1
6.2. Saran.....	VI-2
DAFTAR PUSTAKA .....	xx
DAFTAR RIWAYAT HIDUP	

## DAFTAR GAMBAR

Gambar	Halaman
2.1. Ilustrasi Sistem Temu Kembali Informasi .....	II-2
2.2. Arsitektur Dasar Sistem Temu Kembali Informasi (Cios, 2007) .....	II-5
2.3. Contoh <i>Vector Space Model</i> dengan Dua Dokumen $D_1$ , dan $D_2$ , serta <i>Query</i> Pengguna $Q_1$ .....	II-12
2.4. Representasi Grafis Sudut Vektor Dokumen dan <i>Query</i> .....	II-14
2.5. Hubungan Antar Dokumen yang Relevan .....	II-15
3.1. Tahapan Penelitian .....	III-1
4.1. Konteks Diagram .....	IV-18
4.2. <i>DFD</i> Level 1 Proses 1 .....	IV-19
4.3. <i>DFD</i> Level 2 Proses 2 .....	IV-21
4.4. <i>DFD</i> Level 2 Proses 3 .....	IV-22
4.5. <i>DFD</i> Level 2 Proses 4 .....	IV-23
4.6. <i>DFD</i> Level 2 Proses 5 .....	IV-24
4.7. Rancangan <i>Entity Relationship Diagram</i> .....	IV-26
4.8. Rancangan Menu Awal Pencarian Ayat Al-Qur'an .....	IV-32
4.9. Rancangan Menu Preposesing Korpus .....	IV-32
4.10. Rancangan Menu Daftar Korpus .....	IV-33
4.11. Rancangan Menu Daftar <i>Stopword</i> .....	IV-33
4.12. Rancangan Menu Daftar Tanda Baca .....	IV-34
4.13. Rancangan Menu Kata Dasar .....	IV-34
4.14. Rancangan Menu <i>Indexong</i> .....	IV-35
4.15. Rancangan Menu Pembobotan .....	IV-36
4.16. Rancangan Menu <i>Retrieval</i> Admin .....	IV-36
5.1. Menu Halaman Utama .....	V-2
5.2. Menu Preposesing Korpus .....	V-3

5.3. Tabel Quran .....	V-4
5.4. Tabel <i>Stem</i> .....	V-4
5.5. Menu Daftar Korpus .....	V-5
5.6. Tabel Surah .....	V-5
5.7. Menu Daftar <i>Stopword</i> .....	V-6
5.8. Menu Daftar Tanda Baca .....	V-6
5.9. Menu Kata Dasar.....	V-7
5.10. Menu <i>Indexing</i> .....	V-7
5.11. Tabel Dokumen.....	V-8
5.12. Tabel <i>Indexing</i> .....	V-9
5.13. Menu Pembobotan .....	V-9
5.14. Menu <i>Retrieval Admin</i> .....	V-10
5.15. Tabel <i>wjxwiq</i> .....	V-11
5.16. Dokumen Subtopik Hasil Pencarian .....	V-12
5.17. Tampilan hasil pencarian untuk <i>query</i> "kisah musa" .....	V-14
5.18. Tampilan hasil pencarian untuk <i>query</i> "berpuasa di bulan ramadhan"	V-16
5.19. Tampilan hasil pencarian untuk <i>query</i> "Allah menciptakan langit dan bumi" .....	V-18
5.20. Tampilan hasil pencarian untuk <i>query</i> "pohon zaitun yang menghasilkan minyak" .....	V-20
5.21. Grafik interpolasi <i>recall</i> dan <i>precision</i> pada <i>query</i> "kisah musa" .....	V-22
5.22. Grafik interpolasi <i>recall</i> dan <i>precision</i> pada <i>query</i> "berpuasa di bulan ramadhan" .....	V-23
5.23. Grafik interpolasi <i>recall</i> dan <i>precision</i> pada <i>query</i> "Allah menciptakan langit dan bumi" .....	V-23
5.24. Grafik interpolasi <i>recall</i> dan <i>precision</i> pada <i>query</i> "pohon zaitun yang menghasilkan minyak" .....	V-24

## DAFTAR TABEL

<b>Tabel</b>	<b>Halaman</b>
2.1. Parameter untuk menghitung <i>precision</i> dan <i>recall</i> .....	II-16
4.1. Nilai <i>IDF</i> ( <i>Inverse Document Frequency</i> ).....	IV-10
4.2. Nilai Bobot <i>Term</i> .....	IV-12
4.3. Nilai Normalisasi Bobot <i>Term</i> .....	IV-13
4.4. <i>Query</i> .....	IV-15
4.5. Bobot <i>Query</i> .....	IV-16
4.6. Normalisasi Bobot <i>Query</i> .....	IV-16
4.7. Deskripsi Diagram Konteks .....	IV-18
4.8. Deskripsi <i>DFD</i> Level 1 Proses <i>Login</i> .....	IV-20
4.9. Deskripsi <i>DFD</i> Level 1 Proses Preposesing .....	IV-20
4.10. Deskripsi <i>DFD</i> Level 1 Proses Daftar Korpus.....	IV-20
4.11. Deskripsi <i>DFD</i> Level 1 Proses <i>Indexing</i> .....	IV-20
4.12. Deskripsi <i>DFD</i> Level 1 Proses <i>Retrieval</i> .....	IV-21
4.13. Deskripsi <i>DFD</i> Level 2 Proses 2.1 Preposesing .....	IV-21
4.14. Deskripsi <i>DFD</i> Level 2 Proses 3.1 Pilih Surah.....	IV-22
4.15. Deskripsi <i>DFD</i> Level 2 Proses 3.2 Daftar Korpus.....	IV-22
4.16. Deskripsi <i>DFD</i> Level 2 Proses 4.1 Pilih Subtopik.....	IV-23
4.17. Deskripsi <i>DFD</i> Level 2 Proses 4.2 <i>Indexing</i> .....	IV-23
4.18. Deskripsi <i>DFD</i> Level 2 Proses 5.1 Preposesing <i>Query</i> .....	IV-24
4.19. Deskripsi <i>DFD</i> Level 2 Proses 5.2 <i>Similarity</i> .....	IV-25
4.20. Deskripsi <i>DFD</i> Level 2 Proses 5.3 <i>Retrieval</i> Dokumen .....	IV-25
4.21. Admin.....	IV-27
4.22. Tandabaca .....	IV-27
4.23. Surah .....	IV-27
4.24. <i>Stopword</i> .....	IV-28
4.25. Quran.....	IV-28
4.26. <i>Stemlist</i> .....	IV-28



4.27. <i>Stem</i> .....	IV-29
4.28. Dokumen.....	IV-29
4.29. <i>Indexing</i> .....	IV-30
4.30. <i>Bobotq</i> .....	IV-30
4.31. <i>Normq</i> .....	IV-31
4.32. <i>wjxwiq</i> .....	IV-31
5.1. Perangkat Keras yang Digunakan .....	V-1
5.2. Perangkat Lunak yang Digunakan .....	V-2
5.3. Daftar <i>Query</i> yang Dimasukkan .....	V-13
5.4. Perbandingan Hasil Perhitungan <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_1$ ..	V-14
5.5. Perbandingan Hasil Perhitungan <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_2$ ...	V-16
5.6. Perbandingan Hasil Perhitungan <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_3$ ...	V-18
5.7. Perbandingan Hasil Perhitungan <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_4$ ...	V-21
5.8. Hasil Pengujian <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_1$ .....	V-25
5.9. Hasil Pengujian <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_2$ .....	V-25
5.10. Hasil Pengujian <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_3$ .....	V-26
5.11. Hasil Pengujian <i>precision(P)</i> dan <i>recall(R)</i> pada $Q_4$ .....	V-26
5.12. Presentasi Seluruh Kemampuan Sistem pada <i>Query</i> .....	V-27

## DAFTAR RUMUS

<b>Rumus</b>	<b>Halaman</b>
2.1. <i>Invers Document Frequency</i> .....	II-9
2.2. <i>Bobot Term</i> .....	II-9
2.3. <i>Normalisasi Panjang Term</i> .....	II-13
2.4. <i>Pembobotan Query</i> .....	II-13
2.5. <i>Normalisasi Query</i> .....	II-13
2.6. <i>Similarity</i> .....	II-14
2.7. <i>Precision</i> .....	II-15
2.8. <i>Recall</i> .....	II-15
2.9. <i>Recall non interpolasi</i> .....	II-16
2.10. <i>Precision non interpolasi</i> .....	II-16
2.11. <i>Rata-rata (Mean)</i> .....	II-17
2.12. <i>Persentil</i> .....	II-17

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Ajaran atau tuntunan sebuah agama pada umumnya dituangkan dalam bentuk kitab suci. Al-Qur'an adalah kitab suci agama Islam yang akan selalu terjaga kemurniannya. Allah SWT telah berjanji bahwa Dia yang menurunkan Al-Qur'an dan Dia juga yang menjaganya. Ini ada di dalam QS. Al-Hijr (15) ayat 9. Al-Qur'an merupakan firman Allah SWT yang dibawa malaikat Jibril AS kepada Nabi Muhammad SAW untuk menjadi petunjuk bagi seluruh umat manusia. Al-Qur'an yang terdiri atas 30 juz, 114 surat, dan 6236 ayat, dan ditulis dalam bahasa Arab, menjadi hambatan bagi kebanyakan umat Islam di Indonesia mengetahui kandungan topik dari ayat yang dibacanya.

Para pengembang perangkat lunak telah mencoba mendigitalkan dokumen Al-Qur'an, membuat transliterasi dan terjemahannya ke dalam berbagai bahasa. Beberapa perangkat lunak buatan Indonesia, seperti <http://id.lidwa.com/app/>, juga sudah menyertakan fitur pencarian terhadap ayat Al-Qur'an sesuai dengan masalah yang akan dicari. Namun pada aplikasi ini menyediakan strategi pencarian *query* berbasis kata (*word-matching*) yang menampilkan daftar hasil (*search result*) hanya berupa dokumen ayat berdasarkan faktor kesamaan antara *query* dan terjemahan ayat dalam *database* yang mengandung kata yang ada dalam *query*. Hal ini menyebabkan ada ayat relevan dalam koleksi yang tidak dipanggil karena tidak memuat kata yang ada dalam *query*.

Pendayagunaan sistem temu kembali informasi (*Information Retrieval System*) menjadi penting agar dapat menghemat waktu dan kerja untuk mendapatkan ayat yang relevan dengan kata kunci (*query*) yang dimasukkan oleh pengguna. Untuk itu dalam penelitian ini, akan dibangun suatu media pengaksesan ayat Al-Qur'an digital terjemahan Indonesia yang mengadaptasi dari cara kerja *search engine* yang merupakan salah satu aplikasi dari disiplin ilmu IR (*Information Retrieval*) yang mengkaji metode-metode pencarian di dalam

*database*, berdasarkan representasi kebutuhan informasi yang relevan berupa kata kunci, yaitu *query* atau *keyword*. Peningkatan kinerja dapat dilakukan dengan menerapkan *stemming*. *Stemming* merupakan proses konversi *term* ke bentuk dasarnya. Hal ini bisa dilakukan dengan menghilangkan akhiran atau awalan dari sebuah kata. Menurut Agusta, penggunaan *stemming* dengan algoritma Nazief dan Adriani memiliki kemampuan presentase keakuratan (*precision*) lebih baik dibanding algoritma lainnya.

Menurut Salton, terdapat 3 model yang dapat digunakan dalam representasi *Information Retrieval System* yaitu model *boolean*, model *probabilistic*, dan model ruang vektor. Dari ketiga model tersebut, model ruang vektor merupakan pemodelan yang paling sederhana dan paling produktif. Model ruang vektor merepresentasikan setiap dokumen dan *query* pengguna sebagai suatu vektor *n*-dimensi. Tiap dimensi pada vektor tersebut diwakili oleh satu *term*. *Term* yang digunakan biasanya berpatokan pada *term* yang ada pada *query*, sehingga *term* yang ada pada dokumen tetapi tidak ada pada *query* biasanya diabaikan.

Pendayagunaan IR (*Information Retrieval*) memiliki manfaat yang menjanjikan, namun mesin pencari tidak selalu memberikan informasi yang akurat. Kekurangan ini dapat disebabkan karena bahasa teks terjemahan Al-Qur'an memiliki perbedaan dengan bahasa dokumen lainnya. Dalam terjemahan Al-Qur'an khususnya dalam bahasa Indonesia, ada kaidah-kaidah yang perlu diperhatikan. Kaidah-kaidah tersebut seperti redaksi yang bersifat umum, dhamir (kata ganti), logika bahasa, tujuan kalimat, dan konteks kalimat. Pemahaman terhadap terjemahan bahasa Al-Qur'an harus memperhatikan konteks ayat, konteks kalimat yang berhubungan dengan maksud ayat (Rahmat A., 2020). Kaidah-kaidah dan juga pemahaman Al-Qur'an dapat menjadikan pengguna tidak memasukan *query* sesuai permintaannya dengan benar. Keterbatasan bahasa dan pengguna untuk mengungkapkan kebutuhan informasinya dalam *query* dapat menyebabkan sistem memberikan kinerja yang buruk. Oleh karena itu diperlukan solusi khusus dalam mengatasi kemungkinan kesalahan pengguna dalam memahami bahasa Al-Qur'an.

Penelitian yang menggunakan pemodelan ruang vektor pada *information retrieval* adalah penelitian yang dilakukan oleh Heru Adi Darmawan. Pada penelitian ini, Heru menggunakan pemodelan ruang vektor dalam pencarian terhadap teks tafsir Al-Qur'an. Penelitian *menggunakan* korpus tafsir Al-Misbah untuk surat Al-fatihah dan Al-baqarah. Masukan sistem yang digunakan *query* berbasis teks bahasa Indonesia dan *output* dari sistem berupa terjemahan ayat Al-Qur'an dan detail tafsir ayat. Berdasarkan penelitian ini, tingkat keakuratan aplikasi terhadap hasil pencarian mencapai nilai persentase 73,8%. Sedangkan tingkat kemampuan aplikasi untuk mengambil semua ayat yang relevan dengan *query* sehingga tidak ada ayat yang tertinggal atau hilang, mencapai nilai rata-rata hingga 96,3%.

Oleh karena itu dalam penelitian ini dibangun sebuah aplikasi pencarian ayat Al-Qur'an berdasarkan terjemahan bahasa Indonesia yang lengkap (30 juz) versi Departemen Agama RI dengan menggunakan konsep pemodelan ruang vektor, untuk dapat memberikan informasi relevan berupa dokumen yang menampilkan terjemahan beberapa ayat Al-Qur'an berdasarkan kandungan subtopik dari ayat Al-Qur'an agar mudah dipahami oleh pengguna. Penelitian ini juga menerapkan penggunaan *stemming* untuk mengatasi permasalahan kompleksitas bahasa terjemahan Al-Quran. Maka dari latar belakang yang telah dipaparkan sebelumnya, penelitian ini berjudul "Sistem Pencarian Ayat Al-Qur'an Berdasarkan Terjemahan Bahasa Indonesia dengan Pemodelan Ruang Vektor".

## **1.2. Rumusan Masalah**

Berdasarkan latar belakang telah diuraikan di atas, maka dapat diambil suatu rumusan masalah yaitu "Bagaimana tahapan membangun aplikasi pencarian ayat Al-Qur'an berdasarkan terjemahan bahasa Indonesia dengan konsep *Information Retrieval*."

### 1.3. Batasan Masalah

Batasan masalah dalam laporan tugas akhir ini adalah:

1. Koleksi dokumen (korpus) yang digunakan sebagai bahan penelitian analisa, implementasi, dan pengujian adalah korpus terjemahan ayat Al-Qur'an bahasa Indonesia dan korpus gambar ayat (tulisan arab), yang telah disesuaikan berdasarkan terjemahan yang dikeluarkan oleh yayasan penyelenggara pentafsir Al-Qur'an, Jakarta 1 maret 1971.
2. Masukan pada mesin pencarian ayat Al-Qur'an menggunakan *query* berbasis teks bahasa Indonesia.
3. Hasil pencarian (*search result*) akan menampilkan beberapa ayat Al-Qur'an dan terjemahan dalam satu subtopik surah Al-Qur'an sesuai *query* pengguna untuk mencegah salah tafsir.
4. Sistem tidak menggunakan perluasan *query* pengguna.
5. Sistem tidak membahas tentang pendekatan semantik dalam hubungannya dengan pemahaman kandungan tafsir pada ayat Al-Qur'an.

### 1.4. Tujuan Pembahasan

Tujuan pembahasan yang ingin dicapai oleh penulis dari penelitian dan penyusunan tugas akhir ini adalah:

1. Menganalisis koleksi dokumen (korpus) yang digunakan dalam pembangunan sistem pencarian ayat Al-Qur'an sebagai bahan untuk melakukan *indexing* pada tahapan preprosesing.
2. Menerapkan pemodelan ruang vektor pada mesin pencarian ayat Al-Qur'an sesuai dengan konsep *Information Retrieval*.
3. Mengukur kemampuan sistem dengan nilai *precision* dan *recall* dari penerapan model ruang vektor dalam pencarian ayat Al-Qur'an berdasarkan terjemahan bahasa Indonesia.

## 1.5. Sistematika Penulisan

Laporan tugas akhir ini terdiri dari enam bab, dengan sistematika penulisan tersusun atas bab:

### BAB I Pendahuluan

Membahas mengenai latar belakang, rumusan masalah, batasan masalah, tujuan pembahasan, serta sistematika penulisan.

### BAB II Landasan Teori

Membahas mengenai konsep dasar dan teori dari korpus, *information retrieval system*, arsitektur *information retrieval*, model dalam *information retrieval*, kualitas teks *retrieval*, dan pengujian untuk menilai kemampuan sistem.

### BAB III Metodologi Penelitian

Membahas mengenai bagaimana tahapan-tahapan yang akan dilakukan untuk menjawab perumusan masalah penelitian berupa identifikasi masalah, perumusan masalah, studi literatur, analisa sistem, pengembangan perangkat lunak, implementasi sistem, pengujian sistem, serta kesimpulan dan saran.

### BAB IV Analisa dan Perancangan

Membahas mengenai analisa permasalahan, analisa sistem, analisa tahapan *information retrieval system*, perancangan sistem, perancangan *database*, serta perancangan tampilan menu sistem.

### BAB V Implementasi dan Pengujian

Membahas mengenai tahapan implementasi terhadap sistem yaitu kebutuhan perangkat keras dan perangkat lunak, hasil implementasi *interface* sistem. Serta tahapan implementasi yaitu rencana pengujian, hasil pengujian dan kesimpulan pengujian untuk kerja sistem.

### BAB VI Penutup

Membahas mengenai uraian beberapa kesimpulan dari hasil yang didapatkan selama penelitian dan saran yang dapat digunakan pada penelitian selanjutnya.

## **BAB II**

### **LANDASAN TEORI**

Bab ini membahas mengenai konsep dasar dan teori-teori yang berhubungan dengan topik sistem temu kembali informasi (*Information Retrieval System*) dengan pemodelan ruang vektor.

#### **2.1. Korpus**

Korpus berasal dari bahasa Latin yang berarti “*body*” atau kumpulan yang dalam makna yang terdekat berarti kumpulan teks dari suatu variasi bahasa yang dapat semaksimal mungkin merepresentasikan suatu kecenderungan atau fenomena dari variasi tersebut. Oleh karena itu, korpus tidak terdiri beberapa teks saja, akan tetapi kumpulan teks yang bisa saja terdiri dari ratusan ribu bahkan jutaan kata untuk dapat melihat suatu fenomena bahasa.

Defenisi dari istilah korpus adalah kumpulan dari potongan-potongan teks bahasa dalam bentuk elektronik, dipilih sesuai dengan kriteria eksternal untuk mewakili suatu bahasa atau ragam bahasa sebagai sumber data untuk penelitian linguistik. (Sinclair, 2004)

#### **2.2. *Information Retrieval System***

*Information Retrieval* merupakan suatu bidang ilmu yang mempelajari tahapan temu kembali dan penelusuran dokumen. Sistem ini berfungsi untuk menemukan informasi yang relevan dengan kebutuhan pengguna.

Beberapa defenisi dari istilah *Information Retrieval*:

“*Information Retrieval* adalah proses menemukan material (biasanya dokumen) dari lingkungan bersifat tidak terstruktur (biasanya teks) yang memenuhi kebutuhan informasi dari dalam koleksi yang berukuran besar (biasanya pada komputer)” (Manning, 2009).

“*Information Retrieval* merupakan tindakan, metode, dan prosedur untuk menemukan kembali data yang tersimpan, kemudian menyediakan informasi mengenai subyek yang dibutuhkan. Tindakan tersebut mencakup *text indexing*,

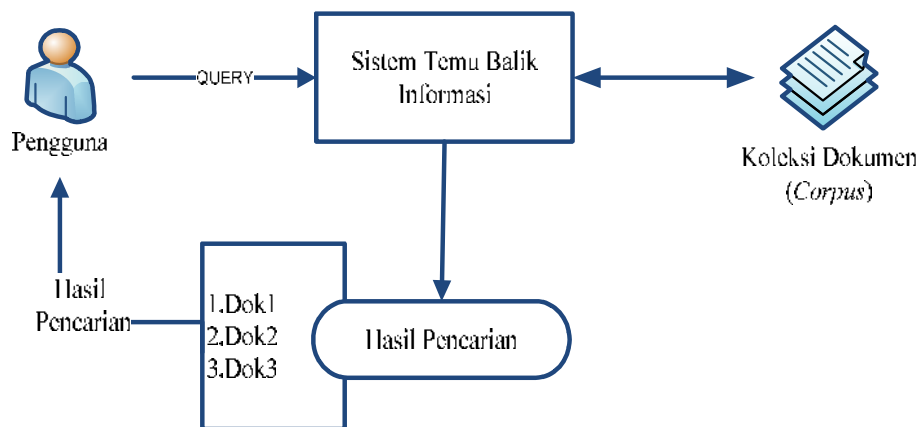


*inquiry analysis*, dan *relevance analysis*. Dokumen yang dimasukkan dapat berupa teks, tabel, diagram, suara, gambar, video.” (Cios, 2007)

Dari dua defenisi di atas dapat dikatakan *Information Retrieval* adalah pencarian informasi pada sebuah koleksi dokumen yang berukuran besar yang meliputi tindakan, metode, dan prosedur pengembalian dokumen yang relevan sesuai dengan kebutuhan pengguna.

*Information Retrieval System* digunakan untuk menemukan kembali (*retrieve*) informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis (Mandala,2006). Sistem temu kembali informasi yang baik memungkinkan pengguna menentukan secara cepat dan akurat apakah isi dari dokumen yang diterima memenuhi kebutuhannya. Agar representasi dokumen lebih baik, dokumen-dokumen dengan topik atau isi yang mirip dikelompokkan bersama-sama (Murad,2007). *Input* dari suatu sistem temu kembali informasi direpresentasikan dengan *query* dari pengguna dan koleksi dokumen, dan *output*-nya adalah dokumen yang dianggap relevan oleh sistem.

Proses dalam *Information Retrieval* dapat digambarkan sebagai sebuah proses untuk mendapatkan dokumen relevan dari koleksi dokumen yang ada melalui pencarian *query* yang dimasukkan pengguna. (Mustaqim,2008)



**Gambar 2.1. Ilustrasi Sistem Temu Kembali Informasi**

Salah satu aplikasi dari *Information Retrieval* adalah mesin pencari (*search engine*) yang dapat diterapkan di berbagai bidang. Pada mesin pencari dengan *Information Retrieval* pengguna dapat memasukkan *query* yang bebas, dalam arti kata *query* yang sesuai dengan bahasa manusia dan sistem dapat menemukan dokumen yang sesuai dengan *query* yang masukkan oleh pengguna.

Ada beberapa istilah penting yang berkaitan dengan sistem temu kembali informasi. Istilah tersebut antara lain: (Ramadhany,2009)

1. *Query*

*Query* adalah ekspresi kebutuhan akan informasi. *Query* bisa berupa kata kunci yang diberikan kepada sistem untuk mendapatkan informasi yang relevan terhadap pengguna *query*. Definisi *query* secara formal adalah kumpulan spesifikasi yang digunakan untuk menggali kumpulan data yang diperlukan dari *database*.

2. Dokumen

Dokumen adalah sebuah objek yang mengandung informasi yang akan dieksplorasi oleh sistem temu kembali informasi. Informasi pada dokumen ini dibatasi hanya berupa teks. Dari dokumen inilah kemudian sistem akan memberikan hasil pencarian.

3. Relevansi

Relevansi adalah tingkat kesesuaian sebuah dokumen dengan *query* yang dimasukkan oleh pengguna. Relevansi dihitung sesuai dengan rumus tertentu, tergantung model yang digunakan dalam sistem temu kembali informasi.

4. Ranking

Ranking adalah pengaturan dokumen-dokumen yang telah dicari sistem temu kembali informasi. Acuan dalam pengurutan ini adalah nilai relevansi tiap dokumen terkait *query* yang diberikan.

5. *Term*

*Term* adalah kata yang memiliki arti yang terdapat pada dokumen dan *query*. Dari semua kata yang ada, kata-kata yang merupakan kata umum dihilangkan sehingga yang tersisa hanya kata yang benar-benar

berhubungan dengan isi dokumen. Kata yang tersisa itulah yang disebut *term*.

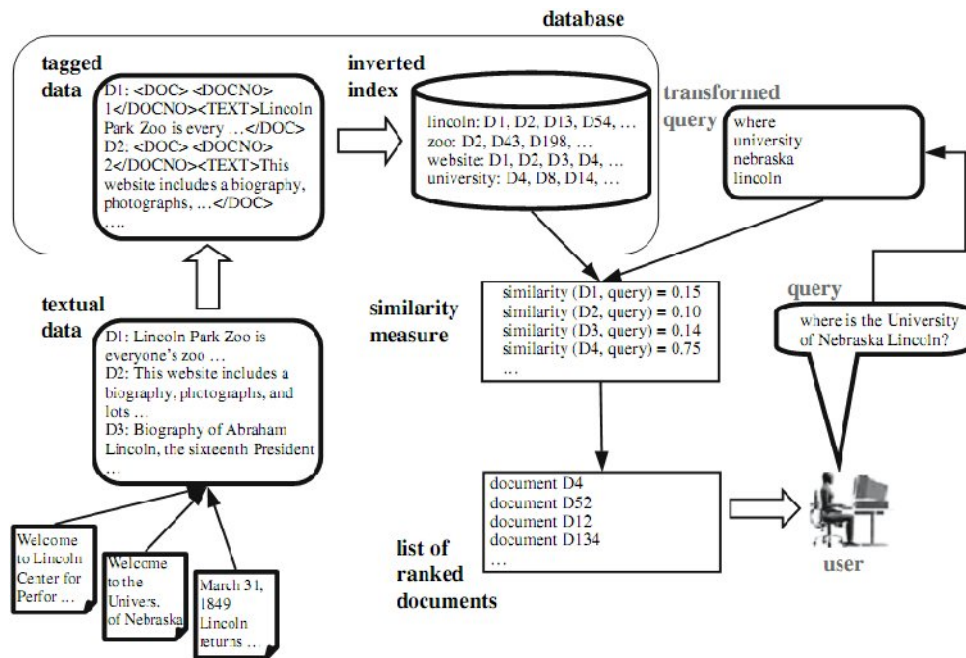
### **2.3. Arsitektur *Information Retrieval***

Pada umumnya, ada dua tahapan dalam membangun *Information Retrieval System*, yaitu melakukan preprosesing terhadap *database* dan menerapkan metode tertentu untuk menghitung relevansi antara dokumen di dalam *database* yang telah dipreprosesing dengan *query* pengguna. (Tronojoyo,2010)

Pada tahapan preprosesing, sistem yang berurusan dengan dokumen semi-*structured* biasanya memberikan *tag* tertentu pada *term* atau bagian dari dokumen, sedangkan pada dokumen tidak terstruktur proses ini dilewati dan membiarkan *term* tanpa imbuhan *tag*.

*Query* yang dimasukkan pengguna dikonversi sesuai aturan tertentu untuk mengekstrak *term-term* penting yang sejalan dengan *term* yang sebelumnya yang telah diekstrak dari dokumen dan menghitung relevansi antara *query* dan dokumen berdasarkan pada *term* tersebut. Sebagai hasilnya, sistem mengembalikan suatu daftar dokumen terurut sesuai nilai kemiripannya dengan *query* pengguna. (Cios,2007)

Setiap dokumen (termasuk *query*) direpresentasikan menggunakan model *bag-of-words* yang mengabaikan urutan dari kata-kata di dalam dokumen, struktur sintaktis dari dokumen dan kalimat. Dokumen ditransformasi ke dalam suatu “tas” berisi kata-kata independen. *Term* disimpan dalam suatu *database* pencarian khusus yang ditata sebagai sebuah *inverted index*. *Index* ini merupakan konversi dari dokumen asli yang mengandung sekumpulan kata ke dalam daftar kata yang berasosiasi dengan dokumen terkait dimana kata-kata tersebut muncul.



**Gambar 2.2. Arsitektur Dasar Sistem Temu Kembali Informasi (Cios,2007)**

Proses yang terjadi dalam *Information Retrieval System* terdiri dari 2 bagian utama, yaitu *indexing subsystem* dan *searching subsystem (matching system)*.

### 1. Proses Membangun *Indexing*

Pembangunan indeks dari koleksi dokumen merupakan tugas pokok pada tahapan *preprocessing* di dalam *Information Retrieval*. *Indexing subsystem* adalah proses merepresentasikan koleksi dokumen ke dalam bentuk tertentu untuk memudahkan dan mempercepat proses pencarian dan penemuan kembali dokumen yang relevan.

Kualitas indeks mempengaruhi efektifitas dan efisiensi sistem temu kembali informasi. Indeks dokumen adalah himpunan *term* yang menunjukkan isi atau topik yang dikandung oleh dokumen. Indeks akan membedakan suatu dokumen dari dokumen lain yang berada di dalam koleksi. Ukuran indeks yang kecil dapat memberikan hasil buruk dan mungkin beberapa *item* yang relevan terabaikan. Indeks yang besar memungkinkan ditemukan banyak dokumen yang

relevan tetapi sekaligus dapat menaikkan jumlah dokumen yang tidak relevan dan menurunkan kecepatan pencarian (*searching*).

Menurut Manning, dkk (2009) dan Garcia (2006) terdapat 5 langkah pembangunan *inverted index*, yaitu :

1. Mengumpulkan dokumen yang akan diindeks (dikenal dengan nama korpus/koleksi dokumen).

2. Penghapusan format dan *markup* dari dalam dokumen.

Pada tahap ini semua *tag markup* dan format khusus dihapus dari dokumen jika ada, terutama pada dokumen yang mempunyai banyak *tag* dan format seperti dokumen HTML.

3. Pemisahan rangkaian kata (*tokenization*).

Pada tahapan ini, seluruh kata di dalam kalimat ataupun paragraf dipisahkan menjadi potongan kata tunggal. Tahapan ini juga menghilangkan karakter-karakter tertentu seperti tanda baca dan mengubah semua kata ke bentuk huruf kecil (*lowercase*).

4. Melakukan *linguistic preprocessing* untuk menghasilkan *token/term* yang telah dinormalisasi. Dua hal yang dilakukan dalam tahap ini adalah:

- a. Penyaringan (*filtration*)

Pada tahapan ini ditentukan *term* mana yang akan digunakan untuk merepresentasikan dokumen sehingga dapat mendeskripsikan isi dokumen dan membedakan dokumen tersebut dari dokumen lain di dalam koleksi. *Term* yang sering dipakai tidak dapat digunakan untuk tujuan ini karena dua alasan. Pertama, jumlah dokumen yang relevan terhadap suatu *query* kemungkinan besar merupakan bagian kecil dari koleksi. *Term* yang efektif dalam pemisahan dokumen yang relevan dari dokumen tidak relevan kemungkinan besar adalah *term* yang muncul pada sedikit dokumen. Kedua, *term* yang muncul dalam banyak dokumen tidak mencerminkan definisi dari topik atau sub-topik dokumen. Karena itu, *term* yang sering digunakan dianggap sebagai *stopwords* dan dihapus dari dokumen.

b. *Stemming*

*Stemming* adalah proses konversi *term* ke bentuk dasarnya. Hal ini bisa dilakukan dengan cara menghilangkan akhiran atau awalan dari sebuah kata. Tidak banyak algoritma yang dikhususkan untuk *stemming* bahasa Indonesia dengan berbagai keterbatasan di dalamnya, diantaranya:

- a. Algoritma Porter, Algoritma ini membutuhkan waktu yang lebih singkat dibandingkan dengan *stemming* menggunakan Algoritma Nazief dan Adriani, namun proses *stemming* menggunakan Algoritma Porter memiliki presentase keakuratan (presisi) lebih kecil dibandingkan dengan *stemming* menggunakan Algoritma Nazief dan Adriani.
- b. Algoritma Nazief dan Adriani, algoritma *stemming* untuk teks berbahasa Indonesia yang memiliki kemampuan presentase keakuratan (presisi) lebih baik dari algoritma lainnya. Algoritma ini sangat dibutuhkan dan menentukan dalam proses sistem temu kembali informasi dalam dokumen Indonesia. Algoritma Nazief dan Adriani mengacu pada aturan morfologi bahasa Indonesia yang mengelompokkan imbuhan, yaitu imbuhan yang diperbolehkan atau imbuhan yang tidak diperbolehkan. Pengelompokan ini termasuk imbuhan di depan (awalan), imbuhan kata di belakang (akhiran), imbuhan kata di tengah (sisipan) dan kombinasi imbuhan pada awal dan akhir kata (konfiks).

Algoritma ini menggunakan kamus kata keterangan yang digunakan untuk mengetahui bahwa proses *stemming* telah mendapatkan kata dasar (Nazief, B.A.A. dan Andriani, M. 1996).

Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani ini memiliki tahap-tahap sebagai berikut :

1. Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tersebut adalah *root word*, maka algoritma berhenti.
2. *Inflection suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa partikel (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus *possesive pronouns* (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus *Derivation suffixes* (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a
  - a. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
  - b. Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
4. Hapus *Derivation prefix*. Jika pada langkah 3 ada *sufiks* yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
  - a. Periksa tabel kombinasi awalan-akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
  - b. *For i = 1 to 3*, tentukan tipe awalan kemudian hapus awalan. Jika *root word* belum juga ditemukan lakukan langkah 5.
5. Jika sudah maka algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama maka algoritma berhenti. Melakukan *recoding*.
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

5. Membangun *indexing* dan pemberian bobot terhadap *term*.

Dalam pencarian yang dilakukan oleh sistem temu kembali informasi, semua *term* yang dicari tidak memiliki bobot yang sama. Untuk itulah dibutuhkan metode pembobotan *term* agar pencarian lebih mudah difokuskan. Pada model ruang vektor, pembobotan terhadap *term* dilakukan dengan mengalikan bobot *tf* dan bobot *idf*, dikenal dengan pembobotan *tf-idf*. Metode pembobotan ini dilakukan dengan memberikan bobot kepada *term* yang penting. Artinya, *term* yang jika muncul di suatu dokumen maka, dokumen tersebut dapat dianggap relevan dengan *query* pengguna.

Term frequency

*Term frequency (tf)* adalah jumlah kemunculan sebuah *term* pada sebuah dokumen. Jika sebuah *term i* sering muncul pada sebuah dokumen, maka *query* yang mengandung *term* harus mendapatkan dokumen tersebut. Nilai sebuah *tf* dihitung berdasarkan kemunculan *term* dalam dokumen.

Inverse Document Frequency

*Inverse Document Frequency (idf)* adalah jumlah dokumen yang mengandung sebuah *term* yang dicari dari kumpulan dokumen yang ada.

$$Idf(i) = \frac{1}{N(i)} \dots\dots\dots(2.1)$$

Dimana:

N = jumlah seluruh dokumen pada koleksi dokumen

(i) = jumlah dokumen koleksi yang mengandung term i

Bobot Term

Pembobotan *tf x idf* untuk sebuah *term i* untuk dokumen *j* didapatkan dari hasil perkalian *tf* dan *idf*.

$$, = , x \dots\dots\dots(2.2)$$



## 2.4. Model dalam *Information Retrieval*

Berkaitan dengan representasi sistem informasi temu kembali, Salton menjelaskan bahwa terdapat 3 model yang dapat digunakan, yakni model *Boolean*, model probabilistik, dan model ruang vektor. (Arifin,2002)

### 2.4.1. Model *Boolean*.

Salah satu model sistem temu kembali informasi yang paling awal digunakan adalah model *boolean*. Model *boolean* merepresentasikan dokumen sebagai suatu himpunan kata-kunci (*set of keywords*). Sedangkan *query* direpresentasikan sebagai ekspresi *boolean*. *Query* dalam ekspresi *boolean* merupakan kumpulan kata kunci yang saling dihubungkan melalui operator *boolean* seperti *AND*, *OR* dan *NOT* serta menggunakan tanda kurung untuk menentukan *scope* operator. Hasil pencarian dokumen dari model *boolean* adalah himpunan dokumen yang relevan.

Kekurangan dari model *boolean* ini antara lain:

- a. Hasil pencarian dokumen berupa himpunan, sehingga tidak dapat dikenali dokumen-dokumen yang paling relevan atau agak relevan (*partial match*).
- b. *Query* dalam ekspresi *boolean* dapat menyulitkan pengguna yang tidak mengerti tentang ekspresi *boolean*. (Mandala,2006)

### 2.4.2. Model Probabilistik

Model probabilistik dasar adalah model sistem temu kembali informasi yang mengurutkan dokumen dalam urutan menurun terhadap peluang relevansi sebuah dokumen terhadap informasi yang dibutuhkan. (Ramadhany, 2009)

Pada model probabilistik dasar ini, perhitungan dilakukan untuk tiap kata yang ada pada *query*. Dari perhitungan tersebut akan diketahui nilai bobot dari kata-kata yang dicari. Kemudian untuk tiap dokumen, akan dijumlahkan nilai logaritma dari bobot tiap kata yang dimiliki oleh dokumen tersebut. Dokumen yang paling relevan didapatkan dengan jumlah nilai yang paling tinggi.

Kekurangan dari model *boolean* ini antara lain:

- a. Informasi relevansi harus diketahui pada awal pencarian.
- b. Pembobotan tidak dipengaruhi oleh *term frequency*.

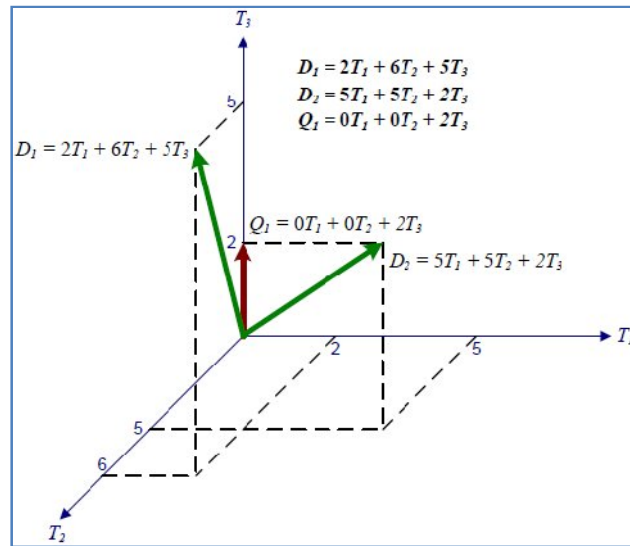
- c. Membutuhkan kemampuan komputasi yang tinggi.
- d. Mengasumsikan *term* tidak saling berkaitan.

### 2.4.3. Model Ruang Vektor

Menurut Salton, model ruang vektor merupakan model paling sederhana dan paling produktif. Model vektor ini merepresentasikan *term* yang digunakan baik oleh dokumen maupun oleh *query* (Arifin,2002). Dalam *Information Retrieval System*, kemiripan antar dokumen didefinisikan berdasarkan representasi *bag-of-words* dan dikonversi ke suatu model ruang vektor (*Vector Space Model*). Pada model ruang vektor, setiap dokumen dan *query* pengguna direpresentasikan oleh suatu vektor  $n$ -dimensi. Tiap dimensi pada vektor tersebut diwakili oleh satu *term*. *Term* yang digunakan biasanya berpatokan pada *term* yang ada pada *query*, sehingga *term* yang ada pada dokumen tetapi tidak ada pada *query* biasanya diabaikan. (Mustaqim,2008)

*Vocabulary* merupakan kumpulan semua *term* berbeda yang tersisa dari dokumen setelah *preprocessing* dan mengandung  $t$  *term* index. Kumpulan *term* ini membentuk suatu ruang vektor.

1. Setiap *term*  $i$  di dalam dokumen atau *query*  $j$ , diberikan suatu bobot (*weight*) bernilai real  $w_{ij}$ .
2. Dokumen dan *query* diekspresikan sebagai vektor  $t$  dimensi  $d_j = (w_{1j}, w_{2j}, \dots, w_{tj})$  dan terdapat  $n$  dokumen didalam koleksi, yaitu  $j = 1, 2, \dots, n$ .
3. Berikut merupakan contoh dari *vector space model* tiga dimensi untuk dua dokumen  $d_1$  dan  $d_2$ , satu *query* pengguna  $q$ , dan tiga *term*  $t_1, t_2, t_3$  :



**Gambar 2.3. Contoh *Vector Space Model* dengan Dua Dokumen , dan , serta *Query* Pengguna**

Berbeda dengan model *boolean* yang menggunakan nilai biner sebagai bobot indeks *term*, model ruang vektor melakukan pembobotan berdasarkan *term* yang sering muncul dalam dokumen atau dikenal dengan sebutan *term frequency* (*tf*) dan jumlah kemunculannya dalam koleksi dokumen yang disebut *inverse document frequency* (*idf*). (Manning,2009)

Rumus Relevansi

Penentuan relevansi dokumen dengan *query* dipandang sebagai pengukuran kesamaan (*similarity measure*) antara vektor dokumen dengan vektor *query*. Semakin “sama” suatu vektor dokumen dengan vektor *query* maka dokumen dapat dipandang semakin relevan dengan *query*.

Model ruang vektor pada sebuah sistem temu kembali informasi sangat mementingkan *frequency* kemunculan *term* yang dicari pada dokumen. Tapi perhitungan tersebut tidak diimbangi dengan faktor panjang dokumen itu sendiri. Akibatnya pada dokumen lebih panjang sebuah *term* memiliki jumlah kemunculan lebih banyak dan kemungkinan *term* yang didapatkan semakin banyak. Hal ini menyebabkan dokumen yang lebih panjang memiliki kemungkinan lebih besar untuk muncul sebagai dokumen yang paling relevan.

Untuk itu perlu dilakukan normalisasi, yaitu membagi relevansi dokumen yang didapatkan dengan perkalian panjang vektor *query* dengan panjang vektor dokumen bersangkutan. Panjang vektor *query* atau dokumen secara matematis adalah akar kuadrat dari penjumlahan nilai kuadrat dari panjang vektor linier pembentuk *query* atau dokumen. Maka diperoleh rumus untuk normalisasi bobot *term*, menghitung bobot masing-masing *query*, serta menghitung normalisasi pembobotan *query* untuk mendapatkan pengukuran kesamaan (*similarity measure*) antara vektor dokumen dengan vektor *query*.

1. Normalisasi panjang *term*, merupakan jumlah perbandingan antara bobot *term* terhadap panjang *term*.

$$w_{t,d} = \frac{w_{t,d}}{\sum_{t \in D} w_{t,d}} \dots \dots \dots (2.3)$$

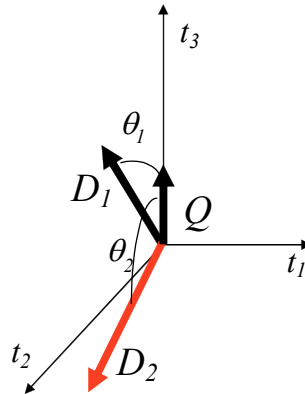
2. Pembobotan masing-masing *query*

$$w_{t,q} = 0.5 + \frac{w_{t,d}}{\sum_{t \in Q} w_{t,d}} \times idf(i) \dots \dots \dots (2.4)$$

3. Normalisasi *query*, merupakan jumlah perbandingan antara bobot *query* terhadap panjang *query*.

$$w_{q,d} = \frac{w_{t,q}}{\sum_{t \in Q} w_{t,q}} \dots \dots \dots (2.5)$$

Dalam gambar 2.3. mengilustrasikan kesamaan antara dokumen dan dengan *query* *Q*. Sudut menggambarkan kesamaan dokumen dengan *query* sedangkan menggambarkan kesamaan dokumen dengan *query*.



**Gambar 2.4. Representasi Grafis Sudut Vektor Dokumen dan *Query***

Perhitungan kesamaan antara vektor *query* dan vektor dokumen dilihat dari sudut yang kecil. Sudut yang dibentuk oleh dua buah vektor dapat dihitung dengan melakukan perkalian dalam (*inner product*), sehingga rumus relevansinya adalah:

$$R(Q,D) = \cos \theta = \frac{Q \cdot D}{|Q| |D|}$$

Dimana:

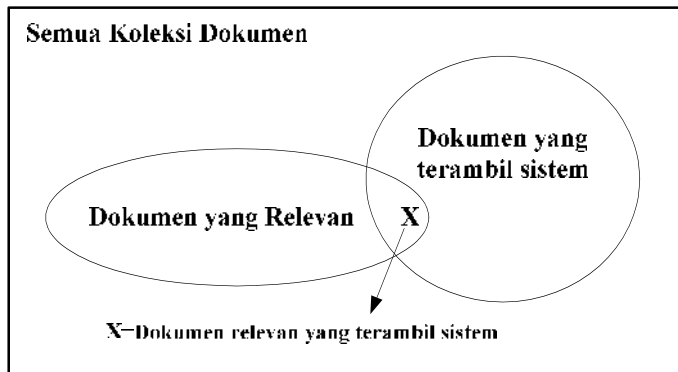
- Q = bobot *query*
- D = bobot dokumen
- |Q| = panjang *query*
- |D| = panjang dokumen

*Cosine Similarity* tidak hanya digunakan untuk menghitung normalisasi panjang dokumen tapi juga menjadi salah satu ukuran kemiripan yang populer. Ukuran ini menghitung nilai kosinus sudut antara dua vektor. Jika terdapat dua vektor dokumen *d* dan *query* *q*, serat *t term* diekstrak dari koleksi dokumen maka nilai kosinus antara *d* dan *q* didefinisikan sebagai satu faktor lagi yang disebut sebagai normalisasi panjang dokumen. Normalisasi yang digunakan adalah normalisasi kosinus. Berdasarkan rumus normalisasi kosinus yaitu :

$$R(Q,D) = \cos \theta = \frac{\sum_{t \in T} q_t \cdot d_t}{\sqrt{\sum_{t \in T} q_t^2} \sqrt{\sum_{t \in T} d_t^2}} \dots \dots \dots (2.6)$$

**2.5. Kualitas *Text Retrieval***

*Information Retrieval System* mengembalikan sekumpulan dokumen sebagai jawaban dari *query* pengguna. Terdapat dua kategori dokumen yang dihasilkan oleh sistem terkait pemrosesan *query*, yaitu dokumen yang relevan dan dokumen yang terambil sistem. Hubungan antara kedua kategori dokumen tersebut diilustrasikan pada Gambar 2.5.



**Gambar 2.5. Hubungan Antar Dokumen yang Relevan**

Ukuran umum yang digunakan untuk mengukur kualitas dari *text retrieval* adalah kombinasi *precision* dan *recall*. *Precision* mengevaluasi kemampuan sistem temu kembali informasi untuk menemukan kembali dokumen *top-ranked* yang paling relevan, dan didefinisikan sebagai persentase dokumen yang dikembalikan yang benar-benar relevan terhadap *query* pengguna. (Cios, 2007)

$$\frac{\text{Dokumen relevan yang terambil sistem}}{\text{Dokumen yang terambil sistem}} \dots\dots\dots(2.7)$$

*Recall* mengevaluasi kemampuan sistem temu kembali informasi untuk menemukan semua item yang relevan dari dalam koleksi dokumen dan didefinisikan sebagai persentase dokumen yang relevan terhadap *query* pengguna dan yang diterima.

$$\frac{\text{Dokumen relevan yang terambil sistem}}{\text{Dokumen yang Relevan}} \dots\dots\dots(2.8)$$

Pada dasarnya, nilai *precision* dan *recall* bernilai antara 0-1. Oleh karena itu, dalam suatu *Information Retrieval System* yang baik diharapkan untuk dapat memberikan nilai *precision* dan *recall* mendekati 1.

Menurut (Hinrich, 2008), pengujian kemampuan sistem *information retrieval* dilakukan dengan menghitung nilai *precision* dan *recall* berdasarkan relevanan sistem menampilkan dokumen sesuai dengan *query*. Nilai *precision* adalah keakurasian atau kecocokan ( antara permintaan informasi dengan jawaban terhadap permintaan itu) jika seseorang mencari informasi di dalam sistem, dan sistem menawarkan beberapa dokumen maka keakurasian ini sebenarnya juga adalah relevansi. Artinya, seberapa persis atau cocok dokumen tersebut untuk keperluan pencari informasi, bergantung kepada seberapa relevan dokumen tersebut bagi pengguna. Kemudian *recall* adalah proporsi jumlah dokumen yang dapat ditemu kembalikan oleh sebuah proses pencarian pada sistem temu kembali informasi. Berikut ini adalah Tabel parameter untuk menghitung *precision* dan *recall* secara non interpolasi:

**Tabel 2.1. Parameter untuk menghitung *precision* dan *recall***

Keterangan	Relevan	Tidak relevan
Terambil	<i>True positive</i> (tp)	<i>False positive</i> (fp)
Tidak terambil	<i>False negative</i> (fn)	<i>True negative</i> (tn)

Rumus untuk menghitung *precision*:

$$P = tp / (tp + fp) \dots\dots\dots (2.9)$$

Keterangan :

P = *Precision*

tp = *true positive*

fp = *false positive*

Rumus untuk menghitung *recall*:

$$R = tp / ( tp + fn) \dots\dots\dots (2.10)$$

Keterangan :

R = *Recall*

$T_p = \text{true positive}$

$F_n = \text{false negative}$

Untuk menentukan nilai rata-rata dari *precision* dan *recall*, diperlukan perhitungan statistik. Salah satu tugas statistik adalah menentukan angka yang menjadi pusat suatu distribusi. Angka/nilai yang menjadi pusat suatu distribusi disebut tendensi sentral atau kecenderungan tengah.

Pengukuran rata-rata (*mean*) digunakan untuk menentukan angka/nilai rata-rata dan secara aritmatik ditentukan dengan cara menjumlah seluruh nilai dibagi banyaknya individu. Pengukuran rata-rata dapat diterapkan dengan asumsi bahwa data yang diperoleh dari hasil pengukuran berskala interval dan rasio. (Harinaldi, 2005)

$$\text{Mean (x)} = \frac{\sum X}{N} \dots\dots\dots (2.11)$$

Dimana;

X = Jumlah nilai

N = Banyak individu atau jumlah frekuensi

Salah satu fungsi statistik yang kerap diterapkan baik dalam aktivitas riset maupun kepentingan praktis adalah menentukan/menyediakan “ukuran”, batas atau norma. Ukuran dapat dihitung dengan mengubah nilai dalam bentuk persen. Berikut cara menentukan harga persentil Jika berhadapan dengan data tunggal atau tanpa frekuensi:

$$P = \frac{i}{N} \times 100 \dots\dots\dots (2.12)$$

Dimana;

i = persentil ke berapa yang hendak dihitung

N = jumlah individu/frekuensi.



## 2.6. Pengujian Untuk Menilai Kemampuan Sistem

Menurut (Ramadhany, 2008) Pengujian kemampuan sistem merupakan pengujian untuk memperoleh nilai kesesuaian sistem yang dimasukkan pengguna. Oleh karena itu, untuk mencapai nilai keakurasian sistem ini dibutuhkan nilai perbandingan antara jumlah dokumen yang relevan dalam koleksi dokumen dengan jumlah dokumen yang berhasil ditemukan sistem ini. Untuk menginterpretasikan nilai kesesuaian, ditetapkan tiga kategori yaitu : kesesuaian rendah, sedang dan tinggi. Kemudian tolak ukur yang digunakan adalah skala interval, dengan mencari selisih kemungkinan angka kesesuaian tinggi (1) dengan kemungkinan kesesuaian rendah (0) dibagi dengan 3 sesuai dengan kategori penilaian seperti  $(1 - 0) : 3 = 0.33$

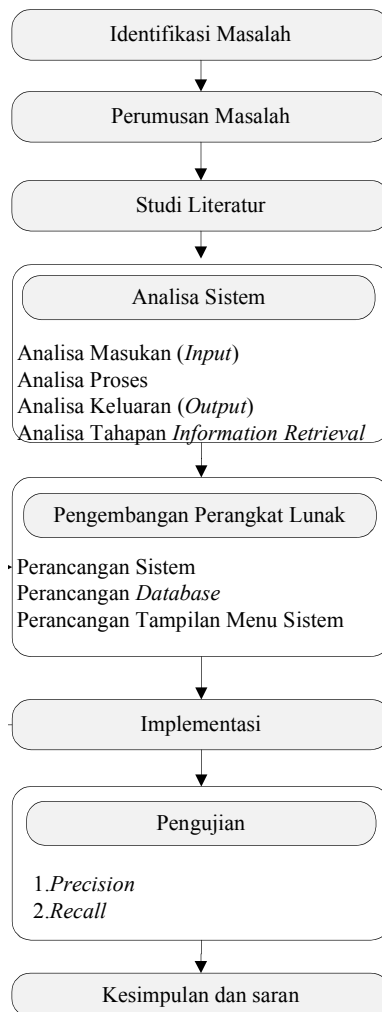
Nilai 1 dan 0 ditetapkan sesuai dengan konsep jarak kedekatan model ruang vektor dalam hal ini menggunakan algoritma genetika. Dengan demikian kelas interval dari ketiga kategori interpretasi ketepatan sistem tersebut adalah:

1. Kesesuaian rendah berada apabila angka pada rentang 0.00- 0.33.
2. Kesesuaian sedang berada apabila angka pada rentang 0.34- 0.66.
3. Kesesuaian tinggi berada apabila angka pada rentang 0.67-1.00.

### BAB 3

## METODE PENELITIAN

Metodologi penelitian menjelaskan bagaimana langkah-langkah atau tahapan-tahapan yang akan dilakukan dalam penelitian untuk dapat menjawab perumusan masalah penelitian sesuai konsep sistem temu balik informasi. Tahapan penelitian yang akan dilakukan dalam tugas akhir ini dapat dilihat pada Gambar 3.1.



**Gambar 3.1. Tahapan Penelitian**

### **3.1. Identifikasi Masalah**

Pada tahapan ini dilakukan identifikasi permasalahan bahwa pentingnya bagi *user* untuk memperoleh dokumen ayat Al-Qur'an dan terjemahan yang relevan sesuai dengan kebutuhannya sesuai dengan masukan *query* pengguna dari sekumpulan informasi (dokumen).

### **3.2. Perumusan Masalah**

Dari tahapan identifikasi masalah, didapatkan rumusan masalah tentang bagaimana proses membangun aplikasi pencarian ayat Al-Qur'an berdasarkan terjemahan bahasa Indonesia dengan konsep *Information Retrieval*.

### **3.3. Studi Literatur**

Studi literatur dilakukan dengan mempelajari konsep-konsep yang berkaitan dengan penelitian ini, seperti sistem temu balik informasi, model ruang vektor, tahapan preprosesing seperti *tokenization*, *stopwords*, *indexing*, *stemming*, pembobotan *tf-idf* melalui literatur-literatur seperti buku, jurnal, dan sumber ilmiah lain seperti situs internet ataupun artikel dokumen teks yang berhubungan.

### **3.4. Analisa Sistem**

Pada tahapan ini, ditentukan analisa masukan (*input*) untuk membangun sistem, analisa proses pada sistem, dan analisa keluaran (*output*) yang diharapkan dari sistem, serta tahapan pembangunan *information retrieval system* sebelum diterapkan Model Ruang Vektor dalam pengukuran kesamaan (*similarity measure*).

Secara garis besar tahapan pembangunan sistem pencarian ayat Al-Qur'an dalam tugas akhir ini adalah:

1. Mengumpulkan dokumen yang akan diindeks (korpus).

Koleksi dokumen (korpus) yang digunakan dalam membangun sistem adalah korpus terjemahan Al-Qur'an dalam bahasa Indonesia dan korpus gambar (tulisan arab) ayat Al-Qur'an berformat *\*.png*. Dari dua koleksi dokumen tadi dilakukan proses untuk penyusunan kembali korpus untuk

membentuk sebuah korpus baru berisi kumpulan subtopik Al-Qur'an. Satu dokumen subtopik mewakili dari beberapa ayat Al-Qur'an. Pembagian subtopik merujuk dari subtopik Al-Qur'an terjemahan versi Departemen Agama RI. Hal ini dilakukan untuk menghindari salah tafsir dari pengguna. Jika dijadikan satu ayat sebagai satu dokumen, ditinjau dari terjemahannya, banyak ayat pendek justru tidak selesai kalimatnya, atau tidak mengandung suatu makna yang jelas. Korpus subtopik Al-Qur'an nantinya akan digunakan sebagai bahan indeks dokumen untuk melakukan tahapan *indexing*.

2. *Tokenization* (pemisahan string menjadi kata/*term*)

Pemrosesan pemisahan rangkaian kata pada teks terjemahan Al-Qur'an pada tahapan *indexing* adalah proses memisahkan kalimat menjadi potongan kata/*term*. Pada proses ini juga menghilangkan karakter tanda baca (*stoplist*) dan mengubah kata ke dalam huruf kecil.

3. Pembuangan *Stopwords*

Setelah proses pemisahan rangkaian kata, dilakukan proses penghapusan *stopwords* untuk kata-kata yang dianggap tidak mencerminkan isi dari dokumen. Proses penghapusan kata yang dianggap tidak mencerminkan isi dari dokumen, yaitu kata sambung, kata depan, dan kata ganti.

4. *Stemming*

Proses *stemming* dilakukan untuk menjadikan kata yang ke bentuk dasar, dengan menghilangkan awalan atau akhiran dari sebuah kata. *Stemming* di sini menggunakan kamus daftar kata berimbunan yang mempunyai kata dasarnya dengan cara membandingkan kata-kata yang ada dalam terjemahan Al-Qur'an dengan daftar kamus *stem*.

5. Membangun *indexing* dan pembobotan *TF-IDF*

Proses membangun *indexing* merupakan pemilihan istilah yang mewakili dokumen menjadi daftar kata/*term* yang berasosiasi dengan asal kemunculan dokumen. Selanjutnya dilakukan pemberian nilai bobot pada tiap *term*. Pembobotan kata sangat berpengaruh dalam menentukan kemiripan antara dokumen dengan *query*. Apabila bobot tiap kata dapat

ditentukan dengan tepat, diharapkan hasil perhitungan kemiripan teks akan menghasilkan perankingan dokumen yang baik.

6. Analisis model ruang vektor.

Setelah menghitung pembobotan, dilakukan perhitungan kemiripan (*similarity*) antara dokumen dengan *query* dengan menerapkan pemodelan ruang vektor. Melakukan analisis terhadap persamaan model ruang vektor untuk mengetahui nilai korelevanan/kemiripan (*similarity*) dokumen terhadap *query* sehingga dokumen dapat diurutkan (dirangkingkan). Setelah dokumen diranking, sejumlah tetap dokumen *top-scoring* dikembalikan kepada pengguna.

### **3.5. Pengembangan Perangkat Lunak**

Pada tahapan ini akan dilakukan analisa pembangunan *information retrieval system* terhadap kebutuhan pengguna dan kebutuhan perangkat lunak, dijelaskan secara rinci tentang proses dari sistem temu kembali informasi yang akan dibangun sehingga mempermudah pemahaman terhadap sistem.

#### **3.5.1. Perancangan Sistem**

Tahap perancangan *information retrieval system* dengan model ruang vektor merupakan tahapan dalam membuat rincian sistem temu balik informasi berdasarkan analisa agar dapat dimengerti oleh pengguna. Tiga rancangan utama yang akan dilakukan, yaitu:

1. Perancangan sistem temu kembali ayat Al-Qur'an yang akan dibangun.
2. Transformasi koleksi dokumen ke dalam *database* yang akan digunakan dalam sistem meliputi preprosesing, *indexing*, *stemming* dan pembobotan.
3. Perancangan antarmuka sistem (*interface*) yang baik sehingga mudah digunakan (*user friendly*).

#### **3.6. Implementasi Sistem**

Pada implementasi sistem akan dilakukan pembuatan modul-modul yang telah dirancang dalam tahap perancangan ke dalam bahasa pemrograman.

### **3.7. Pengujian Sistem**

Pengujian merupakan tahapan sistem akan dijalankan. Tahap pengujian diperlukan untuk menjadi ukuran bahwa sistem dapat dijalankan sesuai dengan tujuan. Pengujian sistem temu balik informasi dalam tugas akhir ini dilakukan dengan cara mengukur kualitas *text retrieval*. Ukuran yang digunakan untuk mengukur kualitas dari *text retrieval* adalah *precision* dan *recall*.

### **3.8. Kesimpulan dan Saran**

Tahapan ini akan membahas tentang kesimpulan yang dihasilkan dari penelitian tugas akhir, kesimpulan diambil dari proses analisa kepada implementasi dan pengujian.

Pada tahapan saran, penelitian ini diharapkan dapat memberikan sebuah catatan rekomendasi untuk menyempurnakan dan mengembangkan penelitian sistem temu kembali pencarian ayat Al-Qur'an.

## **BAB 4**

### **ANALISA DAN PERANCANGAN**

Pada bab ini membahas mengenai analisa dan perancangan alur sistem pencarian ayat Al-Qur'an terhadap ekspresi kebutuhan pengguna (*query*) untuk menguji representasi model ruang vektor pada *Information Retrieval System*.

#### **4.1. Analisa Permasalahan**

Untuk membantu dan memudahkan seseorang dalam proses pencarian informasi terhadap kandungan dari ayat-ayat suci Al-Qur'an, dibutuhkan sebuah media yang mampu mewujudkan proses pencarian ini. Dokumen yang digunakan dalam proses pencarian sebagai sumber informasi adalah koleksi dokumen (korpus) teks terjemahan bahasa Indonesia dari tiap ayat suci Al-Qur'an. Media itu dapat diwujudkan dalam bentuk sebuah aplikasi yang berupa *search engine* (mesin pencari), aplikasi ini akan mengembalikan beberapa atau banyak informasi mengenai subtopik yang berkaitan dengan keinginan pengguna berdasarkan kata kunci (*query*) yang dimasukkan.

Korpus teks terjemahan dari Al-Qur'an yang digunakan sebagai masukan sistem telah disusun dalam bentuk *database*, serta korpus ayat (tulisan arab) dalam bentuk gambar dengan format *\*.png*. Namun pada teks terjemahan ayat masih dalam bentuk *natural language* yang belum terstruktur hingga belum bisa dimengerti oleh sistem. Dibutuhkan sebuah cara atau teknik tertentu untuk membuat teks yang masih berupa *natural language* menjadi lebih terstruktur sehingga dapat dimengerti dan diproses oleh sistem. Pendayagunaan *information retrieval system* merupakan teknik yang bisa digunakan untuk proses ekstraksi teks terjemahan ayat Al-Qur'an ini menjadi lebih terstruktur.

Pada umumnya, ada dua tahapan dalam membangun *Information Retrieval System* (IRS), yaitu melakukan preprosesing terhadap *database* dan kemudian menerapkan metode tertentu untuk menghitung relevansi antara dokumen di dalam *database* yang telah dipreprosesing berdasarkan masukan *query* pengguna.

Dalam tahapan preprosesing, terdapat 4 proses, yaitu *tokenizing* (pemecahan *string* untuk tiap kata/*term*), *filtering* (pembuangan kata/*term* yang dianggap tidak penting) yaitu pembuangan *stopwords*, *stemming* (pengembalian sebuah kata ke dalam bentuk kata dasar itu sendiri tanpa adanya imbuhan), serta membangun *indexing* dan pemberian bobot terhadap *term* dan perhitungan tingkat kemiripan tiap dokumen/ayat berdasarkan *term* yang dikandung dalam ayat tersebut dilakukan dengan algoritma model ruang vektor. Pada proses *stemming* digunakan algoritma Nazief dan Andriani sebagai algoritma *stemmer* yang spesifik untuk bahasa Indonesia.

Setelah melalui proses *stemming*, teks terjemahan Al-Qur'an yang telah menjadi sekelompok atau koleksi kata/*term* yang lebih terstruktur. Namun sampai tahap ini data berupa kumpulan *term* tersebut belum dapat digunakan untuk proses pencarian, maka dibangun *indexing* serta pemberian nilai bobot untuk tiap *term* berdasarkan kemunculan dokumen.

Untuk menghindari salah tafsir dari pengguna terhadap ayat Al-Qur'an, maka sistem tidak menggunakan satu ayat sebagai satu dokumen sebagai hasil pencarian. Karena jika ditinjau dari terjemahannya, banyak ayat pendek justru tidak selesai kalimatnya, atau tidak mengandung suatu makna yang jelas, sehingga tidak cocok dijadikan dokumen. Untuk mengatasi hal ini, disusun dokumen yang terstruktur dengan mengorganisasikan kembali korpus Al-Qur'an.

Tahapan awal dalam penelitian ini adalah membentuk sebuah korpus subtopik yang dilakukan dengan menuliskan kembali topik dan subtopik Al-Qur'an yang dirujuk dari naskah terjemahan Al-Qur'an bahasa Indonesia versi Departemen RI. Hal ini dilakukan agar dokumen menjadi sistematis, dan memudahkan orang untuk mengetahui kandungan dari pembahasan ayat yang dibacanya. Skema pembagian subtopik dibentuk berdasarkan isi kandungan dari tiap-tiap surah Al-Qur'an. Berdasarkan rujukan naskah terjemahan, topik Al-Qur'an diambil dari judul besar dalam tiap surah, dan subtopik diambil dari pembagian pecahan topik-topik besar tadi. Tiap topik menggambarkan tentang tema utama dari kandungan surah Al-Qur'an, dan subtopik menggambarkan satu kandungan naskah cerita dari kumpulan ayat-ayat Al-Qur'an. Hasil pemetaan dari



114 surah di dalam Al-Qur'an tercatat sebanyak 328 topik, dari tiap topik terpecah lagi menjadi 800 dokumen subtopik. Setiap dokumen subtopik terdiri atas jumlah ayat yang bervariasi, namun tetap merupakan satu kesatuan cerita yang utuh dari suatu topik atau subtopik pada Al-Qur'an. Sebagai contoh, pada Surat Al-Baqarah ayat 1 sampai dengan 20 dijadikan 3 dokumen, yaitu berdasarkan subtopik Golongan Mu'min (ayat 1-5), Golongan Kafir (ayat 6-7) dan Golongan Munafik (ayat 8-20) dan selanjutnya hingga ayat terakhir. Korpus subtopik sebanyak 800 daftar tadi disimpan ke dalam *database* yang akan dijadikan acuan indeks dokumen dalam membangun *indexing*.

Setiap kata/*term* dalam dokumen yang diindeks harus diberi nilai bobot berdasarkan jumlah kemunculan di tiap dokumen untuk dapat dianalisa tingkat kemiripan antar tiap ayat yang ada. Untuk tahap pemberian nilai atau bobot dari tiap *term* digunakan algoritma TF-IDF (*Term Frequency - Inverse Document Frequency*), setelah tiap *term* dari masing-masing ayat mempunyai nilai atau bobot, maka dibentuk sebuah matrik *term-document* berdasarkan nilai yang dihasilkan dari proses pembobotan TF-IDF sebelumnya. Proses pembentukan matrik *term-document* ini menggunakan algoritma *Vector Space Model* yang merupakan salah satu model IRS (*Information Retrieval System*). Kemudian dilanjutkan dengan menghitung tingkat kemiripan antar dokumen. *Cosine Similarity* merupakan teknik untuk dapat mengukur tingkat kemiripan (*similarity*) antar dokumen.

Pada proses pencarian yang menjadi masukan adalah teks dari kata kunci pengguna. Secara garis besar alur pengolahan dari *query* sampai menghasilkan daftar pencarian, mirip dengan pemrosesan dari teks terjemahan ayat Al-Qur'an. *Query* dari pengguna diekstraksi terlebih dahulu sesuai dengan tahapan preprosesing dokumen, kemudian tiap kata dari *query* diberi bobot dengan perhitungan TF-IDF dan menghitung tingkat kemiripan kata kunci dengan daftar terjemahan ayat yang ada menggunakan algoritma model ruang vektor dan *Cosine Similarity*.

Setelah semua tahapan di atas dilalui, maka sangat dimungkinkan adanya sebuah media yang mampu untuk membantu dan memudahkan seseorang dalam

proses pencarian informasi mengenai kandungan subtopik dari ayat-ayat suci Al-Qur'an yang diinginkan pengguna.

## 4.2. Analisa Sistem

Analisa adalah kajian yang dilaksanakan terhadap sebuah bahasa untuk meneliti struktur bahasa tersebut secara mendalam. Pada bagian ini akan dilakukan analisa terhadap *input* data, proses dan *output* dari sistem pencarian ayat Al-Qur'an sesuai konsep *information retrieval* dengan menggunakan model ruang vektor. Hasilnya digunakan untuk mengidentifikasi serta mengevaluasi hasil implementasi dan pengujian yang akan dibahas pada bab berikutnya. Dari proses analisis akan dapat dihasilkan dasar dalam perancangan sistem pencarian ayat Al-Qur'an.

### 4.2.1. Analisa Masukan (*Input*)

Data yang digunakan dalam penelitian ini adalah data-data yang digunakan sebagai bahan dasar pembangunan sistem. Data masukan yang digunakan dalam pembuatan TA (Tugas Akhir) ini adalah sebagai berikut:

1. Data *Administrator*
2. Korpus teks terjemahan Al-Qur'an dalam bahasa Indonesia yang letak surat dan ayatnya sudah disusun dalam bentuk *database*. Korpus diambil berdasarkan sumber dari <http://www.qurandownload.com/>
3. Korpus subtopik Al-Qur'an dari tiap surah Al-Qur'an yang disusun ke dalam *database*.
4. Korpus gambar ayat (tulisan arab) Al-Qur'an berformat *\*.png.*, terdiri dari beberapa gambar setiap ayat Al-Qur'an. Korpus diambil berdasarkan sumber dari <http://www.everyayah.com/data/quranpngs/>
5. Data kumpulan daftar *stopwords* berjumlah 352 kata. Daftar *stopwords* ini berdasarkan sumber dari <http://www.scribd.com/doc/61824071/DAFTAR-PUSTAKA>
6. Data kumpulan daftar *stemming*, yang merupakan kumpulan kata dasar digunakan sebagai kamus pada tahapan *stemming*, penggunaan kamus

mengacu kepada penelitian tugas akhir terdahulu yang menggunakan *stemming* algoritma Andriani dan Nazief. (Syahroni, 2012)

#### 4.2.2. Analisa Proses

Proses yang terjadi pada *Informaton Retrieval System* ini adalah:

1. *Maintenance*, dilakukan oleh admin. Mengumpulkan dokumen yang akan diindeks (dikenal dengan nama korpus/koleksi dokumen). Admin dapat melakukan proses *input*, *update*, dan *delete* terhadap data di dalam korpus.
2. Preprosesing, dilakukan oleh admin. Ada 4 proses dalam preprosesing, yaitu *tokenization* (pemecahan *string* untuk tiap kata/*term*), *filtering* (pembuangan kata/*term* yang dianggap tidak penting), *stemming* (pengembalian sebuah kata ke dalam bentuk kata dasar itu sendiri tanpa adanya imbuhan), serta membangun *indexing* dan pemberian bobot terhadap *term*.
3. Menerapkan pemodelan ruang vektor untuk menghitung relevansi antara *query* dan dokumen, dilakukan oleh admin. Setiap dokumen dan *query* pengguna direpresentasikan oleh suatu vektor *n*-dimensi. Tiap dimensi pada vektor tersebut diwakili oleh satu *term*.
4. Pencarian ayat Al-Qur'an, dilakukan oleh pengguna dengan memasukkan teks *query* atau kata kunci pada kolom pencarian.

#### 4.2.3. Analisis Keluaran (*Output*)

*Output* yang diharapkan dari *information retrieval system* ini adalah pengguna dapat menemukan daftar dokumen subtopik yang dianggap relevan dari dokumen yang telah dirangkingkan sesuai dengan kemiripan atau relevansinya terhadap *query* dari pengguna. Tiap dokumen hasil pencarian berisi satu kandungan isi subtopik dari beberapa ayat Al-Qur'an di dalamnya dengan menampilkan panduan terjemahan dalam bentuk teks latin, dan ayat (tulisan arab) dalam bentuk gambar.

### 4.3. Analisa Tahapan *Information Retrieval System*

Secara garis besar, ada dua tahapan yang ditangani oleh sistem ini, yaitu melakukan preprosesing terhadap dokumen dan menerapkan metode tertentu dalam hal ini menggunakan pemodelan ruang vektor untuk menghitung kedekatan (relevansi/*similarity*) antara dokumen dan *query* tersebut.

Proses awal dilakukan pemisahan kata (*tokenization*) terhadap *query* dengan menghilangkan beberapa tanda baca dan karakter spasi. Proses pemisahan kata ini dilakukan menjadi unit paling kecil yang disebut dengan kata atau *term*. Selanjutnya dilakukan proses pembuangan *stopwords*, yaitu pembuangan kata yang dianggap tidak penting. Dilakukan proses *stemming* pada tiap *term* yang mempunyai imbuhan dengan menggunakan algoritma Nazief dan Adriani. Setiap *term* hasil preprosesing akan dihitung pembobotan terhadap *term* dan menerapkan model ruang vektor untuk mencari nilai *similarity*/kesamaan antara dokumen dan *query* yang dimasukkan pengguna. Hasil pencarian dokumen berdasarkan *query* pengguna akan menampilkan urutan daftar dokumen dari tingkat korelevanan yang paling tinggi. Berikut merupakan contoh kasus penerapan algoritma model ruang vektor pada dokumen teks terjemahan ayat Al-Qur'an:

#### Contoh Kasus

Terdapat 4 dokumen ( ) yang diambil dari korpus terjemahan Al-Qur'an yaitu:

- : Allah telah mengunci-mati hati dan pendengaran mereka, dan penglihatan mereka ditutup. Dan bagi mereka siksa yang amat berat.
- : Hampir-hampir kilat itu menyambar penglihatan mereka. setiap kali kilat itu menyinari mereka, mereka berjalan di bawah sinar itu, dan bila gelap menimpa mereka, mereka berhenti. jika Allah menghendaki, niscaya dia melenyapkan pendengaran dan penglihatan mereka. sesungguhnya Allah berkuasa atas segala sesuatu.
- : Maka tatkala mereka melupakan peringatan yang telah diberikan kepada mereka, Kami-pun membukakan semua pintu-pintu kesenangan untuk mereka; sehingga apabila mereka bergembira dengan apa yang telah

diberikan kepada mereka, Kami siksa mereka dengan sekonyong-konyong, maka ketika itu mereka terdiam berputus asa.

: Katakanlah: "Terangkanlah kepadaku jika Allah mencabut pendengaran dan penglihatan serta menutup hatimu, siapakah tuhan selain Allah yang kuasa mengembalikannya kepadamu?" Perhatikanlah, bagaimana Kami berkali-kali memperlihatkan tanda-tanda kebesaran (Kami), kemudian mereka tetap berpaling (juga).

Q : Allah mencabut pendengaran dan penglihatan.

Diketahui:

D = 4 (Jumlah Dokumen terjemahan ayat)

Q = *query* yang dimasukkan pengguna

Langkah-langkah yang harus dilakukan pada contoh di atas untuk tahap preprosesing, pembuatan indeks, pencarian relevansi menggunakan pemodelan ruang vektor adalah sebagai berikut:

1. Penghilangan tanda baca dan mengubah ke huruf kecil (*lower case*). Untuk kasus ini tanda baca yang dihilangkan adalah tanda baca “ - . , ? ; : " ( ) ” .

: allah telah mengunci mati hati dan pendengaran mereka dan penglihatan mereka ditutup dan bagi mereka siksa yang amat berat

: hampir hampir kilat itu menyambar penglihatan mereka setiap kali kilat itu menyinari mereka mereka berjalan di bawah sinar itu dan bila gelap menimpa mereka mereka berhenti jika allah menghendaki niscaya dia melenyapkan pendengaran dan penglihatan mereka sesungguhnya allah berkuasa atas segala sesuatu

: maka tatkala mereka melupakan peringatan yang telah diberikan kepada mereka kami pun membukakan semua pintu-pintu kesenangan untuk mereka sehingga apabila mereka bergembira dengan apa yang telah diberikan kepada mereka kami siksa mereka dengan sekonyong-konyong maka ketika itu mereka terdiam berputus asa

: katakanlah terangkanlah kepadaku jika allah mencabut pendengaran dan penglihatan serta menutup hatimu siapakah tuhan selain allah yang kuasa mengembalikannya kepadamu perhatikanlah bagaimana kami berkali kali memperlihatkan tanda tanda kebesaran kami kemudian mereka tetap berpaling juga

2. Penghapusan *stopwords*, yaitu penghapusan *term* yang paling sering muncul pada dokumen. Untuk kasus ini *term* yang mengalami *stopwords* yaitu: “ apa, atas, bagaimana, bagi, bila, dan, dapat, dengan, di, dia, itu, hampir, jika, juga, kali, kami, kemudian, kepada, ketika, lagi, maka, mereka, oleh, paling, pun, segala, sehingga, selain, semua, serta, sesuatu, setiap, telah, tetap, tidak, untuk, yang”.

Berikut merupakan hasil teks terjemahan ayat yang sudah di melalui tahapan pembuangan *stopwords*:

: allah mengunci mati hati pendengaran penglihatan ditutup siksa amat berat  
: kilat menyambar penglihatan kilat menyinari berjalan bawah sinar gelap menimpa berhenti allah menghendaki niscaya melenyapkan pendengaran penglihatan sesungguhnya allah berkuasa  
: tatkala melupakan peringatan diberikan membukakan pintu pintu kesenangan bergembira diberikan siksa sekonyong konyong terdiam berputus asa  
: terangkanlah allah mencabut pendengaran penglihatan menutup hatimu siapakah tuhan allah kuasa mengembalikannya perhatikanlah memperlihatkan tanda tanda kebesaran

3. *Stemming*, yaitu proses konversi *term* ke bentuk dasarnya, merupakan tahapan *information retrieval* untuk meningkatkan performansi sistem temu kembali yang akan dibangun. Adapun algoritma *stemming* yang akan akan digunakan yaitu algoritma Nazief dan Adriani.

Langkah-langkah pada Algoritma Nazief dan Adriani adalah:

1. Kata yang belum distem dicari pada kamus. Jika kata itu langsung ditemukan, berarti kata tersebut adalah kata dasar. Kata tersebut dikembalikan dan algoritma dihentikan.
2. Hilangkan *inflectional suffixes* terlebih dahulu. Jika hal ini berhasil dan *suffix* adalah partikel (“lah” atau ”kah”), langkah ini dilakukan lagi untuk menghilangkan *inflectional possessive pronoun suffixes* (“ku”, “mu” atau ”nya”).
3. *Derivational suffix* kemudian dihilangkan. Lalu langkah ini dilanjutkan lagi untuk mengecek apakah masih ada *derivational suffix* yang tersisa, jika ada maka dihilangkan. Jika tidak ada lagi maka lakukan langkah selanjutnya.
4. Kemudian *derivational prefix* dihilangkan. Lalu langkah ini dilanjutkan lagi untuk mengecek apakah masih ada *derivational prefix* yang tersisa, jika ada maka dihilangkan. Jika tidak ada lagi maka lakukan langkah selanjutnya.
5. Setelah tidak ada lagi imbuhan yang tersisa, maka algoritma ini dihentikan kemudian kata dasar tersebut dicari pada kamus, jika kata dasar tersebut ketemu berarti algoritma ini berhasil tapi jika kata dasar tersebut tidak ketemu pada kamus, maka dilakukan *recoding*.
6. Jika semua langkah telah dilakukan tetapi kata dasar tersebut tidak ditemukan pada kamus juga maka algoritma ini mengembalikan kata yang asli sebelum dilakukan *stemming*.

Berikut merupakan hasil teks terjemahan ayat yang sudah di melalui tahapan *stemming*:

- : allah kunci mati hati dengar lihat tutup siksa amat berat
- : kilat sambar lihat kilat sinar jalan bawah sinar gelap timpa henti allah  
hendak niscaya lenyap dengar lihat sungguh allah kuasa
- : tatkala lupa ingat beri buka pintu pintu senang gembira beri siksa  
konyong konyong diam putus asa

: terang allah cabut dengar lihat tutup hati siapa tuhan allah kuasa kembali  
hati lihat tanda tanda besar

4. Membangun *indexing* dan pemberian bobot terhadap *term*.

Setiap kata/*term* dalam dokumen yang akan diindeks harus diberi nilai bobot berdasarkan jumlah kemunculan di tiap dokumen untuk dapat dianalisa tingkat kemiripan antar tiap ayat yang ada. Untuk tahap pemberian nilai atau bobot dari tiap *term* digunakan algoritma TF-IDF (*Term Frequency - Inverse Document Frequency*). Pemberian bobot pada masing-masing *term* didefinisikan melalui perhitungan frekuensi kemunculan dokumen yang mengandung sebuah *term* yaitu DF (*document frequency*); penghitungan frekuensi kemunculan *term* di dalam dokumen yaitu TF (*term frequency*) dan perhitungan jumlah dokumen yang mengandung sebuah *term* yang dicari dari kumpulan dokumen yang ada yaitu IDF (*Inverse Document Frequency*).

$$Idf(i) = \frac{1}{( )}$$

Hasil perhitungan nilai TF-IDF pada term tiap dokumen dapat dilihat pada Tabel 4.1 berikut.

**Tabel 4.1. Nilai TF-IDF (*Inverse Document Frequency*)**

<i>Term</i>	no dokumen	<i>Tf</i>	<i>Df</i>	<i>Idf</i>
allah	1	1	3	0,12494
kunci	1	1	1	0,60206
mati	1	1	1	0,60206
hati	1	1	2	0,30103
dengar	1	1	3	0,12494
lihat	1	1	3	0,12494
tutup	1	1	2	0,30103
siksa	1	1	2	0,30103
amat	1	1	1	0,60206
berat	1	1	1	0,60206
kilat	2	2	1	0,60206
sambar	2	1	1	0,60206
lihat	2	2	3	0,12494



sinar	2	1	2	0,30103
jalan	2	1	1	0,60206
bawah	2	1	1	0,60206
sinar	2	1	2	0,30103
gelap	2	1	1	0,60206
timpa	2	1	1	0,60206
henti	2	1	1	0,60206
allah	2	2	3	0,12494
niscaya	2	1	1	0,60206
dengar	2	1	3	0,12494
lenyap	2	1	1	0,60206
sungguh	2	1	1	0,60206
kuasa	2	1	2	0,30103
tatkala	3	1	1	0,60206
lupa	3	1	1	0,60206
ingat	3	1	1	0,60206
beri	3	2	1	0,60206
buka	3	1	1	0,60206
pintu	3	2	1	0,60206
senang	3	1	1	0,60206
gembira	3	1	1	0,60206
siksa	3	1	2	0,30103
konyong	3	2	1	0,60206
diam	3	1	1	0,60206
putus	3	1	1	0,60206
asa	3	1	1	0,60206
terang	4	1	1	0,60206
allah	4	2	3	0,12494
cabut	4	1	1	0,60206
dengar	4	1	3	0,12494
lihat	4	1	3	0,12494
tutup	4	1	2	0,30103
hati	4	2	2	0,30103
siapa	4	1	1	0,60206
tuhan	4	1	1	0,60206
kuasa	4	1	2	0,30103
kembali	4	1	1	0,60206
tanda	4	2	1	0,60206
besar	4	1	1	0,60206

### Pemberian bobot terhadap term

Di bawah ini adalah Tabel 4.2 pencarian nilai bobot *term* dengan menggunakan

rumus  $tf = \frac{1}{x}$ ,

**Tabel 4.2. Nilai Bobot Term**

<i>Term</i>	no dokumen	<i>Tf</i>	<i>Df</i>	<i>Idf</i>	<i>Bobot Term</i>
allah	1	1	3	0,12494	0,12494
kunci	1	1	1	0,60206	0,60206
mati	1	1	1	0,60206	0,60206
hati	1	1	2	0,30103	0,30103
dengar	1	1	3	0,12494	0,12494
lihat	1	1	3	0,12494	0,12494
tutup	1	1	2	0,30103	0,30103
siksa	1	1	2	0,30103	0,30103
amat	1	1	1	0,60206	0,60206
berat	1	1	1	0,60206	0,60206
kilat	2	2	1	0,60206	1,20412
sambar	2	1	1	0,60206	0,60206
lihat	2	2	3	0,12494	0,24988
sinar	2	1	2	0,30103	0,30103
jalan	2	1	1	0,60206	0,60206
bawah	2	1	1	0,60206	0,60206
sinar	2	1	2	0,30103	0,30103
gelap	2	1	1	0,60206	0,60206
timpa	2	1	1	0,60206	0,60206
henti	2	1	1	0,60206	0,60206
allah	2	2	3	0,12494	0,24988
niscaya	2	1	1	0,60206	0,60206
dengar	2	1	3	0,12494	0,12494
lenyap	2	1	1	0,60206	0,60206
sungguh	2	1	1	0,60206	0,60206
kuasa	2	1	2	0,30103	0,30103
tatkala	3	1	1	0,60206	0,60206
lupa	3	1	1	0,60206	0,60206
ingat	3	1	1	0,60206	0,60206
beri	3	2	1	0,60206	1,20412
buka	3	1	1	0,60206	0,60206
pintu	3	2	1	0,60206	1,20412
senang	3	1	1	0,60206	0,60206
gembira	3	1	1	0,60206	0,60206

siksa	3	1	2	0,30103	0,30103
konyong	3	2	1	0,60206	1,20412
diam	3	1	1	0,60206	0,60206
putus	3	1	1	0,60206	0,60206
asa	3	1	1	0,60206	0,60206
terang	4	1	1	0,60206	0,60206
allah	4	2	3	0,12494	0,24988
cabut	4	1	1	0,60206	0,60206
dengar	4	1	3	0,12494	0,12494
lihat	4	1	3	0,12494	0,12494
tutup	4	1	2	0,30103	0,30103
hati	4	2	2	0,30103	0,60206
siapa	4	1	1	0,60206	0,60206
tuhan	4	1	1	0,60206	0,60206
kuasa	4	1	2	0,30103	0,30103
kembali	4	1	1	0,60206	0,60206
tanda	4	2	1	0,60206	1,20412
besar	4	1	1	0,60206	0,60206

### Normalisasi bobot term

Di bawah ini adalah Tabel 4.3 pencarian nilai normalisasi bobot *term* dengan

rumus 
$$= \frac{\text{Bobot Term}}{\Sigma \text{Bobot Term}}$$

**Tabel 4.3. Nilai Normalisasi Bobot Term**

<i>Term</i>	no dokumen	<i>Tf</i>	<i>Df</i>	<i>Idf</i>	<i>Bobot Term</i>	<i>Normalisasi Term</i>
allah	1	1	3	0,12494	0,12494	0,09395
kunci	1	1	1	0,60206	0,60206	0,45272
mati	1	1	1	0,60206	0,60206	0,45272
hati	1	1	2	0,30103	0,30103	0,22636
dengar	1	1	3	0,12494	0,12494	0,09395
lihat	1	1	3	0,12494	0,12494	0,09395
tutup	1	1	2	0,30103	0,30103	0,22636
siksa	1	1	2	0,30103	0,30103	0,22636
amat	1	1	1	0,60206	0,60206	0,45272
berat	1	1	1	0,60206	0,60206	0,45272
kilat	2	2	1	0,60206	1,20412	0,53192

sambar	2	1	1	0,60206	0,60206	0,26596
lihat	2	2	3	0,12494	0,24988	0,11038
sinar	2	1	2	0,30103	0,30103	0,13298
jalan	2	1	1	0,60206	0,60206	0,26596
bawah	2	1	1	0,60206	0,60206	0,26596
sinar	2	1	2	0,30103	0,30103	0,13298
gelap	2	1	1	0,60206	0,60206	0,26596
timpa	2	1	1	0,60206	0,60206	0,26596
henti	2	1	1	0,60206	0,60206	0,26596
allah	2	2	3	0,12494	0,24988	0,11038
niscaya	2	1	1	0,60206	0,60206	0,26596
dengar	2	1	3	0,12494	0,12494	0,05519
lenyap	2	1	1	0,60206	0,60206	0,26596
sungguh	2	1	1	0,60206	0,60206	0,26596
kuasa	2	1	2	0,30103	0,30103	0,13298
tatkala	3	1	1	0,60206	0,60206	0,21693
lupa	3	1	1	0,60206	0,60206	0,21693
ingat	3	1	1	0,60206	0,60206	0,21693
beri	3	2	1	0,60206	1,20412	0,43386
buka	3	1	1	0,60206	0,60206	0,21693
pintu	3	2	1	0,60206	1,20412	0,43386
senang	3	1	1	0,60206	0,60206	0,21693
gembira	3	1	1	0,60206	0,60206	0,21693
siksa	3	1	2	0,30103	0,30103	0,10847
konyong	3	2	1	0,60206	1,20412	0,43386
diam	3	1	1	0,60206	0,60206	0,21693
putus	3	1	1	0,60206	0,60206	0,21693
asa	3	1	1	0,60206	0,60206	0,21693
terang	4	1	1	0,60206	0,60206	0,29163
allah	4	2	3	0,12494	0,24988	0,12104
cabut	4	1	1	0,602 06	0,60206	0,29163
dengar	4	1	3	0,12494	0,12494	0,06052
lihat	4	1	3	0,12494	0,12494	0,06052
tutup	4	1	2	0,30103	0,30103	0,14581
hati	4	2	2	0,30103	0,60206	0,29163
siapa	4	1	1	0,60206	0,60206	0,29163
tuhan	4	1	1	0,60206	0,60206	0,29163
kuasa	4	1	2	0,30103	0,30103	0,14581
kembali	4	1	1	0,60206	0,60206	0,29163
tanda	4	2	1	0,60206	1,20412	0,58325
besar	4	1	1	0,60206	0,60206	0,29163

5. Pemrosesan terhadap *query*

*Query* yang digunakan : Allah mencabut pendengaran dan penglihatan.

Ketika pengguna memasukkan teks *query* pada mesin pencarian, *query* diekstrak kembali menjadi *term* tunggal melalui proses preprosesing untuk dihitung relevansi antara *term query* dengan *term* yang ada di dokumen. hasil dari pemrosesan terhadap teks *query* tersebut adalah:

= allah

= cabut

= dengar

= lihat

**Pembobotan Query**

Setelah teks *query* diekstrak menjadi *term* tunggal, maka dilakukan proses

pembobotan *term query* dengan rumus  $W_{ij} = 0.5 + \frac{f_{ij}}{f_i} \times idf(i)$

**Tabel 4.4. Query**

<i>Term / kata</i>	<i>Frequency</i>
Allah	3
cabut	1
dengar	3
Lihat	3

$$W_{11} = 0.5 + \frac{1}{3} \times 0.12494 = 0.37482$$

$$W_{12} = 0.5 + \frac{1}{1} \times 0.60206 = 0.60206$$

$$W_{13} = 0.5 + \frac{1}{3} \times 0.12494 = 0.24988$$

$$W_{14} = 0.5 + \frac{1}{3} \times 0.12494 = 0.24988$$

**Tabel 4.5. Bobot Query**

<i>Query</i>	<b>Bobot</b>
(allah)	0.37482
(cabut)	0.60206
(dengar)	0.24988
(lihat)	0.24988

**Normalisasi Bobot Query**

Setelah menentukan bobot *query*, selanjutnya dilakukan normalisasi terhadap

bobot *query*, dengan rumus 
$$b_{ij} = \frac{b_{ij}}{\sum b_{ij}}$$

$$\sum b_{ij} = 0.81423$$

$$= \frac{0.37482}{0.81423} = 0.46033$$

$$= \frac{0.60206}{0.81423} = 0.73943$$

$$= \frac{0.24988}{0.81423} = 0.30689$$

$$= \frac{0.24988}{0.81423} = 0.38361$$

**Tabel 4.6. Normalisasi Bobot Query**

<i>Query</i>	<b>Bobot</b>
(allah)	0.46033
(cabut)	0.73943
(dengar)	0.30689
(lihat)	0.38361

6. Perhitungan relevansi/kemiripan (*similarity*) dengan rumus

$$\text{Sim}(D, q) = \sum(D_i \cdot x_i)$$

$$\begin{aligned} \text{Sim}(D, Q) &= \sum(D_i \cdot x_i) \\ &= 0.04325 + 0 + 0.02883 + 0.03604 = 0.10812 \end{aligned}$$

$$\begin{aligned} \text{Sim}(D, Q) &= \sum(D_i \cdot x_i) \\ &= 0.05081 + 0 + 0.01694 + 0.04234 = 0.11009 \end{aligned}$$

$$\begin{aligned} \text{Sim}(D, Q) &= \sum(D_i \cdot x_i) \\ &= 0 + 0 + 0 = 0 \end{aligned}$$

$$\begin{aligned} \text{Sim}(D, Q) &= \sum(D_i \cdot x_i) \\ &= 0.05572 + 0.21564 + 0.01857 + 0.02322 = 0.31314 \end{aligned}$$

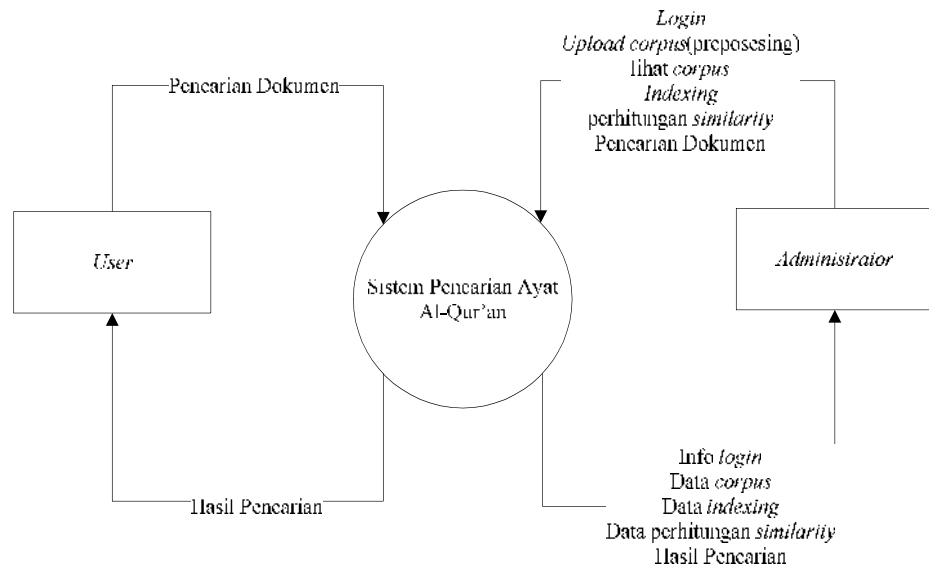
Dari hasil penghitungan relevansi pada langkah sebelumnya, jika pengguna memasukkan *query* “Allah mencabut pendengaran dan penglihatan mereka”, maka sistem akan mengembalikan dokumen yang dapat dirankingkan dari yang paling relevan yaitu Dokumen 1, 2, 3, 4. Dokumen yang paling relevan berdasarkan *query* pengguna yaitu 1 : Katakanlah: "Terangkanlah kepadaku jika Allah mencabut pendengaran dan penglihatan serta menutup hatimu, siapakah tuhan selain Allah yang kuasa mengembalikannya kepadamu?" Perhatikanlah, bagaimana Kami berkali-kali memperlihatkan tanda-tanda kebesaran (Kami), kemudian mereka tetap berpaling (juga).”

#### 4.4. Perancangan Sistem

Setelah dilakukan tahapan analisa *Information Retrieval System* dengan pemodelan ruang vektor yang akan dibangun, maka akan dilanjutkan dengan tahapan perancangan sistem.

##### 4.4.1 Context Diagram

*Context Diagram* digunakan untuk menggambarkan proses kerja sistem secara umum. *Context Diagram* merupakan DFD (*Data Flow Diagram*) yang menggambarkan garis besar operasional sistem.



**Gambar 4.1. Konteks Diagram**

**Tabel 4.7. Deskripsi Diagram Konteks**

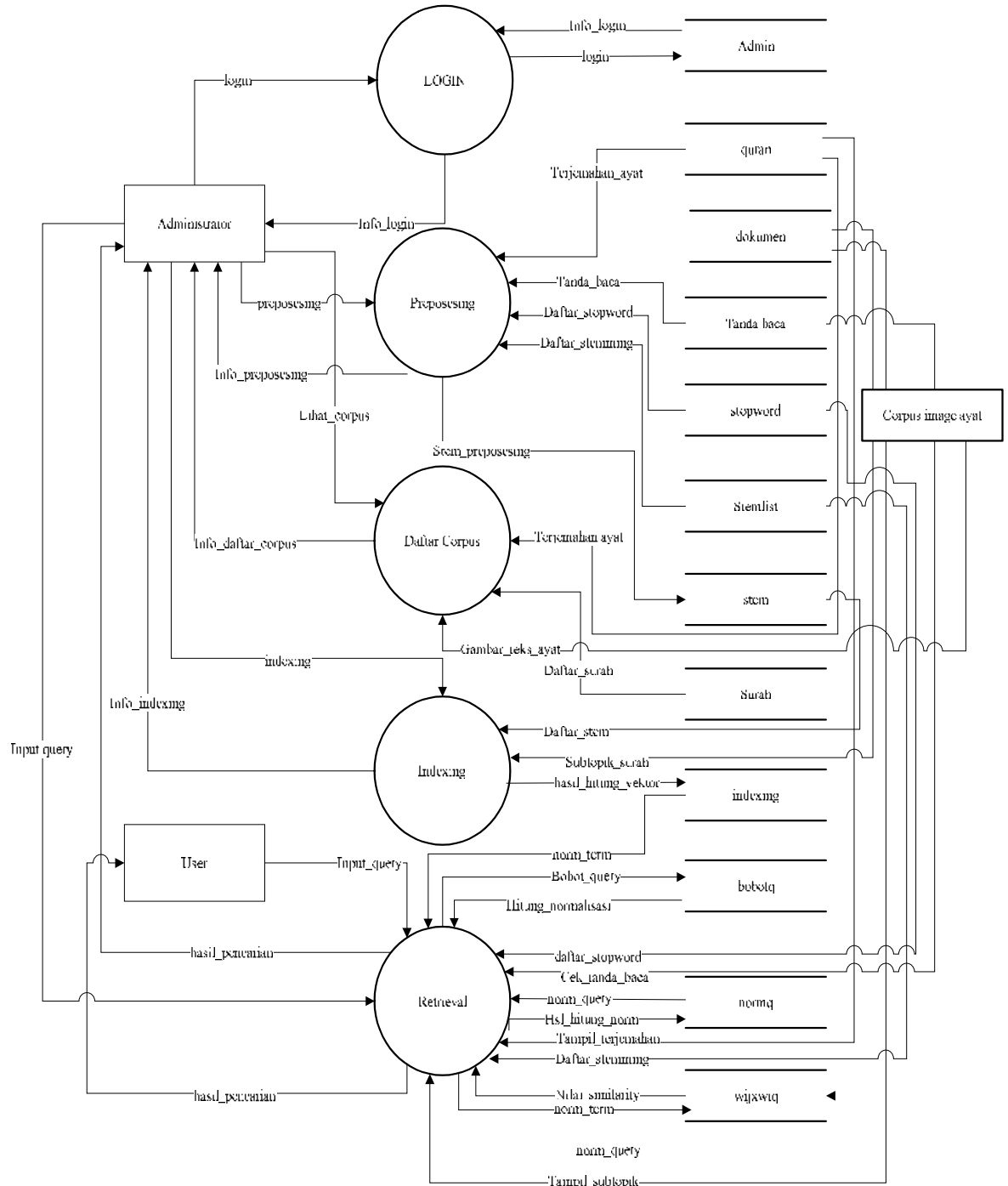
No	Entitas	Masukan	Keluaran	Keterangan
1.	<i>Administrator</i>	<i>Login, upload korpus (preprocessing), lihat korpus, indexing, perhitungan similarity, pencarian dokumen</i>	<i>Info login, data korpus, data indexing, data perhitungan similarity, hasil pencarian</i>	<i>Administrator adalah orang yang mengelola sistem.</i>
2.	<i>User</i>	<i>Pencarian data dokumen</i>	<i>Hasil pencarian</i>	<i>User adalah orang melakukan pencarian pada sistem</i>

#### 4.4.2 Diagram Aliran Data (*Data Flow Diagram*)

*Data Flow Diagram* merupakan penjabaran dari *Context Diagram* secara lebih terperinci. Semua proses yang terjadi dapat dilihat pada analisa *Data Flow Diagram*.



1. DFD Level 1. Gambar DFD level 1 sistem pencarian ayat Al-Qur'an dengan menggunakan algoritma ruang vektor adalah sebagai berikut:



Gambar 4.2. DFD Level 1 Proses 1

**Tabel 4.8. Deskripsi DFD Level 1 Proses Login**

No. Proses	: 1
Nama proses	: <i>Login</i>
Deskripsi	: Proses <i>login</i> oleh <i>administrator</i>
Masukan	: <i>Login</i>
Keluaran	: Info <i>login</i>

**Tabel 4.9. Deskripsi DFD Level 1 Proses Preposesing**

No. Proses	: 2
Nama proses	: Preposesing
Deskripsi	: Proses preposesing dari korpus terjemahan ayat Al-Qur'an
Masukan	: Preposesing
Keluaran	: Info preposesing, tabel <i>stem</i>

**Tabel 4.10. Deskripsi DFD Level 1 Proses Daftar Korpus**

No. Proses	: 3
Nama proses	: Daftar Korpus
Deskripsi	: Proses menampilkan ayat Al-Qur'an dan terjemahan
Masukan	: Lihat korpus
Keluaran	: Info daftar korpus

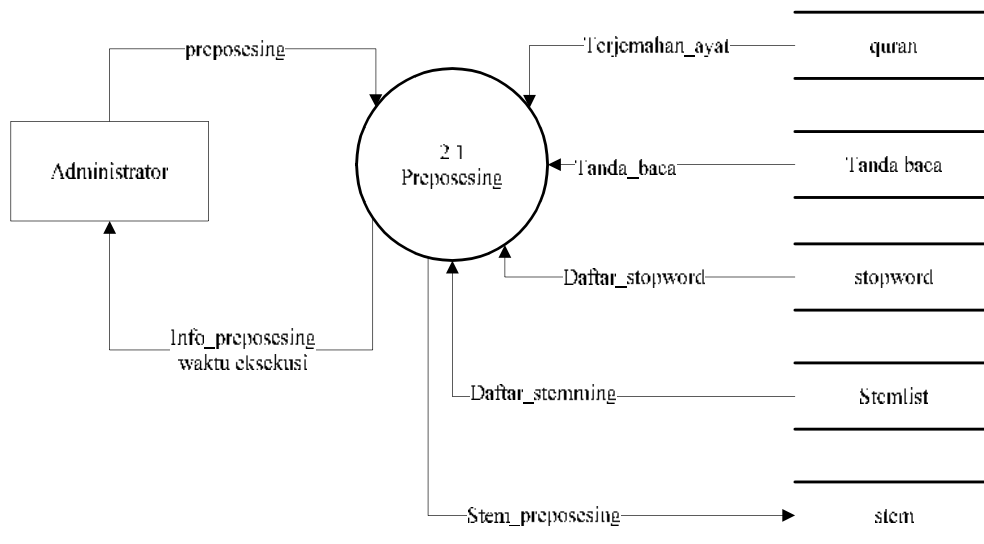
**Tabel 4.11. Deskripsi DFD Level 1 Proses Indexing**

No. Proses	: 4
Nama proses	: <i>Indexing</i>
Deskripsi	: Proses pengindeksian <i>term</i> dan pemberian bobot <i>term</i>
Masukan	: <i>Indexing</i>
Keluaran	: Info <i>indexing</i> , tabel <i>indexing</i>

**Tabel 4.12. Deskripsi DFD Level 1 Proses Retrieval**

No. Proses	: 5
Nama proses	: <i>Retrieval</i>
Deskripsi	: Proses pencarian dokumen relevan berdasarkan rumus ruang vektor
Masukan	: <i>Input query</i>
Keluaran	: Tabel <i>bobotq, normq, wijxwiq</i> , hasil pencarian

**1. DFD Level 2 Proses Preposinging**

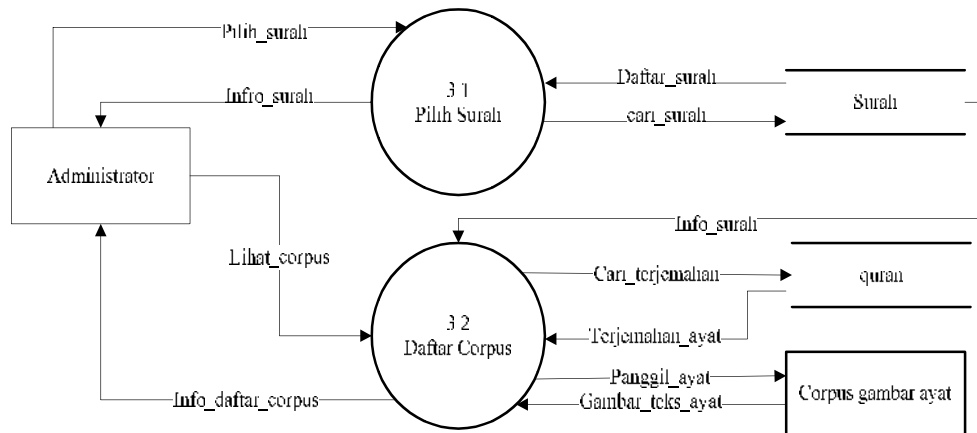


**Gambar 4.3. DFD Level 2 Proses 2**

**Tabel 4.13. Deskripsi DFD Level 2 Proses 2.1 Preposinging**

No. Proses	: 2.1
Nama proses	: Preposinging
Deskripsi	: Proses preposinging koleksi dokumen terjemahan ayat Al-Qur'an
Masukan	: Preposinging
Keluaran	: Info preposinging, waktu eksekusi

## 2. DFD Level 2 Proses Daftar Korpus



Gambar 4.4. DFD Level 2 Proses 3

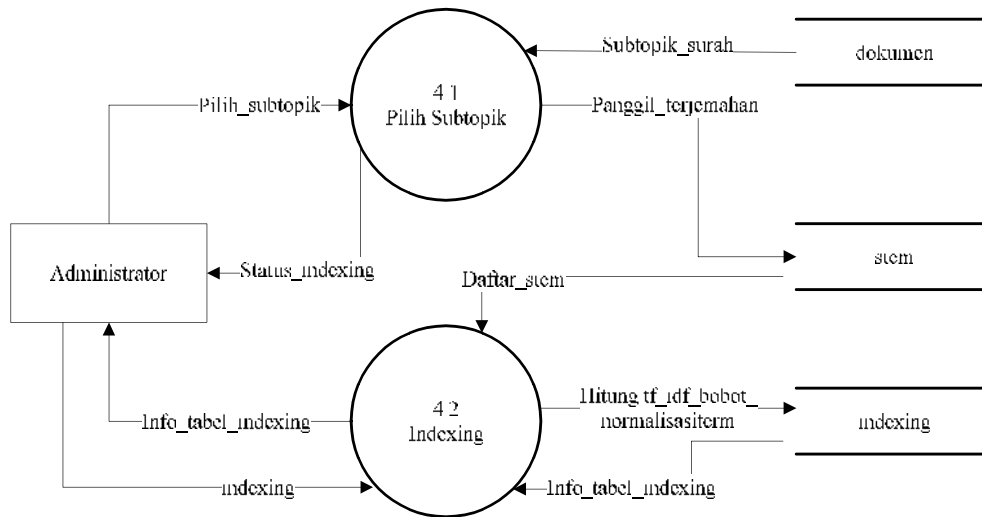
Tabel 4.14. Deskripsi DFD Level 2 Proses 3.1 Pilih Surah

No. Proses	: 3.1
Nama proses	: Pilih Surah
Deskripsi	: Proses memilih yang akan ditampilkan dalam daftar korpus
Masukan	: Pilih surah
Keluaran	: Info surah

Tabel 4.15. Deskripsi DFD Level 2 Proses 3.2 Daftar Korpus

No. Proses	: 3.2
Nama proses	: Daftar Korpus
Deskripsi	: Proses menampilkan surah dan ayat yang telah dipilih
Masukan	: Lihat korpus
Keluaran	: Info daftar korpus

### 3. DFD Level 2 Proses Indexing



Gambar 4.5. DFD Level 2 Proses 4

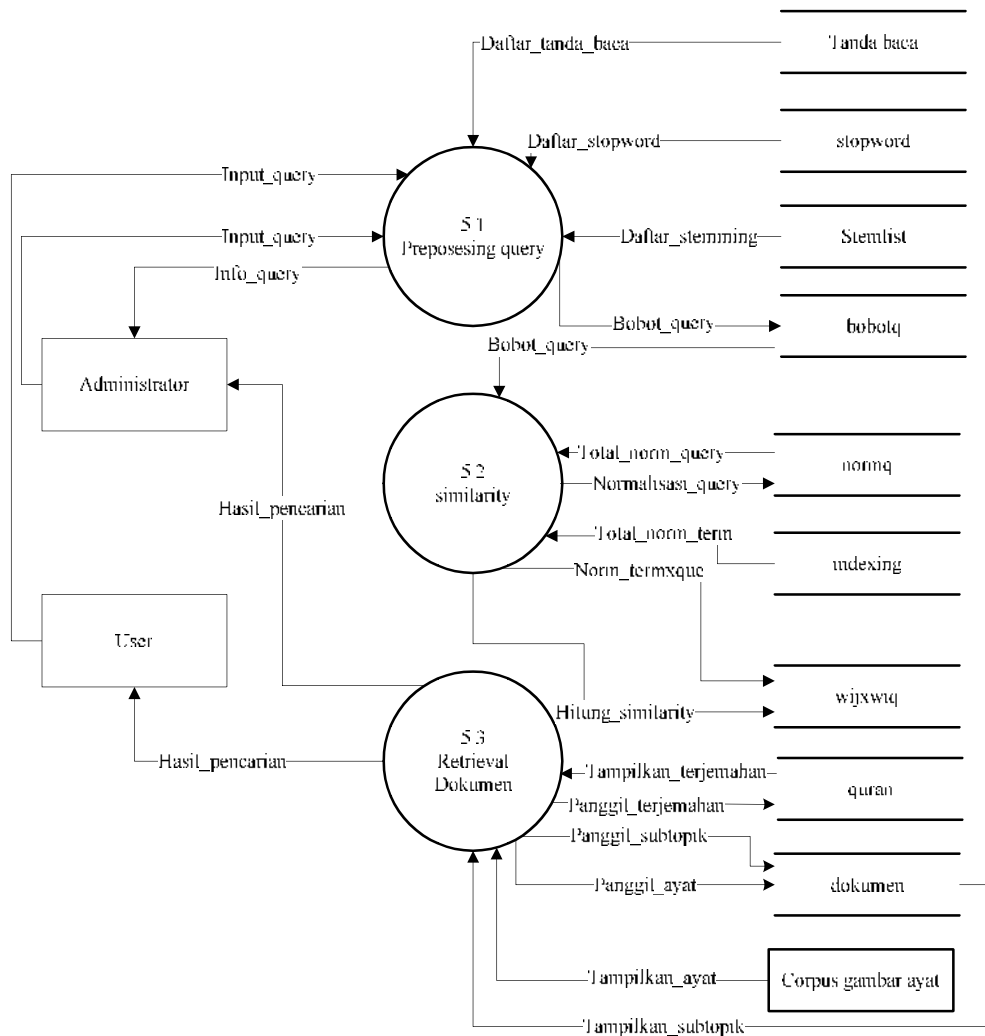
Tabel 4.16. Deskripsi DFD Level 2 Proses 4.1 Pilih Subtopik

No. Proses	: 4.1
Nama proses	: Pilih Subtopik
Deskripsi	: Proses memilih Subtopik yang akan diindeks
Masukan	: Pilih subtopik
Keluaran	: Status <i>indexing</i>

Tabel 4.17. Deskripsi DFD Level 2 Proses 4.2 Indexing

No. Proses	: 4.2
Nama proses	: <i>Indexing</i>
Deskripsi	: Proses memulai <i>indexing</i> subtopik dan pembobotan <i>term</i>
Masukan	: <i>Indexing</i>
Keluaran	: Info tabel <i>indexing</i>

4. *DFD Level 2 Proses Retrieval*



Gambar 4.6. *DFD Level 2 Proses 5*

Tabel 4.18. Deskripsi *DFD Level 2 Proses 5.1 Preprocessing Query*

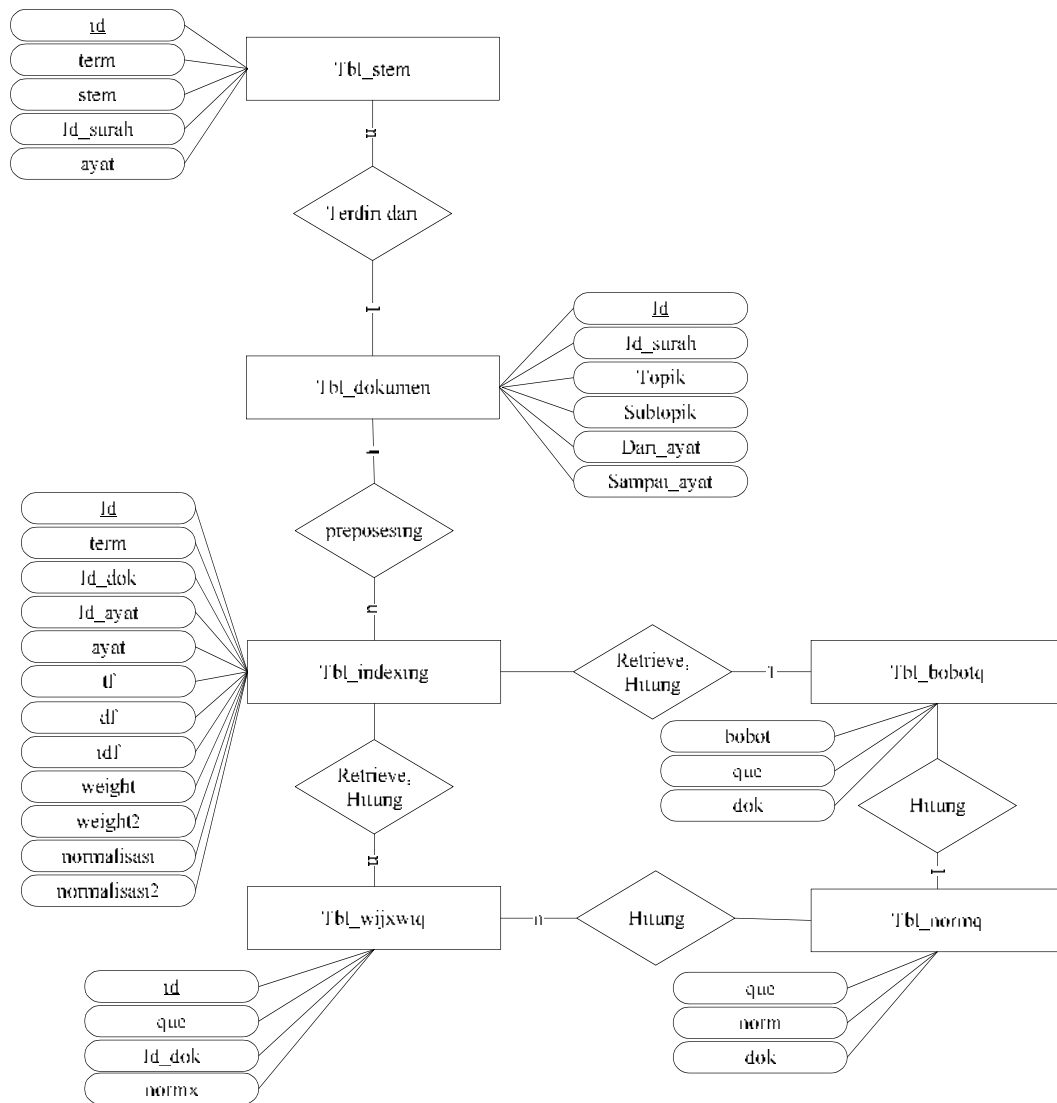
No. Proses	: 5.1
Nama proses	: Preprocessing <i>query</i>
Deskripsi	: Proses melakukan preprocessing pada <i>query</i> yang dimasukkan
Masukan	: <i>Input query</i>
Keluaran	: <i>Info query</i>

**Tabel 4.19. Deskripsi DFD Level 2 Proses 5.2 Similarity**

No. Proses	:	5.2
Nama proses	:	Similarity
Deskripsi	:	Proses perhitungan nilai kesamaan antara dokumen dan <i>query</i> .
Masukan	:	Bobot <i>query</i>
Keluaran	:	Total bobot normalisasi <i>term</i> dokumen dan <i>query</i>

**Tabel 4.20. Deskripsi DFD Level 2 Proses 5.3 Retrieval Dokumen**

No. Proses	:	5.3
Nama proses	:	<i>Retrieval</i> Dokumen
Deskripsi	:	Proses pencarian dokumen relevan berdasarkan perhitungan ruang vektor
Masukan	:	Total bobot normalisasi <i>term</i> dokumen dan <i>query</i>
Keluaran	:	Hasil pencarian



**Gambar 4.7.**Rancangan *Entity Relationship Diagram*



#### 4.5. Perancangan *Database*

Dalam perancangan *database*, penelitian ini untuk pangkalan data pencarian dokumen terdiri dari beberapa tabel yaitu:

##### Tabel 4.21. Admin

Tabel ini memuat tentang informasi pengguna sistem

Atribut	Tipe data	Keterangan
Username	<i>Varchar(20)</i>	<i>Primary</i>
Password	<i>Varchar(20)</i>	

##### Tabel 4.22. Tandabaca

Tabel ini memuat tentang kumpulan tanda baca yang digunakan pada tahapan preposesing

Atribut	Tipe data	Keterangan
Id	<i>Int(10)</i>	<i>Primary</i>
Tandabaca	<i>Varchar(10)</i>	-

##### Tabel 4.23 Surah

Tabel ini memuat tentang kumpulan detail surah di dalam Al-Qur'an

Atribut	Tipe data	Keterangan
Id_surah	<i>Int(10)</i>	<i>Primary</i>
Nama_surah	<i>Varchar(255)</i>	-
Jumlah_ayat	<i>Int(10)</i>	-
Turun	<i>Varchar(50)</i>	Tempat dimana ayat Al-Qur'an diturunkan

**Tabel 4.24. Stopword**

Tabel ini memuat tentang kumpulan daftar *stopwords*, yaitu kata yang tidak memiliki makna dalam dokumen.

Atribut	Tipe data	Keterangan
Id	<i>Int(10)</i>	<i>Primary</i>
Stopword	<i>Varchar(100)</i>	-

**Tabel 4.25. Quran**

Tabel ini memuat koleksi terjemahan ayat Al-Qur'an yang dijadikan sebagai korpus penelitian.

Atribut	Tipe data	Keterangan
Id	<i>Int(11)</i>	<i>Primary</i>
DatabaseID	<i>Smallint(6)</i>	-
SuraID	<i>Int(11)</i>	Indeks Surah
VerseID	<i>Int(11)</i>	Indeks ayat ke- <i>n</i> dalam satu surah
AyahText	<i>Text</i>	Teks terjemahan ayat Al-Qur'an

**Tabel 4.26. Stemlist**

Tabel ini memuat tentang kumpulan daftar kata dasar yang digunakan pada proses *stemming*.

Atribut	Tipe data	Keterangan
Id	<i>Int(10)</i>	<i>Primary</i>
Stem	<i>Varchar(20)</i>	Kata dasar
Stemto	<i>Varchar(20)</i>	Jenis kata

**Tabel 4.27. Stem**

Tabel ini memuat tentang kumpulan *term* hasil preprosesing seluruh dokumen terjemah ayat Al-Qur'an.

Atribut	Tipe data	Keterangan
Id	<i>Int(11)</i>	<i>Primary</i>
Term	<i>Varchar(50)</i>	Kata dasar
Stem	<i>Varchar(50)</i>	Kata asal sebelum di <i>stemming</i>
Id_surah	<i>Varchar(50)</i>	Indeks Surah
Ayat	<i>Int(5)</i>	Indeks ayat berdasarkan surah

**Tabel 4.28. Dokumen**

Tabel ini memuat tentang kumpulan subtopik yang menjadi dokumen bahan pada proses *indexing*.

Atribut	Tipe data	Keterangan
Id	<i>Int(11)</i>	<i>Primary</i>
Id_surah	<i>Int(10)</i>	
Topik	<i>Varchar(250)</i>	
Subtopik	<i>Text</i>	
Dari_ayat	<i>Int(5)</i>	Awal nomor ayat dalam subtopik surah
Sampai ayat	<i>Int(5)</i>	Akhir nomor ayat dalam subtopik surah

**Tabel 4.29. Indexing**

Tabel ini memuat tentang kumpulan *term* yang telah diindeks beserta perhitungan pembobotan *TF-IDF* masing-masing *term*.

Atribut	Tipe data	Keterangan
Id	<i>Int(11)</i>	<i>Primary</i>
Term	<i>Varchar(200)</i>	
Id_dok	<i>Int(11)</i>	
Id_ayat	<i>Text</i>	
Tf	<i>Int(11)</i>	<i>Term Frequency</i>
Df	<i>Int(11)</i>	<i>Document Frequency</i>
Idf	<i>Float</i>	<i>Inverse Document Frequency</i>
Weight	<i>Float</i>	Bobot <i>term</i>
Weight2	<i>Float</i>	Bobot <i>term</i> kuadrat
Normalisasi	<i>Float</i>	Normalisasi <i>term</i>
Normalisasi2	<i>Float</i>	Normalisasi <i>term</i> kuadrat

**Tabel 4.30. Bobotq**

Tabel ini memuat nilai masing-masing bobot *term query* yang dimasukkan pengguna

Atribut	Tipe data	Keterangan
Que	<i>Varchar(300)</i>	<i>Primary</i>
Bobot	<i>Float</i>	
Dok	<i>Varchar(300)</i>	

**Tabel 4.31. Normq**

Tabel ini memuat nilai masing-masing bobot *term query* yang telah dinormalisasi.

Atribut	Tipe data	Keterangan
Que	<i>Varchar(300)</i>	<i>Primary</i>
Norm	<i>Float</i>	
Dok	<i>Varchar(300)</i>	

**Tabel 4.32. wijxwiq**

Tabel ini memuat penjumlahan seluruh bobot *term* dan *query* yang sama pada sebuah dokumen.

Atribut	Tipe data	Keterangan
Id	<i>Int(20)</i>	<i>Primary</i>
Que	<i>Varchar(300)</i>	
Id_dok	<i>Int(10)</i>	
Normx	<i>Float</i>	

#### **4.6. Perancangan Tampilan Menu Sistem**

Perancangan tampilan menu sistem penelitian ini dibuat dengan tujuan sebagai acuan dari tampilan implementasi dari sistem yang akan dibangun.

##### *1. Form Halaman Awal*

*Form* ini merupakan halaman awal sistem. Halaman memiliki menu *login* bagi *admin* untuk mengakses seluruh sistem dan menu pencarian bagi *user*.

SISTEM PENCARIAN AYAT ALQURAN DENGAN PEMODELAN RUANG VEKTOR  
dengan Pemodelan Ruang Vektor

Username .

Password .

SUBMIT

RETRIEVAL

Silahkan Inputkan Keywords .

CARI

**Gambar 4.8. Rancangan Menu Halaman Awal Pencarian Ayat Al-Qur'an**

2. *Form Upload* Korpus

*Form* ini merupakan *form* yang digunakan oleh admin untuk melakukan proses preprosesing dari koleksi dokumen (korpus) terjemahan ayat Al-Qur'an bahasa Indonesia yang sebelumnya telah tersimpan di dalam *database*.

SISTEM PENCARIAN AYAT ALQURAN DENGAN PEMODELAN RUANG VEKTOR  
dengan Pemodelan Ruang Vektor

Welcome  
 Username = admin

MENU ADMIN

- » Upload Corpus
- Daftar Corpus
- Daftar Stopword
- Daftar Tanda Baca
- Kata Dasar

Indexing  
 Pembobotan

Retrieval

LOGOUT

CORPUS

Silahkan klik tombol di samping untuk Preprosesing dari corpus quran

PROSES

**Gambar 4.9. Rancangan Menu Preprosesing Korpus**

### 3. *Form* Menu Daftar Korpus

*Form* ini digunakan untuk menampilkan ayat Al-Qur'an beserta terjemahan berdasarkan surah yang telah dipilih oleh pengguna

SISTEM PENCARIAN AYAT AL-QURAN DENGAN PEMODELAN RUANG VEKTOR  
dengan Pemodelan Ruang Vektor

**Welcome**  
Username = admin

**MENU ADMIN**  
[Upload Corpus](#)  
[Daftar Corpus](#)  
[Daftar Stopword](#)  
[Daftar Tanya Baca](#)  
[Kata Dasar](#)

[Indexing](#)  
[Pembaharuan](#)

[Renewal](#)

[LOGOUT](#)

**LIST CORPUS**  
Pilih Surat :

**Gambar 4.10. Rancangan Menu Daftar Korpus**

### 4. *Form* Menu Daftar *Stopword*

*Form* ini berisi tabel daftar *stopword* yang digunakan pada proses preprosesing sebelumnya.

SISTEM PENCARIAN AYAT AL-QURAN DENGAN PEMODELAN RUANG VEKTOR  
dengan Pemodelan Ruang Vektor

**Welcome**  
Username = admin

**MENU ADMIN**  
[Upload Corpus](#)  
[Daftar Corpus](#)  
[Daftar Stopword](#)  
[Daftar Tanya Baca](#)  
[Kata Dasar](#)

[Indexing](#)  
[Pembaharuan](#)

[Renewal](#)

[LOGOUT](#)

**LIST STOPWORD**  
[Tambahkan Stopword Baru](#)

Id	Stopword	Type	Edit	Delete

Page :

**Gambar 4.11. Rancangan Menu Daftar *Stopword***

5. *Form* Menu Daftar *Stopword*

*Form* ini berisi tabel daftar tanda baca yang digunakan pada proses preprosesing sebelumnya.

The screenshot shows a web application interface with a title bar: "SISTEM PENCARIAN AYAT ALQURAN DENGAN PEMODELAN RUANG VEKTOR dengan Pemodelan Ruang Vektor". On the left, there is a "Welcome" section with "Username = admin" and a "MENU ADMIN" list containing: "Upload Corpus", "Daftar Corpus", "Daftar Stopword", ">>Daftar Tanda Baca", "Kata Dasar", "Indexing", "Penbobotan", "Retrieval", and "Logout". The main content area is titled "LIST STOPWORD" and includes a link "Tambahkan Tanda Baca Baru". Below this is a table with the following structure:

Id	Tanda Baca	Type	Edit	Delete

At the bottom of the table area, there is a "Page" indicator with a value of "1".

**Gambar 4.12. Rancangan Menu Daftar Tanda baca**

6. *Form* Menu Kata Dasar

*Form* ini berisi tabel daftar kata dasar yang digunakan pada proses *stemming* sebelumnya.

The screenshot shows the same web application interface as Gambar 4.12. The "MENU ADMIN" list now includes ">>Kata Dasar" instead of ">>Daftar Tanda Baca". The main content area is titled "LIST STOPWORD" and includes a link "Tambahkan Kata Dasar Baru". Below this is a table with the following structure:

Id	Kata Dasar	Type	Edit	Delete

At the bottom of the table area, there is a "Page" indicator with a value of "1".

**Gambar 4.13. Rancangan Menu Kata Dasar**



### 7. *Form Menu Indexing*

*Form* ini berisi tabel yang menampilkan 800 subtopik dokumen yang siap di *indexing*, ditampilkan juga status pada tiap subtopik yang telah belum melalui tahapan *indexing* maupun yang selesai diindeks beserta jumlah ayat yang berhasil diindeks.

SISTEM PENCARIAN AYAT ALQURAN DENGAN PEMODELAN RUANG VEKTOR  
dengan Pemodelan Ruang Vektor

**Welcome**  
Username= admin

---

**MENU ADMIN**

- [Upload Corpus](#)
- [Daftar Corpus](#)
- [Daftar Stopword](#)
- [Daftar Tanda Baca](#)
- [Kata Dasar](#)

---

**>>Indexing**  
[Pembobotan](#)

---

[Rampal](#)

---

[LOGOUT](#)

**INDEXING PROGRESS**

Silahkan Klik Di Status Untuk Melakukan Indexing

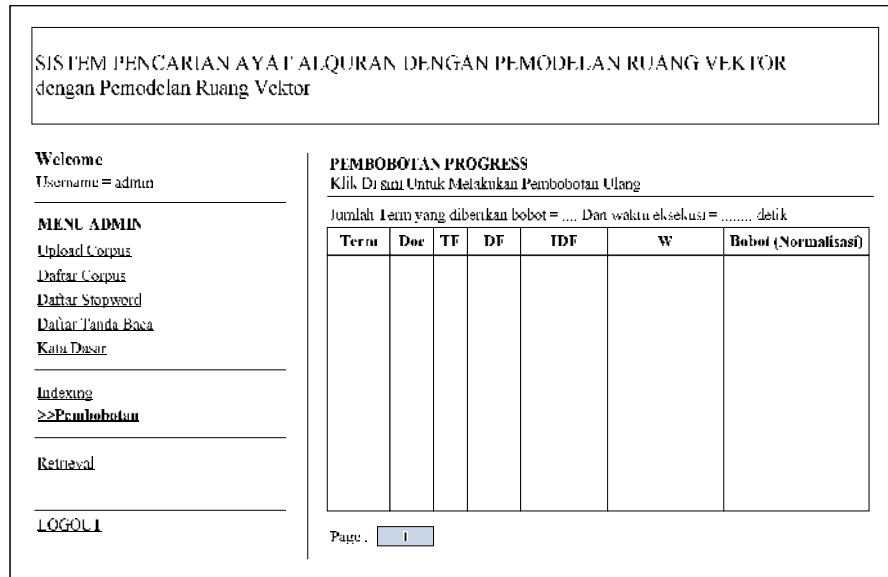
No	Subtopik	Status	Jumlah Teraf Di Index	Total Ayat

Page .

**Gambar 4.14. Rancangan Menu *Indexing***

### 8. *Form Menu Pembobotan*

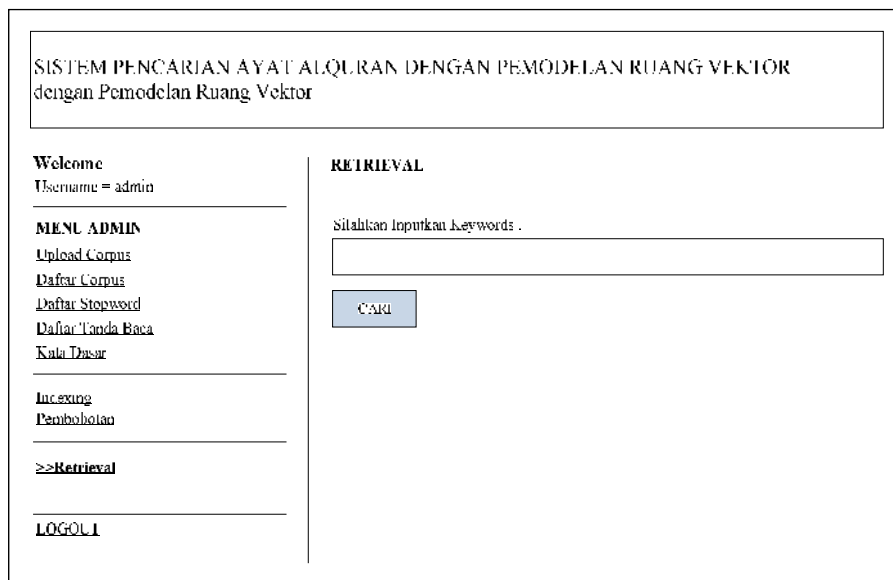
*Form* ini berisi tabel seluruh hasil perhitungan pembobotan dan normalisasi *term* pada saat tahapan *indexing* berdasarkan algoritma ruang vektor.



**Gambar 4.15. Rancangan Menu Pembobotan**

9. *Form Retrieval*

*Form* ini merupakan halaman *retrieval* bagi admin. Hasil pencarian dokumen berdasarkan *query* pengguna akan menampilkan urutan daftar subtopik dokumen dari tingkat kerelevanan yang paling tinggi.



**Gambar 4.16. Rancangan Menu *Retrieval* Admin**

## **BAB V**

### **IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

#### **5.1 Implementasi**

Setelah menganalisis dan merancang sistem yang akan dibangun secara rinci, maka selanjutnya menuju tahap implementasi. Tujuan implementasi adalah untuk mengkonfirmasi modul program perancangan kepada para pengguna sistem, sehingga pengguna dapat memberikan masukan terhadap pembangunan sistem.

##### **5.1.1. Kebutuhan Perangkat Keras**

Perangkat keras yang diperlukan untuk mengimplementasikan perangkat lunak dari Aplikasi *Information Retrieval System* adalah sebagai berikut :

**Tabel 5.1. Perangkat Keras yang Digunakan**

No	Perangkat Keras	Spesifikasi
1	<i>Processor</i>	<i>Intel Pentium processor T4200</i>
2	<i>RAM</i>	<i>2.0 GHz</i>
3	<i>HDD</i>	<i>320 GB</i>
4	Monitor	<i>15.6" HD LCD</i>

### 5.1.2. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan aplikasi *Information Retrieval System* ini adalah sebagai berikut :

**Tabel 5.2. Perangkat Lunak yang Digunakan**

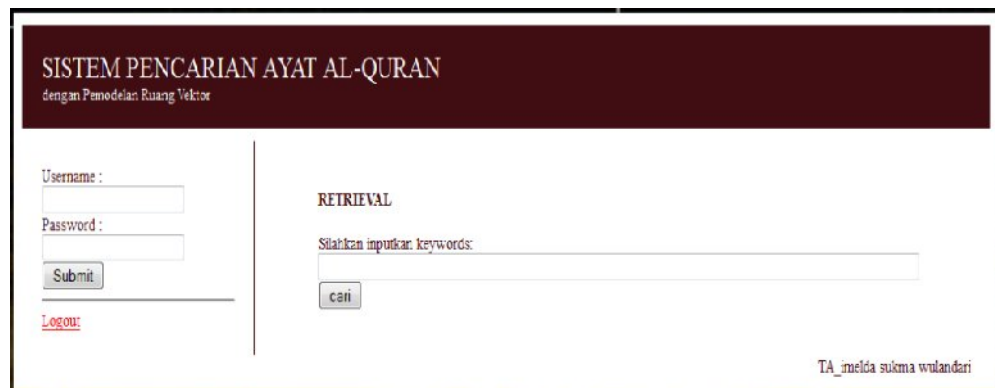
No	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	<i>Windows 7 Ultimate</i>
2	Bahasa Pemrograman	<i>Php</i>
3	<i>Database</i>	<i>MySQL</i>
4	<i>Tools Perancangan</i>	<i>Notepad++</i>
5	<i>Browser</i>	<i>Mozilla Firefox</i>

### 5.1.3. Hasil Implementasi *Interface* Sistem

Adapun hasil implementasi *interface* pada sistem pencarian ayat Al-Qur'an ini mengacu pada rancangan *interface* pada bab analisa dan perancangan.

#### 1. *Form* Halaman Awal

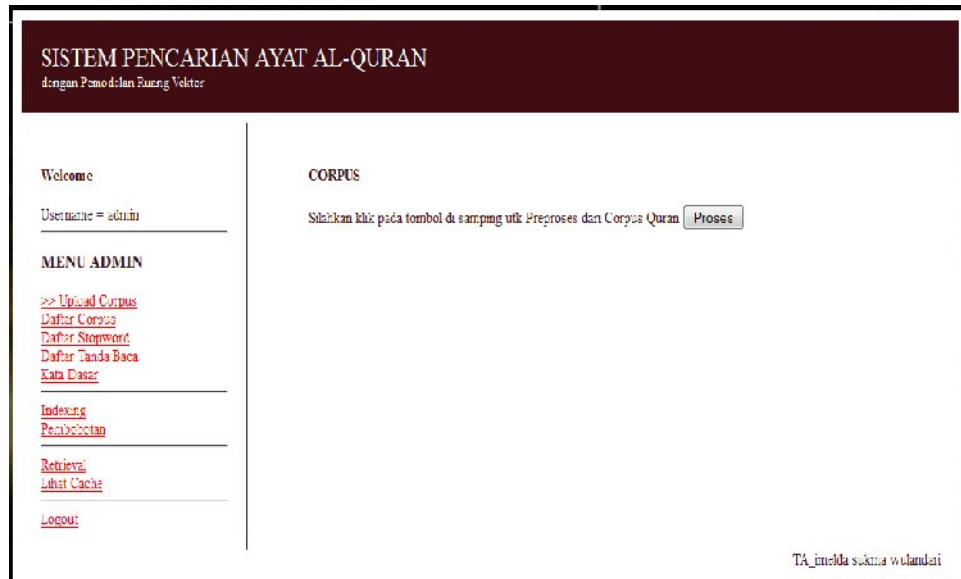
*Form* ini merupakan halaman awal sistem. Halaman memiliki menu *login* bagi admin untuk mengakses seluruh sistem dan menu pencarian bagi pengguna. Admin harus mengisi *username* dan *password* untuk dapat mengakses halaman utama. Jika terjadi kesalahan *input* baik *username* maupun *password*, aplikasi akan memberi konfirmasi dan mempersilahkan untuk mengulangi masukan.



**Gambar 5.1. Menu Halaman Utama**

## 2. Form Upload Corpus

Form ini merupakan *form* yang digunakan oleh admin untuk melakukan proses preprosesing yaitu *tokenizing*, *filtering*, dan *stemming* dari koleksi dokumen (korpus) yang sebelumnya telah tersimpan di dalam *database*. Tombol Proses digunakan untuk memulai proses preprosesing.



**Gambar 5.2. Menu Preprosesing Corpus**

Saat tahapan preprosesing berlangsung, data yang digunakan sebagai bahan pengolahan adalah data korpus teks terjemahan ayat Al-Qur'an yang disimpan dalam tabel *quran*. Pada proses *filtering*, data yang digunakan adalah data pada tabel *tandabaca*, *stopword*, dan *stemlist*. Hasil pengolahan preprosesing teks terjemahan ayat sejumlah 74.199 buah *term* disimpan dalam tabel *stem* di dalam *database*. Gambar tabel *quran* dan tabel *stem* dapat dilihat pada Gambar 5.3 dan 5.4.

Showing rows 0 - 29 (6,236 total, Query took 0.0021 sec)

SQL query:

```
SELECT *
FROM 'quran'
LIMIT 0 , 30
```

[ Edit ] [ Explain SQL ] [ Create PHP Code ] [ Refresh ]

Show: 30 row(s) starting from record # 30

in horizontal mode and repeat headers after 100 cells

Sort by key: None

	ID	DatabaseID	SuraID	VerseID	AyahText		
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	1	68	1	1	Dengan menyebut nama Allah Yang Maha Pemurah lagi ...
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2	68	1	2	Segala puji bagi Allah, Tuhan semesta alam.
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	3	68	1	3	Maha Pemurah lagi Maha Penyayang.
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	4	68	1	4	Yang menguasai di Hari Pembalasan.
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	5	68	1	5	Hanya Engkaulah yang kami sembah, dan hanya kepada...

Gambar 5.3. Tabel Quran

Showing rows 0 - 29 (74,199 total, Query took 0.0018 sec)

SQL query:

```
SELECT *
FROM 'stem'
LIMIT 0 , 30
```

[ Edit ] [ Explain SQL ] [ Create PHP Code ] [ Refresh ]

Show: 30 row(s) starting from record # 30

in horizontal mode and repeat headers after 100 cells

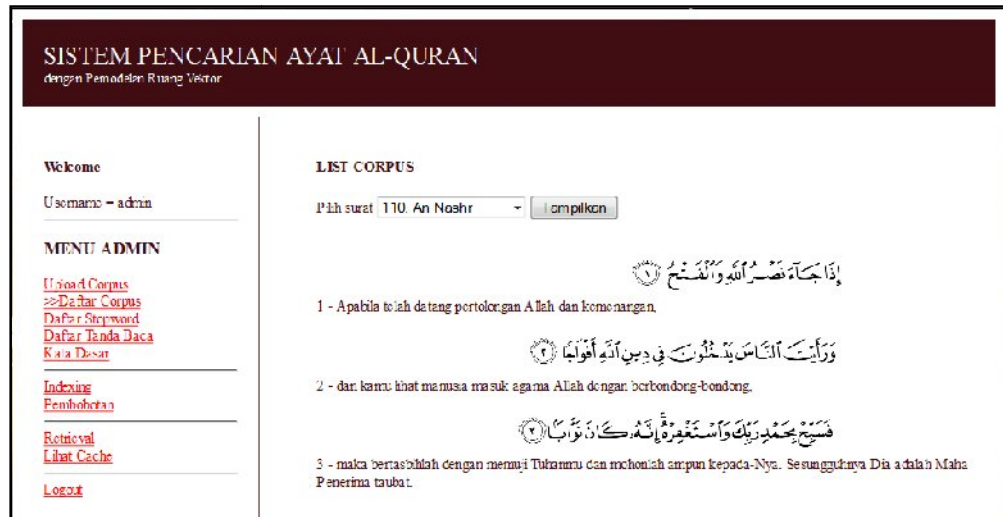
Sort by key: None

	id	term	stem	id_surah	ayat
<input type="checkbox"/>	1	sebut	menyebut	1	1
<input type="checkbox"/>	2	nama	nama	1	1
<input type="checkbox"/>	3	allah	allah	1	1
<input type="checkbox"/>	4	maha	maha	1	1
<input type="checkbox"/>	5	murah	pemurah	1	1
<input type="checkbox"/>	6	maha	maha	1	1
<input type="checkbox"/>	7	sayang	penyayang	1	1
<input type="checkbox"/>	8	puji	puj	1	2
<input type="checkbox"/>	9	allah	allah	1	2
<input type="checkbox"/>	10	tuhan	tuhan	1	2
<input type="checkbox"/>	11	semesta	semesta	1	2
<input type="checkbox"/>	12	alam	alam	1	2
<input type="checkbox"/>	13	murah	pemurah	1	3
<input type="checkbox"/>	14	maha	maha	1	3
<input type="checkbox"/>	15	sayang	penyayang	1	3
<input type="checkbox"/>	16	kuasa	menguasai	1	4

Gambar 5.4. Tabel Stem

### 3. Form Menu Daftar Corpus

Form ini digunakan untuk menampilkan ayat Al-Qur'an beserta terjemahan berdasarkan surah yang telah dipilih oleh pengguna.



Gambar 5.5. Menu Daftar Korpus

Pada implementasi menampilkan daftar ayat dan terjemahan surah, data yang digunakan dalam pengolahan adalah data dari tabel *surah* berelasi dengan tabel *quran* yang disimpan dalam *database* dan korpus gambar ayat (tulisan arab) yang disimpan terpisah dalam *folder eksternal*. Tabel *surah* dapat dilihat dari Gambar 5.6.

id_surah	nama_surah	jumlah_ayat	turun
1	Al-Fatihah	7	Makiyah
2	Al-Baqarah	286	Madinah
3	Ali Imran	200	Madinah
4	An-Nisaa	176	Madinah
5	Al-Maidah	120	Madinah

Gambar 5.6. Tabel Surah

#### 4. Form Menu Daftar Stopword

Form ini berisi tabel daftar *stopword* yang digunakan pada proses preprosesing sebelumnya.

**SISTEM Pencarian Ayat Al-Quran**  
dengan Pemodelan Ruang Vektor

Welcome  
Username = admin

**MENU ADMIN**

- [Upload Corpus](#)
- [Daftar Corpus](#)
- [->Daftar Stopword](#)
- [Daftar Tanda Baca](#)
- [Kata Dasar](#)

---

- [indexing](#)
- [Pembobotan](#)

---

- [Retrieval](#)
- [Lihat Cache](#)

---

- [Logout](#)

**LIST STOP WORD**

[Tambahkan Stopword Baru](#)

ID	Stem	Type	Edit	Delete
1	ada		<a href="#">Edit</a>	<a href="#">Delete</a>
2	adalah		<a href="#">Edit</a>	<a href="#">Delete</a>
3	adanya		<a href="#">Edit</a>	<a href="#">Delete</a>
4	adapte		<a href="#">Edit</a>	<a href="#">Delete</a>
5	anak		<a href="#">Edit</a>	<a href="#">Delete</a>
6	anaknya		<a href="#">Edit</a>	<a href="#">Delete</a>
7	agar		<a href="#">Edit</a>	<a href="#">Delete</a>
8	akan		<a href="#">Edit</a>	<a href="#">Delete</a>
9	akanlah		<a href="#">Edit</a>	<a href="#">Delete</a>
10	akhirnya		<a href="#">Edit</a>	<a href="#">Delete</a>
11	amat		<a href="#">Edit</a>	<a href="#">Delete</a>
12	amatlah		<a href="#">Edit</a>	<a href="#">Delete</a>
13	antar		<a href="#">Edit</a>	<a href="#">Delete</a>

**Gambar 5.7. Menu Daftar Stopword**

#### 5. Form Menu Daftar Tanda Baca

Form ini berisi tabel daftar tanda baca yang digunakan pada proses preprosesing sebelumnya.

**SISTEM Pencarian Ayat Al-Quran**  
dengan Pemodelan Ruang Vektor

Welcome  
Username = admin

**MENU ADMIN**

- [Upload Corpus](#)
- [Daftar Corpus](#)
- [Daftar Stopword](#)
- [->Daftar Tanda Baca](#)
- [Kata Dasar](#)

---

- [indexing](#)
- [Pembobotan](#)

---

- [Retrieval](#)
- [Lihat Cache](#)

---

- [Logout](#)

**LIST TANDABACA**

[Tambahkan Tanda Baca Baru](#)

ID	Tandabaca	Type	Edit	Delete
1	!		<a href="#">Edit</a>	<a href="#">Delete</a>
2	&		<a href="#">Edit</a>	<a href="#">Delete</a>
3	/		<a href="#">Edit</a>	<a href="#">Delete</a>
4	\		<a href="#">Edit</a>	<a href="#">Delete</a>
5	?		<a href="#">Edit</a>	<a href="#">Delete</a>
6	.		<a href="#">Edit</a>	<a href="#">Delete</a>
7	(		<a href="#">Edit</a>	<a href="#">Delete</a>
8	)		<a href="#">Edit</a>	<a href="#">Delete</a>
9	-		<a href="#">Edit</a>	<a href="#">Delete</a>
10	)		<a href="#">Edit</a>	<a href="#">Delete</a>
11	'		<a href="#">Edit</a>	<a href="#">Delete</a>
12			<a href="#">Edit</a>	<a href="#">Delete</a>
13	?		<a href="#">Edit</a>	<a href="#">Delete</a>

**Gambar 5.8. Menu Daftar Tanda baca**



## 6. Form Menu Kata Dasar

Form ini berisi tabel daftar kata dasar yang digunakan pada proses *stemming* sebelumnya.

ID	Stem	Type	Edit	Delete
1	a	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
2	ab	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
3	aba	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
4	aba-aba	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
5	abad	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
6	abad.	Adjektiva	<a href="#">Edit</a>	<a href="#">Delete</a>
7	abadah	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
8	abah	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
9	aba	Adjektiva	<a href="#">Edit</a>	<a href="#">Delete</a>
10	abamara	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
11	abaka	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>
12	abaktinal	Adjektiva	<a href="#">Edit</a>	<a href="#">Delete</a>
13	abalisa	Nomina	<a href="#">Edit</a>	<a href="#">Delete</a>

Gambar 5.9. Menu Kata Dasar

## 7. Form Menu Indexing

Form ini berisi tabel yang menampilkan daftar sebanyak 800 subtopik dokumen yang siap diindeks, ditampilkan juga status pada tiap subtopik yang telah selesai melalui tahapan *indexing* maupun yang belum selesai diindeks beserta jumlah ayat yang berhasil diindeks.

No	Subtopik	Status	Jumlah Ayat di-index	Jumlah Total
1	Golongan muslimin	Selesai	5	5
2	Golongan kafir	Selesai	2	2
3	Golongan munafik	Selesai	13	13
4	Perintah menyembah Tuhan Yang Maha Esa	Selesai	2	2
5	Tantangan kepada kaum Musyrikin mengenai al-Quran	Selesai	2	2
6	Belasah terhadap orang-orang yang beriman	Selesai	3	3
7	Bukti bukti kekuasaan Tuhan	Selesai	2	2
8	Penciptaan Tuhan dan Penguasaannya di Bumi	Selesai	10	10
9	Beberapa perintah dan larangan Tuhan kepada Bani Israil	Selesai	9	9
10	Perincian nikmat Tuhan kepada Bani Israil	Selesai	12	12

Gambar 5.10. Menu Indexing

Pada Implementasi tahapan *indexing*, data yang digunakan sebagai bahan pengolahan adalah data dari tabel *dokumen* yang berisi seluruh subtopik Al-Qur'an berelasi dengan tabel *stem* yang berisi seluruh *term* hasil preprosesing. Tabel dokumen dapat dilihat pada Gambar 5.11.

Showing rows 0 - 25 (800 total, Query took 2.0015 sec)

SQL query:

```
SELECT *
FROM 'dokumen'
LIMIT 0, 30
```

in horizontal mode and repeat headers after 100 cells

Sort by key: None

	id	id_surah	topik	subtopik	dari_ayat	sampai_ayat
	1	2	Tiga golongan manusia dalam menghadapi al-Qur'an	Golongan mukmin	1	5
	2	2	Tiga golongan manusia dalam menghadapi al-Qur'an	Golongan kafir	6	7
	3	2	Tiga golongan manusia dalam menghadapi al-Qur'an	Golongan munafik	8	20
	4	2	Keesaan dan kekuasaan Tuhan	Perintah menyembah Tuhan Yang Maha Esa	21	22
	5	2	Keesaan dan kekuasaan Tuhan	Tantangan kepada kaum Musyrikin mengenal al-Quran	23	24
	6	2	Keesaan dan kekuasaan Tuhan	Balasan terhadap orang-orang yang beriman	25	27
	7	2	Keesaan dan kekuasaan Tuhan	Bukti-bukti kekuasaan Tuhan	28	29
	8	2	Keesaan dan kekuasaan Tuhan	Pendataan Tuhan dan Penguasaannya di Bumi	30	39
	9	2	Peringatan Tuhan Kepada Ban Israil	Beberapa perintah dan larangan Tuhan kepada Sari I...	40	48

**Gambar 5.11. Tabel Dokumen**

Pada proses implementasi *indexing*, sistem telah menerapkan perhitungan pembobotan *term* dengan algoritma model ruang vektor, yaitu perhitungan *tf*, *idf*, bobot, serta normalisasi bobot *term* yang nantinya digunakan untuk menghitung nilai kesamaan (*similarity*) antara *term* dengan *query*. Hasil perhitungan *indexing* akan disimpan dalam tabel *indexing*. Tabel *indexing* dapat dilihat pada Gambar 5.12.

Showing rows 450 - 479 (61.850 total, Query took 0.0021 sec)

```

--SQL query--
SELECT *
FROM indexing
LIMIT 30

```

Page number: 18

id	term	id_dok	id_suat	suat	tf	df	idf	weight	weight1	normalisasi	normalisasi1
451	bulan	9	46	2.200-2.200-2.200-2.205-2.205-2.200-2.202-2.277-2...	1	420	0.27578	0.27675	0.076504	0.0244264	0.003900581
452	yakini	9	46	2.260-2.260-2.205-2.249-2.218-2.102-2.45-2.26-2.4...	1	31	1.19302	1.19302	1.42927	0.100319	0.0111342
453	...	9	47	2.264-2.264-2.264-2.247-2.47-2.40-2.3...	1	31	1.19302	1.19302	1.84694	0.111949	0.0148979
454	atas	9	47	2.284-2.284-2.260-2.253-2.253-2.251-2.250-2.246-2...	1	256	0.49482	0.49482	0.244677	0.0436763	0.03190762
455	bon	9	47	2.246-2.211-2.102-2.106-2.86-2.86-2.17-2.10...	1	11	1.29301	1.29301	1.66189	0.118990	0.0129696
456	inact	9	47	2.282-2.260-2.281-2.211-2.213-2.152-2.152-2.127-2...	0	227	0.547061	1.09412	1.19712	0.0965598	0.03932572
457	israil	9	47	2.240-2.211-2.122-2.103-2.05-2.03-2.47-2.40	1	37	1.32405	1.32405	1.70100	0.11702	0.0130015
458	niku nat	9	47	2.231-2.211-2.211-2.102-2.101-2.103-2.122-2.47-2.4...	1	100	0.90305	0.90305	0.310072	0.0797083	0.00603341
459	...	9	47	2.251-2.215-2.148-2.143-2.143-2.142-2.141-2.134-2...	1	71	1.19302	1.19302	1.42927	0.100319	0.0111342
460	azab	9	48	2.281-2.148-	1	207	0.58712	0.58712	0.344706	0.0518202	0.00268534
461	hala	9	48	2.281-2.148-	1	14	1.19302	1.19302	0.20115	0.0324518	0.0046567

Gambar 5.12. Tabel *Indexing*

8. *Form* Menu Pembobotan

*Form* ini berisi tabel seluruh hasil perhitungan pembobotan dan normalisasi *term* pada saat tahapan *indexing* berdasarkan perhitungan algoritma ruang vektor.

SISTEM PENCARIAN AYAT AL-QURAN  
dengan Pemodelan Ruang Vektor

Welcome  
Username = admin

MENU ADMIN  
[Upload Corpus](#)  
[Daftar Corpus](#)  
[Daftar Stopword](#)  
[Daftar Tanda Baca](#)  
[Kata Dasar](#)  
[Indexing](#)  
[>>Pembobotan](#)  
[Retrieval](#)  
[Lihat Cache](#)  
 Logout

PEMBBOTAN PROGRESS  
 Klik di [sini](#) untuk melakukan pembobotan  
 Jumlah term yang diberikan bobot = 61890 dan waktu eksekusi 0.001881 detik

Term	Doc	TF	IDF	IDF	W	Bobot (Normalisasi)
bulan	1-1	1	13	1.78915	1.78915	0.725146
mim	1-1	1	16	1.69597	1.69597	0.688596
ai	1-2	1	193	0.617533	0.617533	0.310903
quran	1-2	1	158	0.709965	0.709965	0.357138
ragu	1-2	1	27	1.47173	1.47173	0.740956
talwa	1-2	1	158	0.709965	0.709965	0.357138
hunik	1-2	1	190	0.624316	0.624316	0.314328
arungerah	1-3	1	35	1.35902	1.35902	0.446815
bin	1-3	1	787	0.445708	0.445708	0.146174
ghaib	1-3	1	32	1.39794	1.39794	0.459511

Gambar 5.13. Menu Pembobotan

## 9. Form Retrieval

Form ini merupakan halaman *retrieval* bagi admin. Hasil pencarian dokumen berdasarkan *query* pengguna akan menampilkan urutan daftar subtopik dokumen dari tingkat kerelevanan yang paling tinggi.

SISTEM PENCARIAN AYAT AL-QURAN  
dengan Pemodelan Ruang Vektor

Website  
Username - admin

MENU ADMIN

- [Upload Corpus](#)
- [Daftar Corpus](#)
- [Daftar Stopword](#)
- [Daftar Tanda Baca](#)
- [Kata Dasar](#)
- [Indexing](#)
- [Pemeliharaan](#)
- [>> Retrieval](#)
- [Chat Admin](#)
- [Logout](#)

RETRIEVAL  
waktu proses - 0.002022 detik  
Silahkan inputkan keywords:  
kasah musa  
car

HASIL QUERY  
Maka ketika datang kepada mereka kebonaran dari sisi Kami, mereka berkata: "Mengapakah tidak diurunkan kepadanya (Muhammad) seperti yang telah diberikan kepada Musa dahulu?" Dan bukankah mereka itu telah mengkar (juga) kepada apa yang telah diberikan kepada Musa dahulu?; mereka dahulu telah berkata: "Musa dan Harun adalah dua ahli sukur yang bantu membantu". Dan mereka (juga) berkata: "Sesungguhnya kamu tidak mempercayai masing-masing mereka itu".  
Similarity: 0.31985068321228  
Topik: Kisah Musa A.S. Dan Fu'ad: Sebagai Bukti Kebenaran Al-Quran  
SubTopik: Oran-Oran Kahu Selalu Merupakan Bukti, Sekalipun Dahulu Dirimanya (Al-Qashash: 48 - 51)

Mengapa kamu datang lebih cepat daripada kaummu, hai Musa?  
Similarity: 0.237539306282997  
Topik: Kisah Nabi Musa a.s  
SubTopik: Teguran Allah kepada Nabi Musa a.s. ( Taaha: 83 - 84)

Gambar 5.14. Menu *Retrieval* Admin

Pada implementasi tahapan *retrieval*, setiap *query* yang dimasukkan akan diekstrak kembali melalui tahapan preprosesing hingga dihasilkan *term-term* tunggal yang akan dihitung nilai *similarity* antara dokumen dan *query*. Hasil pencarian dokumen subtopik Al-Qur'an relevan akan diurutkankan dari tingkat nilai *similarity* yang paling tinggi. Pengolahan data yang digunakan pada perhitungan nilai *similarity* adalah pada tabel *bobotq* yang menyimpan nilai bobot tiap *query* hasil ekstrak dari proses preprosesing, tabel *normq* menyimpan hasil perhitungan normalisasi *query*, dan tabel *wijxwiq* yang menyimpan hasil perkalian antara bobot *query* yang dinormalisasi dengan normalisasi bobot *term* untuk seluruh dokumen yang mengandung *term* berdasarkan *query* yang dimasukkan pengguna. Tabel *wijxwiq* dapat dilihat pada Gambar 5.15.

Showing rows 0 - 29 (98 total, Query took 0.0011 sec)

**SQL query:**

```
SELECT *
FROM 'wijxwiq'
ORDER BY 'id_dok' ASC
LIMIT 0 , 30
```

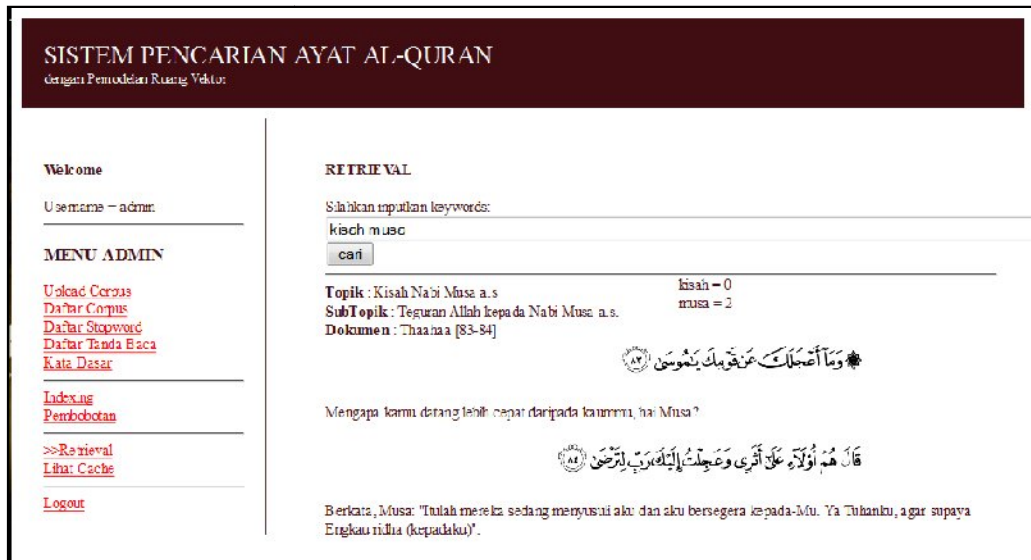
[ Edit ] [ Explain SQL ] [ Create PHP Code ] [ Refresh ]

Show : 30 row(s) starting from record # 30  
 in horizontal mode and repeat headers after 100 cells  
 Sort by key: None Go

	id	que	id_dok	normx
<input type="checkbox"/>	1	musa	10	0.0570444
<input type="checkbox"/>	2	musa	11	0.168731
<input type="checkbox"/>	3	musa	14	0.0591002
<input type="checkbox"/>	4	musa	17	0.0894847
<input type="checkbox"/>	5	musa	18	0.0775302
<input type="checkbox"/>	6	musa	22	0.0720795
<input type="checkbox"/>	7	musa	26	0.0561228
<input type="checkbox"/>	8	musa	46	0.0359493
<input type="checkbox"/>	9	kisah	65	0.0188766
<input type="checkbox"/>	10	musa	70	0.0718316
<input type="checkbox"/>	11	musa	116	0.102926
<input type="checkbox"/>	13	musa	117	0.0781392
<input type="checkbox"/>	12	kisah	117	0.0654044
<input type="checkbox"/>	14	musa	124	0.0787566
<input type="checkbox"/>	15	kisah	125	0.019068

Gambar 5.15. Tabel *wijxwiq*

Berdasarkan nilai *similarity* yang telah di dapat, untuk menampilkan dokumen hasil pencarian bagi pengguna, maka tabel *wijxwiq* akan berelasi dengan tabel *quran*, *dokumen*, dan korpus gambar ayat (tulisan arab). Dokumen relevan hasil pencarian akan menampilkan satu subtopik surah, ayat-ayat Al-Qur'an beserta terjemahan Al-Qur'an bahasa Indonesia. Gambar dokumen hasil pencarian dapat dilihat pada Gambar 5.16.



**Gambar 5.16. Dokumen Subtopik Hasil Pencarian**

## 5.2. Pengujian Kemampuan Sistem

Dalam tahapan ini, sistem akan dijalankan dan diuji cobakan untuk mengetahui apakah sistem berjalan sesuai dengan hasil analisa dan tujuan yang diharapkan. Untuk mengetahui kemampuan sistem yang telah dibangun, maka akan dilakukan pengujian dengan mengukur kualitas *retrieval*, yaitu dengan menghitung nilai *precision* dan *recall*.

### 5.2.1. Rencana Pengujian

Rencana pengujian untuk sistem tugas akhir ini mengacu kepada pengujian terdahulunya, yaitu pengujian terhadap penelitian tugas akhir Sistem Temu Kembali Informasi dengan Metode *TF-IDF* Menggunakan Model *Neural Network* (Suryani, 2012). Berikut rencana pengujian tersebut.

1. Memasukkan 12 *query* yang berbeda.
2. Dokumen yang akan diujicobakan adalah sesuai dengan keseluruhan subtopik ada pada sistem.
3. Menilai kemampuan sistem dalam mengembalikan dokumen yang relevan dengan *query* pengguna dengan cara menghitung nilai *precision* dan *recall* sebagai parameter kualitas *retrieval*.



### 5.2.2. Hasil Pengujian

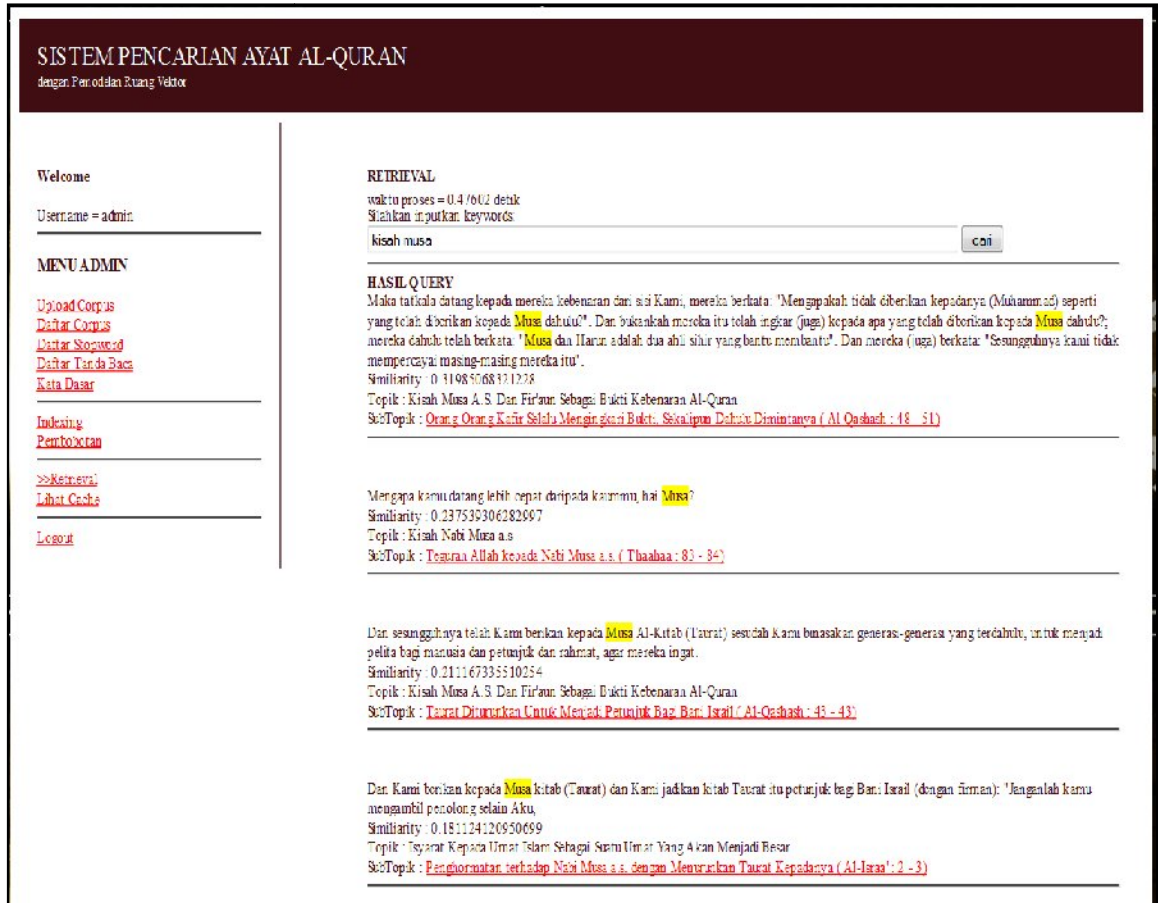
Berikut daftar *query* yang dimasukkan untuk dilakukan pengujian.

**Tabel 5.3. Daftar *Query* yang Dimasukkan**

No	<i>Query</i>
1	Kisah Musa
2	Berpuasa di bulan ramadhan
3	Allah menciptakan langit dan bumi
4	Pohon zaitun yang menghasilkan minyak
5	Menunaikan zakat
6	Tidak ada keraguan dalam alquran
7	Berbuat riba
8	Arah kiblat saat sholat
9	Perceraian
10	Nazar
11	Perkawinan
12	Berlaku Adil

Untuk memperoleh nilai *precision* dan *recall*, digunakan Persamaan 2.7, Persamaan 2.8 sebagai acuan dalam penghitungannya. Hasil perhitungan *recall* dan *precision* beserta profil hubungan antara *precision-recall* yang ditampilkan berupa grafik data interpolasi akan dilampirkan pada perhitungan untuk *query* 1 (menunaikan zakat), *query* 2 (berpuasa di bulan ramadhan), *query* 3 (Allah menciptakan langit dan bumi), dan *query* 4 (pohon zaitun yang menghasilkan minyak) dapat dilihat pada Tabel 5.4 sampai Tabel 5.7 dan Gambar 5.21 Sampai 5.24.

Tampilan hasil pencarian pada sistem untuk *query* “kisah musa” dapat dilihat pada Gambar 5.17.



Gambar 5.17. Tampilan hasil pencarian untuk *query* “kisah musa”

Tabel 5.4. Perbandingan Hasil Perhitungan *presicion(p)* dan *recall(r)* pada  $Q_1$

NO	Subtopik	Surah	Relevan?	Recall	Precision
1	Orang-Orang Kafir Selalu Mengingkari Bukti, Sekalipun Dahulu Dimintanya (dok.450)	Al-Qashash 48 - 51	Ya	1/20= 0,05	1/1= 1,00
2	Teguran Allah kepada Nabi Musa a.s. (dok.353)	Thaahaa 83 - 84	Ya	2/20= 0,10	2/2= 1,00
3	Taurat Diturunkan Untuk Menjadi Petunjuk Bagi Bani Israil (dok.448)	Al-Qashash 43 - 43	Ya	3/20= 0,15	3/3= 1,00



4	Penghormatan terhadap Nabi Musa a.s. dengan Menurunkan Taurat kepadanya (dok.302)	Al-Israa' 2 - 3	Ya	4/20= 0,20	4/4= 1,00
5	Takwa Kepada Allah Membawa Kepada Perbaikan Amal Dan Ampunan Dosa (dok.518)	Al-Ahzab 69 - 71	Ya	5/20= 0,25	5/5= 1,00
6	Cobaan Terhadap Nabi Musa a.s. (dok.470)	Al-Ankabuut 39 - 39	Ya	6/20= 0,30	6/6= 1,00
7	Pembalasan terhadap sikap dan perbuatan Bani Israil (dok.11)	Al-Baqarah 61 - 61	Ya	7/20= 0,35	7/7= 1,00
8	Hardikan Musa a.s terhadap Samiri (dok.356)	Thaahaa 95 - 96	Ya	8/20= 0,40	8/8= 1,00
9	Nabi Musa a.s. Menerima Permulaan Wahyu (dok.345)	Thaahaa 9 - 16	Ya	9/20= 0,45	9/9= 1,00
10	Kehancuran Fir'aun Hendaklah Menjadi Pelajaran Bagi Umat Yang Datang Kemudian (dok.615)	Az-Zukhruf 46 - 56	Ya	10/20= 0,50	10/10= 1,00
11	Kekejaman Fir'aun dan Pertolongan Allah kepada Bani Israil, Kaum Yang Tertindas (dok.441)	Al-Qashash 1 - 6	Ya	11/20= 0,55	11/11= 1,00
12	Kisah Musa A.S. Dan Fir'aun Sebagai Penghibur Bagi Nabi Muhammad SAW. (dok.746)	An-Naazi'at 15 - 26	Ya	12/20= 0,60	12/12= 1,00
13	Persamaan pokok-pokok agama yang diwahyukan kepada para Rasul (dok.117)	An-Nisaa 163 - 170	Ya	13/20= 0,65	13/13= 1,00
14	Azab Yang Ditimpakan kepada Samiri (dok.357)	Thaahaa 97 - 98	Ya	14/20= 0,70	14/14= 1,00
15	Khidir Membunuh Seorang Anak (dok.325)	Al-Kahfi 74 - 76	Ya	15/20= 0,75	15/15= 1,00
16	Musa a.s. Diberi Ilham dan Hikmah Sebagai Persiapan Untuk Menjadi Rasul (dok.443)	Al-Qashash 14 - 19	Ya	16/20= 0,80	16/16= 1,00
17	Pengkhianatan Samiri (dok.354)	Thaahaa 85 - 91	Ya	17/20= 0,85	17/17= 1,00
18	Kisah Nabi Musa a.s (dok.237)	Huud 96 - 99	Ya	18/20= 0,90	18/18= 1,00

19	Musa a.s. Dicampakkan ke Dalam Sungai Nil Untuk Menyelamatkan Kaumnya dari Kerajaan Fir'aun (dok.442)	Al-Qashash 7 – 13	Ya	19/20= 0,95	19/19= 1,00
20	Teguran Musa a.s. kepada Harun a.s. dan Balasan Harun a.s. (dok.355)	Thaahaa 92 – 94	Ya	20/20= 1,00	20/20= 1,00

Tampilan hasil pencarian pada sistem untuk *query* “Berpuasa di bulan ramadhan” dapat dilihat pada Gambar 5.18.

Gambar 5.18. Tampilan hasil pencarian untuk *query* “berpuasa di bulan ramadhan”

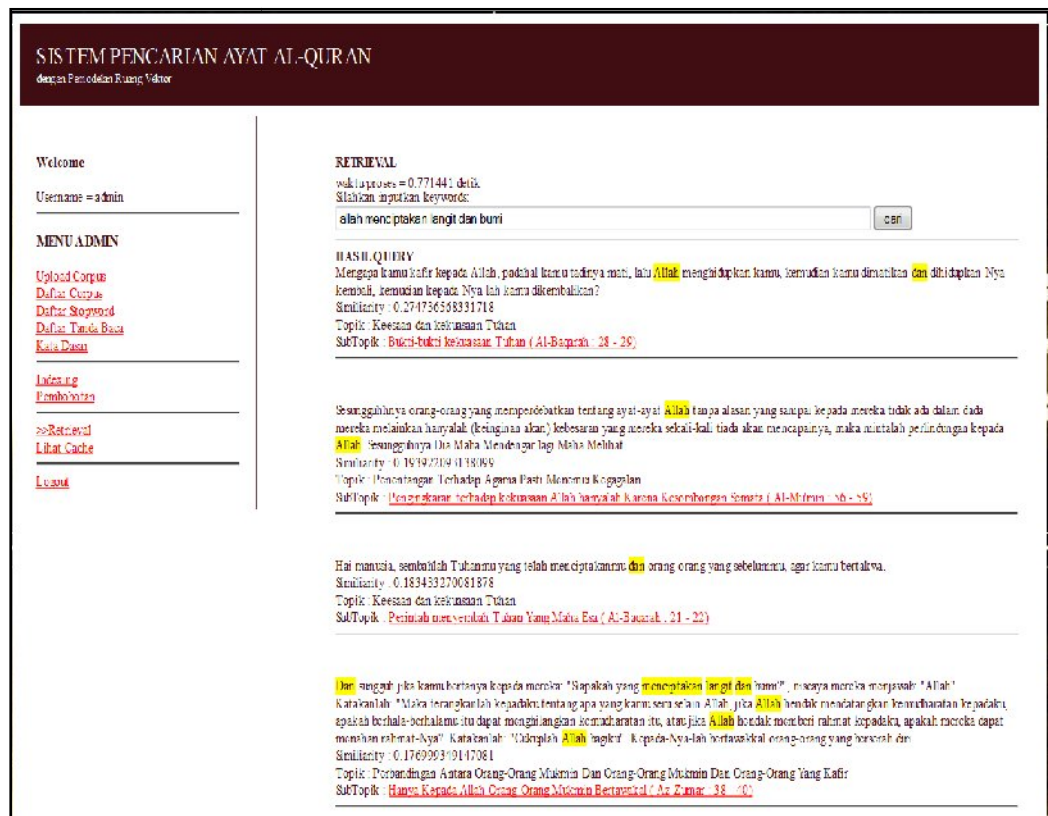
Tabel 5.5. Perbandingan Hasil Perhitungan *presicion(p)* dan *recall(r)* pada  $Q_2$

NO	Subtopik	Surah	Relevan?	Recall	Precision
1	Puasa (dok.35)	Al-Baqarah 183 - 188	Ya	1/1= 1,00	1/1= 1,00
2	Hukum perang dalam Islam (dok.41)	Al-Baqarah 216 - 218	Tidak	1/1= 1,00	1/2= 0,50

3	Bulan-bulan yang Dihormati (dok.198)	At-Taubah 36 - 37	Tidak	1/1= 1,00	1/3= 0,33
4	Wahyu dan Dasar-dasar Kebenarannya (dok.217)	Yunus 1 - 6	Tidak	1/1= 1,00	1/4= 0,25
5	Kemuliaan Lailatulqadar (dok.782)	Al-Qadr 1 - 5	Tidak	1/1= 1,00	1/5= 0,20
6	Allah Menentukan Rezeki Tiap-Tiap Makhluk (dok.476)	Al- 'Ankabuut 60 - 63	Tidak	1/1= 1,00	1/6= 0,17
7	Cara Nabi Ibrahim A.S. Memimpin Kaumnya Kepada Agama Tauhid (dok.149)	Al-An'am 74 - 83	Tidak	1/1= 1,00	1/7= 0,14
8	Kepandaian-Kepandaian Yang Diberikan Kepada Daud A.S. Dan Kekuasaan Yang Diberikan Kepada Sulaiman a.s (dok.521)	Saba' 10 - 14	Tidak	1/1= 1,00	1/8= 0,13
9	Manusia Mengalami Proses Kehidupan Tingkat Demi Tingkat (dok.761)	( Al- Insyiqaaq : 16 - 25)	Tidak	1/1= 1,00	1/9= 0,11
10	Hukum Zihar (dok.680)	Almujudilah 1 - 6	Tidak	1/1= 1,00	1/10= 0,10
11	Siapakah Tuhan Semesta Alam dan Bagaimana Bermohon Kepada-Nya? (dok.160)	Al-A'raf 54 - 56	Tidak	1/1= 1,00	1/11= 0,09
12	Beberapa Nikmat Allah yang Dilimpahkan kepada Hamba-hamba-Nya (dok.271)	Ibrahim 32 - 34	Tidak	1/1= 1,00	1/12= 0,08
13	Pertolongan Allah Pasti Datang (dok.371)	Al-Hajj 15 - 18	Tidak	1/1= 1,00	1/13= 0,08
14	Kekuasaan Allah Menghidupkan Manusia Seperti Semula (dok.731)	Al- Qiyaamah 1 - 15	Tidak	1/1= 1,00	1/14= 0,07
15	Manusia Diilhami Allah Jalan Yang Buruk Dan Yang Baik (dok.774)	Asy-Syams 1 - 15	Tidak	1/1= 1,00	1/15= 0,07
16	Haji (dok.37)	Al-Baqarah 196 - 203	Tidak	1/1= 1,00	1/16= 0,07
17	Berita Kehancuran Musuh Nabi Muhammad SAW (dok.660)	Al-Qamar 1 - 7	Tidak	1/1= 1,00	1/17= 0,06
18	Al-Quranul Karim adalah Petunjuk dan Penawar (dok.600)	Fushshilat 33 - 54	Tidak	1/1= 1,00	1/18= 0,06

19	Orang Islam Bebas dari Tanggung Jawab terhadap Perjanjian dengan Kaum Musyrikin (dok.191)	At-Taubah 1 – 4	Tidak	1/1= 1,00	1/19= 0,05
20	Berjihad dengan jiwa dan harta di jalan Allah SWT (dok.36)	Al-Baqarah : 189 – 195	Tidak	1/1= 1,00	1/20= 0,05

Tampilan hasil pencarian pada sistem untuk *query* “Allah menciptakan langit dan bumi” dapat dilihat pada Gambar 5.19.



Gambar 5.19. Tampilan hasil pencarian untuk *query* “Allah menciptakan langit dan bumi”

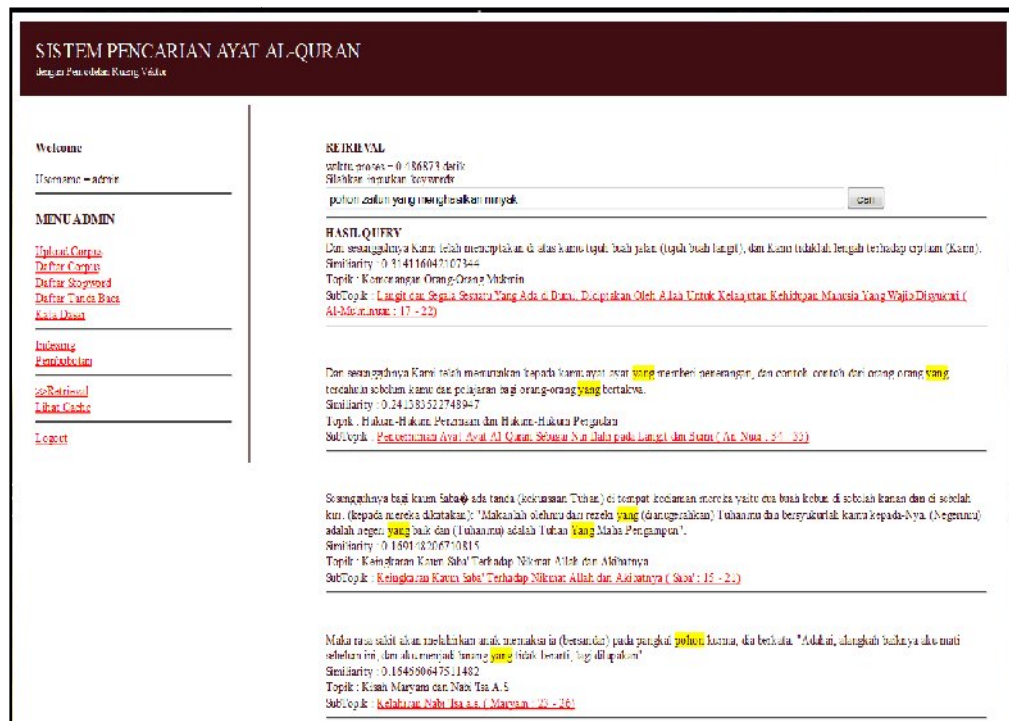
Tabel 5.6. Perbandingan Hasil Perhitungan *presicion(p)* dan *recall(r)* pada  $Q_3$

NO	Subtopik	Surah	Relevan?	Recall	Precision
1	Bukti-bukti kekuasaan Tuhan (dok.7)	Al-Baqarah 28 – 29	Ya	1/16= 0,06	1/1= 1,00

2	Pengingkaran terhadap kekuasaan Allah hanyalah Karena Kesombongan Semata (dok.589)	Al-Mu'min 56 – 59	Ya	2/16= 0,13	2/2= 1,00
3	Perintah menyembah Tuhan Yang Maha Esa (dok.4)	Al-Baqarah 21 – 22	Tidak	2/16= 0,13	2/3= 0,67
4	Hanya Kepada Allah Orang-Orang Mukmin Bertawakal (dok.574)	Az-Zumar 38 – 40	Ya	3/16= 0,19	3/4= 0,75
5	Ayat Kursi (dok.49)	Al-Baqarah 255 – 255	Tidak	3/16= 0,19	3/5= 0,60
6	Siapakah Tuhan Semesta Alam dan Bagaimana Bermohon Kepada-Nya? (dok.160)	Al-A'raf 54 – 56	Ya	4/16= 0,25	4/6= 0,67
7	Anjuran Memperhatikan Alam Semesta (dok.768)	Al-Ghaasyiyah 17 – 26	Ya	5/16= 0,31	5/7= 0,71
8	Keharusan bertakwa (dok.111)	An-Nisaa 131 – 134	Ya	6/16= 0,38	6/8= 0,75
9	Anjuran Allah Agar Mengadakan Perlawatan Di Muka Bumi Untuk Membuktikan Kekuasaan Allah (dok.538)	Faathir 44 – 45	Tidak	6/16= 0,38	6/9= 0,67
10	Allah Adalah Pencipta, Penguasa, Dan Pemberi Rahmat (dok.527)	Faathir 1 – 4	Ya	7/16= 0,44	7/10= 0,70
11	Tindakan-tindakan menghalangi beribadah (dok.23)	Al-Baqarah 114 – 118	Ya	8/16= 0,50	8/11= 0,73
12	Perbedaan Keadaan Orang yang Bertakwa dengan Orang Yang Mendustakan Hari Kiamat (dok.581)	Az-Zumar 60 – 63	Ya	9/16= 0,56	9/12= 0,75
13	Dalil-Dalil tentang keesaan Allah (dok.142)	Al-An'am 1 – 3	Ya	10/16= 0,63	10/13= 0,77
14	Al-Quranul Karim Menjamin Suksesnya Orang-Orang Yang Beriman (dok.491)	Luqman 1 – 11	Ya	11/16= 0,69	11/14= 0,79
15	Al-Quran Diturunkan Sebagai Peringatan Bagi Manusia (dok.344)	Thaahaa 1 – 8	Ya	12/16= 0,75	12/15= 0,80
16	Kekuasaan Allah dan Keharmonisan Ciptaan-Nya (dok.410)	Al-Furqaan 1 – 3	Ya	13/16= 0,81	13/16= 0,81

17	Beberapa Nikmat Allah yang Dilimpahkan kepada Hamba-hamba-Nya (dok.271)	Ibrahim : 32 – 34	Ya	14/16= 0,88	14/17= 0,82
18	Doa Yusuf a.s. (dok.248)	Yusuf 101 – 101	Tidak	14/16= 0,88	14/18= 0,78
19	Wahyu Allah kepada Muhammad SAW. Sama Dengan Wahyu-Nya kepada Rasuk-Rasul Sebelumnya (dok.601)	Asy-Syuura 1 – 4	Ya	15/16= 0,94	15/19= 0,79
20	Beberapa Al-Asma-ul Husna (dok.688)	Al-Hasyr 22 – 24	Ya	16/16= 1,00	16/20= 0,80

Tampilan hasil pencarian pada sistem untuk *query* “pohon zaitun menghasilkan yang” dapat dilihat pada Gambar 5.20.



**Gambar 5.20.** Tampilan hasil pencarian untuk *query* “pohon zaitun yang menghasilkan minyak”

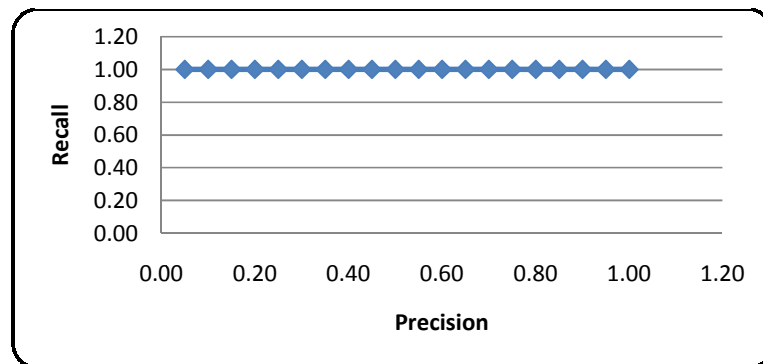
**Tabel 5.7. Perbandingan Hasil Perhitungan *presicion(p)* dan *recall(r)* pada  $Q_4$**

NO	Subtopik	Surah	Relevan?	Recall	Precision
1	Langit dan Segala Sesuatu Yang Ada di Bumi, Diciptakan Oleh Allah Untuk Kelanjutan Kehidupan Manusia Yang Wajib Disyukuri (dok.384)	Al-Mu'minuun 17 – 22	Ya	1/2= 0,50	1/1= 1,00
2	Pencerminan Ayat-Ayat Al-Quran Sebagai Nur Ilahi pada Langit dan Bumi (dok.402)	An-Nuur 34 – 35	Ya	2/2= 1,00	2/2= 1,00
3	Keingkaran Kaum Saba' Terhadap Nikmat Allah dan Akibatnya (dok.522)	Saba' 15 - 21	Tidak	2/2= 1,00	2/3= 0,67
4	Kelahiran Nabi 'Isa a.s. (dok.337)	Maryam 23 – 26	Tidak	2/2= 1,00	2/4= 0,50
5	Perintah menyembah Tuhan Yang Maha Esa (dok.4)	Al-Baqarah 21 – 22	Tidak	2/2= 1,00	2/5= 0,40
6	Perbuatan Jelek dan Amal Yang Saleh Akan Mendapat Pembalasan Yang Setimpal (dok.622)	Ad-Dukhaan 43 – 59	Tidak	2/2= 1,00	2/6= 0,33
7	Perumpamaan Tentang Kebenaran dan Kebatilan (dok.268)	Ibrahim 24 – 27	Tidak	2/2= 1,00	2/7= 0,29
8	Buah Pohon Zaqqum Makanan Ahli Neraka (dok.553)	Ash-Shaaffat 63 – 70	Tidak	2/2= 1,00	2/8= 0,25
9	Kejadian-Kejadian dalam Alam Membuktikan Kebenaran Adanya Hari Berbangkit (dok.646)	Qaaf 6 – 11	Tidak	2/2= 1,00	2/9= 0,22
10	Azab atas Golongan Kiri dan Tempelak untuk Mereka (dok.675)	Al-Waaqi'ah 41 – 74	Tidak	2/2= 1,00	2/10= 0,20
11	Keadaan Penghuni-Penghuni Neraka Dan Penghuni-Penghuni Surga (dok.767)	Al-Ghaasyiyah 1 - 16	Tidak	2/2= 1,00	2/11= 0,18
12	Pengusiran Bani Nadhir dari Madinah (dok.684)	Al-Hasyr 1 - 5	Tidak	2/2= 1,00	2/12= 0,13
13	Tukang-Tukang Sihir Fir'aun Menjadi Orang-Orang Yang Beriman (dok.351)	Thaahaa 70 – 76	Tidak	2/2= 1,00	2/13= 0,15
14	Orang-Orang yang Mendustakan Kebenaran Pasti Binasa (dok.713)	Al-Haaqqah 1 – 12	Tidak	2/2= 1,00	2/14= 0,14
15	Balasan kepada Golongan Kanan (dok.674)	Al-Waaqi'ah 27 – 40	Tidak	2/2= 1,00	2/15= 0,13

16	Tamsil Kehidupan Dunia dan Orang-Orang yang Tertipu Padanya (dok.320)	Al-Kahfi 32 – 46	Tidak	2/2= 1,00	2/16= 0,13
17	Kisah Nabi Adam A.S. Dan Pembangkangan Iblis (dok.360)	Thaahaa 115 – 127	Tidak	2/2= 1,00	2/17= 0,12
18	Keadaan Orang-Orang Mukmin Di Syurga (dok.552)	Ash-Shaaffat 40 – 62	Tidak	2/2= 1,00	2/18= 0,11
19	Derajat Seseorang Seimbang Dengan Amalnya (dok.153)	Al-An'am 130 – 135	Tidak	2/2= 1,00	2/19= 0,11
20	Kaum yang Ingkar Pasti Dapat Hukuman (dok.310)	Al-Israa' 58 – 65	Tidak	2/2= 1,00	2/20= 0,10

Untuk mengetahui unjuk kerja sistem yang dibangun, profil hubungan *recall-precision* dibuat dalam bentuk data interpolasi yang digambarkan secara grafik. Data interpolasi *recall* dan *precision* pada *query* 1 (menunaikan zakat), *query* 2 (berpuasa di bulan ramadhan), *query* 3 (Allah menciptakan langit dan bumi), dan *query* 4 (pohon zaitun yang menghasilkan minyak) dapat dilihat pada Gambar 5.21 sampai Gambar 5.24.

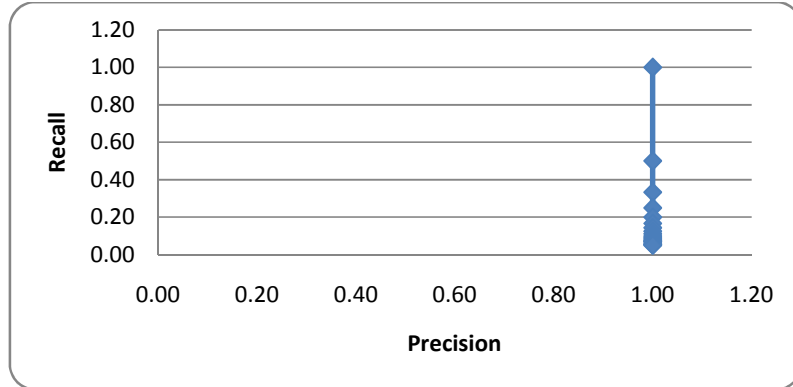
Data interpolasi *recall* dan *precision* pada Tabel 5.4 dapat digambarkan secara grafik garis sebagai berikut.



**Gambar 5.21. Grafik interpolasi *recall* dan *precision* pada *query* “kisah musa”**

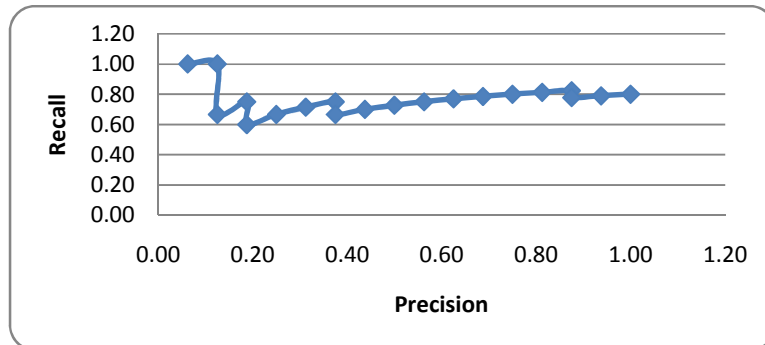


Data interpolasi *recall* dan *precision* pada Tabel 5.5 dapat digambarkan secara grafik garis sebagai berikut.



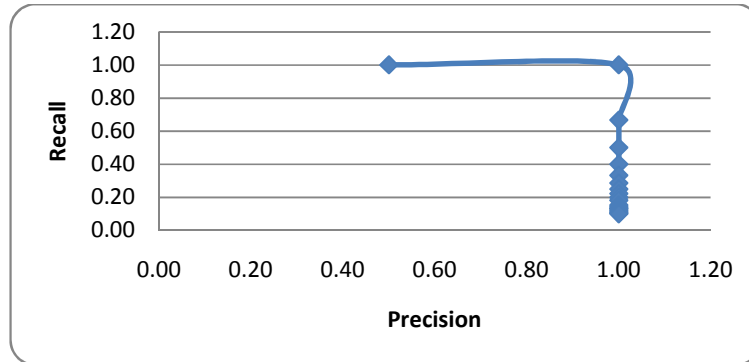
**5.22. Grafik interpolasi *recall* dan *precision* pada query “berpuasa di bulan ramadhan”**

Data interpolasi *recall* dan *precision* pada Tabel 5.6 dapat digambarkan secara grafik garis sebagai berikut.



**5.23. Grafik interpolasi *recall* dan *precision* pada query “Allah menciptakan langit dan bumi”**

Data interpolasi *recall* dan *precision* pada Tabel 5.7 dapat digambarkan secara grafik garis sebagai berikut.



#### 5.24. Grafik interpolasi *recall* dan *precision* pada *query* “pohon zaitun yang menghasilkan minyak”

Gambar 5.21 menunjukkan bahwa dokumen-dokumen yang dihasilkan dari pencarian nilai *recall* dan *precision* untuk *query* “menunaikan zakat” merupakan dokumen yang relevan di urutan teratas, direpresentasikan dengan gambar grafik yang mendatar/horizontal. Ini menandakan sistem memberikan hasil yang sangat baik, ditunjukkan dengan nilai rata-rata recall adalah 1. Gambar 5.22 untuk grafik interpolasi *query* “berpuasa di bulan ramadhan” dapat dikatakan dari segi presisi, sistem telah memberikan hasil yang relevan pada urutan pertama, yaitu dengan subtopik “puasa”, dan tidak ada lagi topik mengenai puasa ramadhan di ayat-ayat lainnya. Gambar 5.23 untuk grafik interpolasi *query* “Allah menciptakan langit dan bumi”, direpresentasikan dengan grafik yang menampilkan dokumen relevan di urutan teratas. Beberapa penurunan *precision* menggambarkan adanya dokumen yang tidak relevan terselip di antara dokumen yang dihasilkan mesin pencari. Sedangkan untuk grafik interpolasi *query* “pohon zaitun yang menghasilkan minyak” pada Gambar 5.24 direpresentasikan dengan garis lurus di awal grafik yang menandakan kemampuan sistem menemukan dokumen relevan, namun penurunan *precision* karena memang tidak adalagi dokumen yang membahas mengenai pohon zaitun di ayat-ayat lainnya.

Berikut pengujian yang tidak menggunakan *presicion* yang tidak terinterpolasi. Pada pengujian akan terlihat jumlah dokumen yang dikembalikan

baik yang relevan maupun tidak dan jumlah dokumen yang tidak dikembalikan baik yang relevan maupun tidak, dimana data tersebut akan digunakan untuk menghitung nilai *precision* dan *recall* berdasarkan masing-masing *query*.

**Tabel 5.8. Hasil Pengujian *precision(P)* dan *recall(R)* pada Q<sub>1</sub> (kisah musa)**

Keterangan	Relevan	Tidak relevan
Terambil	78 ( <i>tp</i> )	12 ( <i>fp</i> )
Tidak terambil	0 ( <i>fn</i> )	0 ( <i>tn</i> )

Berdasarkan Tabel 5.8 di atas dapat dicari nilai *precision* dan *recall* dengan rumus di bawah ini:

$$P = tp / (tp + fp)$$

$$= \frac{78}{78 + 12} = \frac{78}{90} = 0.87$$

$$R = tp / (tp + fn)$$

$$= \frac{78}{78 + 0} = \frac{78}{78} = 1$$

Dari hasil di atas didapat nilai *precision* adalah 0.87 dan nilai dari *recall* adalah 1.

**Tabel 5.9. Hasil Pengujian *precision(P)* dan *recall(R)* pada Q<sub>2</sub> (berpuasa di bulan ramadhan)**

Keterangan	Relevan	Tidak relevan
Terambil	42( <i>tp</i> )	41 ( <i>fp</i> )
Tidak terambil	0 ( <i>fn</i> )	0 ( <i>tn</i> )

Berdasarkan Tabel 5.9 di atas dapat dicari nilai *precision* dan *recall* dengan rumus di bawah ini:

$$P = tp / (tp + fp)$$

$$= \frac{42}{42 + 41} = \frac{42}{83} = 0.51$$

$$R = tp / (tp + fn)$$

$$= \frac{42}{42 + 0} = \frac{42}{42} = 1$$

Dari hasil di atas didapat nilai *precision* adalah 0.51 dan nilai dari *recall* adalah 1.

**Tabel 5.10. Hasil Pengujian *presicion(P)* dan *recall(R)* pada Q<sub>3</sub> ( Allah menciptakan langit dan bumi)**

Keterangan	Relevan	Tidak relevan
Terambil	668( <i>tp</i> )	603 ( <i>fp</i> )
Tidak terambil	0 ( <i>fn</i> )	0 ( <i>tn</i> )

Berdasarkan Tabel 5.10 di atas dapat dicari nilai *precision* dan *recall* dengan rumus di bawah ini:

$$P = tp / (tp + fp)$$

$$= \frac{668}{668 + 603} = \frac{668}{1271} = 0.53$$

$$R = tp / ( tp + fn)$$

$$= \frac{668}{668 + 0} = \frac{668}{668} = 1$$

Dari hasil di atas didapat nilai *precision* adalah 0.53 dan nilai dari *recall* adalah 1.

**Tabel 5.11. Hasil Pengujian *presicion(P)* dan *recall(R)* pada Q<sub>4</sub> (Pohon zaitun yang menghasilkan minyak)**

Keterangan	Relevan	Tidak relevan
Terambil	43 ( <i>tp</i> )	41 ( <i>fp</i> )
Tidak terambil	0 ( <i>fn</i> )	0 ( <i>tn</i> )

Berdasarkan Tabel 5.11 di atas dapat dicari nilai *precision* dan *recall* dengan rumus di bawah ini:

$$P = tp / (tp + fp)$$

$$= \frac{43}{43 + 41} = \frac{43}{84} = 0.51$$

$$R = tp / ( tp + fn)$$

$$= \frac{43}{43 + 0} = \frac{43}{43} = 1$$

Dari hasil di atas didapat nilai *precision* adalah 0.51 dan nilai dari *recall* adalah 1.

Berdasarkan perhitungan pengujian yang tidak menggunakan *presicion* yang tidak terinterpolasi di atas, maka hasil perhitungan pengujian *precision* dan *recall* untuk seluruh *query* akan dilampirkan di dalam Tabel 5.12.

**Tabel 5.12. Persentasi Seluruh Kemampuan Sistem pada *Query***

<i>Query</i>	<i>Precision</i>	<i>Recall</i>
$Q_1$ (Kisah musa)	0.87	1
$Q_2$ (Berpuasa di bulan ramadhan)	0.51	1
$Q_3$ (Allah menciptakan langit dan bumi)	0.53	1
$Q_4$ (Pohon zaitun yang menghasilkan minyak)	0.51	1
$Q_5$ (Menunaikan zakat)	0.82	1
$Q_6$ (Tidak ada keraguan dalam Al-Qur'an)	0.62	1
$Q_7$ (Berbuat riba)	1	1
$Q_8$ (Arah kiblat saat sholat)	0.52	1
$Q_9$ (Perceraian)	0.79	1
$Q_{10}$ (Nazar)	1	1
$Q_{11}$ (Perkawinan)	1	1
$Q_{12}$ (Berlaku Adil)	0.66	1

Dari hasil seluruh pengujian maka dapat dihasilkan kesimpulan pada rata-rata berikut ini:

Rata-rata *Precision* =

$$= \frac{0.87 + 0.51 + 0.53 + 0.51 + 0.82 + 0.62 + 1 + 0.52 + 0.79 + 1 + 1 + 0.66}{12} = \frac{9.84}{12}$$

$$= 0.82$$

Rata-rata *Recall* =

$$= \frac{1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1}{12} = \frac{12}{12}$$

$$= 1$$

### 5.2.3. Kesimpulan Pengujian Unjuk Kerja Sistem

Hasil pengujian yang diperoleh dari sistem pencarian ayat Al-Qur'an dengan menggunakan metode *tf-idf* menggunakan model NN (*Neural Network*) menggunakan 4 *query* pengujian dengan menggunakan rumus persamaan 2.7 dan 2.8 untuk menghitung nilai *recall* dan *precision*, relevan atau tidaknya dokumen dievaluasi oleh pengguna berdasarkan apakah kata-kata yang ada di dalam *query* terdapat di dalam dokumen. Dokumen subtopik hasil pencarian yang ditampilkan merupakan dokumen yang diurutkan berdasarkan nilai kesamaan/*similarity* antara *term* yang muncul pada *query* dan dokumen teks terjemahan Al-Qur'an. Namun, nilai relevansi dokumen terhadap *query* diserahkan kepada pengguna karena merupakan penilaian subyektif. Diharapkan pengguna membuka dokumen hasil pencarian dimulai dari urutan teratas hingga dokumen hasil pencarian terakhir untuk mendapatkan dokumen yang relevan sesuai dengan *query* yang dimasukkan. Sedangkan 12 *query* pengujian yang telah dilakukan berdasarkan tabel *query* yang terdapat pada tabel 5.3 dengan menggunakan rumus persamaan 2.9 dan 2.10 untuk menghitung nilai *recall* dan *precision* yang tidak terinterpolasi, dapat diambil kesimpulan bahwa sistem temu kembali informasi ini memiliki korelevansi tinggi, karena nilai rata-rata *precision* sebesar 0.74 dan *recall* sebesar 1.00 berada pada rentang kesesuaian tinggi yaitu pada rentang 0,67 – 1,00.

## **BAB VI**

### **PENUTUP**

Pada bab ini akan diuraikan beberapa kesimpulan dari hasil yang didapatkan selama penelitian dan saran yang dapat digunakan pada penelitian selanjutnya.

#### **6.1. Kesimpulan**

Setelah menyelesaikan tahapan-tahapan penelitian dalam *Information Retrieval System*, dapat diambil beberapa kesimpulan, yaitu:

1. Koleksi dokumen (korpus) terjemahan Al-Qur'an dan korpus gambar ayat (tulisan Arab) yang digunakan sebagai bahan dasar pembangunan sistem telah direorganisasi dengan membentuk sebuah korpus baru yaitu korpus subtopik Al-Qur'an untuk digunakan pada tahapan membangun *indexing*.
2. Metode ruang vektor yang diterapkan dalam sistem pencarian ayat Al-Qur'an mampu memberikan hasil pencarian berupa daftar dokumen subtopik yang terangkai berdasarkan perhitungan relevansi antara dokumen teks terjemahan ayat dan *query* yang dimasukkan pengguna.
3. Berdasarkan penelitian yang telah dilakukan dengan menggunakan metode ruang vektor pada kasus pencarian ayat Al-Qur'an memberikan hasil yang baik yaitu dengan rata-rata *precision* 74% dan *recall* 100% dari 12 kali percobaan terhadap *query*. Setiap pengguna memiliki sudut pandang yang berbeda dalam menilai kerelevanan dokumen terhadap *query* (subyektif).
4. Panjang *query* yang dimasukkan untuk pencarian ayat Al-Qur'an mempengaruhi kualitas kerelevanan hasil pencarian. Semakin panjang *query* yang dimasukkan akan meningkatkan nilai *precision* sistem dan menurunkan nilai *recall*.

## 6.2. Saran

Berdasarkan penelitian yang telah dilakukan, saran-saran yang dapat dilakukan untuk perbaikan dan pengembangan *information retrieval system* mendatang, yaitu:

1. Penulis menyarankan untuk perlu dilakukan perluasan *query* yang merupakan suatu proses menambahkan sejumlah kata dari dokumen yang relevan terhadap *query* awal, agar *query* menjadi lebih spesifik dan mendapatkan hasil yang lebih tepat sasaran. Metode yang disarankan adalah dengan menggunakan metode *thesaurus* yang melakukan ekspansi *query* dengan menggunakan sinonim dalam membentuk basis data kata benda. *Thesaurus* menyediakan informasi berdasarkan sinonim dan kata-kata yang saling berhubungan serta frase-frase.
2. Implementasi perangkat lunak ke depan sebaiknya lebih dikembangkan, yaitu dalam hal antar muka yang lebih lengkap fiturnya dan lebih interaktif, juga dalam kinerja sistem secara keseluruhan.



## DAFTAR PUSTAKA

- Agusta, Lady. 2009. “Perbandingan Algoritma Stemming Porter Dengan Algoritma Nazief & Adriani Untuk Stemming Dokumen Teks Bahasa Indonesia”. Konferensi Nasional Sistem dan Informatika 2009, Bali, November 14, 2009.
- Arifin, Agus Zainal. *Penggunaan Digital Tree Hibrida pada Aplikasi Information Retrieval untuk Dokumen Berita*, Proseding Seminar Nasional Sains dan Teknologi 2002, Lembaga Penelitian. ITS, 31 Juli 2002.
- Cios, Krzyztof J. Etc. *Data Mining A Knowledge Discovery Approach*. Spinger, 2007.
- Darmawan, Heru Adi, Dkk. *Rancang Bangun Aplikasi Search Engine Tafsir Al-Qur'an Menggunakan Teknik Text Mining Dengan Algoritma VSM (Vector Space Model)*. Program Studi Sistem Informasi, STIKOM Surabaya.
- Garcia. E. Dr. *Cosine Similarity Term Weight Tutorial*. 2006. [online] available <http://www.miislita.com/information-retrieval-tutorial/indexing.html>, diakses 28 maret 2012.
- Harinaldi. *Prinsip-Prinsip Statistik untuk Teknik dan Sains*. Jakarta. Erlangga, 2005.
- Mandala, Rila dan Hendra Setiawan. *Peningkatan Performansi Sistem Temu Kembali Informasi dengan Perluasan Query Secara Otomatis*, Laboratorium Keahlian Informatika teori Department Teknik Informatika, Institut Teknologi Bandung, 2006.
- Manning, Christopher D, Ragnavan Prabhakar, Schütze, Hinrich. *Introduction to Information Retrieval*. England: Cambridge University Press, 2009.
- Murad, AzmiMA., Martin, Trevor. 2007. *Word Similarity for Document Grouping using Soft Computing*. IJCSNS International Journal of Computer Science and Network Security, Vol.7 No.8, August 2007, pp. 20- 27.
- Mustaqim, Taufik. *Analisa dan Implementasi Kombinasi Model Ruang Vektor dan Model Probabilistik Pada Sistem Temu Kembali Informasi*. Institut Teknologi Bandung, 2008.
- Ramadhany, Taufik. *Analisa Dan Implementasi Penerapan Algoritma Genetika Pada Information Retrieval*, IT Telkom bandung, 2009.
- Sinclair, John. *a Guide to Good Practice Corpus and Text — Basic Principles*. Tuscan Word Centre, 2004.
- Trunojoyo, Husni. *Sistem Temu Balik Informasi (sebuah contoh implementasi)*, 2010.