

SKRIPSI

PENERAPAN TOKENISASI KALIMAT DAN METODE TF (TERM FREQUENCY) PADA PERINGKAS TEKS OTOMATIS

ARTIKEL BERITA BERBAHASA INDONESIA

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar

Sarjana Teknik Informatika



Disusun Oleh:

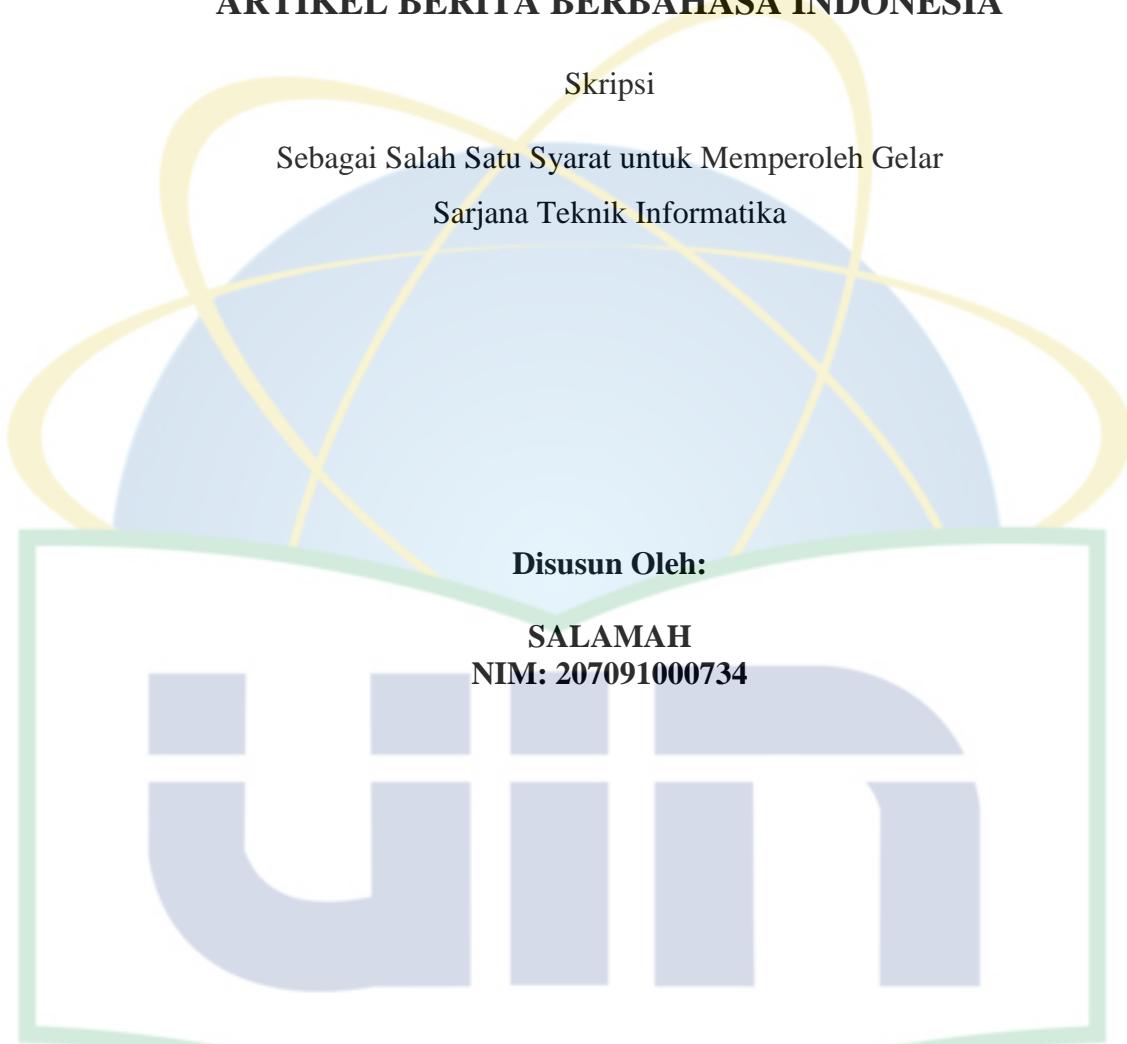
SALAMAH
NIM: 207091000734

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH
JAKARTA
2014 M/1435 H

SKRIPSI

PENERAPAN TOKENISASI KALIMAT DAN METODE TF (TERM FREQUENCY) PADA PERINGKAS TEKS OTOMATIS

ARTIKEL BERITA BERBAHASA INDONESIA



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH
JAKARTA
2014 M/1435 H

LEMBAR PENGESAHAN SKRIPSI

JUDUL

**PENERAPAN TOKENISASI KALIMAT DAN METODE TF
(TERM FREQUENCY) PADA PERINGKAS TEKS OTOMATIS ARTIKEL
BERITA BERBAHASA INDONESIA**

Disusun Oleh :

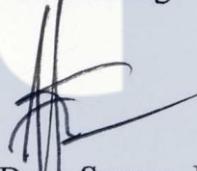
SALAMAH

NIM : 207091000734

Disetujui Dan Disahkan Sebagai Salah Satu Syarat Menyelesaikan Studi Pada
Program Strata Satu Program Studi Teknik Informatika
Universitas Islam Negeri Jakarta

Jakarta, Juli 2014

Pembimbing I



Hendra Bayu Suseno, M.Kom
NIP. 19620714 198903 1 004

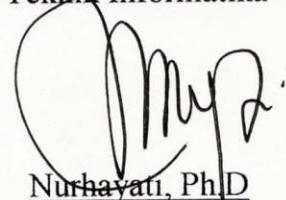
Pembimbing II



Victor Amrizal, M.Kom
NIP. 19740624 200710 1 001

Mengetahui,

Ketua Program Studi
Teknik Informatika


Nurhayati, Ph.D

NIP. 19690316 199903 2 002

PENGESAHAN UJIAN

Skripsi yang berjudul “**Penerapan Tokenisasi Kalimat dan Metode Tf (Term Frequency) pada Peringkas Teks Otomatis Artikel Berbahasa Indonesia**” yang ditulis oleh **Salamah (207091000734)**, telah diuji dan dinyatakan lulus dalam Sidang Munaqosah Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta, pada hari Senin, 14 Juli 2014. Skripsi ini telah diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) Program Studi Teknik Informatika.

Jakarta, Agustus 2014

Menyetujui,

Penguji I



Andrew Fiade, M.Kom
NIP. 19820811 200912 1 004

Penguji II



Anif Hanifa Setyaningrum, M.Si
NIDN. 410 116 402

Pembimbing I



Hendra Bayu Suseno, M.Kom
NIP. 19620714 198903 1 004

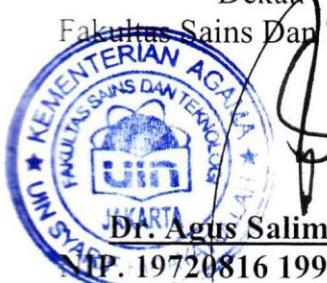
Pembimbing II



Victor Amrizal, M.Kom
NIP. 19740624 200710 1 001

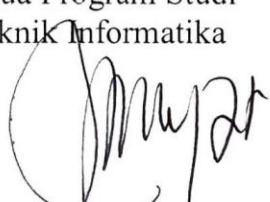
Mengetahui,

Dekan



Fakultas Sains Dan Teknologi
Dr. Agus Salim, M.Si
NIP. 19720816 199903 1 003

Ketua Program Studi
Teknik Informatika



Nurhayati, Ph.D
NIP. 19690316 199903 2 002

PERNYATAAN

DENGAN INI SAYA MENYATAKAN BAHWA SKRIPSI INI BENAR-BENAR HASIL KARYA SENDIRI YANG BELUM PERNAH DIAJUKAN SEBAGAI SKRIPSI ATAU KARYA ILMIAH PADA PERGURUAN TINGGI MANAPUN.



Jakarta, Juli 2014

Salamah

ABSTRAK

SALAMAH (207091000734). Penerapan Tokenisasi Kalimat dan Metode Tf (*Term Frequency*) Pada Peringkas Teks Otomatis Artikel Berbahasa Indonesia. Dibawah Bimbingan Bapak **HENDRA BAYU SUSENO, M.Kom** dan Bapak **VICTOR AMRIZAL, M.Kom**

Peringkas teks otomatis (*Automatic Text Summarization*) adalah pembuatan versi yang lebih singkat dari sebuah teks dengan memanfaatkan aplikasi yang dijalankan pada komputer. Peringkasan teks otomatis berguna untuk membantu manusia dalam mendapatkan ringkasan dari suatu bacaan tanpa harus membaca keseluruhan isi bacaan. Tokenisasi kalimat dan TF (*Term Frequency*) adalah Metode yang digunakan dalam pembuatan aplikasi peringkas dokumen ini. Tokenisasi kalimat adalah sebuah proses pemecahan kalimat menjadi kata yang nantinya akan diproses pada tahapan pembobotan. Sedangkan metode *Term Frequency* untuk menghitung frekuensi kemunculan kata pada dokumen dan seluruh koleksi dokumen yang ada.

Kata kunci : Peringkas Otomatis, *Term Frequency* (TF), *Inverse Document Frequency* (IDF), Algoritma *Stemming* Porter.

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalaamu'alaikum warahmatullaahi wabarakaaatuh.

Segala puji senantiasa tercurah kepada Allah SWT, Dzat Maha Sempurna yang lautan ilmu dan ampunan-Nya tidak akan pernah habis. Shalawat serta salam bagi baginda rasulullah Muhammad SAW beserta keluarganya, para sahabatnya, dan para pengikutnya hingga akhir zaman.

Skripsi ini disusun sebagai tugas akhir dalam masa belajar di Universitas Islam Negeri Syarif Hidayatullah Jakarta, serta menjadi syarat kelulusan program Strata Satu (S1), adapun skripsi ini berjudul "**Penerapan Tokenisasi Kalimat dan Metode Tf (Term Frequency) pada Peringkas Teks Otomatis Artikel Berita Berbahasa Indonesia**".

Penulis berharap penulisan dan pembuatan skripsi ini dapat berguna bagi banyak orang khususnya dalam bidang pendidikan, semoga skripsi ini tidak hanya dilihat dari sisi kekurangannya karena tidak ada yang sempurna di dunia ini, kekurangan yang banyak terdapat dalam skripsi ini mudah-mudahan bisa menjadi acuan untuk membuat sesuatu yang lebih baik.

Tidak lupa penulis ingin mengucapkan banyak terimakasih kepada semua pihak yang membantu hingga terselesaikannya penulisan skripsi ini, diantaranya :

1. Prof. Dr. Komarudin Hidayat selaku Rektor Universitas Islam Negeri Syarif Hidayatullah Jakarta. Dr. Agus Salim, M.Si., selaku Dekan

Fakultas Sains dan Teknologi, Universitas Islam Negeri Syarif Hidayatullah Jakarta.

2. Ibu Nurhayati, Ph.D dan Bapak Hendra Bayu Suseno, M.kom, selaku Ketua Program Studi dan Sekretaris Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
3. Bapak Hendra Bayu Suseno, M.kom dan Bapak Victor Amrizal, M.Kom, selaku dosen pembimbing yang selalu memberikan bimbingan dengan sabar kepada penulis dalam penulisan skripsi ini.
4. Ibu Anif Hanifa Setyaningrum, M.Si dan Bapak Andrew Fiade, M.Kom, selaku dosen penguji yang memudahkan penulis pada saat pengujian dan perevisian skripsi hingga rampung.
5. Seluruh dosen serta karyawan Universitas Islam Negeri Syarif Hidayatullah Jakarta.
6. Ayahanda dan Ibundaku tercinta yang selalu mendo'akan penulis dalam kebaikan dan kesuksesan penulis, serta dukungan baik secara materil maupun moril. Kepada kedua kakakku tersayang Waziruddin Mughny dan Mutia Tsani Mughny yang selalu memberikan motivasi dan semangat kepada penulis. Juga keponakanku tersayang Icam, Ara, Fiza Senyum dan tawa mereka adalah spirit bagi penulis.
7. Kepada Abang uwiie tersayang yang selalu memberikan dukungan baik materil dan moril "Haadza Min Fadli Rabbi".
8. Sahabat-sahabatku Georgeous Girl (Khadijah Efrison, Mahasika Dian, Eka Julianti, Fitriyanti Pondok, Merry Tyas Utami, Indah Putri Utami,

Suci Putri Herizona dan Julianti) terima kasih banyak untuk dukungan dan Do'a kalian. Semoga *ukhuwah* ini bisa terjalin selama-lamanya.

9. Keluarga kecil dikosan Ibu Endang, Nisa, Cimud, Ifa, Naya terima kasih atas keceriaan dan semangatnya.
10. Laptop tercinta.

Penulis menyadari skripsi ini masih banyak kekurangannya, oleh karena itu penulis mengharapkan kritik dan sarannya. Akhir kata, dengan segala kerendahan hati penulis mengucapkan terima kasih kepada semua pihak yang telah membantu penyelesaian skripsi ini. Penulis juga berharap semoga skripsi ini bermanfaat bagi pembaca umumnya dan penulis khususnya.

Wassalaamu'alaikum warahmatullaahi wabarakatuh.

Jakarta, Agustus 2014

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
LEMBAR PERNYATAAN	iv
ABSTRAK	v
KATA PENGANTAR.....	vi
DAFTAR ISI	ix
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
DAFTAR LAMPIRAN	xvi
BAB I PENDAHULUAN	
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
1.6 Metode Penelitian	5
1.6.1 Metode Pengumpulan Data	5
1.6.2 Metode Pengembangan Sistem	5
1.7 Sistematika Penulisan	6

BAB II LANDASAN TEORI

2.1 Temu Kembali Informasi (<i>Information Retrieval</i>)	8
2.2 Peringkas Dokumen Otomatis (<i>Automatic Text Summarization</i>)	9
2.3 Penelitian Peringkas Otomatis	12
2.3.1 Metode Signifikasi Kata Dan Kalimat	13
2.3.2 Metode Pemilihan Kalimat	14
2.3.3 Metode Statis Dan <i>Query Biased</i>	14
2.4 Pemrosesan Kalimat	15
2.4.1. <i>Tokenizing</i>	16
2.4.2 Penghilangan <i>Stopword</i>	16
2.4.3 Pembobotan	17
2.4.3.1 Perencanaan Kebutuhan (<i>Requirements Planning</i>)	17
2.4.4 Evaluasi	18
2.5 Model Pengembangan Sistem <i>Rapid Application Development</i> (RAD)	18
2.5.1 Pengertian <i>Rapid Application Development</i> (RAD)	18
2.5.2 Tujuan RAD	20
2.5.3 Tahapan RAD (<i>Rapid Application Development</i>)	21
2.5.3.1 Perencanaan Kebutuhan (<i>Requirements Planning</i>)	22
2.5.3.2 Proses Desain (<i>Workshop Design</i>)	22
2.5.3.3 Implementasi (<i>Implementation System</i>)	23
2.5.4 Keuntungan RAD	24
2.5.5 Kondisi Penunjang Keberhasilan RAD	25

2.6 Model Perancangan Sistem	25
2.6.1 Bagan Alir (<i>Flowchart</i>)	25
2.6.2 <i>State Transition Diagram (STD)</i>	28
2.6.2.1 Pendekatan untuk Membuat STD	28
2.6.2.2 Notasi State Transition Diagram (STD)	29
2.7 MySQL	30
2.8 PHP	32
2.8.1 Sejarah PHP	32
2.8.2 Pengertian PHP	33
2.9 Storyboard	34
2.10 Pengujian	34
BAB III METODE PENELITIAN	
3.1 Metode Pengumpulan Data	36
3.1.1 Wawancara	36
3.1.2 Studi Pustaka	37
3.1.3 Studi Literatur	37
3.2 Metode Pengembangan Sistem	38
3.3.1 Fase <i>Requirements Planning</i>	39
3.3.2 Fase <i>Workshop Design</i>	40
3.3.3 Fase <i>Implementation</i>	41
3.4 Kerangka Berpikir Penelitian	41

BAB IV PEMBAHASAN

4.1 <i>Requirements Planning</i> (Perencanaan Kebutuhan)	43
4.1.1 Tujuan Pengembangan Sistem.....	43
4.1.2 Kebutuhan Sistem	43
4.1.3 Identifikasi Masalah.....	44
4.2 <i>Workshop Design</i> (Proses Desain).....	45
4.2.1 Perancangan Sistem Peringkas	45
4.2.1.1 <i>Tokenizing</i>	46
4.2.1.2 Pehilangan <i>Stopword</i>	48
4.2.1.3 Pembobotan	49
4.2.2 Desain Sistem	52
4.2.2.1 <i>State Transition Diagram</i> (STD).....	52
4.2.2.2 Desain Basis Data.....	53
4.2.2.3 Desain Antarmuka.....	54
4.2.3 <i>Coding</i> (Pengkodean)	54
4.3 Implementasi.....	57
4.3.1 Pengujian Sistem.....	57
4.3.1.1 <i>Black Box Testing</i>	57
4.3.1.2 Evaluasi (Kuisisioner)	59

BAB V PENUTUP

5.1 Kesimpulan	61
5.2 Saran	61

DAFTAR PUSTAKA**63**



DAFTAR GAMBAR

Gambar 2.1 Tahapan <i>Rapid Application Development (RAD)</i>	21
Gambar 2.2 Notasi <i>State Transition Diagram (STD)</i>	30
Gambar 2.3 Kondisi dan Aksi <i>State Transition Diagram (STD)</i>	30
Gambar 2.4 Tampilan MySQL	32
Gambar 2.5 Contoh <i>Storyboard</i>	34
Gambar 2.6 Skema Pengujian <i>Black-Box</i>	35
Gambar 3.1 Fase-Fase Model RAD	39
Gambar 3.2 Kerangka Berfikir Penelitian	42
Gambar 4.1 <i>Flowchart</i> Penyusunan Ringkasan	45
Gambar 4.2 <i>Flowchart</i> Pembobotan Kata	50
Gambar 4.3 STD (<i>State Transition Diagram</i>) halaman Utama Peringkas	52
Gambar 4.4 Tabel kumpulan <i>Stoplist</i>	54
Gambar 4.5 Desain Antarmuka Pengguna	54
Gambar 4.6 Tampilan Hasil Ringkasan	58
Gambar 4.7 Grafik Tingkat Pemahaman Pengguna.....	60

DAFTAR TABEL

Tabel 2.1 Simbol-Simbol <i>Flowchart</i>	26
Tabel 3.1 Studi Literatur	37
Tabel 4.1 Contoh <i>Tokenizing</i>	46
Tabel 4.2 Contoh Penghilangan <i>Stopword</i>	48
Tabel 4.3 Contoh Artikel yang diinputkan	50
Tabel 4.4 Proses Tokenisasi, Penghilangan Stopword,dan Pembobotan.....	51
Tabel 4.5 Uji Coba Modul Peringkas Teks Otomatis	58

DAFTAR LAMPIRAN

Lampiran 1 Surat Keputusan Pembimbing	A
Lampiran 2 Wawancara	B
Lampiran 3 Daftar Kata <i>Stopword</i>	C
Lampiran 4 Kuisioner Penelitian	D
Lampiran 5 Source Code Program	E



BAB I

PENDAHULUAN

1.1. Latar Belakang

Sejak diperkenalkannya internet, dokumen digital dan informasi yang tersedia secara *online* meningkat dengan pesat jumlahnya. Jumlah surat kabar dan majalah yang tersedia di internet bertambah banyak. Dokumen ilmiah seperti jurnal dan makalah penelitian dapat dengan mudah diakses melalui internet. Nurhasanah seorang editor penerbitan buku di Bandung mengaku selalu kebanjiran naskah yang dikirim para penulis melalui surat elektronik. Wanita 29 tahun ini mengeluhkan tak sempat membaca setiap naskah tuntas satu per satu. “bayangkan, untuk membaca naskah puluhan halaman dibutuhkan waktu minimal setengah hari, mengikhtisarkan atau membaca cepat adalah salah satu solusinya” ujarnya melalui Tempo Interaktif Jakarta Maret 2010. *Summary*, atau ringkasan didefinisikan sebagai penyajian bacaan dalam bentuk singkat dengan mengambil inti sari bacaan, mempetahankan urutan isi dan sudut pandang tanpa mengubah struktur wacana (Irman, 2008). Dengan ringkasan dapat diketahui informasi-informasi penting dari suatu dokumen tanpa perlu membaca dokumen secara keseluruhan.

Ringkasan biasanya dibuat oleh seorang abstraktor profesional, namun banyak dokumen digital yang tidak mempunyai ringkasan atau abstraknya sehingga jika dibuat secara manual oleh manusia akan memakan banyak waktu

dan biaya. Peringkas teks otomatis (*Automatic Text Summarization*) adalah pembuatan ringkasan dari sebuah teks secara otomatis dengan memanfaatkan aplikasi yang dijalankan pada komputer untuk menghasilkan informasi yang paling penting dari dokumen aslinya. Sistem diberi masukan berupa teks, kemudian melakukan peringkasan dan menghasilkan keluaran berupa teks yang lebih singkat dari teks asli.

Studi tentang peringkas otomatis pertama kali dilakukan oleh Luhn pada tahun 1958 berdasarkan tingkat pentingnya suatu kata dan kalimat (Luhn, 1958). Kemudian pada tahun 1995, Brandow menggunakan metode *term frequency* (tf) dan *inverse document frequency* (idf) dalam pembuatan peringkas otomatis (Brandow, 1995). Metode tf merupakan cara pembobotan kata dengan menghitung frekuensi kemunculan kata dalam dokumen. Penelitian yang dilakukan Brandow merupakan penelitian peringkas otomatis yang berhasil dikomersilkan dengan sistemnya yang bernama ANES (*Automatic News Extraction System*). Dengan mengacu pada solusi yang diberikan oleh Sistem Peringkas Teks Otomatis (*Automatic Text Summarization*), diharapkan dapat membantu pengguna dalam menghasilkan ikhtisar tanpa harus membaca seluruh isi dokumen.

Oleh karena itu, peneliti mengusulkan penelitian yang berjudul “**PENERAPAN TOKENISASI KALIMAT DAN METODE TF (TERM FREQUENCY) PADA PERINGKAS TEKS OTOMATIS ARTIKEL BERITA BERBAHASA INDONESIA**”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka didapat rumusan masalah sebagai berikut:

1. Bagaimana merancang aplikasi yang dapat membantu meringkas dokumen?
2. Bagaimana tahap *Preprocessing* kata dan penerapan metode *Term Frequency* (Tf) pada aplikasi peringkas dokumen berbahasa Indonesia?

1.3 Batasan Masalah

Dari rumusan masalah yang ada maka, dapat diberi batasan-batasan sehingga pembahasannya lebih terarah. Adapun batasan-batasan masalah yang mencadi acuan dalam penelitian skripsi ini adalah sebagai berikut:

1. Tidak dilakukan proses *Stemming* kata dalam aplikasi peringkas ini.
2. Tahap *Preprocessing* kata hanya *Tokenizing* dan penghapusan Stopword.
3. Metode pembobotan kalimat yang digunakan dalam penelitian peringkas otomatis ini hanya menggunakan metode Tf (*Term Frequency*).
4. Difokuskan penggunaan aplikasi peringkas teks otomatis hanya pada dokumen berbahasa Indonesia.
5. Hasil ringkasan berupa ekstrak, yaitu ringkasan yang seluruh hasilnya bedasarkan dari dokumen sumber.
6. Aplikasi peringkas dokumen ini tidak mendukung pencarian file untuk artikel masukan dan penyimpanan file untuk hasil ringkasan.
7. Dokumen masukan terbatas pada bidang ekonomi, olahraga, teknologi, kesehatan dan metropolitan.

8. Disarankan untuk dapat menghasilkan ringkasan dengan kompresi 10%, artikel yang dimasukkan memiliki jumlah kalimat minimal 10 kalimat dan untuk efisiensi waktu, dokumen masukan hanya artikel berita yang jumlah kalimatnya sekitar 200 kalimat.
9. Aplikasi tidak dapat bekerja sempurna pada artikel ilmiah.
10. Metode pengembangan sistem yang digunakan adalah RAD dengan 3 tahapan yaitu *Requirement Planning, Workshop Design, Implementation.*
11. Sistem yang dibangun menggunakan bahasa pemrograman PHP dan menggunakan database MySQL.
12. Perancangan sistem menggunakan bagan alir *Flowchart* dan *State Transition Diagram.*
13. Evaluasi peringkasan dilakukan hanya dengan menyebarkan kuisioner untuk mengetahui tingkat pemahaman pengguna.

1.4 Tujuan Penelitian

Adapun tujuan penelitian skripsi ini adalah merancang aplikasi peringkas dokumen otomatis (*Automatic Text Summarization*) dengan menggunakan metode Tf (*Term frequency*).

1.5 Manfaat Penelitian

Manfaat dari penelitian skripsi ini adalah:

1. Menambah wawasan dalam bidang ilmu pengetahuan tentang merancang sistem aplikasi peringkas dokumen.
2. Dapat menjadi salah satu referensi bagi penelitian berikutnya di bidang kajian pengembangan *Automatic Text Summarization*.

1.6 Metode Penelitian

1.6.1 Metode Pengumpulan Data

1. Wawancara

Mengumpulkan data dengan mewawancara orang yang terkait langsung dengan kajian penelitian yang sedang dilakukan (Jogiyanto, 2005).

2. Studi Pustaka

Pengumpulan data dan informasi dengan cara membaca buku-buku terkait yang dapat dijadikan acuan penelitian (Keraf, 1994).

3. Studi Literatur

Peneliti membaca dan mempelajari buku serta jurnal yang berhubungan dengan perancangan sistem, serta penelitian sejenis untuk mendukung topik dan memperkuat argumen dalam penyusunan skripsi ini.

1.6.2 Metode Pengembangan dan Perancangan Sistem

Metode pengembangan sistem menggunakan model *Rapid Application Development* (RAD). Terdapat 3 fase utama dalam model ini yaitu *Requirement*

Planning, Workshop Design, dan Implementation. Metode perancangan sistem dalam penelitian ini menggunakan bagan alir *Flowchart* dan *STD (Start Transition Diagram)*.

1.7 Sistematika Penulisan

Dalam penyusunan laporan skripsi ini sistematika penulisan terdiri dari 5 bab, adapun uraian masing-masing bab tersebut adalah:

BAB I PENDAHULUAN

Bab ini membahas secara keseluruhan mengenai penulisan laporan. Bab ini berisi latar belakang, rumusan masalah, ruang lingkup, tujuan, manfaat, metodologi penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisikan berbagai teori yang mendasari analisis permasalahan dan berhubungan dengan topik yang dibahas.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang metodologi penelitian yang digunakan dalam penulisan laporan.

BAB IV PEMBAHASAN

Bab ini berisi tentang masalah dari sistem yang sedang berjalan, tujuan dibuatnya sistem, menentukan batasan ruang lingkup sistem yang akan dibangun, merancang sistem yang diusulkan, perancangan dan desain sistem yang digambarkan dengan menggunakan *Flowchart*.

BAB V**PENUTUP**

Bab ini berisi kesimpulan yang didapat dari hasil analisis yang dilakukan dan saran-saran yang dapat digunakan sebagai bahan masukan untuk pengembangan lebih lanjut.



BAB II

LANDASAN TEORI

2.1 Temu Kembali Informasi (*Information Retrieval*)

Temu kembali informasi atau perolehan informasi atau yang lebih dikenal sebagai *Information Retrieval* (IR). Mendefinisikan IR merupakan suatu bidang yang berhubungan dengan representasi, penyimpanan, pengorganisasian dan pengaksesan suatu informasi. Tujuan dari representasi dan pengorganisasian informasi adalah untuk mempermudah pengaksesan informasi yang dibutuhkan oleh pengguna.

Salah satu permasalahan yang dihadapi dalam IR atau perolehan informasi adalah sulitnya mengkarakterisasi kebutuhan informasi pengguna. Untuk mendapatkan perolehan informasi, pengguna harus menterjemahkan kebutuhan informasinya ke dalam bentuk kata kunci (*keyword*) agar dapat diproses lebih lanjut. Lalu, *keyword* ini akan diterjemahkan menjadi suatu *query*. *Query* adalah ekspresi dari informasi yang dibutuhkan pengguna dalam bahasa yang disediakan oleh sistem informasi. Dengan *query* tersebut akan didapatkan sejumlah dokumen yang mungkin berguna atau relevan sesuai dengan kebutuhan pengguna (Yates dan Neto, 1999).

Information Retrieval (IR) adalah perolehan informasi dan bukan perolehan data. Dalam perolehan informasi dengan menggunakan *query* tertentu dapat ditemukan sejumlah dokumen. Akan tetapi tidak dapat dipastikan bahwa dokumen yang ditemukan relevan dengan informasi yang diinginkan oleh

pengguna. Ada kemungkinan dokumen yang ditemukan tidak relevan atau dokumen yang relevan justru tidak ditemukan. Hal itu disebabkan karena perolehan informasi memiliki data yang tidak terstruktur baik, semantiknya bersifat ambigu. Jika terjadi kesalahan kecil terhadap *query*, maka *query* tersebut tidak akan langsung ditolak. Sementara itu dalam perolehan data, bila *query* sesuai dengan nilai atribut yang ada dalam basis data maka akan ditemukan data yang relevan, dan bila *query* tidak sesuai dengan nilai atribut yang ada dalam basis data maka tidak akan ditemukan informasi apapun. Perolehan data mempunyai data yang terstruktur dan jelas semantiknya (Yates dan Neto, 1999).

Berbagai riset terus dilakukan untuk mengatasi permasalahan dalam perolehan informasi. Mulai tahun 2000 sampai saat ini, riset yang dilakukan berfokus pada ekstraksi informasi secara otomatis, Multimedia IR, *Cross-Language IR*, Ringkasan Dokumen dan lain-lain (Adriani, 2008).

2.2 Peringkas Dokumen Otomatis (*Automatic Text Summarization*)

Peringkas dokumen otomatis (*Automatic Text Summarization*) adalah salah satu solusi untuk mengatasi masalah membanjirnya informasi. Memberikan ringkasan dokumen diharapkan dapat membantu pengguna memilih dokumen yang paling relevan sesuai dengan kebutuhannya (Husni dan Kurniawan, 2003).

Menurut Kupiec 2005, ringkasan didefinisikan sebagai teks yang singkat yang didalamnya mengandung arti penting dari dokumen sumber. Ringkasan harus dapat mengurangi kompleksitas dan panjang dokumen. Judul, kata kunci dan isi harus dipertimbangkan dalam membuat ringkasan. Panjang ringkasan

diukur dengan rasio kompresi dan dapat dinyatakan dengan jumlah kata atau jumlah kalimat. Rasio kompresi dapat dihitung dari perbandingan antara panjang ringkasan dengan panjang dokumen sumber (Hovy, 1998).

Ringkasan dapat dilakukan secara manual atau otomatis. Ringkasan secara manual ditulis oleh penulis itu sendiri atau ditulis oleh abstraktor (pembuat abstrak) profesional. Sedangkan ringkasan secara otomatis dilakukan dengan menggunakan bantuan komputer. Ringkasan secara manual memiliki beberapa masalah, yaitu :

- a. membutuhkan abstraktor profesional atau orang yang ahli dalam bidang dokumen yang akan dibuat ringkasannya (Luhn, 1958).
- b. terkadang ringkasan yang dihasilkan oleh abstraktor menjadi bias karena dipengaruhi oleh latar belakang, sikap dan minatnya (Luhn, 1958).
- c. ringkasan yang dihasilkan oleh orang yang sama memiliki kemungkinan hasil yang tidak sama. Jika orang yang sama membuat abstrak dari artikel yang sama dan waktu yang berbeda, kemungkinan memiliki hasil yang tidak sama (Luhn, 1958).
- d. ringkasan secara manual lebih mahal dan menghabiskan banyak waktu karena jumlah informasi yang semakin meningkat (Jaruskulchai, 2003).

Masalah-masalah dalam ringkasan manual tersebut dapat diatasi dengan membuat ringkasan secara otomatis dengan menggunakan bantuan komputer. Ringkasan secara manual ataupun otomatis terdiri dari beberapa jenis, yaitu (Hovy, 1998) :

1. Tujuan

Ringkasan terbagi atau ringkasan indikatif dan ringkasan informatif.

Ringkasan indikatif digunakan untuk kategorisasi cepat yang mengindikasikan subjek permasalahan dari dokumen. Sedangkan ringkasan informatif berisi semua informasi penting yang terdapat dalam dokumen.

2. Bentuk

Bentuk ringkasan terdiri dari dua yaitu ekstrak dan abstrak. Ekstrak adalah ringkasan yang berisi kalimat-kalimat yang ada dalam dokumen sumber tanpa ada perubahan pada kalimat-kalimat tersebut. Sedangkan abstrak adalah menyimpulkan kembali isi dari dokumen, kalimatnya tidak sama dengan dokumen sumber (Husni dan Kurniawan, 2003).

3. Konteks

Konteks ringkasan terbagi menjadi dua, yaitu ringkasan *generic* dan *query-oriented*. Ringkasan *generic* mengikuti pandangan penulis secara umum. Contohnya adalah ringkasan yang biasanya disediakan oleh penulis. Sedangkan *query-oriented* merupakan ringkasan yang merefleksikan minat dari pembaca. Pembaca memasukkan *query* (pertanyaan), kata-kata apa saja yang ingin diperoleh informasinya dalam bentuk ringkasan.

4. Latar Belakang

Ringkasan dibagi menjadi ringkasan *background* dan *just-the news*.

Ringkasan *background* mengasumsikan jika pengetahuan pembaca sebelumnya tidak banyak. Contohnya adalah ringkasan yang ada dalam buku-buku pelajaran. Sedangkan *just-the news* merupakan ringkasan yang meliputi perkembangan sesuai fakta terbaru. Contohnya adalah ringkasan yang ada dalam surat kabar.

5. Dimensi

Dimensi ringkasan terdiri dari dua, yaitu ringkasan *single-document* dan *multi-document source*. Ringkasan *single-document* berdasarkan pada suatu dokumen saja. Contohnya dokumen X yang diringkas menjadi ringkasan Y. Sedangkan ringkasan *multi-document source* adalah ringkasan yang menggabungkan beberapa dokumen. Contohnya dokumen X dan dokumen Y yang diringkas menjadi ringkasan Z.

2.3 Penelitian Peringkas Otomatis

Penelitian mengenai peringkas otomatis sudah pernah dilakukan oleh peneliti-peneliti lain. Mereka mengembangkan berbagai macam metode dalam menghasilkan ringkasan secara otomatis. Metode tersebut dikembangkan untuk berbagai jenis dokumen dan bahasa.

2.3.1 Metode Signifikasi Kata Dan Kalimat

Penelitian yang dilakukan oleh Luhn 1958, merupakan pendekatan klasik dalam menghasilkan ringkasan secara otomatis dengan bahasa inggris. Metode yang digunakan adalah metode signifikasi kata dan kalimat.

Metode ini menetukan kalimat mana yang terbaik dalam artikel yang dapat dijadikan sebagai ringkasan otomatis. Dibutuhkan pengukuran dengan membandingkan dan mengurutkan isi informasi dari semua kalimat. Sebuah nilai diberikan kepada masing-masing kalimat sesuai dengan kriteria kualitas signifikasi.

Faktor signifikasi dari sebuah kalimat diperoleh dari analisis kata. Analisis kata menggunakan perhitungan frekuensi kemunculan kata. Kata-kata yang dihasilkan akan membentuk bagian yang memiliki frekuensi tinggi, menengah, dan rendah. Kata-kata yang memiliki frekuensi tinggi dianggap terlalu umum untuk memiliki signifikasi. Kata-kata dengan frekuensi rendah dianggap tidak berguna sehingga harus dibuang. Dengan demikian, kata-kata yang dianggap signifikan adalah kata-kata yang memiliki frekuensi menengah.

Diantara kelompok kata yang signifikan terdapat 4 atau 5 kata yang tidak signifikan yang akan digolongkan atau dibentuk *cluster*. *Cluster* yang memiliki nilai signifikasi paling tinggi akan dimasukkan sebagai nilai signifikasi kalimat. Setelah itu, semua kalimat diurutkan berdasarkan nilai signifikasinya. Nilai signifikasi kalimat tertinggi akan dijadikan ringkasan.

2.3.2 Metode Pemilihan Kalimat

Brando 1995 melakukan penelitian peringkas dokumen otomatis artikel berita dengan menggunakan metode pemilihan kalimat berbahasa Inggris. Penelitiannya menghasilkan sistem yang disebut ANES (*Automatic News Extraction System*). Sistem ini mengambil contoh materi berita mencakup tiga kategori, yaitu berita umum, majalah dan surat kabar.

ANES memiliki empat proses utama, yaitu analisa korpus, pemilihan kata-kata yang penting (*signature words*), pembobotan kalimat dan pemilihan kalimat. Analisa korpus dihitung terlebih dahulu sebagai acuan untuk melakukan pembuatan ringkasan, yang berisi daftar kata-kata yang penting yang mengindikasikan topic (*signature words*). Masing-masing kata penting tersebut dihitung dengan menggunakan metode frekuensi kemunculan kata berdasarkan rumus tf (*term frequency*) dan idf (*inverse document frequency*). Kemudian hasil frekuensi kata tersebut dijumlahkan sehingga mendapatkan bobot kalimat. Bobot kalimat digunakan untuk memilih kalimat yang akan dijadikan ringkasan. Kalimat yang telah dibobot diurutkan sesuai bobot tertinggi sampai terendah. Bobot kalimat yang tertinggi diambil untuk dijadikan ringkasan (Brando, 1995).

2.3.3 Metode Statis Dan *Query Biased*

Penelitian ini merupakan penelitian peringkas otomatis artikel berita yang dilakukan oleh Melani 2007. Penelitian ini merupakan pengembangan dari penelitian Tombros 1998 dengan menggunakan pendekatan statis dan *query-biased*.

Metode statis merupakan suatu metode yang menggunakan kalimat yang terdapat pada awal dokumen sebagai ringkasan. Dengan demikian, ringkasan yang dihasilkan akan selalu sama untuk setiap dokumen. Namun, tidak semua kalimat yang terdapat pada awal dokumen dapat mengindikasikan topik dari dokumen tersebut. Metode *query-biased* memungkinkan isi ringkasan menjadi dinamis. Sebab ringkasan yang dihasilkan melalui metode ini merefleksikan minat pembaca.

Pada metode *query-biased*, dilakukan pencarian dokumen pada sistem perolehan informasi. Pencarian tersebut berdasarkan *query* yang diberikan. Dokumen yang diperoleh dari proses pencarian tersebut menjadi masukan dalam pembuatan ringkasan. Pada metode *query-biased* ini membutuhkan *input query* yang sebelumnya telah disimpan sehingga jika pengguna harus memasukkan *query* yang sesuai dengan yang sudah ada barulah ringkasan akan didapatkan.

2.4 Pemrosesan Kalimat

Dalam sebuah dokumen mengandung banyak kata. Akan tetapi, tidak semua kata itu penting dan tidak semua kata itu dapat mewakilkan isi dari dokumen tersebut. Oleh karena itu, dibutuhkan pemrosesan kata dari koleksi dokumen untuk menetapkan kata yang akan digunakan. Proses tersebut bertujuan untuk menyeragamkan kata, menurunkan *noise*, dan mengurangi volume kosakata (Yates dan Neto, 1999).

2.4.1 *Tokenizing*

Tokenizing adalah proses memecah aliran teks menjadi kata-kata, frase, simbol, atau elemen bermakna lain yang disebut token. Daftar token menjadi masukan untuk diproses lebih lanjut. *Tokenizing* berguna baik dalam linguistik (bentuk segmentasi teks) dan dalam ilmu komputer ia merupakan bagian dari analisis leksikal (Huang dan Prevot, 2007). Menurut Weiss *et al.* (2005), proses ini cukup rumit untuk sebuah program komputer karena beberapa karakter dapat dijadikan sebagai pembatas (*delimiter*) dari token-token itu sendiri. Pembatas dari token tersebut antara lain spasi, tab dan baris baru sedangkan karakter () < ! ? “ . , terkadang dianggap sebagai pembatas dan juga bukan pembatas tergantung pada kondisi pemakainya.

Pemilahan ini biasanya dilakukan dengan cara memisahkan kalimat menjadi kata-kata dan menghilangkan kata-kata yang bukan merupakan alfabet dan angka. Semua huruf kapital diubah menjadi huruf kecil agar token dapat diurutkan secara alfabet dan diperlakukan sama dengan token-token lain.

2.4.2 Penghilangan *Stopword*

Kata-kata yang terlalu sering ada dikoleksi dokumen bukanlah diskriminator yang baik. Pada kenyataannya, 80% kata-kata yang ada dikoleksi dokumen adalah tidak berguna untuk perolehan informasi. Kata-kata tersebut juga dikenal sebagai *stopwords* yang bertujuan untuk menyaring kata-kata yang berguna.

Penghilangan *stopwords* mempunyai manfaat tambahan yang penting, mengurangi beban ketika proses indexing berjalan dan menghemat penyimpanan dari index yang panjang. Pada kenyataannya, proses ini bertujuan untuk mengurangi ukuran struktur index sampai 40% atau lebih.

Sejak penghapusan *stopwords* disediakan, daftar *stopwords* diperluas dari kata-kata yang meliputi kata depan dan kata penghubung seperti “yang”, “dengan”, “akan” menjadi beberapa kata kerja, kata keterangan, dan kata sifat bisa dijadikan *stopwords* (Yates, 1999). Daftar *stopwords* yang dipakai dalam tugas akhir ini adalah *stopwords* berdasarkan penelitian Tala 2003.

2.4.3 Pembobotan

Pembobotan kata sangat berpengaruh dalam menentukan kemiripan antara dokumen dengan *query*. Jika bobot tiap kata dan kalimat dapat ditentukan dengan tepat, diharapkan hasil perhitungan kemiripan teks akan menghasilkan perngkingan dokumen yang baik (Mandala, 2004).

2.4.3.1 Metode Tf (*Term Frequency*)

Cara termudah untuk memberi bobot pada kata adalah dengan menghitung jumlah kemunculan dari kata tersebut pada dokumen. Jumlah kemunculan kata disebut dengan *Term Frequency* (tf). Tf pertama kali diperkenalkan dalam penelitian Luhn pada tahun 1958. Tf ini didefinisikan sebagai berikut (Yates dan Neto, 1999) :

$$Tf_{ij} = \text{frekuensi kemunculan kata } i \text{ pada dokumen } j$$

Kata-kata yang frekuensinya lebih tinggi pada dokumen seharusnya akan lebih mengindikasikan topik dan menggambarkan dengan baik apa isi dari dokumen tersebut.

2.4.4 Evaluasi

Evaluasi terhadap peringkas dilakukan untuk mengetahui seberapa baik ringkasan yang dihasilkan. Evaluasi ini dilakukan dengan cara menyebarluaskan kuisioner dengan tujuan untuk mengetahui tingkat pemahaman pengguna terhadap ringkasan yang dihasilkan oleh sistem peringkas otomatis.

2.5 Model Pengembangan Sistem *Rapid Application Development* (RAD)

2.5.1 Pengertian *Rapid Application Development* (RAD)

Menurut Agustinus (2002) *Rapid Application Development* (RAD) adalah salah satu metode pengembangan suatu sistem informasi dengan waktu yang relatif singkat. Untuk pengembangan suatu sistem informasi yang normal membutuhkan waktu minimal 180 hari, akan tetapi dengan menggunakan metode RAD suatu sistem dapat diselesaikan hanya dalam waktu 30-90 hari.

Tujuan utama dari semua metode sistem *development* adalah memberikan suatu sistem yang dapat memenuhi harapan dari para pemakai, akan tetapi sering kali di dalam melakukan pengembangan suatu sistem tidak melibatkan para pemakai sistem secara langsung, sehingga hal ini menyebabkan sistem informasi yang dibuat jauh dari harapan pemakai yang dapat berakibat sistem tersebut

walaupun dapat diterima tetapi para pemakai enggan untuk menggunakannya atau bahkan para pemakai menolak untuk menggunakannya.

Pada saat RAD diimplementasikan, maka para pemakai bisa menjadi bagian dari keseluruhan proses pengembangan sistem dengan bertindak sebagai pengambil keputusan pada setiap tahapan pengembangan. RAD bisa menghasilkan suatu sistem dengan cepat karena sistem yang dikembangkan dapat memenuhi keinginan dari para pemakai sehingga dapat mengurangi waktu untuk pengembangan ulang setelah tahap implementasi.

Tahapan pengembangan aplikasi cepat (RAD) terdiri atas perencanaan kebutuhan, rancangan pemakai, konstruksi dan *cut over*. RAD memerlukan empat unsur penting, yaitu (McLeod, 2008):

1. Manajemen

Manajemen, khususnya manajemen puncak, harus merupakan orang yang suka bereksperimen, yang suka melakukan hal yang baru atau orang yang cepat tanggap, yang cepat belajar menggunakan metode baru. Manajemen harus mendukung RAD sepenuhnya dan menyediakan lingkungan kerja yang membuat kegiatan tersebut sangat menyenangkan.

2. Manusia

Dari pada menggunakan satu tim tunggal untuk mengerjakan semua kegiatan SLC, RAD menyadari efisiensi yang dicapai dengan penggunaan beberapa tim yang terspesialisasi. Tim untuk perencanaan kebutuhan, rancangan pemakai dan *cut over* dapat dibentuk. Anggota tim ini adalah para ahli metodologi dan peralatan yang diperlukan untuk melaksanakan tugas khusus

mereka. Untuk menggambarkan tim, Martin menggunakan istilah tim SWAT (*skill with advanced tools*).

3. Metodologi

Metodologi dalam RAD adalah siklus hidup RAD, yaitu terdiri dari empat tahap, yaitu perencanaan kebutuhan, rancangan pemakai, konstruksi, dan *cut over*. Tahap-tahap ini mencerminkan pendekatan sistem. Pemakai berperan penting dalam setiap tahap, bekerja sama dengan spesialis informasi.

4. Peralatan.

Peralatan RAD terutama terdiri dari bahasa-bahasa pemrograman generasi keempat (*fourth-generation language*) dan peralatan CASE yang memudahkan *prototyping* dan pembuatan kode. Bahasa pemrograman generasi keempat memungkinkan spesialis informasi atau pemakai untuk menghasilkan kode komputer tanpa menggunakan bahasa pemrograman konvensional.

2.5.2 Tujuan RAD

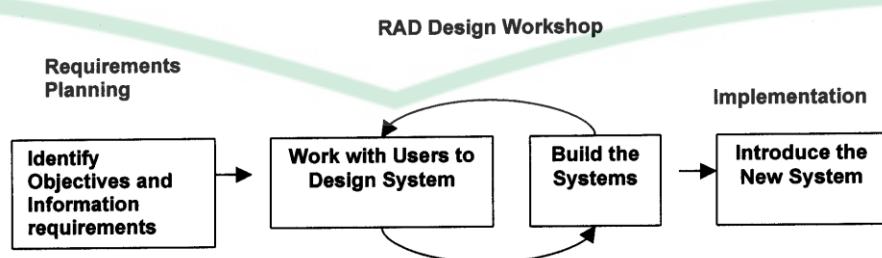
Dengan menggunakan RAD, terdapat beberapa tujuan berikut ini yang tidak akan dapat dicapai secara bersama-sama yaitu (Agustinus, 2002):

- a. Kemungkinan terjadi kesalahan yang kecil, karena pihak pengembang tidak mempunyai hak untuk mengubah komponen-komponen yang digunakan dalam mengembangkan suatu sistem.

- b. Tingkat kepuasan konsumen yang tertinggi, karena kebutuhan-kebutuhan sekunder dari konsumen harus dikorbankan supaya suatu sistem dapat diselesaikan sesuai jadwal.
- c. Biaya pengembangan yang termurah, karena dengan menggunakan komponen yang sudah ada dapat menyebabkan biaya yang lebih besar apabila dibandingkan dengan mengembangkan komponen sendiri.

2.5.3 Tahapan RAD (*Rapid Application Development*)

Menurut Kendall & Kendall (2008) RAD (*Rapid Application Development*) adalah pendekatan berorientasi objek untuk pengembangan sistem yang meliputi metode pengembangan sebagai serta perangkat lunak. Beberapa pengembang melihat RAD sebagai pendekatan yang dapat membantu dalam membuat *e-commerce*, *environments* berbasis *web* yang disebut sebagai penggerak bisnis pertama dimana statusnya mungkin penting. Terdapat tiga fase dalam model pengembangan sistem RAD yang melibatkan pengguna dan analis dalam penilaian, perancangan, implementasi, diantaranya dapat dilihat pada gambar berikut:



Gambar 2.1 Tahapan *Rapid Application Development* (RAD).

2.5.3.1 Perencanaan Kebutuhan (*Requirements Planning*)

Merupakan sebuah tahap awal untuk suatu proyek, dimana pengguna dan analis mengidentifikasi kebutuhan untuk memenuhi tujuan sistem, mengidentifikasi kebutuhan informasi yang timbul dari tujuan tersebut, serta menentukan batasan-batasan sistem yang akan dibuat, kendala serta alternatif masalah. Fase ini membutuhkan keterlibatan intens dari kedua kelompok, bukan terletak pada penandatanganan pada proposal atau dokumen.

Pada tahap ini, *user* dan *analyst* melakukan semacam pertemuan untuk melakukan identifikasi tujuan dari aplikasi atau sistem dan melakukan identifikasi kebutuhan informasi untuk mencapai tujuan. Pada tahap ini hal terpenting adalah adanya keterlibatan dari kedua belah pihak, bukan hanya sekedar persetujuan akan proposal yang sudah dibuat. Untuk lebih jauh lagi, keterlibatan user bukan hanya dari satu tingkatan pada suatu organisasi, melainkan beberapa tingkatan organisasi sehingga informasi yang dibutuhkan untuk masing-masing *user* dapat terpenuhi dengan baik. Di samping itu, dapat juga melakukan koordinasi dengan *Chief Information Office* (CIO) atau bagian perencana strategis terutama untuk mengembangkan suatu aplikasi *E-commerce* berbasis *Web* untuk mendapatkan informasi yang lebih detail akan tujuan dari suatu organisasi. Pertemuan semacam ini seringkali disebut *Joint Application Development*.

2.5.3.2 Proses Desain (*Workshop Design*)

Merupakan tahap lanjutan dari tahap perencanaan kebutuhan (*Requirements Planning*), dimana dilakukan pengidentifikasi dari solusi

alternatif yang ada dengan pemilihan solusi terbaik. Setelah itu, dilanjutkan dengan melakukan pemodelan proses peringkasan dan desain pemrograman untuk data-data yang telah diperoleh yang nantinya akan dimodelkan dalam arsitektur informasi.

Untuk tahap ini maka keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain. Biasanya, *user* dan *analyst* berkumpul menjadi satu dan duduk di meja melingkar dimana masing-masing orang bisa melihat satu dengan yang lain tanpa ada halangan. Pada tahap desain ini membutuhkan waktu beberapa hari, akan tetapi bisa semakin lebih lama, tergantung dari besar kecilnya sistem yang dibuat. Pada selang waktu tersebut, *user* bisa memberikan tanggapan akan sistem yang sudah dikembangkan untuk selanjutnya dilakukan perbaikan-perbaikan. Dengan demikian proses pengembangan suatu sistem membutuhkan waktu yang cepat.

2.5.3.3 Implementasi (*Implementation System*)

Tahap ini terdiri atas dua tahapan, yaitu tahap pengimplementasian sistem ke dalam bahasa pemrograman (*coding*) dan tahap pengujian sistem oleh beberapa *owner*, *analyst*, dan *developer* dengan tujuan apakah sistem yang dibangun sudah berjalan dengan baik pada saat pengoperasiannya atau masih terdapat kesalahan (*error*).

Adapun hal terpenting dalam tahapan ini adalah keerlibatan *user*, agar sistem yang akan dikembangkan dapat memberikan kepuasan kepada *user*, dan sistem yang lama tidak perlu dijalankan secara paralel dengan sistem yang baru.

2.5.4 Keuntungan RAD

Beberapa keuntungan dalam menggunakan metode RAD adalah sebagai berikut (Agustinus, 2002):

- a. Membeli sistem yang baru memungkinkan untuk lebih menghemat biaya ketimbang mengembangkan sendiri.
- b. Proses pengiriman menjadi lebih mudah, hal ini dikarenakan proses pembuatan lebih banyak menggunakan potongan-potongan *script*.
- c. Mudah untuk diamati karena menggunakan model *prototype*, sehingga user lebih mengerti akan sistem yang dikembangkan.
- d. Lebih fleksibel karena pengembang dapat melakukan proses desain ulang pada saat yang bersamaan.
- e. Bisa mengurangi penulisan kode yang kompleks karena menggunakan *wizard*.
- f. Keterlibatan *user* semakin meningkat karena merupakan bagian dari tim secara keseluruhan.
- g. Mampu meminimalkan kesalahan-kesalahan dengan menggunakan alat-alat bantuan (*CASE Tools*).

2.5.5 Kondisi Penunjang Keberhasilan RAD

Beberapa kondisi yang dapat menunjang keberhasilan dari RAD adalah sebagai berikut (Agustinus, 2002):

- a. Sistem berjalan sendiri (*standalone*).
- b. Kinerja dari sistem bukan faktor terpenting.
- c. Distribusi produk yang bersifat sempit.
- d. Ruang lingkup yang terbatas.
- e. Kehandalan dari sistem bukan faktor terpenting.
- f. Membutuhkan teknologi yang tidak terlalu baru (lebih dari 1 tahun).
- g. Sistem dapat dipecah-pecah menjadi bagian-bagian yang lebih kecil.

2.6 Model Perancangan Sistem

Alat-alat perancangan sistem diantaranya adalah *Flowchart Diagram* dan *State Transition Diagram* (STD).

2.6.1 Bagan Alir (Flowchart)

Bagan alir (*flowchart*) adalah suatu bagan yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir. *Flowchart* berguna bagi programmer untuk mempersiapkan program yang rumit. Bagan alir terdiri dari simbol-simbol yang mewakili fungsi-fungsi langkah program dan garis alir menunjukkan urutan dari simbol-simbol yang akan

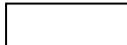
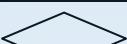
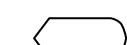
dikerjakan (Suyanto, 2004). Adapun alasan penggunaan *flowchart* sebagai berikut (Palupi, 2007) :

1. *flowchart* dapat memberikan gambaran yang efektif, jelas, dan ringkas tentang prosedur logic. Teknik penyajian yang bersifat grafis jelas akan lebih baik daripada uraian-uraian yang bersifat teks khususnya dalam menyajikan logika-logika bersifat kompleks.
2. dengan adanya pengungkapan yang jelas dalam model atau chart maka para pembaca dapat dengan mudah melihat permasalahan atau memfokuskan perhatian pada area-area tertentu sistem informasi.
3. karena simbol-simbol yang digunakan mengikuti suatu standar tertentu yang sudah diakui secara umum, maka *flowchart* dapat merupakan alat bantu yang sangat efektif dalam mengkomunikasi logika suatu masalah atau dalam mendokumentasikan logika tersebut.

Berikut ini adalah simbol-simbol program *flowchart* (Suyanto, 2004)

Tabel 2.1 Simbol-Simbol Flowchart.

Nama	Simbol	Fungsi
Terminator	○	Permulaan atau akhir program
Garis alir (Flow line)	→	Arah aliran program
Preparation	◇	Proses inisialisasi atau pemberian harga awal

Proses		Proses perhitungan atau proses pengolahan data
Input/Output data		Proses input atau output data, parameter, informasi
Predefined proses (sub program)		Permulaan sub program atau proses menjalankan sub program
Decision		Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
On Page connector		Penghubung bagian-bagian flowchart yang berada pada satu halaman
Off page Connector		Penghubung bagian-bagian flowchart yang berada pada halaman berbeda
Document		Input berasal dari dokumen dalam bentuk kertas atau output dicetak dikertas
Display		Symbol visual display unit atau cathode ray tube sebagai input atau output

2.6.2 State Transition Diagram (STD)

Menurut Pressman (2001 : 302), STD mengindikasikan bagaimana perangkat lunak berlaku sebagai konsekuensi dari kejadian eksternal yang menyebabkan perubahan suatu kondisi. Untuk merealisasikannya, STD menghadirkan model dari suatu kejadian yang disebut dengan *state*. Dalam STD, proses yang terjadi digambarkan dengan transisi antar *state*.

Ada dua macam simbol yang menggambarkan proses dalam STD, yaitu :

1. Gambar persegi panjang yang menunjukkan kondisi (*state*) dari sistem.
2. Gambar panah yang menunjukkan transisi antar *state*. Tiap panah diberi label dengan ekspresi aturan. Label yang diatas menunjukkan kejadian yang menyebabkan transisi terjadi. Label yang bawah menunjukkan aksi yang terjadi akibat dari kejadian tadi.

2.6.2.1 Pendekatan untuk Membuat STD

Ada dua pendekatan dalam membuat STD, yaitu :

- a. Identifikasi setiap kemungkinan state dari sistem dan gambarkan masing-masing pada state sebuah kotak, kemudian tentukan hubungan antar state tersebut.
- b. Dimulai dengan state P1 dan dilanjutkan dengan state P2, berikutnya dilanjutkan sesuai flow yang diinginkan.

2.6.2.2 Notasi State Transition Diagram (STD)

Notasi STD terdiri dari state dan *transition state*. *State* adalah kumpulan keadaan atau atribut yang mencirikan seseorang atau suatu benda pada waktu tertentu. Bentuk state dibagi menjadi dua, yaitu *Initial State* dan *Final State*. *Initial state* menyatakan awal dari suatu state (hanya ada satu state), sedangkan *Final state* menyatakan aktif dari suatu state (bisa lebih dari satu state).

Transition State terdiri dari kondisi dan aksi. Kondisi adalah suatu kejadian pada lingkaran luar yang dapat dideteksi oleh sistem. Sedangkan aksi adalah yang dilakukan oleh sistem bila terjadi perubahan state atau merupakan reaksi terhadap kondisi.

a. Keadaan Sistem

Setiap kontak mewakili suatu keadaan dimana sistem mungkin berada di dalamnya. State disimbolkan dengan segi empat.

Simbol State :



b. Perubahan Sistem

Ini digunakan untuk menghubungkan suatu keadaan dengan keadaan lain. Jika sistem memiliki transisi dalam prilakunya, maka suatu keadaan dapat berubah menjadi keadaan tertentu.

Simbol transition state : →

[Empty rectangle] = State

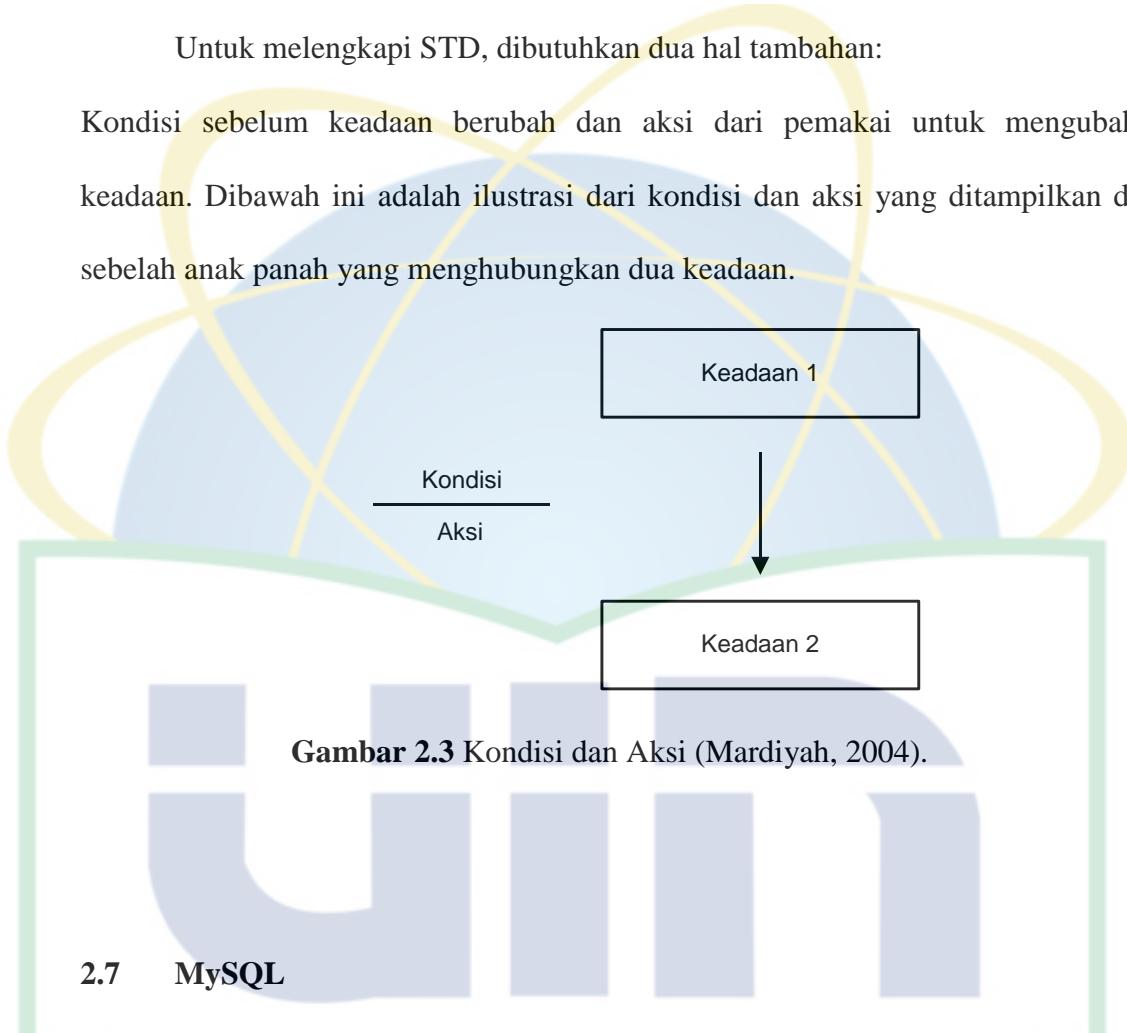
→ = Transition State

Gambar 2.2 Notasi STD (Mardiyah, 2004).

c. Kondisi dan Aksi

Untuk melengkapi STD, dibutuhkan dua hal tambahan:

Kondisi sebelum keadaan berubah dan aksi dari pemakai untuk mengubah keadaan. Dibawah ini adalah ilustrasi dari kondisi dan aksi yang ditampilkan di sebelah anak panah yang menghubungkan dua keadaan.



Gambar 2.3 Kondisi dan Aksi (Mardiyah, 2004).

2.7 MySQL

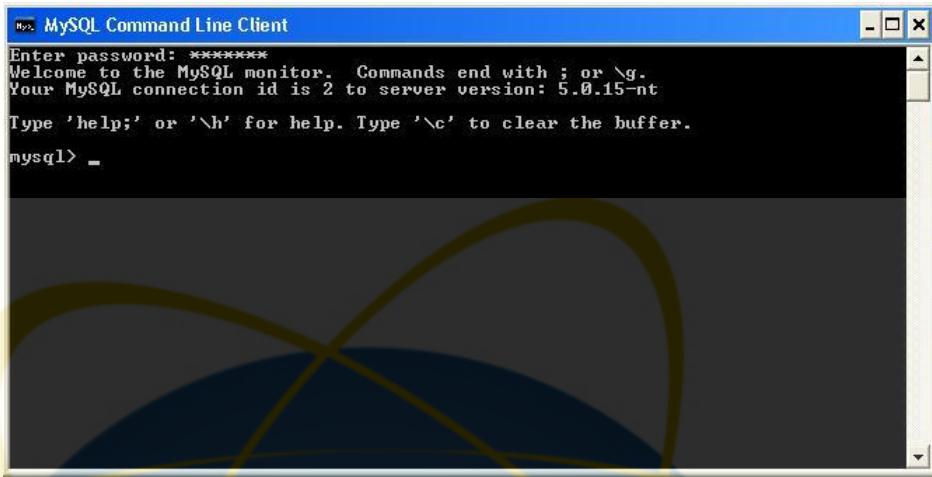
MySQL merupakan *database* yang dikembangkan dari bahasa SQL (*Structure Query Language*). SQL sendiri merupakan bahasa yang terstruktur yang digunakan untuk interaksi antara *script* program dengan *database sever* dalam hal pengolahan data. Dengan SQL, kita dapat membuat tabel yang nantinya akan diisi dengan data, memanipulasi data (misalnya menambah data, menghapus data dan memperbarui data), serta membuat suatu perhitungan dengan berdasarkan data yang ditemukan (Prasetyo, 2004: 11-14).

Oleh karena hal tersebut, SQL tidak hanya terbatas digunakan untuk mendapatkan suatu tampilan dari *database* yang statis. Saat ini juga sedang dikembangkan standar baru yang dikenal sebagai SQL3, yang berencana membuat SQL menjadi bahasa yang mendekati mesin *tuning*.

MySQL dikembangkan sekitar tahun 1994 oleh sebuah perusahaan pengembang *software* sekaligus konsultan *database* bernama MySQL AB yang bertempat di Swedia. Waktu itu perusahaan tersebut masih bernama TcX DataKonsult AB dan tujuan awal dikembangkannya MySQL adalah untuk mengembangkan aplikasi berbasis *web* pada *client*.

Awalnya Michael Widnus “Monty”, pengembang satu-satunya di TcX, memiliki sebuah aplikasi *UNIREG* dan rutin *ISAM* buatannya sendiri dan sedang mencari antarmuka SQL yang cocok untuk diimplementasikan ke dalamnya. Mula-mula Monty memakai miniSQL (mSQL) pada eksperimennya itu, namun mSQL dirasa kurang sesuai, karena lambat dalam pemrosesan *query*.

MySQL adalah *database* yang memiliki kecepatan yang tinggi dalam melakukan pemrosesan data, dapat diandalkan, dan mudah digunakan serta mudah dipelajari. MySQL mendukung banyak bahasa pemrograman seperti C, C++, Perl, Python, Java dan PHP. Selain itu, dengan bantuan *driver* ODBC, MySQL juga mampu berinteraksi dengan berbagai pemrograman visual seperti Delphi, Visual Basic dan sebagainya.



Gambar 2.4 Tampilan MySQL 5.

2.8 PHP

2.8.1 Sejarah PHP

PHP pertama kali dikenal pada musim gugur tahun 1994 oleh Rasmus Lerdoff, awalnya digunakan pada *website*-nya untuk mencatat siapa saja yang berkunjung dan melihat biodatanya. Versi pertama yang di-*release* tersedia pada awal tahun 1995, dikenal sebagai *tool personal home page*, yang terdiri atas *engine parser* yang sangat sederhana yang hanya mengerti beberapa makro khusus dan sejumlah utilitas yang sering digunakan pada halaman-halaman *web*.

Parser diprogram ulang pada pertengahan 1995 dan diberi nama PHP/F1 versi 2.0. F1 bersal dari paket Ramus lainnya yang ditulis untuk menginterpretasi data dari *form* yang kemudian dikombinasikan dengan *tool personal home page* dan ditambahkan dukungan untuk *database mSQL (mini SQL)* (Sidik, 2006).

2.8.2 Pengertian PHP

PHP merupakan *script* untuk pemrograman *script web server side*, *script* yang membuat dokumen HTML secara *on the fly*, dokumen HTML yang dihasilkan dari suatu aplikasi buatan dokumen HTML yang dibuat dengan menggunakan editor teks dan editor HTML (Sidik, 2006).

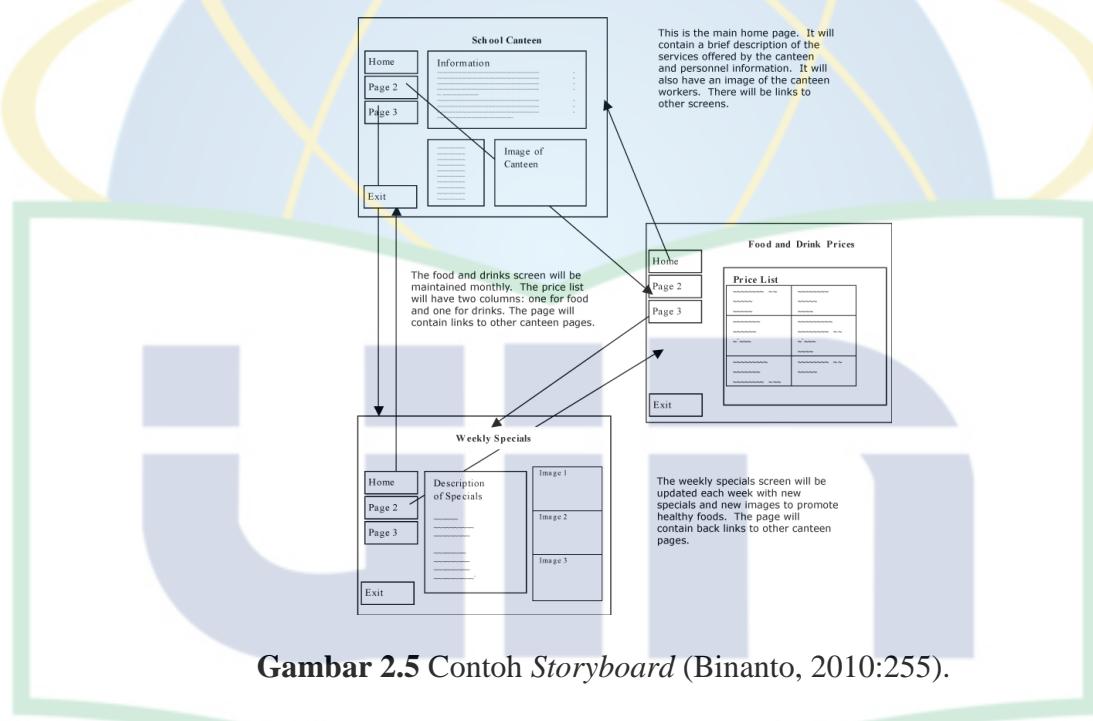
PHP telah tersedia pada hampir semua sistem operasi jaringan yang menyediakan *web server* terutama *web server Apache*. *Web server* berbasis windows non apache juga telah mendukungnya, seperti IIS, PWS, atau Xitami dari Windows 98/ME sampai dengan Windows NT 4/2000 dan XP.

Halaman web yang menggunakan PHP sebagai *script* pengembangan aplikasinya memungkinkan membuat situs yang memiliki sajian informasi yang:

- a. Interaktif, *server* dapat menerima masukan dari pemakai kemudian memproses masukan tersebut kemudian memberikannya lagi kepada pemakai berupa data hasil pengolahan.
- b. Halaman dibuat berdasarkan suatu permintaan dari pengunjung atau pemakai. Pemakai dapat memasukan kriteria informasi yang diinginkan, PHP akan mengakses ke dalam *database* seperti MySQL untuk menampilkan informasi yang diminta tersebut.
- c. Menampilkan informasi terbaru secara otomatis. Halaman *web* akan secara otomatis menampilkan informasi berdasarkan kriteria yang digunakan untuk mengotomatisasi berita terbaru (Sidik, 2005).

2.9 Storyboard

Membuat *storyboard* mempunyai peran yang sangat penting dalam pengembangan multimedia. *Storyboard* digunakan sebagai alat bantu pada tahap perancangan multimedia. *Storyboard* merupakan peng-organisasian grafik, contohnya adalah sederetan ilustrasi atau gambar yang ditampilkan berurutan untuk keperluan *visualisasi* awal dari suatu file, animasi, atau urutan media interaktif termasuk interaktivitas web. (Binanto, 2010:255)

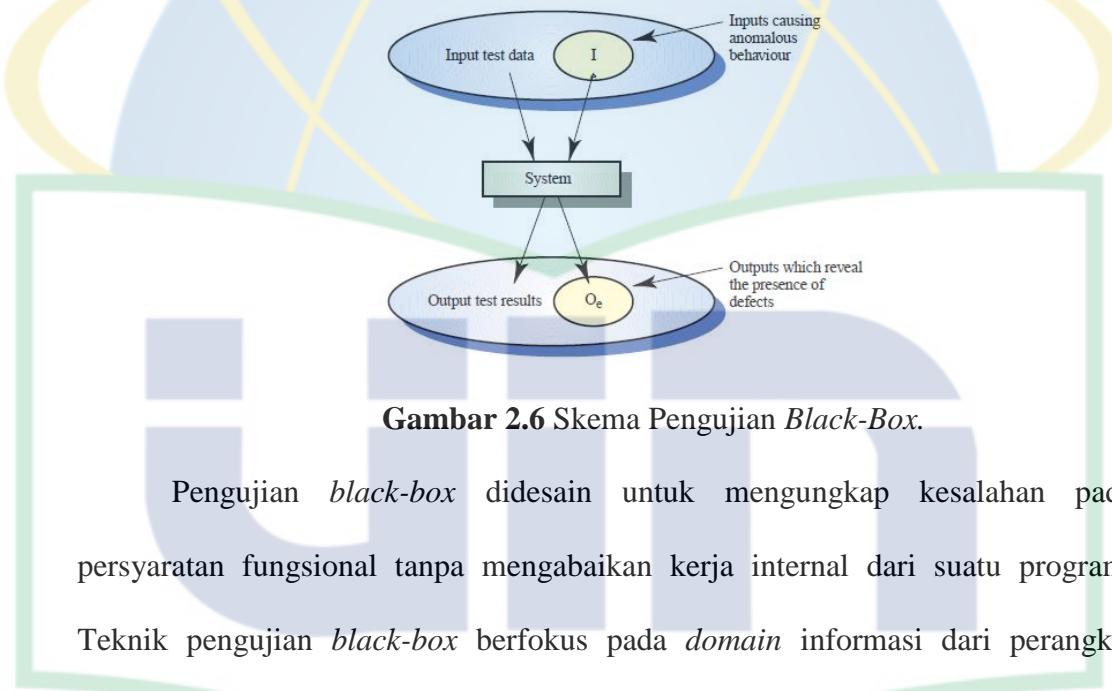


2.10 Pengujian

Pengujian merupakan bagian dari proses rekayasa perangkat lunak secara terintegrasi untuk memastikan kualitas dari perangkat lunak dengan melakukan

verifikasi untuk mencari kesalahan dengan asumsi bahwa hasilnya didokumentasikan dan direview lebih lanjut (Soetam, 2011).

Pengujian *black-box* merupakan Pendekatan pengujian dimana program dianggap sebagai suatu “*black-box*”(kotak hitam), program *test case* berbasiskan spesifikasi, *test planning* dapat dimulai sejak awal proses pengembangan sistem. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Berikut ini adalah skema mengenai pengujian *black box* adalah (Pressman, 2002):



Pengujian *black-box* didesain untuk mengungkap kesalahan pada persyaratan fungsional tanpa mengabaikan kerja internal dari suatu program. Teknik pengujian *black-box* berfokus pada *domain* informasi dari perangkat lunak, dengan melakukan *test case* dengan mempartisi *domain input* dan *output* dari suatu program dengan cara memberikan cangkupan pengujian yang mendalam.

BAB III

METODE PENELITIAN

Dalam penyusunan skripsi ini diperlukan data-data informasi sebagai bahan yang dapat mendukung kebenaran materi uraian pembahasan. Untuk menyelesaikan masalah yang ada dalam sebuah perancangan perangkat lunak ada beberapa tahap yang harus dilakukan. Dalam bab ini dijelaskan mengenai metode pengumpulan data dan metode pengembangan sistem yang digunakan penulis.

3.1 Metode Pengumpulan Data

3.1.1 Wawancara

Metode wawancara dilakukan untuk mendapatkan informasi dari orang yang terkait langsung dengan penelitian. Penulis melakukan wawancara dengan Dzakiah Nurfadhilah peneliti yang membahas tema sejenis dengan judul Peringkas Otomatis Artikel Berbahasa Indonesia Menggunakan Metode TF-IDF (*Term Frequency-Inverse Document Frequency*). Dari wawancara tersebut diperoleh informasi mengenai pemecahan kalimat menggunakan teknik tokenisasi dan bagaimana pembobotan dilakukan setelah proses tokenisasi.

3.1.2 Studi Pustaka

Pada tahapan ini penulis melakukan pengumpulan data yang terkait dari berbagai sumber. Sumber bacaan berasal dari buku-buku ilmu komputer, jurnal, karya ilmiah, dan buku elektronik yang berstandar. Adapun buku-buku yang dipelajari dan situs-situs internet yang dikunjungi dapat dilihat pada daftar pustaka.

3.1.3 Studi Literatur

Dari hasil pengumpulan penelitian sejenis yang dilakukan oleh penulis maka dapat di analisa kelebihan dan kekurangan antara penelitian tersebut dengan penelitian yang penulis lakukan. Berikut adalah tabel perbedaan antara penelitian ini dengan penelitian yang sejenis :

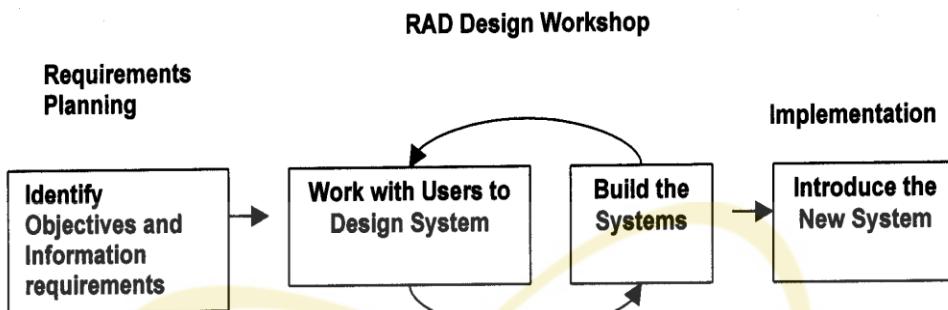
Tabel 3.1 Studi Literatur.

Nama	Judul	Kelebihan	Kekurangan
Dzakiah Nur Fadilah (UIN Syarif Hidayatullah Jakarta)	Peringkas Otomatis Artikel Berbahasa Indonesia Menggunakan Metode TF-IDF (<i>Term Frequency-Inverse Document Frequency</i>)	Kapasitas aplikasi kecil	Terlalu banyak form-form aplikasi yang tidak dibutuhkan

Nama	Judul	Kelebihan	Kekurangan
Aristoteles (FMIPA Universitas Lampung)	Penerapan Algoritma Genetika Pada Peringkasan Teks Dokumen Bahasa Indonesia	Kapasitas aplikasi kecil	Tampilan aplikasi peringkas sangat sederhana
Haryani Diah Sitawati (Universitas Indonesia)	Peringkas Otomatis Untuk Dokumen Berbahasa Indonesia Dengan Metode Frase Penunjuk Dan Tf-Idf	Kapasitas aplikasi kecil	Nilai hasil Evaluasi (<i>Recall</i> dan <i>Precision</i>) aplikasi kecil

3.2 Metode Pengembangan Sistem

Dalam melakukan penelitian terhadap rancang aplikasi peringkas otomatis dilakukan beberapa tahap pada pengembangan RAD (*Rapid Application Development*), yaitu:



Gambar 3.1 Fase - Fase Model RAD.

3.2.1 Fase *Requirements Planning* (Perencanaan Kebutuhan)

Merupakan sebuah tahap awal untuk suatu proyek, dimana pengguna sistem dan pembuat sistem mengidentifikasi kebutuhan untuk memenuhi tujuan sistem, mengidentifikasi kebutuhan informasi yang timbul dari tujuan tersebut, serta menentukan batasan-batasan sistem pendukung keputusan yang dibuat, kendala serta alternatif masalah.

Dalam fase ini diperoleh perencanaan yang dibuat dalam aplikasi peringkas otomatis, yaitu:

1. Tujuan aplikasi peringkas otomatis
2. Analisis kebutuhan peringkas otomatis
3. definisi masalah

3.2.2 Fase Desain

Merupakan tahap lanjutan dari tahap perencanaan kebutuhan (*Requirements Planning*), dimana dilakukan pengidentifikasi dari solusi alternatif yang ada dengan pemilihan solusi terbaik. Setelah itu, dilanjutkan dengan melakukan pemodelan proses peringkasan dan desain pemrograman untuk data-data yang telah diperoleh yang nantinya akan dimodelkan dalam arsitektur informasi. Untuk tahap ini keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain. Fase ini merupakan fase untuk merancang dan memperbaiki yang dapat digambarkan sebagai *workshop*.

Setelah proses pemodelan dilakukan, kemudian proses desain dilanjutkan dengan :

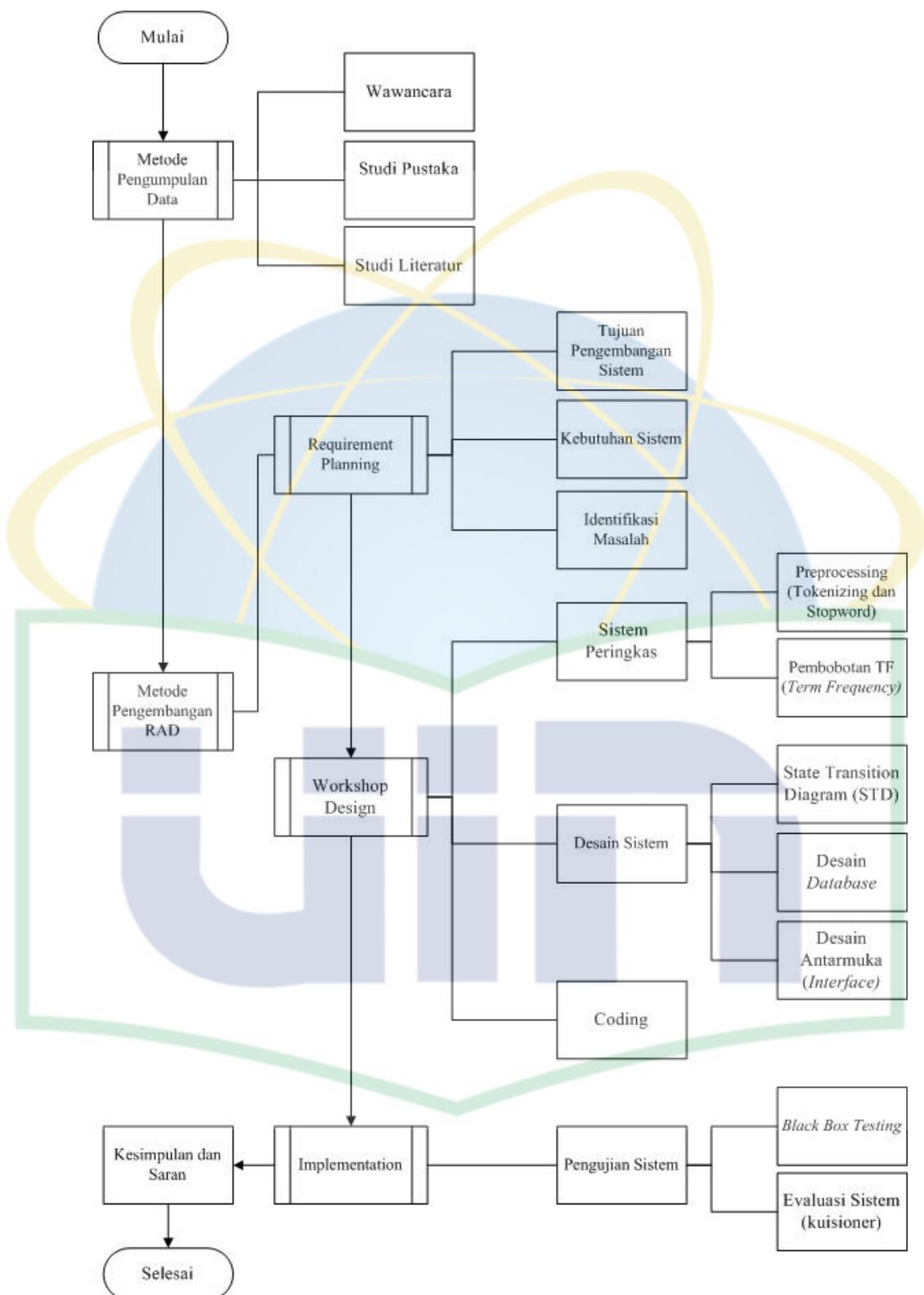
1. merancang sistem dengan menggunakan *Flowchart Diagram* dan *State Transition Diagram* (STD).
2. Desain proses, meliputi desain tentang proses-proses apa saja yang dibutuhkan oleh sistem.
3. Desain basis data, merupakan desain tabel-tabel yang dibutuhkan dalam pengolahan data. Tabel-tabel ini akan diimplementasikan kedalam bentuk *database*, yang menggunakan program MYSQL.
4. Desain antarmuka, merupakan desain untuk menentukan masukan dan keluaran yang akan digunakan untuk mengopersikan sistem (Ladjamudin, 2005).

3.2.3 Fase Implementasi

Tahap ini terdiri atas dua tahapan, yaitu tahap pengimplementasian sistem ke dalam bahasa pemrograman (*coding*) dan tahap pengujian sistem dengan tujuan apakah sistem yang dibangun sudah berjalan dengan baik pada saat pengoperasiannya atau masih terdapat kesalahan (*error*). Adapun hal terpenting dalam tahapan ini adalah keterlibatan *user*, agar sistem yang akan dikembangkan dapat memberikan kepuasan kepada *user*. Pada tahap ini pengujian dan *testing* dilakukan terhadap aplikasi dan sistem, serta melakukan sosialisasi terhadap aplikasi dan sistem yang telah dibuat kepada para pengguna jaringan publik. Dengan selesainya tahap ini maka berakhirlah proses rancang aplikasi peringkas otomatis.

3.3 Kerangka Berfikir Penelitian

Adapun pelaksanaan kegiatan dalam penelitian ini, tertuang dalam kerangka berfikir yang dapat dilihat dalam *flowchart* berikut:



Gambar 3.2 Kerangka Berfikir Penelitian.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan tahapan pengembangan sistem yang digunakan dalam penulisan skripsi. Dalam penelitian ini, menggunakan metode pengembangan sistem RAD.

4.1 *Requirement Planning* (Perencanaan Kebutuhan)

Fase ini adalah fase dimana *analyst* sistem mengidentifikasi tujuan, dan menentukan syarat-syarat dari kebutuhan sistem yang ditimbulkan atas tujuan sistem yang dirumuskan, serta mengidentifikasi masalah yang menjadi latar belakang dalam perancangan sistem.

4.1.1 Tujuan Pengembangan Sistem

Tujuan dari pengembangan aplikasi peringkas dokumen ini adalah untuk membantu para pengguna jaringan publik untuk mendapatkan ringkasan dari suatu bacaan tanpa harus menghilangkan poin-poin penting dari bacaan tersebut. Aplikasi peringkasan dokumen berbasis web yang dibangun ini diharapkan akan mempermudah dan mempercepat pengguna dalam proses meringkas dokumen tanpa harus membaca keseluruhan isi dokumen.

4.1.2 Kebutuhan Sistem

Proses yang dibutuhkan agar sistem peringkas dapat berjalan dengan baik adalah sebagai berikut:

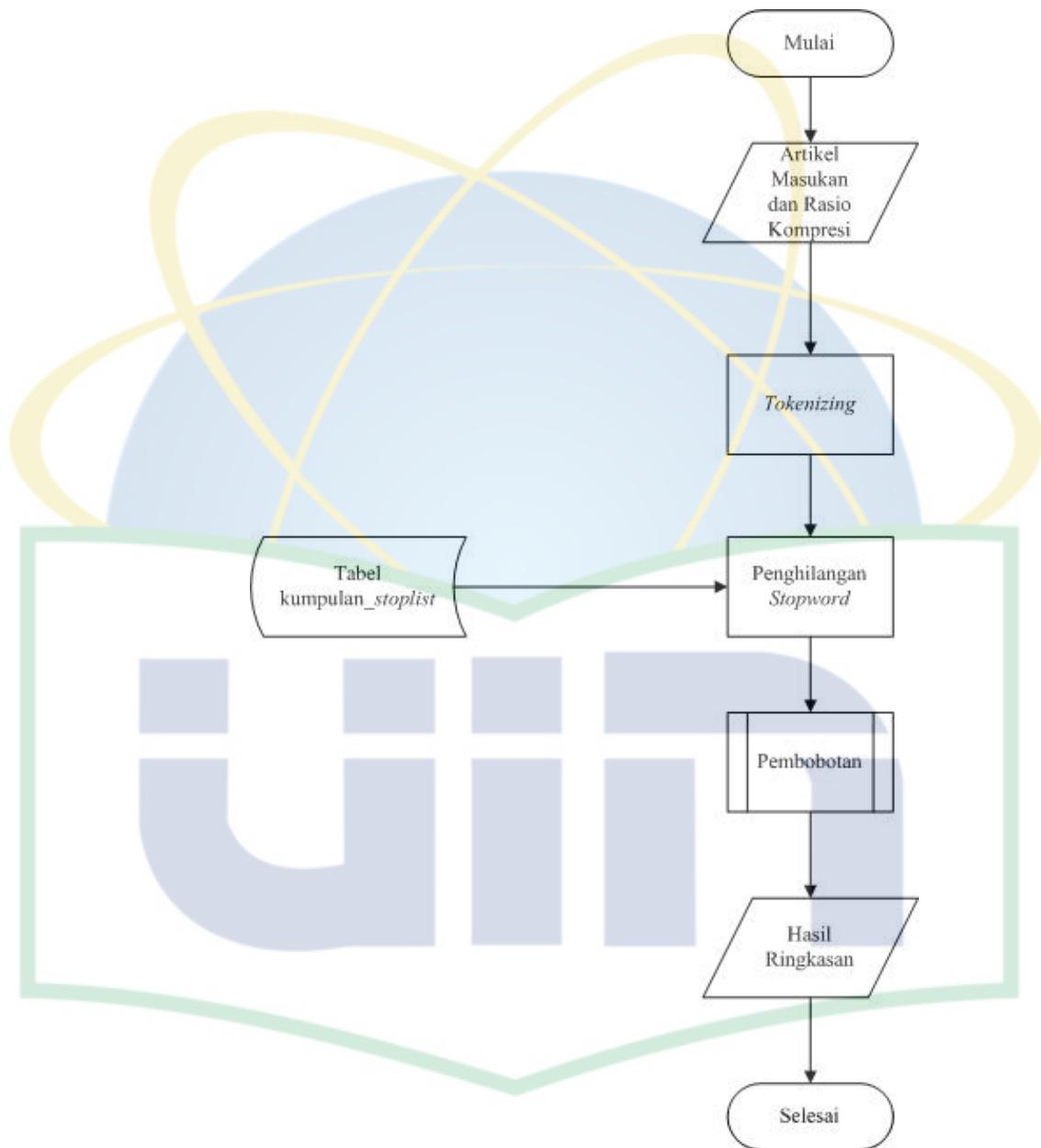
1. Koleksi dokumen berupa artikel berita yang diambil oleh penulis dari artikel beritaonline kompas yang terdiri dari lima kategori (ekonomi, olahraga, teknologi dan informasi, kesehatan dan metropolitan).
2. Kumpulan kata penghubung bahasa Indonesia. Kumpulan kata ini diperlukan untuk proses penghilangan *stopword*.

4.1.3 Identifikasi Masalah

Banyaknya informasi yang tersedia saat ini diperlukan suatu cara yang mudah dan cepat untuk mengetahui isi suatu dokumen atau menemukan dokumen yang relevan. Berkembangnya informasi tersebut membuat banyak orang yang ingin memperoleh informasi secara cepat dan tepat tanpa harus membaca keseluruhan informasi tersebut. Dari permasalahan tersebut dapat disimpulkan bahwa para pembaca artikel berita terutama artikel di media online membutuhkan sebuah sistem yang dapat memudahkan dalam meringkas isi dokumen sehingga membantu para pencari dokumen sehingga sesuai atau relevan dengan kebutuhan.

4.2 Proses Desain (*Workshop Design*)

4.2.1 Perancangan Sistem Peringkas



Gambar 4.1 Flowchart Penyusunan Ringkasan.

4.2.1.1 *Tokenizing*

Tokenizing adalah proses memecah aliran teks menjadi kata-kata, frase, simbol, atau elemen bermakna lain yang disebut token. Daftar token menjadi masukan untuk diproses lebih lanjut.

Contoh artikel yang di inputkan adalah artikel yang didapatkan dari berita online yang dipublikasikan oleh Kompas.com.

Tabel 4.1 Contoh *Tokenizing*.

Sebelum di <i>Tokenizing</i>	Debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Debat kali ini diwarnai pantun dari moderator acara.
Sesudah di <i>Tokenizing</i>	Debat calon presiden dan wakil presiden periode terakhir



diwarnai
hal
berbeda
dari
awal
hingga
akhir
acara
debat
kali
ini
diwarnai
pantun
dari
moderator
acara

4.2.1.2 Pehilangan Stopword

Proses selanjutnya adalah penghapusan *stopword*. Daftar kata-kata *stopword* didapat dari penelitian (Tala, 2003) dan daftar kata-kata *stopword* terlampir pada lampiran. Dengan contoh kata yang sama pada sebelumnya, maka didapat hasil sebagai berikut :

Tabel 4.2 Contoh Penghilangan *Stopword*.

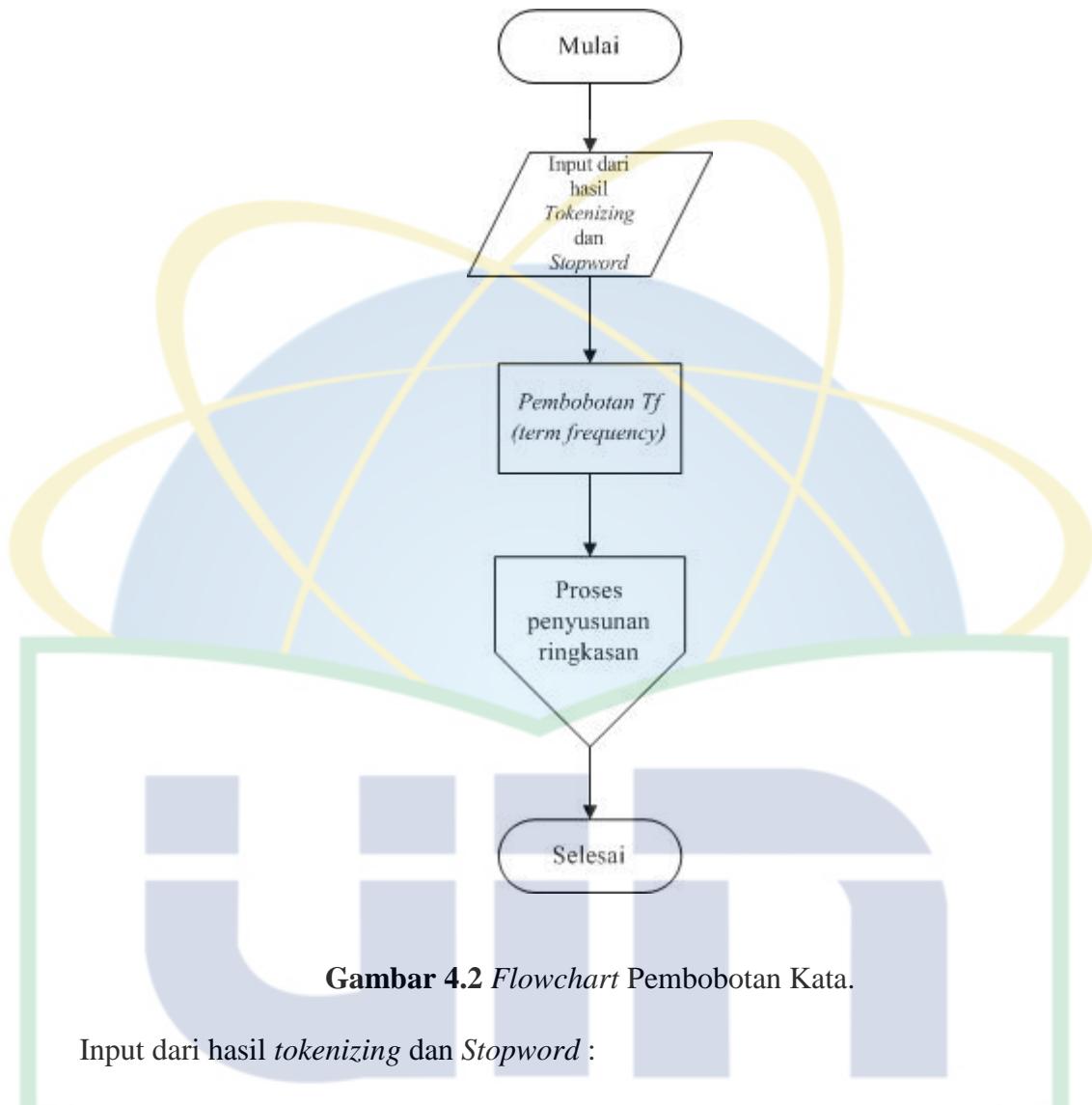
Proses Tokenisasi	Pehilangan <i>Stopword</i>
Debat	Debat
calon	calon
presiden	presiden
dan	-
wakil	wakil
presiden	presiden
periode	periode
terakhir	-
diwarnai	diwarnai
hal	-
berbeda	berbeda
dari	-
awal	-

hingga	-
akhir	-
acara	acara
debat	debat
kali	kali
ini	-
diwarnai	diwarnai
pantun	pantun
dari	-
moderator	moderator
acara	acara

4.2.1.3 Pembobotan Tf (*Term Frequency*)

Pembobotan kata sangat berpengaruh dalam menentukan kemiripan antar dokumen dengan *query*. Pada proses ini, kata dihitung menggunakan metode pembobotan Tf (*term frequency*), yaitu dengan menghitung frekuensi kemunculan kata yang terdapat dalam dokumen. Adapun langkah-langkah dari proses ini adalah sebagai berikut :

a. Pembobotan Tf (*Term Frequency*)



Artikel yang di inputkan	Debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Debat kali ini diwarnai pantun dari moderator acara.
--------------------------	--

Input dari hasil *tokenizing* :

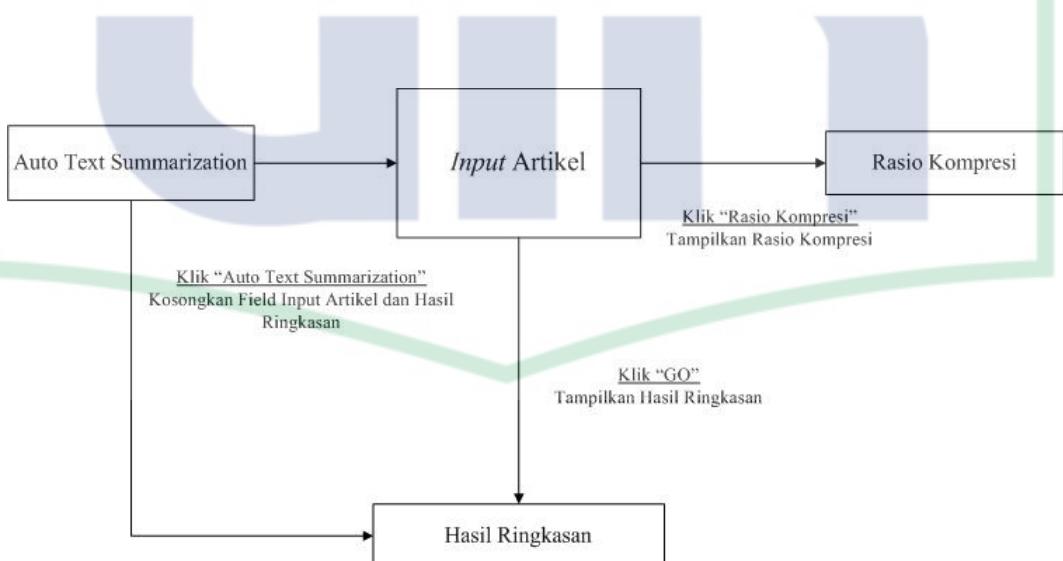
Tabel 4.4 Proses Tokenisasi, Penghapusan *Stopword*, dan Pembobotan.

Proses Tokenisasi	Penghapusan Stopword	Pembobotan
Debat	Debat	2
calon	calon	1
presiden	presiden	1
dan	-	1
wakil	wakil	1
presiden	presiden	1
periode	periode	1
terakhir	-	1
diwarnai	diwarnai	2
hal	-	1
berbeda	berbeda	1
dari	-	2
awal	-	1
hingga	-	1
akhir	-	1
acara	acara	2

debat	debat	2
kali	kali	1
ini	-	1
diwarnai	diwarnai	2
pantun	pantun	1
dari	-	2
moderator	moderator	1
acara	acara	2

4.2.2 Desain Sistem

4.2.2.1 State Transition Diagram (STD)



Gambar 4.3 STD (State Transition Diagram) halaman Utama Peringkas.

State transition diagram halaman utama peringkas, *user* dapat menginput artikel pada field teks asli, kemudian hasil dari ringkasan akan didapatkan pada field hasil ringkasan. Pada bagian rasio kompresi, *user* dapat memilih rasio kompresi yang diinginkan dengan range rasio kompresi yang disediakan 10%, 20% dan 30%. State Auto Text Summarization berfungsi sebagai tombol *reset* dimana *user* dapat menggunakan untuk mengosongkan field input artikel dan hasil ringkasan.

4.2.2.2 Desain Basis Data

Proses desain basis data ini menjelaskan data-data yang terlibat dalam proses-proses. Tetapi proses ini tidak menggambarkan bagaimana data diorganisir dan dikelompokkan (Al-Fatta, 2007). Desain basis data ini dirancang menggunakan *database MySQL* sebagai bagian dari proses ringkasan. Pada aplikasi ini *database* dibuat dengan menggunakan nama “kamus” dan mempunyai tabel-tabel sebagai berikut :

a. Tabel kumpulan *stoplist*

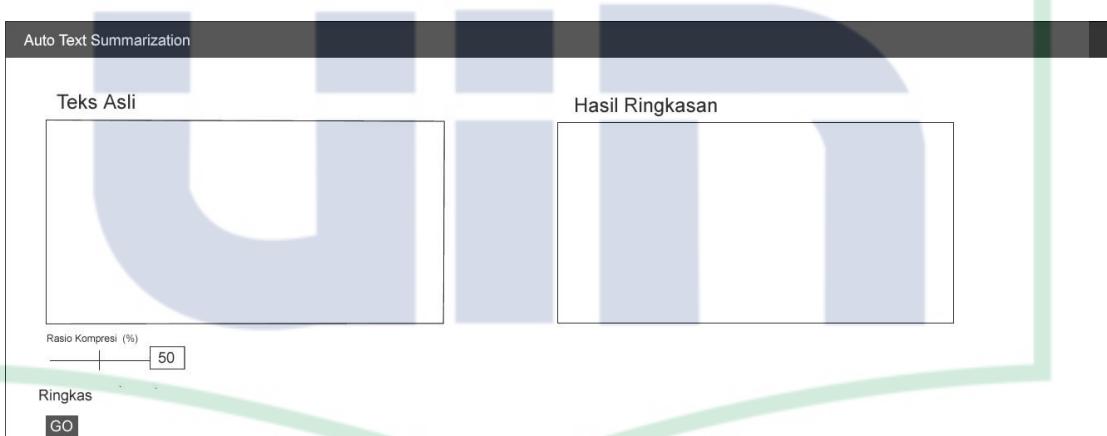
tabel ini menyimpan kata-kata penghubung yang tidak mempunyai arti penting atau *stopword*. Tabel kumpulan_stoplist ini hanya memiliki satu atribut yaitu nama_stoplist dengan *type char(30)* sebagai *primary key*.

			namo_stoplist
<input type="checkbox"/>			acapkali
<input type="checkbox"/>			ada
<input type="checkbox"/>			adalah
<input type="checkbox"/>			adanya
<input type="checkbox"/>			adapun
<input type="checkbox"/>			agak
<input type="checkbox"/>			agaknya
<input checked="" type="checkbox"/>			akan
<input type="checkbox"/>			akankah
<input type="checkbox"/>			akhir
<input type="checkbox"/>			akhiri
<input type="checkbox"/>			akhirnya

Gambar 4.4 Tabel kumpulan *Stoplist*.

4.2.2.3 Desain Antarmuka

Desain antarmuka pengguna aplikasi peringkas otomatis dokumen berita berbahasa Indonesia ini adalah :



Gambar 4.5 Desain Antarmuka Pengguna.

4.2.3 Coding (Pengkodean)

Pada tahap ini dilakukan pengkodean (*coding*) terhadap rancangan sistem yang telah didefinisikan sebelumnya. Pengkodean dilakukan dengan

menggunakan bahasa pemrograman PHP, dan MySQL sebagai basis datanya serta berbasis *web* dan sesuai dengan rancangan yang telah dibuat sebelumnya.

Summarize

```
<?php
    include "./indexer.php";
    include "koneksi.php";
    function summarize ($filename, $compress) {
        // proses indexing
        $inv_index = indexer();

        // load file dan daftar stopwords
        $load_file = $filename;
        // $word = str_word_count($load_file,1);
        $word = preg_split("/ /", $load_file);
        $rr = array();
        foreach ($word as $key=>$val) {
            if($val == 'Prof.Dr.Ir'
|| $val == 'Dr.' || $val == 'Ir.' || $val == 'Prof.')
            {
                $val =
str_replace('. ', '__', $val);
                $rr = implode(
", $val");
            }
            // print_r($val);
        }
        // $sentence = preg_split("/ /",
$load_file);
        $sentence = preg_split("/[.]+/", $load_file);
        $sentence = array_slice($sentence, 0,
sizeof($sentence)-1); // buang array terakhir (kosong)

        $stp=mysql_query("SELECT * from
tbl_stopword");
        while($k=mysql_fetch_array($stp)){
            $stopwords=$k[stop_word];
            // $stopwords1 =
file_get_contents("./stopwords.txt");
            $stopwords = preg_split("/[\s]+/", $stopwords);
            // print_r($stopwords);
        }
    }
}
```

```

    // jumlah kalimat yang diringkas
    $compression_rate = $compress/100;
    $max_sentence =
floor(sizeof($sentence)*$compression_rate);

    // inisialisasi
$sentence_weight = array();

    // menghitung bobot tf.idf tiap kalimat
foreach ($sentence as $key => $value) {
    // tokenisasi dengan membuang stopwords
    $word = preg_split("/[\d\W\s]+/", 
strtolower($value));
    $word = array_diff($word, $stopwords);

    $word = array_values($word); // perbaiki
indeks

    // inisialisasi bobot dan hitung
frekuensi token
    $tf_idf = 0;
    $freq_word =
array_count_values($word);

    // hitung bobot tf.idf
    foreach ($freq_word as $token => $tf)
        $tf_idf += $tf *
$inv_index[$token]['idf'];

    // simpan nilai bobot kalimat
    array_push($sentence_weight, $tf_idf);
}

// end stopword
// sorting bobot tertinggi -> potong array ->
sorting urutan kalimat
arsort($sentence_weight);
$sorted = array_slice($sentence_weight, 0,
$max_sentence, true);
ksort($sorted);
// gabungkan ringkasan
$summary = "";
foreach ($sorted as $key => $value)
    $summary = $summary.$sentence[$key] .".

";

    // return teks asli dan hasil ringkasan
$output = array();

```

```
$output['original'] = $load_file;  
$output['summary'] = $summary;  
  
    //print_r($sentence);  
    return $output;  
}
```

4.3 Implementasi

4.3.1 Pengujian Sistem

4.3.1.1 *Black Box Testing*

Pada tahap ini dilakukan pengujian terhadap perangkat lunak sistem yang telah dibuat. Pada pengujian sistem, peneliti melakukan *testing* terhadap sistem pendukung keputusan dengan metode *black box testing*. Cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit-unit, kemudian diamati apakah hasil dari unit itu sesuai atau tidak dengan proses yang diinginkan.

Untuk memastikan bahwa program yang dibuat bebas dari kesalahan (*bug*) maka peneliti melakukan pengujian secara sistem yaitu menggunakan sistem *black box*. Walaupun tidak seratus persen bebas dari *bug*, namun setidaknya bisa meminimalkan kesalahan yang akan terjadi. Berikut ini adalah beberapa pengujian yang dilakukan peneliti :

Tabel 4.5 Uji Coba Modul Peringkas Teks Otomatis

No	Unit Program	Hasil Yang Diharapkan	Hasil Akurat	Keterangan
1.	Input Artikel	Memasukan Artikel	SESUAI	
2.	Rasio Kompresi	Memilih rasio kompresi yang diinginkan	SESUAI	
3.	Auto Text Summarization	Menghapus Artikel yang telah di input dan yang telah diringkas	SESUAI	
4.	GO	Menampilkan hasil ringkasan	SESUAI	

Auto Text Summarization

Teks Asli

Tidak seperti debat-debat sebelumnya, debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Debat kali ini diwarnai pantun dari moderator acara.

Debat terakhir capres-cawapres ini diselenggarakan oleh dua stasiun televisi, yakni Kompas TV dan TVRI. Debat ini diawali dengan penampilan duet Rio Febrian dan Lea Simanjuntak yang menyanyikan lagu "Rayuan Pula Kelapa". Acara kemudian dibuka oleh Ketua Umum Komisi Pemilihan Umum Husein Malik.

Adu pendapat antarpasangan calon dipandu oleh Rektor Universitas Diponegoro Prof. Sudharto P Hadi, MES, PhD. Sudharto menjalankan perannya sebagai moderator dengan baik hingga debat selesai.

Setelah kedua capres menyampaikan pemaparan penutup, Sudharto menutup acara debat dengan menyampaikan pantun. "Bunga mawar bunga melati. Tumbuh subur di depan rumah. Kita sukseskan pemilu nanti. Untuk Indonesia lebih cerah," kata

Hasil Ringkasan

Tidak seperti debat-debat sebelumnya, debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Sudharto menjalankan perannya sebagai moderator dengan baik hingga debat selesai.

Debat kelima ini merupakan debat terakhir dalam rangkaian debat capres-cawapres. Debat tersebut menutup rangkaian kampanye Pemilu Presiden 2014 sebelum memasuki masa tenang.

Ringkas
GO

Gambar 4.6 Tampilan Hasil Ringkasan.

4.3.1.2 Evaluasi Sistem (Kuisisioner)

Evaluasi dilakukan dengan cara meyebarkan kuisioner dengan tujuan untuk mengetahui tingkat pemahaman pengguna terhadap ringkasan yang dihasilkan. Kuisioner dibagi menjadi tiga bagian berdasarkan rasio kompresi. Satu rasio kompresi terdiri dari 10 artikel hasil ringkasan dengan rasio kompresi 10%, 20% dan 30%. Total kuisioner yang disebar sebanyak 30 kuisioner dengan 30 responden.

Nilai akurasi untuk tingkat pemahaman responden terhadap hasil ringkasan sesuai rasio kompresi dapat dihitung sebagai berikut :

$$\text{akurasi} = \frac{\text{jumlah ringkasan yang dipahami}}{\text{total ringkasan yang ada}} \times 100\%$$

- a. Rasio Kompresi 10%

$$\text{akurasi} = \frac{6}{10} \times 100\%$$

= 66,6 % Artikel berita yang dapat dipahami.

- b. Rasio Kompresi 20%

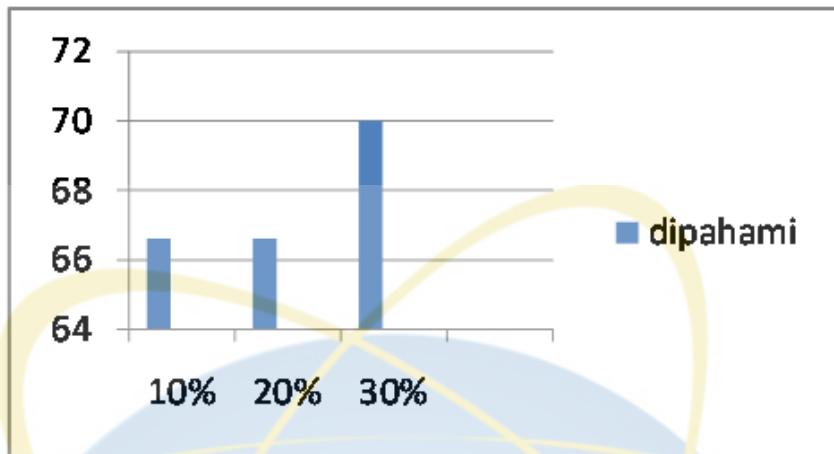
$$\text{akurasi} = \frac{6}{10} \times 100\%$$

= 66,6 % Artikel berita yang dapat dipahami.

- c. Rasio Kompresi 30%

$$\text{akurasi} = \frac{7}{10} \times 100\%$$

= 70 % Artikel berita yang dapat dipahami.



Gambar 4.7 Grafik Tingkat Pemahaman Pengguna

Hasil yang didapatkan melalui perhitungan nilai akurasi untuk tingkat pemahaman responden adalah 77,7% dengan rasio kompresi 20% dan 30%. Dapat disimpulkan bahwa tingkat pemahaman responden lebih baik pada ringkasan yang dihasilkan dengan rasio 30%.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang dapat diambil setelah melakukan pengujian terhadap hasil ringkasan dan perorma sistem yaitu :

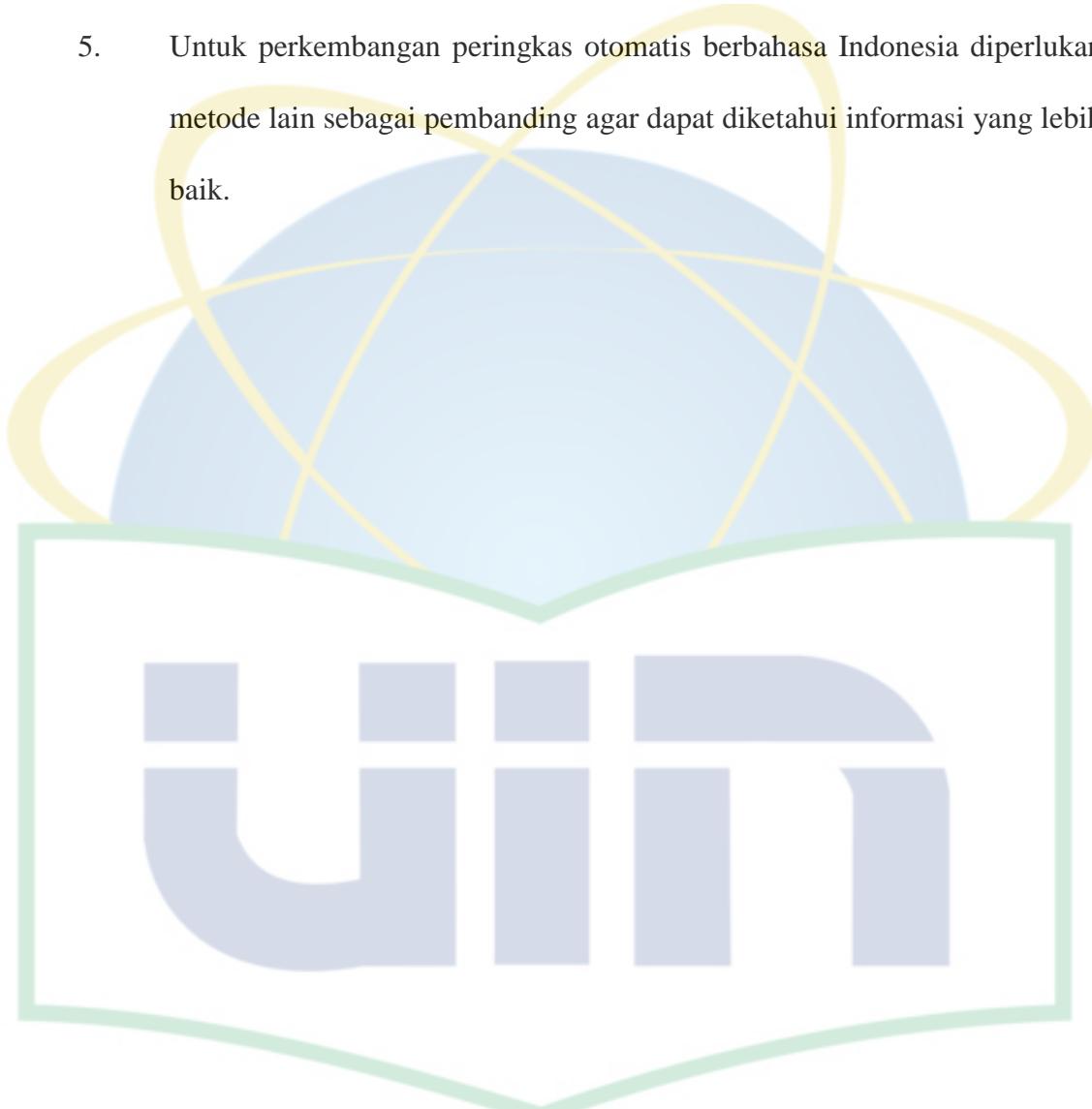
1. Pengimplementasian pembuatan sistem peringkas dokumen otomatis berita berbahasa Indonesia mempunyai proses dalam melakukan peringkasan. Tahapan dalam meringkas dokumen dimulai dari tahap *Preprocessing* yaitu : *Tokenizing*, penghapusan *Stopword* Dilanjutkan dengan *get frequent* dengan metode Tf (*Term Frequency*). Tahapan-tahapan tersebut yang menjadi dasar pembuatan atau implementasi dari sistem peringkas dokumen otomatis ini.

5.2 Saran

Berdasarkan kesimpulan yang telah dipaparkan sebelumnya, terdapat saran guna pengembangan sistem selanjutnya, yaitu :

1. Menambahkan tahap preprocessing yaitu pemilihan kata, *Case Folding*, *Stemming*. Pembobotan kata menambah metode Idf (*Inverse Document Frequency*).
2. Memperbanyak jumlah koleksi dokumen dan kategori dokumen yang akan digunakan benar-benar unik.
3. Menggunakan jenis dokumen selain artikel berita, seperti dokumen ilmiah.

4. Untuk evaluasi sistem peringkas ada baiknya bila rancangan eksperimen menggunakan evaluasi lain. Dengan demikian dapat terlihat lebih jelas akurasi kinerja metode yang digunakan.
5. Untuk perkembangan peringkas otomatis berbahasa Indonesia diperlukan metode lain sebagai pembanding agar dapat diketahui informasi yang lebih baik.



Daftar Pustaka

- Adriani, Mirna. 2008. *Information Retrieval*. Modul Kuliah Pemrosesan Teks Fakultas Ilmu Komputer UI Semester Ganjil 2009.
- Agustinus N. 2002. Studi Analisis Rapid Application Development Sebagai Salah Satu Alternatif Metode Pengembangan Perangkat Lunak. *Jurnal Informatika*, Vol.3, 2002. Hal 74-79. Universitas Kristen Petra.
- Al Fatta, Hanif. 2007. *Analisis dan Perancangan Sistem Informasi Untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern*. Yogyakarta : Andi.
- Alwi, Hasan dkk. 2003. *Tata Bahasa Baku Bahasa Indonesia*. Edisi kelima. Jakarta : Balai Pustaka.
- Arifin, Agus Zainal. 2003. *Aplikasi Information Retrieval untuk Pembentukan Tesaurus Berbahasa Indonesia Secara Otomatis*. Dalam : Seminar Nasional Ilmu Komputer dan Teknologi Informasi IV.
- Baeza-Yates, R, and Ribiero-Neto, B. 1999. *Modern Information Retrieval*. Addison-Wesley. New York.
- Binanto, Iwan. 2010. *Multimedia Digital-Dasar Teori dan Pengembangannya*. Andi. Yogyakarta.
- Brandow, Ronald dkk. 1995. *Automatic Condensation of Electronic Publications by Sentence Selection*. Dalam : Information Processing dan Management.
- Fattah, M. A. dan Ren, F. 2008. *Automatic Text Summarization*. World Academy of Science. Engineering and Technology.
- Gregorius, S. Budhi, Ibnu Gunawan dan Ferry Yuwono. 2006. *Algoritma Porter Stemmer For Bahasa Indonesia Untuk Pre-Processing Text Mining Berbasis Metode Market Basket Analysis*. Pakar Jurnal Teknologi Informasi dan Bisnis.
- Gregorius, S, Budhi. Intan, Rolly, Silvia dan Stevanus. 2007. *Indonesian Automated Text Summarization*. Petra Christian University. Informatics Engineering Dept.
- Hariwijaya, M. dan Djaelani, Bisri M. 2005. *Teknik Penulisan Skripsi dan Tesis*. Zenith Publisher. Yogyakarta.

- Hassel, M. 2004. *Summaries and the Process of Summarization from Evaluation of Automatic Text Summarization – A practical Implementation.* Liccentiate Thesis. KTH NADA.
- Hovy, E, Yew LC. 1998. *Automatic Text Summarization and Summarist System.* Fips ter III. Final Report SUMMAC.
- Huang, C., Simon, P., Hsieh, S., & Prevot, L. (2007) Rethinking Chinese Word Segmentation: Tokenization, Character Classification, or Word break Identification prosiding Prague, Association for Computational Linguistics <http://en.wikipedia.org/wiki/Tokenization>
- Husni, Muchammad dan Kurniawan, Arif. 2003. *Implementasi Sistem Peringkas Otomatis Dokumen Berita Berbahasa Indonesia Dengan Metode Extraction.* Dalam : Seminar Nasional Ilmu Komputer dan Teknologi Informasi IV.
- Irman, M., Prastowo, T. W., dan Nurdin, 2008. *Bahasa Indonesia 2: Untuk SMK/MAK Semua Program Keahlian Kelas XI.* Pusat Perbukuan: Departemen Pendidikan Nasional.
- Jaruskulchai, Chuleerat dan Kruengkrai, Canasai. 2003. *A Practical Text Summarizer by Paragraph Extraction for Thai.* Dalam : Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages.
- Ladjamudin, Al-Bahra. 2005. *Analisis dan Desain Sistem Informasi.* Yogyakarta : Graha Ilmu.
- Luhn, H.P. 1958. *The Automatic Creation of Literature Abstracts.* Dalam : IBM Journal Of Research and Development.
- Mandala, Rila. 2004. *Pengaruh Pembobotan Kata Terhadap Kinerja Search-Engine.* Dalam : The 1st Conference on Telematics Systems, Services, and Applications.
- Mani, Indrajeet. 2001. *Summarization Evaluation : an Overview.* Dalam : Proceedings of the NAACL 2001 Workshop an Automatic Summarization.
- Manning, Christopher. 2008. *An Introduction to Information Retrieval.* England : Cambridge University Press.
- Mardiyah, Dini. 2004. *Applikasi Pembelajaran Matematika Berbasis Multimedia.* FST UIN. Jakarta

- McLeod JR, dan Schell GP. 2008. *Management Information System*. Pearson Education, New Jersey: Pearson Prentice Hall, Inc., Edisi kesepuluh, 2007. Jakarta: Salemba Empat.
- Melani, Ade. 2007. *Peringkas Otomatis Untuk Dokumen Bahasa Indonesia Menggunakan Metode Statis dan Query Biased*. Depok : Fakultas Ilmu Komputer Universitas Indonesia.
- Prasetyo, Didik Dwi. 2004. *Aplikasi Database Client/Server Menggunakan Delphi dan MySQL*. PT. Elex Media Komputindo. Jakarta
- Pressman RS. 2002. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Sarkar, Kamal. 2009. *Sentences Clustering-Based Summarization of Multiple Text Document*. International Journal of Computing Science and Communication of Technologies.
- Sarwono J. 2012. Analisis Data Penelitian Menggunakan SPSS. ANDI. Yogyakarta
- Sidik B. 2006. *Pemrograman Web dengan PHP*. Bandung : Informatika.
- Soetam R. 2011. *Konsep Dasar Rekayasa Perangkat Lunak (Software Engineering)*. Jakarta : Prestasi Pustaka Publisher.
- Steinbach, M., G, Karypis and Vipin Kumar. 2000. *A Comparison Of Document Clustering Techniques*. University Of Minessota, Departemen of Computer Science and Engineering : Minessota.
- Tala, Z Fadhilah. 2003. *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. Netherlands : Universiteit van Amsterdam.
- Visser, W.T dan Wielling, M.B. 2007. *Sentence-Based Summarization of Scientific Documents, The Design and Implementation of an Online Available Automatic Summarizer*.
<http://home.hccnet.nl/m.b.wieling/files/wielingvisser05automaticsummarization.pdf>;
- Yates, Ricardo Baeza dan Neto, Berthier Ribiero. 1999. Modern Information Retrieval. New York : Addison Wesley.
- Jogiyanto. 2005. *Analisis dan Desain: Sistem Informasi Pendekatan Struktur Teori dan Praktik Aplikasi Bisnis*. ANDI - Yogyakarta.



LAMPIRAN A

Surat Keputusan Pembimbing





**KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI (UIN)
SYARIF HIDAYATULLAH JAKARTA
FAKULTAS SAINS DAN TEKNOLOGI**

Jl. Ir. H. Juanda No. 95 Ciputat 15412 Indonesia
Telp.: (62-21) 7493606, 7493547 Fax.: (62-21) 7493315

Email : uinjkt@cabi.net.id
Website : <http://www.fst.uinjkt.ac.id>

Nomor : Un.01/F9/PP.00.9/5149/2014
Lampiran : -
Perihal : Bimbingan Skripsi

Jakarta, 28 August 2014

Kepada Yth.

1. Hendra Bayu Suseno, M.Kom

2. Victor Amrizal, MKom

Dosen Pembimbing Skripsi

Assalamu'alaikum Wr.Wb.

Dengan ini diharapkan kesediaan Saudara untuk menjadi pembimbing I/II/ (Materi/Teknis)* penulisan skripsi mahasiswa:

Nama	:	Salamah
NIM	:	207091000734
Program Studi	:	Teknik Informatika
Judul Skripsi	:	“Penerapan Tokenisasi Kalimat Dan Metode TF (Term Frequency) Pada Peringkas Teks Otomatis Artikel Berita Berbahasa Indonesia”

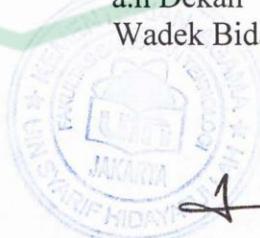
Judul tersebut telah disetujui oleh Program Studi bersangkutan pada tanggal 16 Januari 2014 dengan outline, abstraksi dan daftar pustaka terlampir. Bimbingan skripsi ini diharapkan selesai dalam waktu 6 (enam) bulan setelah ditandatanganinya surat penunjukan pembimbing skripsi.

Apabila terjadi perubahan terkait dengan skripsi tersebut selama proses pembimbingan, harap segera melaporkan kepada Program Studi bersangkutan.

Demikian atas kesediaan Saudara, kami ucapan terima kasih.

Wassalamu'alaikum Wr.Wb.

a.n Dekan
Wadek Bidang Akademik



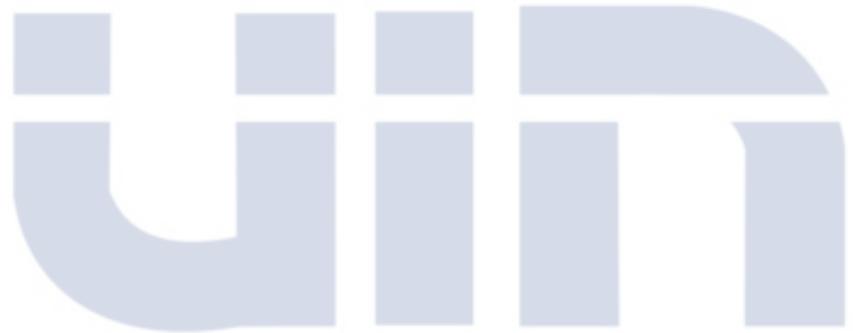
Lily Surayya EP, Dr., M.Env.Stud
NIP. 19690404200501 2 005

Tembusan:
Dekan (sebagai laporan)



LAMPIRAN B

Wawancara



WAWANCARA

Nara Sumber : Dzakiah Nurfadilah (Peneliti Dengan Tema Yang Sama)

Topik : Mempelajari Teknik Pemecahan Kalimat (Tokenizing) Dan Pembobotan.

Tanggal :

1. Tahapan *text mining* apa yang bisa memecah kalimat menjadi kata?

Jawab :

Tahapan *text mining* dalam preprocessing kata yang memecah kalimat menjadi kata adalah tokenisasi kalimat.

2. Bagaimana melakukan pembobotan terhadap kata yang dihasilkan dari proses tokenisasi?

Jawab :

Menghitung kemunculan frekuensi kata yang terdapat didalam artikel yang telah di input, sesuai dengan hasil token yang didapatkan.

Rangkuman : Berdasarkan hasil wawancara di atas, peneliti berinisiatif membuat aplikasi peringkas otomatis sederhana dengan tahap preprocessing yang singkat dan metode pembobotan hanya menghitung kemunculan kata dalam suatu term / artikel yang telah di input.



LAMPIRAN C

Daftar Kata Stopword

Daftar Kumpulan Stopword

a	apalagi	baiknya	berakhirnya
abad	apatah	bakal	berapa
acara	api	bakalan	berapakah
aceh	april	balik	berapalah
ada	artikel	bandar	berapapun
adalah	artinya	bangsa	berarti
adanya	as	bank	berasal
adapun	asal	banyak	berat
agak	asalkan	bapak	berawal
agaknya	asas	barang	berbagai
agama	asia	barangan	berbanding
agar	asing	barat	berbeda
agustus	atas	baru	berdasarkan
air	atau	baru-baru	berdatangan
akan	ataukah	bawah	berharap
akankah	ataupun	beberapa	berhasil
akhir	australia	begini	beri
akhiri	awal	beginian	berikan
akhirnya	awalnya	beginikah	berikut
akibat	awam	beginilah	berikutnya
aku	b	begitu	berita
akulah	badan	begitukah	berjalan
alam	bagai	begitulah	berjaya
album	bagaikan	begitupun	berjumlah
amat	bagaimana	bekas	berkaitan
amatlah	bagaimanakah	bekerja	berkali
amerika	bagaimanapun	belakang	berkali-kali
anak	bagainamakah	belakangan	berkata
and	bagi	belanda	berkehendak
anda	bagian	beli	berkeinginan
andalah	bahagian	beliau	berkenaan
anggota	bahan	belum	berlainan
antar	baharu	belumlah	berlaku
antara	bahasa	benar	berlalu
antarabangsa	bahawa	benarkah	berlangsung
antaranya	bahkan	benarlah	berlebihan
apa	bahwa	bentuk	bermacam
apaan	bahwasannya	berada	bermacam-
apabila	bahwasanya	berakhir	macam
apakah	baik	berakhirlah	

bermain
bermaksud
bermula
bernama
bernilai
bersama
bersama-sama
bersiap
bertanya
bertemu
berturut
bertutur
berubah
berujar
berupa
besar
besok
betul
betulkah
bhd
biasa
biasanya
bidang
bila
bilakah
bilion
bintang
bisa
bisakah
blog
bn
bola
boleh
bolehkah
bolehlah
buat
bukan
bukankah
bukanlah
bukannya

buku
bulan
bumi
bung
bursa
cadangan
cara
caranya
catch
china
click
code
copyright
cukup
cukupkah
cukuplah
cuma
daerah
dagangan
dahulu
dalam
dan
dana
dapat
dari
daripada
dasar
data
datang
datuk
dekat
demi
demikian
demikianlah
dengan
depan
derivatives
desa
desember
detik

dewan
di
dia
diadakan
diakhiri
diakhirinya
dialah
dianggap
diantara
diantaranya
diberi
diberikan
diberikannya
dibuat
dibuatnya
dibuka
dicatatkan
didapat
didatangkan
didirikan
diduga
digunakan
diibaratkan
diibaratkannya
diingat
diingatkan
diinginkan
dijangka
dijawab
dijelaskan
dijelaskannya
dikarenakan
dikatakan
dikatakannya
dikenal
dikerjakan
diketahui
diketahuinya
dikira
dilakukan

dilalui
dilihat
dimaksud
dimaksudkan
dimaksudkann
ya
dimaksudnya
dimana
diminta
dimintai
dimisalkan
dimulai
dimulailah
dimulainya
dimungkinkan
dini
diniagakan
dipastikan
diperbuat
diperbuatnya
dipergunakan
diperkirakan
diperlihatkan
diperlukan
diperlukannya
dipersoalkan
dipertanyakan
dipunyai
diri
dirilis
dirinya
dis
disampaikan
disebut
disebutkan
disebutkannya
disember
disini
disinilah
distrik

ditambahkan	err	ibaratkan	januari
ditandaskan	faedah	ibaratnya	jauh
ditanya	feb	ibu	jawa
ditanyai	film	ii	jawab
ditanyakan	gat	iklan	jawaban
ditegaskan	gedung	ikut	jawabnya
ditemukan	gelar	ilmu	jawatan
ditujukan	gettracker	indeks	jawatankuasa
ditunjuk	global	india	jelas
ditunjuki	grup	indonesia	jelaskan
ditunjukkan	guna	industri	jelaslah
ditunjukkanny	gunakan	informasi	jelasnya
a	gunung	ingat	jenis
ditunjuknya	hadap	inggris	jepang
ditutup	hadapan	ingin	jepun
dituturkan	hal	inginkah	jerman
dituturkannya	hampir	inginkan	jika
diucapkan	hanya	ini	jikalau
diucapkannya	hanyalah	inikah	jiwa
diungkapkan	harga	inilah	jual
document.writ	hari	internasional	jualan
e	harian	islam	juga
dolar	harus	isnin	julai
dong	haruslah	isu	jumaat
dr	harusnya	italia	jumat
dua	hasil	itu	jumlah
dulu	hendak	itukah	jumlahnya
dunia	hendaklah	itulah	jun
effective	hendaknya	jabatan	juni
ekonomi	hidup	jadi	justru
eksekutif	hingga	jadilah	juta
eksport	https	jadinya	kabar
empat	hubungan	jakarta	kabupaten
enam	hukum	jalan	kadar
enggak	hutan	jalur	kala
enggaknya	i	jaman	kalangan
entah	ia	jan	kalau
entahlah	iaitu	jangan	kalaualah
era	ialah	jangankan	kalaupun
eropa	ibarat	janganlah	kali

kalian	kelihatannya	kewangan	lamanya
kalimantan	kelima	khamis	langkah
kami	kelompok	khusus	langsung
kamilah	keluar	khususnya	lanjut
kamis	keluarga	kini	lanjutnya
kamu	kelurahan	kinilah	laporan
kamulah	kembali	kira	laut
kan	kementerian	kira-kira	lebih
kantor	kemudahan	kiranya	lembaga
kapal	kemudian	kita	lepas
kapan	kemungkinan	kitalah	lewat
kapankah	kemungkinann	klci	lima
kapanpun	ya	klibor	lingkungan
karena	kenaikan	klik	login
karenanya	kenapa	km	lokasi
karya	kenyataan	kok	lot
kasus	kepada	komentar	luar
kata	kepadanya	kompas	luas
katakan	kepala	komposit	lumpur
katakanlah	kepentingan	kondisi	mac
katanya	keputusan	kontrak	macam
kaunter	kerajaan	korban	mahkamah
kawasan	kerana	korea	mahu
ke	kereta	kos	majlis
keadaan	kerja	kota	maka
kebetulan	kerjasama	kuala	makanan
kebutuhan	kes	kuasa	makanya
kecamatan	kesampaian	kukuh	makin
kecil	keselamatan	kumpulan	maklumat
kedua	keseluruhan	kurang	malah
kedua-dua	keseluruhanny	kurangnya	malahan
keduanya	a	lagi	malam
kedudukan	kesempatan	lagian	malaysia
kegiatan	kesihatan	lagu	mampu
kehidupan	keterangan	lah	mampukah
keinginan	keterlaluan	lain	mana
kejadian	ketiga	lainnya	manakala
kekal	ketika	laku	manalagi
kelamaan	ketua	lalu	mantan
kelihatan	keuntungan	lama	manusia

masa	an	mengalami	menunjuk
masalah	memperkiraka	mengambil	menunjuki
masalahnya	n	mengapa	menunjukkan
masih	memperlihatka	mengatakan	menunjuknya
masihkah	n	mengatakanny	menurut
masing	mempersiapka	a	menuturkan
masing-	n	mengenai	menyaksikan
masing	mempersoalka	mengerjakan	menyampaika
masuk	n	mengetahui	n
masyarakat	mempertanyak	menggalakkan	menyangkut
mata	an	menggunakan	menyatakan
mau	mempunyai	menghadapi	menyebabkan
maupun	mulai	menghendaki	menyebutkan
measure	memungkinka	mengibaratkan	menyediakan
media	n	mengibaratkan	menyeluruh
mei	menaiki	nya	menyiapkan
melainkan	menambah	mengikut	merasa
melakukan	menambahkan	mengingat	mereka
melalui	menandaskan	mengingatkan	merekalah
melawan	menanti	menginginkan	erosot
melihat	menantikan	mengira	merupakan
melihatnya	menanya	mengucapkan	meski
memandangka	menanyai	mengucapkan	meskipun
n	menanyakan	nya	mesyuarat
memang	menarik	mengumumka	metrov
memastikan	menawarkan	n	meyakini
membantu	mencapai	mengungkapk	meyakinkan
membawa	mencari	an	milik
memberi	mencatatkan	mengurangkan	militer
memberikan	mendapat	meninggal	minat
membolehkan	mendapatkan	meningkat	minggu
membuat	mendatang	meningkatkan	minta
memerlukan	mendatangi	jadi	minyak
memihak	mendatangkan	menjalani	mirip
memiliki	menegaskan	menjawab	misal
meminta	menerima	menjelang	misalkan
memintakan	menerusi	menjelaskan	misalnya
memisalkan	mengadakan	menokok	mobil
memperbuat	mengakhiri	menteri	modal
mempergunak	mengaku	menuju	mohd

mudah	orang	pemerintah	pertandingan
mula	organisasi	pemerintahan	pertanyaan
mulai	pada	pemimpin	pertanyakan
mulailah	padahal	pendapatan	pertubuhan
mulanya	padanya	pendidikan	pertumbuhan
muncul	pagetracker	penduduk	perubahan
mungkin	pagi	penerbangan	perusahaan
mungkinkah	pak	pengarah	pesawat
musik	paling	pengeluaran	peserta
musim	pameran	pengerusi	petang
nah	panjang	pengguna	pihak
naik	pantas	penggunaan	pihaknya
nama	papan	pengurusan	pilihan
namun	para	peniaga	pinjaman
nanti	paras	peningkatan	polis
nantinya	parlimen	penting	polisi
nasional	partai	pentingnya	politik
negara	parti	per	pos
negara-negara	particle	perancis	posisi
negeri	pasar	perang	presiden
new	pasaran	peratus	prestasi
niaga	password	percuma	produk
nilai	pasti	perdagangan	program
nomor	pastilah	perdana	projek
noun	pasukan	peringkat	pronomia
nov	paticle	perjanjian	pronoun
november	pegawai	perkara	proses
numeral	pejabat	perkhidmatan	proton
numeralia	pekan	perlادangan	provinsi
nya	pekerja	perlu	pt
nyaris	pelabur	perlukah	pubdate
nyatanya	pelaburan	perlunya	pukul
of	pelancongan	permintaan	pula
ogos	pelanggan	pernah	pulau
okt	pelbagai	perniagaan	pun
oktober	peluang	persekituan	punya
olah	pemain	persen	pusat
oleh	pembangunan	persidangan	rabu
olehnya	pemberita	persoalan	radio
operasi	pembinaan	pertama	raja

rakan	saya	segera	selatan
rakyat	sayalah	segi	selepas
ramai	sdn	seharusnya	seluruh
rantau	se	sehingga	seluruhnya
rasa	sebab	seingat	semacam
rasanya	sebabnya	sejak	semakin
rata	sebagai	sejarah	semalam
raya	sebagaimana	sejauh	semampu
rendah	sebagainya	sejenak	semampunya
republik	sebagian	sejumlah	semasa
resmi	sebahagian	sekadar	semasih
ribu	sebaik	sekadarnya	semata
ringgit	sebaiknya	sekali	semaunya
root	sebaliknya	sekali-kali	sementara
ruang	sebanyak	sekalian	semisal
rumah	sebarang	sekaligus	semisalnya
rupa	sebegini	sekalipun	sempat
rupanya	sebegitu	sekarang	semua
saat	sebelah	sekaranglah	semuanya
saatnya	sebelum	sekecil	semula
sabah	sebelumnya	seketika	sen
sabtu	sebenarnya	sekiranya	sendiri
sahaja	seberapa	sekitar	sendirian
saham	sebesar	sekitarnya	sendirinya
saja	sebetulnya	sekolah	senin
sajalah	sebisanya	sektor	seolah
sakit	sebuah	sekurang	seolah-olah
salah	sebut	sekurangnya	seorang
saling	sebutlah	sekuriti	sepak
sama	sebutnya	sela	sepanjang
sama-sama	secara	selagi	sepantasnya
sambil	secukupnya	selain	sepantasnyala
sampai	sedang	selaku	h
sampaikan	sedangkan	selalu	seperlunya
sana	sedemikian	selama	seperti
sangat	sedikit	selama-	sepertinya
sangatlah	sedikitnya	lamanya	sepikhak
sarawak	seenaknya	selamanya	sept
satu	segala	selanjutnya	september
sawit	segalanya	selasa	serangan

serantau	sinilah	tanyakan	terkira
seri	sistem	tanyanya	terlalu
serikat	soal	tapi	terlebih
sering	soalnya	tawaran	terletak
seringnya	sokongan	tegas	terlihat
serta	sri	tegasnya	termasuk
serupa	stasiun	teknologi	ternyata
sesaat	suara	telah	tersampaikan
sesama	suatu	televisi	tersebut
sesampai	sudah	teman	tersebutlah
sesegera	sudahkah	tempat	tertentu
sesekali	sudahlah	tempatan	tertuju
seseorang	sukan	tempo	terus
sesi	suku	tempoh	terutama
sesuai	sumber	tenaga	testimoni
sesuatu	sungai	tengah	testimony
sesuatunya	supaya	tentang	tetap
sesudah	surat	tentara	tetapi
sesudahnya	susut	tentu	the
setelah	syarikat	tentulah	tiada
setempat	syed	tentunya	tiap
setengah	tadi	tepat	tiba
seterusnya	tadinya	terakhir	tidak
setiap	tahap	terasa	tidakkah
setiausaha	tahu	terbaik	tidaklah
setiba	tahun	terbang	tidaknya
setibanya	tak	terbanyak	tiga
setidak	tama	terbesar	tim
setidaknya	tambah	terbuka	timbalan
settinggi	tambahnya	terdahulu	timur
seusai	tampak	terdapat	tindakan
sewaktu	tampaknya	terdiri	tinggal
siap	tampil	terhadap	tinggi
siapa	tan	terhadapnya	tingkat
siapakah	tanah	teringat	toh
siapapun	tandas	terjadi	tokoh
siaran	tandasnya	terjadilah	try
sidang	tanggal	terjadinya	tun
singapura	tanpa	terkait	tunai
sini	tanya	terkenal	tunjuk

turun
turut
tutur
tuturnya
tv
uang
ucap
ucapnya
udara
ujar
ujarnya
umum
umumnya
unescape
ungkap
ungkapnya
unit
universitas
untuk
untung
upaya
urus
usah
usaha
usai
user
utama
utara
var
versi
waduh
wah
wahai
wakil
waktu
waktunya
walau
walaupun
wang
wanita

warga
warta
wib
wilayah
wong
word
ya
yaitu
yakin
yakni
yang
zaman



KUISIONER

PENERAPAN TERM FREQUENCY- INVERSE DOCUMENT

FREQUENCY DAN ALGORITMA STEMMING PORTER PADA

PERINGKAS OTOMATIS ARTIKEL BERITA BERBAHASA INDONESIA

Kuisisioner ini merupakan bagian dari tahapan analisis penyusunan Skripsi Mahasiswa program S1 Jurusan Teknik Informatika Fakultas Sains Dan Teknologi UIN Syarif Hidayatullah Jakarta. Kuisisioner ini bertujuan untuk memperoleh data mengenai tingkat pemahaman pengguna terhadap hasil ringkasan sistem peringkas otomatis artikel berita berbahasa Indonesia.

Untuk kebutuhan di atas, mohon kiranya Bapak/Ibu, Saudara/I sebagai responden dapat memberikan pilihan sebagai jawaban atas pernyataan yang diberikan dalam kuisisioner ini untuk kemudian dapat diolah dalam proyek evaluasi ini.

Data Responden		
Nama		
Jenis Kelamin / Umur	a. Pria	b. Wanita / thn
Status		

PETUNJUK PENGISIAN

Responden dapat memilih salah satu jawaban yang dianggap tepat dengan memberikan tanda (✓) pada tempat yang tersedia.

Apakah hasil ringkasan dibawah ini dapat dipahami oleh anda?

Judul	Hasil Ringkasan 10%	Ya	Tidak
Debat Capres Ditutup dengan Pantun Mawar	Tidak seperti debat-debat sebelumnya, debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara.		
Stok Melimpah, Kebutuhan Daging di Jember Aman hingga Lebaran	Tetapi stok daging yang ada kami sudah pastikan cukup hingga lebaran, bahkan hingga akhir tahun nanti," kata kasi informasi pasar dan pengembangan investasi Disnakkankel Jember, Moch Chotib, Minggu (6/7/2014).		
Harga BBM Non-subsidi Mulai Naik	Pertamax naik Rp 350 per liter, sementara harga Pertamax Plus dengan oktan 95 hanya naik Rp 300 per liter. Sementara itu, Communication Manager PT Shell Indonesia, Inggit Notosusanto mengungkapkan, pihaknya tidak menaikkan harga BBM jenis Super.		
Hatta dan JK Sepakat Investigasi Mafia Migas	Tema yang diangkat yakni terkait pangan, energi, dan lingkungan hidup.		

Jokowi: Mau Tambah Sawah Perlu, Airnya dari Mana?	Sebelum membangun sawah baru, maka sumber air harus terlebih dahulu dibangun.	
Mentan: Harga Cabai Jatuh, Petani Menangis	<p>Sebab, tingginya konsumsi cabai segar malah membuat harga menjadi fluktuatif.</p> <p>Menurut dia, kondisi harga cabai saat ini secara langsung atau tidak langsung karena pedagang yang mungkin saja mengambil keuntungan.</p>	
2013, Jakarta Raih 4 Adipura	Adipura merupakan penghargaan yang diberikan pemerintah kepada pemerintah kota/kabupaten yang memiliki komitmen dalam mewujudkan kota yang bersih dan hijau (clean and green city).	
Roy Suryo: Kami Merasa Dikambinghitamkan atas Molornya MRT	Menurut Undang-Undang Sistem Keolahragaan Nasional, ada empat syarat yang harus dipenuhi untuk membongkar stadion, yakni adanya lahan pengganti tempat yang dibongkar, lahan penggantinya harus memiliki sertifikat yang jelas, sertifikat lahan pengganti tidak boleh bermasalah, dan adanya rencana anggaran untuk pembangunan lahan pengganti.	

Dengan e-KTP, Buka Rekening Bank Bisa di Mana Saja	Kabar baik bagi Anda yang kini berdomisili di daerah yang berbeda dengan alamat kartu tanda penduduk (KTP). Aturan tersebut mewajibkan setiap nasabah harus menyertakan dokumen tambahan jika alamat tinggal saat ini berbeda dengan data yang ada di dalam KTP.	
Argentina Kelelahan, Jerman Luar Biasa	"Pertandingan melawan Jerman akan sangat sulit dan saya ingatkan fakta bahwa mereka tak pernah memainkan babak tambahan dan kami telah memainkan dua babak tambahan.	

Judul	Hasil Ringkasan 20%	Ya	Tidak
Debat Capres Ditutup dengan Pantun Mawar	Tidak seperti debat-debat sebelumnya, debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Debat kelima ini merupakan debat terakhir dalam rangkaian debat capres-cawapres. Debat tersebut menutup rangkaian kampanye Pemilu Presiden 2014 sebelum memasuki masa tenang.		
Stok Melimpah, Kebutuhan Daging di Jember Aman hingga Lebaran	Tetapi stok daging yang ada kami sudah pastikan cukup hingga lebaran, bahkan hingga akhir tahun nanti," kata kasi informasi pasar dan pengembangan investasi Disnakkankel Jember, Moch Chotib, Minggu		

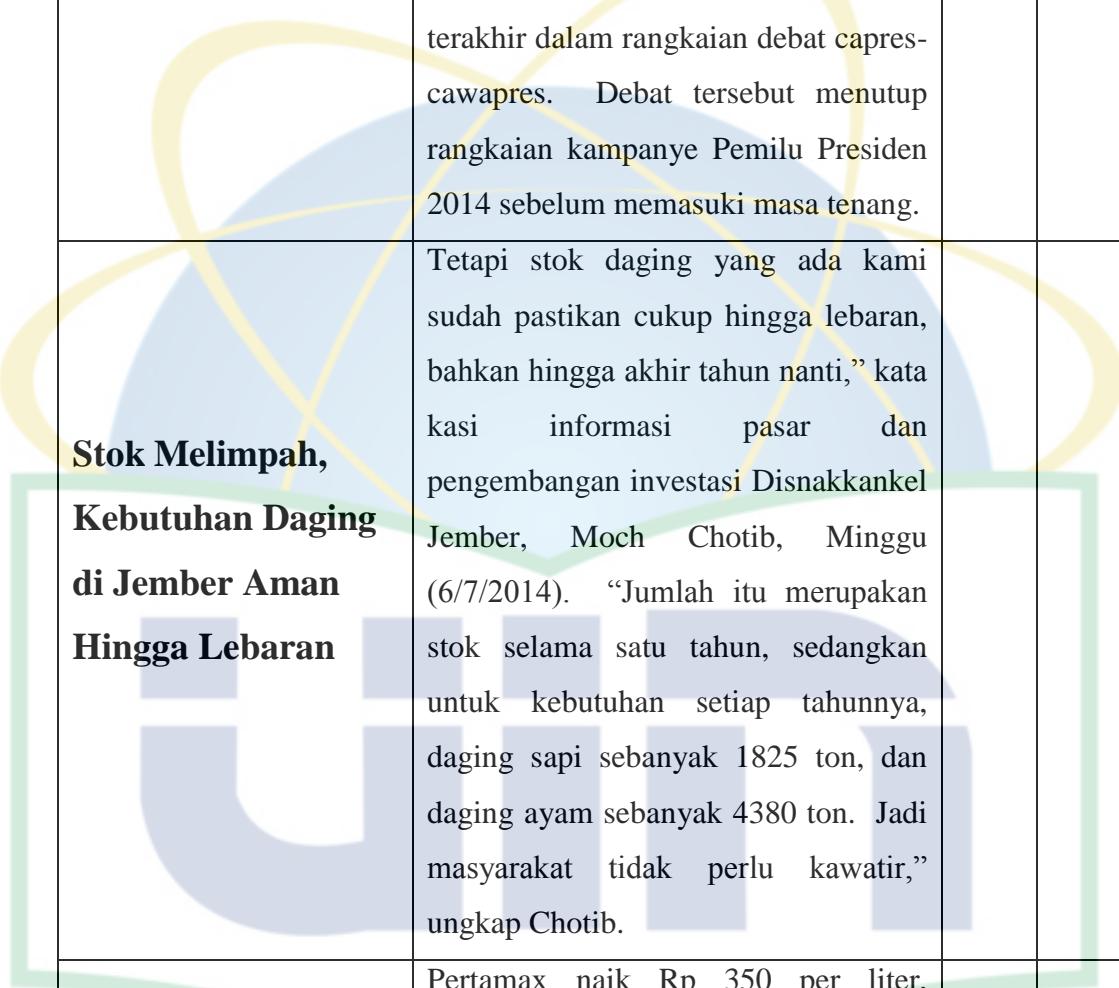
	(6/7/2014). "Jumlah itu merupakan stok selama satu tahun, sedangkan untuk kebutuhan setiap tahunnya, daging sapi sebanyak 1825 ton, dan daging ayam sebanyak 4380 ton.	
Harga BBM Non-Subsidi Mulai Naik	Pertamax naik Rp 350 per liter, sementara harga Pertamax Plus dengan oktan 95 hanya naik Rp 300 per liter. "Kami mengacu pada dua minggu sebelumnya, kemudian seminggu ini kami naik harga mengikuti harga tukar rupiah ke dollar AS yang meningkat," jelas Iskandar kepada KONTAN, Minggu (6/7/2014). Sebab pada saat itu nilai tukar rupiah terhadap dollar akan stabil karena tak banyak aktivitas ekonomi karena libur panjang Lebaran. Sementara itu, Communication Manager PT Shell Indonesia, Inggita Notosusanto mengungkapkan, pihaknya tidak menaikkan harga BBM jenis Super.	
Hatta dan JK Sepakat Investigasi Mafia Migas	"Pak JK, tadi banyak masalah mafia migas, setuju kah bapak dengan pandangan bapak bahwa setiap perpanjangan kontrak, itu apa perlu dilakukan investasi atas perpanjangan itu agar dilakukan pengecekan ulang," kata Hatta. Tema yang diangkat yakni terkait pangan, energi, dan lingkungan	

	hidup.		
Jokowi: Mau Tambah Sawah Perlu, Airnya dari Mana?	<p>Apa Bapak Jokowi sependapat ekstensi lahan harus besar-besaran dan berani tambah lahan hingga 2 juta hektar?" tanya Prabowo, Sabtu (5/7/2014). Sebelum membangun sawah baru, maka sumber air harus terlebih dahulu dibangun.</p>		
Harga Cabai Jatuh, Petani Menangis	<p>Kenapa bisa sampai 4 hingga 5 kali lipat," kata Suswono di Kantor Pusat Bank Indonesia (BI), Jumat (4/7/2014). Sebab, tingginya konsumsi cabai segar malah membuat harga menjadi fluktuatif. "Masyarakat juga perlu dididik agar tidak perlu selalu makan cabe segar.</p> <p>Menurut dia, kondisi harga cabai saat ini secara langsung atau tidak langsung karena pedagang yang mungkin saja mengambil keuntungan.</p>		
2013, Jakarta Raih 4 Adipura	<p>Adipura merupakan penghargaan yang diberikan pemerintah kepada pemerintah kota/kabupaten yang memiliki komitmen dalam mewujudkan kota yang bersih dan hijau (clean and green city). Saat menyampaikan pertanyaannya, Hatta mengatakan bahwa Kalpataru merupakan salah satu penghargaan tertinggi yang banyak diinginkan kota-</p>		

	<p>kota.</p> <p>Jokowi juga menilai alangkah lebih baik jika tidak hanya diberikan penghargaan berupa piala, tetapi juga insentif dalam bentuk dana atau anggaran.</p>	
<p>Roy Suryo: Kami Merasa Dikambinghitamkan atas Molornya MRT</p> 	<p>Menurut Undang-Undang Sistem Keolahragaan Nasional, ada empat syarat yang harus dipenuhi untuk membongkar stadion, yakni adanya lahan pengganti tempat yang dibongkar, lahan penggantinya harus memiliki sertifikat yang jelas, sertifikat lahan pengganti tidak boleh bermasalah, dan adanya rencana anggaran untuk pembangunan lahan pengganti.</p> <p>Roy menyatakan bahwa lahan yang disebut Pemprov DKI sebagai Taman BMW adalah tanah sengketa karena perizinannya bermasalah. Kita tidak perlu menanggapi statement-nya BTP (Basuki) yang dilemparkan di berbagai media.</p>	
<p>Dengan e-KTP, Buka Rekening Bank Bisa di Mana Saja</p>	<p>Kabar baik bagi Anda yang kini berdomisili di daerah yang berbeda dengan alamat kartu tanda penduduk (KTP). Mulai sekarang, Anda bebas membuka rekening baru di kantor cabang bank manapun. "Diharapkan</p>	

	<p>tidak ada nasabah yang punya banyak identitas," ujar Nelson, Jumat (4/7/2014). Direktur Bisnis Konsumen Bank Rakyat Indonesia (BRI) Toni Soetirto, mengungkapkan, sejak Desember tahun lalu, BRI telah memanfaatkan e-KTP sebagai sumber data untuk pembukaan rekening lewat hybrid lounge. Aturan tersebut mewajibkan setiap nasabah harus menyertakan dokumen tambahan jika alamat tinggal saat ini berbeda dengan data yang ada di dalam KTP.</p>	
<p>Argentina Kelelahan, Jerman Luar Biasa</p>	<p>Namun, dengan kerja keras, kerendahan hati, dan keseriusan, kami akan melakukan yang bisa kami lakukan untuk sampai ke puncak," ujar pelatih Argentina, Alejandro Sabella. "Pertandingan melawan Jerman akan sangat sulit dan saya ingatkan fakta bahwa mereka tak pernah memainkan babak tambahan dan kami telah memainkan dua babak tambahan.</p>	

Judul	Hasil Ringkasan 30%	Ya	Tidak
<p>Debat Capres Ditutup dengan Pantun Mawar</p>	<p>Tidak seperti debat-debat sebelumnya, debat calon presiden dan wakil presiden periode terakhir diwarnai hal berbeda dari awal hingga akhir acara. Acara kemudian dibuka oleh Ketua</p>		

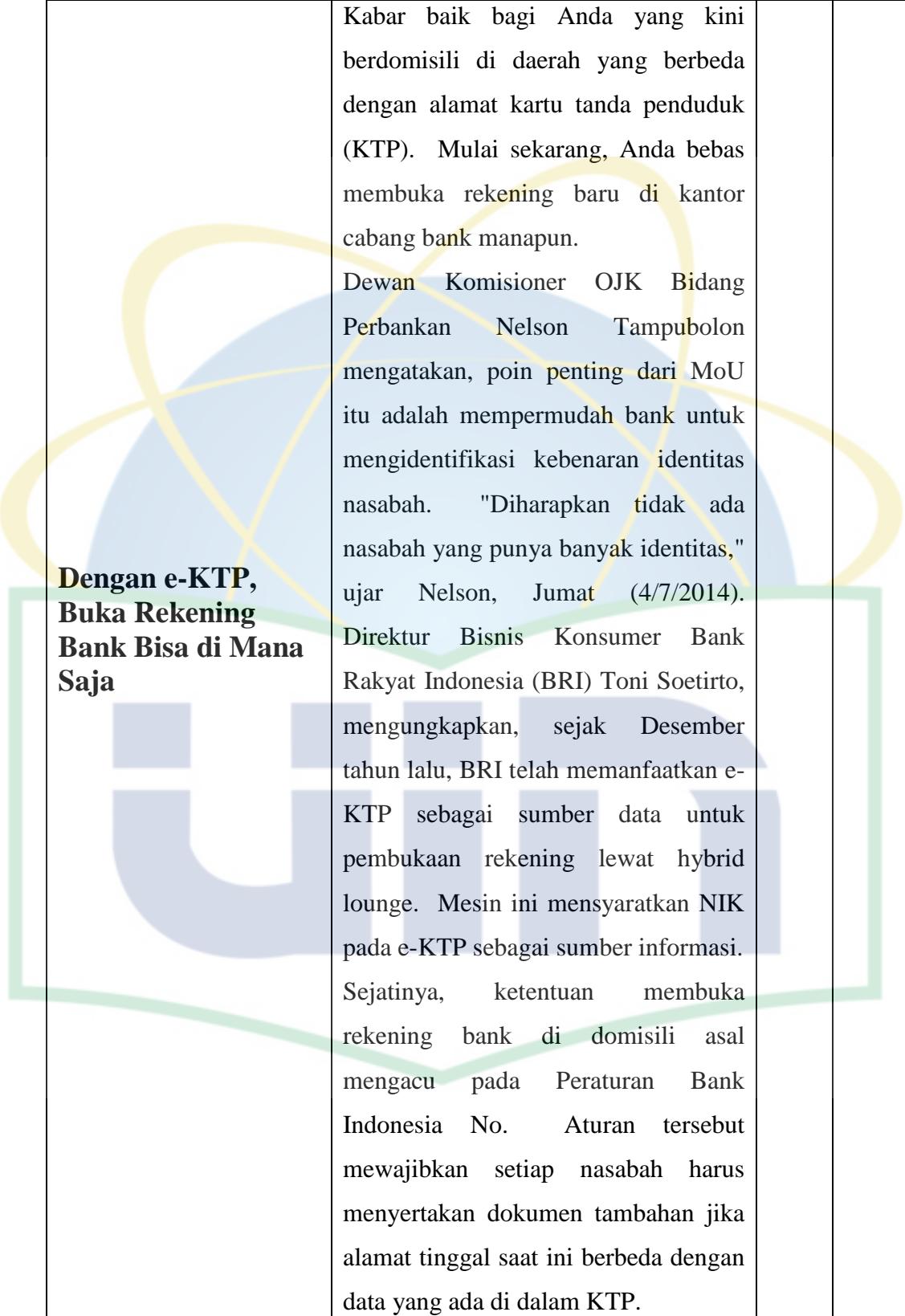
	<p>Umum Komisi Pemilihan Umum Husni Malik. Sudharto menjalankan perannya sebagai moderator dengan baik hingga debat selesai.</p> <p>Debat kelima ini merupakan debat terakhir dalam rangkaian debat capres-cawapres. Debat tersebut menutup rangkaian kampanye Pemilu Presiden 2014 sebelum memasuki masa tenang.</p>	
<p>Stok Melimpah, Kebutuhan Daging di Jember Aman Hingga Lebaran</p> 	<p>Tetapi stok daging yang ada kami sudah pastikan cukup hingga lebaran, bahkan hingga akhir tahun nanti," kata kasi informasi pasar dan pengembangan investasi Disnakkankel Jember, Moch Chotib, Minggu (6/7/2014). "Jumlah itu merupakan stok selama satu tahun, sedangkan untuk kebutuhan setiap tahunnya, daging sapi sebanyak 1825 ton, dan daging ayam sebanyak 4380 ton. Jadi masyarakat tidak perlu kawatir," ungkap Chotib.</p>	
<p>Harga BBM Non-Subsidi Mulai Naik</p>	<p>Pertamax naik Rp 350 per liter, sementara harga Pertamax Plus dengan oktan 95 hanya naik Rp 300 per liter. 300 per liter dari sebelumnya hanya Rp 10.</p> <p>Vice President Fuel Retail Marketing PT Pertamina Muhammad Iskandar membenarkan adanya kenaikan harga</p>	

	<p>tersebut.</p> <p>"Kami mengacu pada dua minggu sebelumnya, kemudian seminggu ini kami naik harga mengikuti harga tukar rupiah ke dollar AS yang meningkat," jelas Iskandar kepada KONTAN, Minggu (6/7/2014). Sebab pada saat itu nilai tukar rupiah terhadap dollar akan stabil karena tak banyak aktivitas ekonomi karena libur panjang Lebaran.</p> <p>Meski harga Pertamax dan Pertamax Plus naik, Pertamina berharap masyarakat tetap membeli BBM non-subsidi.</p> <p>Sementara itu, Communication Manager PT Shell Indonesia, Inggit Notosusanto mengungkapkan, pihaknya tidak menaikkan harga BBM jenis Super.</p>	
<p>Hatta dan JK Sepakat Investigasi Mafia Migas</p>	<p>Dua calon wakil presiden yang tengah berkompetisi yakni Hatta Rajasa dan Jusuf Kalla sama-sama sepakat perlunya dilakukan investigasi mafia migas dalam proses perpanjangan kontrak karya perusahaan-perusahaan migas asing di Indonesia.</p> <p>"Pak JK, tadi banyak masalah mafia migas, setuju kah bapak dengan pandangan bapak bahwa setiap perpanjangan kontrak, itu apa perlu</p>	

	dilakukan investasi atas perpanjangan itu agar dilakukan pengecekan ulang," kata Hatta. Tema yang diangkat yakni terkait pangan, energi, dan lingkungan hidup. Rektor Universitas Diponegoro Prof Prof Sudharto P Hadi menjadi moderator dalam debat kali ini.	
Jokowi: Mau Tambah Sawah Perlu, Airnya dari Mana?	"Kami punya program menambah 2 juta hektar lahan untuk pangan. Apa Bapak Jokowi sependapat ekstensi lahan harus besar-besaran dan berani tambah lahan hingga 2 juta hektar?" tanya Prabowo, Sabtu (5/7/2014). Sebelum membangun sawah baru, maka sumber air harus terlebih dahulu dibangun. Sekarang di Papua sudah dibuka tapi dibiarkan begitu saja karena tidak pernah dipikir airnya dimana, bendungannya dimana," ujar dia.	
Harga Cabai Jatuh, Petani Menangis	Kenapa bisa sampai 4 hingga 5 kali lipat," kata Suswono di Kantor Pusat Bank Indonesia (BI), Jumat (4/7/2014). Suswono mengatakan, dengan kondisi seperti ini, pihak yang menikmati untung justru adalah pedagang. Sebab, tingginya konsumsi cabai segar malah membuat harga menjadi fluktuatif. "Masyarakat juga perlu	

	<p>dididik agar tidak perlu selalu makan cabe segar.</p> <p>Menurut dia, kondisi harga cabai saat ini secara langsung atau tidak langsung karena pedagang yang mungkin saja mengambil keuntungan. Sebab saat ini pasokan cabai sedang melimpah pasca panen raya.</p>		
<p>Jakarta Raih 4 Adipura</p>	<p>Adipura merupakan penghargaan yang diberikan pemerintah kepada pemerintah kota/kabupaten yang memiliki komitmen dalam mewujudkan kota yang bersih dan hijau (clean and green city). Prestasi ini diberitakan secara masif oleh media-media di Indonesia. Saat menyampaikan pertanyaannya, Hatta mengatakan bahwa Kalpataru merupakan salah satu penghargaan tertinggi yang banyak diinginkan kota-kota.</p> <p>Jokowi juga menilai alangkah lebih baik jika tidak hanya diberikan penghargaan berupa piala, tetapi juga insentif dalam bentuk dana atau anggaran.</p> <p>Setelah itu, Jokowi menjawab bahwa Solo pernah mendapatkan penghargaan sebagai kota bersih atau clean city dari Kementerian Lingkungan Hidup dan Kementerian</p>		

	Kehutanan.		
Roy Suryo: Kami Merasa Di kambing Hitamkan atas Molornya MRT	<p>Menurut Roy, Pemprov DKI menganggap Kemenpora sebagai dalang terhambatnya pembangunan MRT itu karena tidak kunjung membongkar Stadion Lebak Bulus yang akan dijadikan depo MRT. Menurut Undang-Undang Sistem Keolahragaan Nasional, ada empat syarat yang harus dipenuhi untuk membongkar stadion, yakni adanya lahan pengganti tempat yang dibongkar, lahan penggantinya harus memiliki sertifikat yang jelas, sertifikat lahan pengganti tidak boleh bermasalah, dan adanya rencana anggaran untuk pembangunan lahan pengganti.</p> <p>Roy menyatakan bahwa lahan yang disebut Pemprov DKI sebagai Taman BMW adalah tanah sengketa karena perizinannya bermasalah.</p> <p>"Kalau yang bersangkutan (Basuki) pernah mengatakan 'Ini bukan Hambalang, bos', malah kalau kami tidak melapor itu bisa jadi berpotensi seperti kasus Hambalang," ujarnya. Kita tidak perlu menanggapi statement-nya BTP (Basuki) yang dilemparkan di berbagai media.</p>		

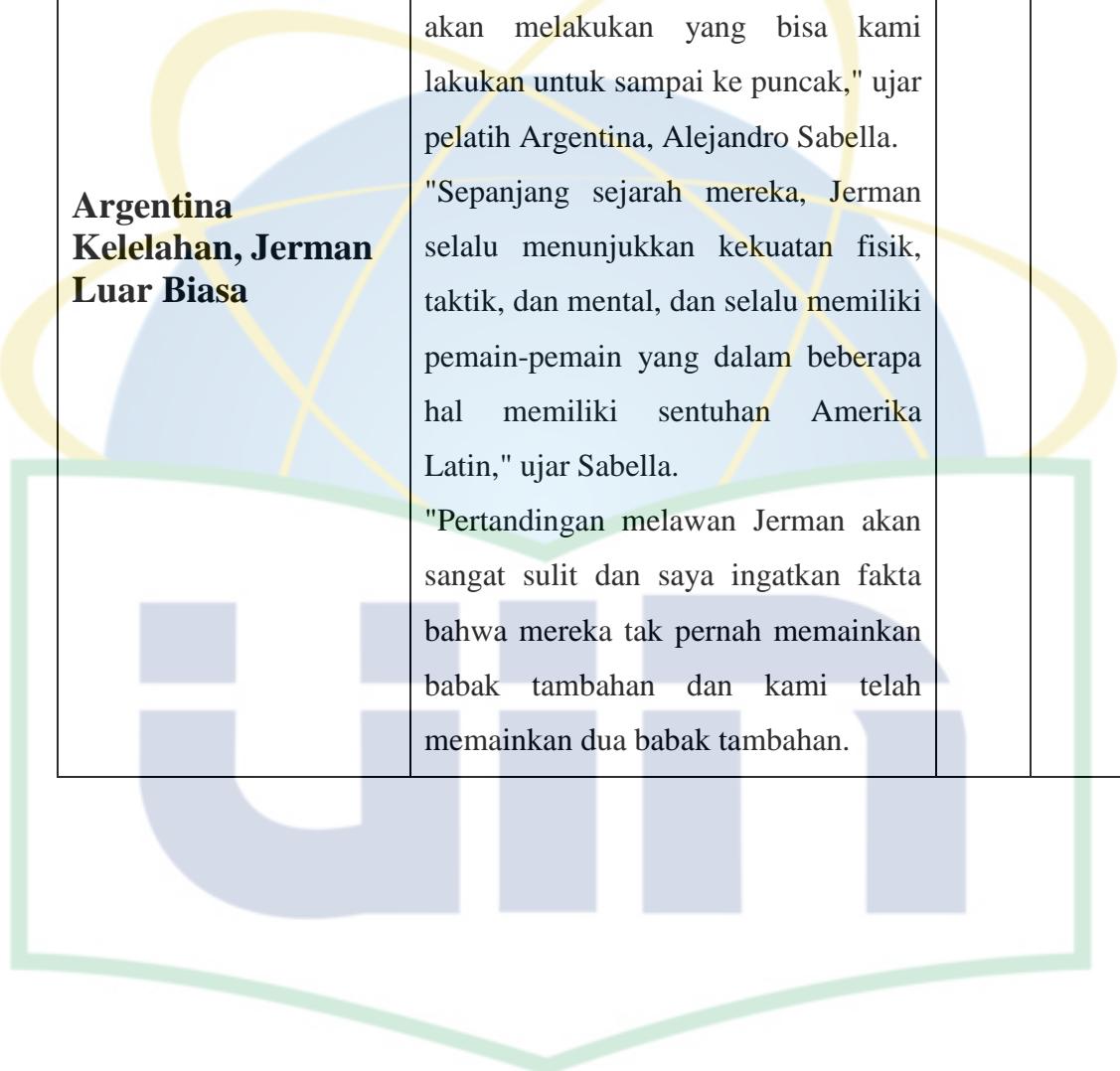


Dengan e-KTP, Buka Rekening Bank Bisa di Mana Saja

Kabar baik bagi Anda yang kini berdomisili di daerah yang berbeda dengan alamat kartu tanda penduduk (KTP). Mulai sekarang, Anda bebas membuka rekening baru di kantor cabang bank manapun.

Dewan Komisioner OJK Bidang Perbankan Nelson Tampubolon mengatakan, poin penting dari MoU itu adalah mempermudah bank untuk mengidentifikasi kebenaran identitas nasabah. "Diharapkan tidak ada nasabah yang punya banyak identitas," ujar Nelson, Jumat (4/7/2014).

Direktur Bisnis Konsumen Bank Rakyat Indonesia (BRI) Toni Soetirto, mengungkapkan, sejak Desember tahun lalu, BRI telah memanfaatkan e-KTP sebagai sumber data untuk pembukaan rekening lewat hybrid lounge. Mesin ini mensyaratkan NIK pada e-KTP sebagai sumber informasi. Sejatinya, ketentuan membuka rekening bank di domisili asal mengacu pada Peraturan Bank Indonesia No. Aturan tersebut mewajibkan setiap nasabah harus menyertakan dokumen tambahan jika alamat tinggal saat ini berbeda dengan data yang ada di dalam KTP.

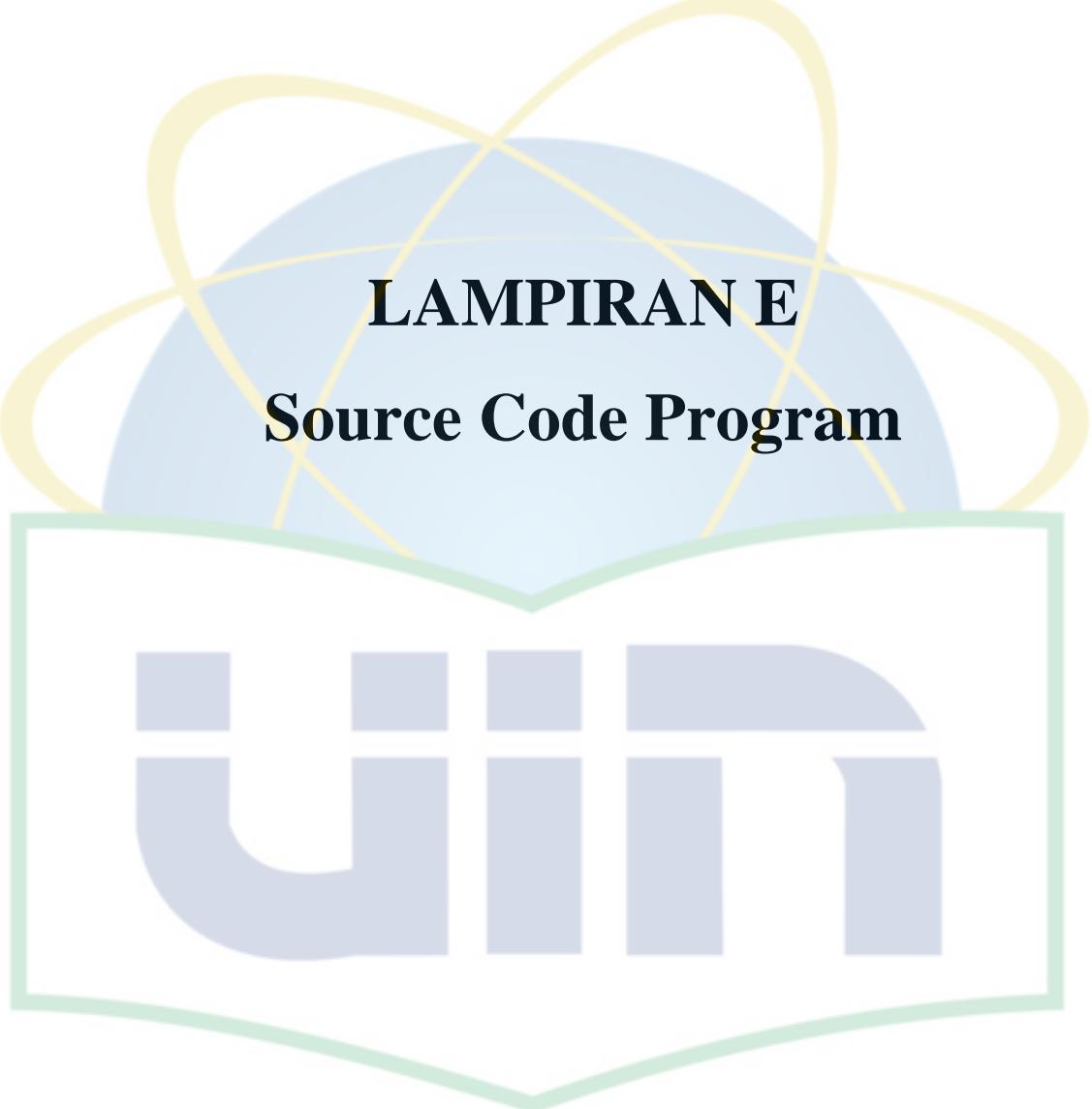


Argentina Kelelahan, Jerman Luar Biasa

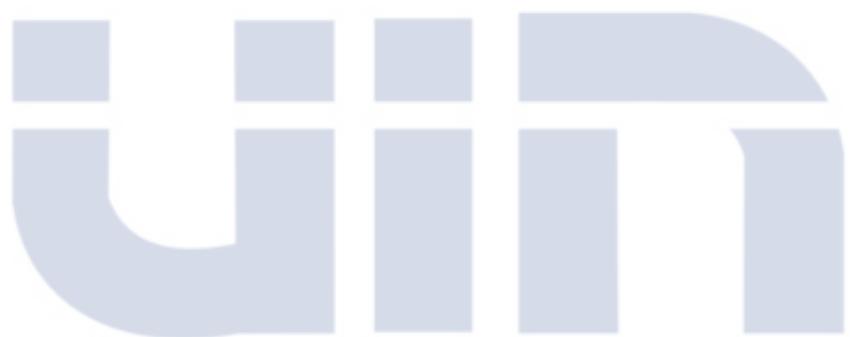
Kami masih punya satu pertandingan, dengan masa persiapan satu hari lebih pendek dan melawan tim seperti Jerman. Namun, dengan kerja keras, kerendahan hati, dan keseriusan, kami akan melakukan yang bisa kami lakukan untuk sampai ke puncak," ujar pelatih Argentina, Alejandro Sabella.

"Sepanjang sejarah mereka, Jerman selalu menunjukkan kekuatan fisik, taktik, dan mental, dan selalu memiliki pemain-pemain yang dalam beberapa hal memiliki sentuhan Amerika Latin," ujar Sabella.

"Pertandingan melawan Jerman akan sangat sulit dan saya ingatkan fakta bahwa mereka tak pernah memainkan babak tambahan dan kami telah memainkan dua babak tambahan.



LAMPIRAN E
Source Code Program



KODE PROGRAM

index.php

```
<html>
    <head>
        <title>Peringkas Teks</title>
        <!-- Bootstrap -->
        <link href=".//css/bootstrap.min.css" rel="stylesheet" media="screen">
        <link href=".//css/bootstrap-responsive.min.css" rel="stylesheet" media="screen">
        <link href=".//css/bootswatch.css" rel="stylesheet" media="screen">
        <script type="text/javascript">
            function printValue(sliderID, textbox) {
                //alert('jalan');
                var x = document.getElementById(textbox);
                var y = document.getElementById(sliderID);
                x.value = y.value;
            }
        </script>
    </head>
    <body>
        <?php
            error_reporting(E_ERROR | E_PARSE);
            include ".//summarize.php";
            // scan nama file korpus
            $dir_corpus = ".//corpus";
            $files      = scandir($dir_corpus);
            $files      = array_slice($files, 2);
            // hasil
            if(isset($_REQUEST["txtasli"]) &&
            isset($_REQUEST["compression"])){
                $filename   = $_REQUEST["txtasli"];
                $compression = $_REQUEST["compression"];
                $output     = summarize($filename, $compression);
                $title      = substr($filename, 0, -4);
            }
        ?>
        <div class="navbar navbar-inverse navbar-fixed-top">
            <div class="navbar-inner">
                <div class="container-fluid">
```

```
<a class="brand" href="index.php">Auto  
Text Summarization</a>  
<div class="nav-collapse collapse">  
    <ul class="nav">  
<li><a class="brand" style="margin-left:85px;"><?php// echo $title;?></a></li>  
    </ul>  
</div>  
</div>  
<div class="container-fluid">  
    <div class="row-fluid">  
        <div class="span1" style="display:none;">  
            <!--<div class="well sidebar-nav">  
                <ul class="nav nav-list">  
                    <form action="" name="frm" method="post">  
                        <li class="nav-header">Select File</li>  
                        <li>  
                            <textarea name="txtasli" id="txtasli" class="input-large" cols="60" rows="15"><?php echo !empty($output['original'])? $output['original'] : "";?></textarea>  
                        </li>  
                        <li class="nav-header">Set Compression Rate (%)</li>  
                        <li>  
                            <table border="0">  
                                <tr>  
                                    <td><input type="range" id="slider" min="1" max="100" value="<?php echo !empty($compression)? $compression : "?>" step="1" style="width:100%;" onChange="printValue('slider','rangeValue')"/></td>  
                                    <td><input type="text" id="rangeValue" name="compression" value="<?php echo !empty($compression)? $compression : '50';?>" style="width:35px;" /></td>  
                                </tr>  
                            </table>  
                        </li>  
                    <li class="nav-header">Summarize</li>  
                    <li><input class="btn btn-primary" type="submit" value="Go!" style="float: right;"></li>  
                    <li>&nbsp;</li>  
                </form>  
            </ul>  
        </div>--><!--.well -->  
    </div><!--span-->  
    <div class="span11">
```

```

<div class="row-fluid" style="margin-top:0px;">
    <form action="" name="frm" method="post">
        <div class="span6">
            <h2>Teks Asli</h2>
            <textarea name="txtasli" id="txtasli" cols="60" rows="12" class="input-xxlarge"><?php echo !empty($output['original'])? $output['original'] : "";"?></textarea>
        </div><!--/span-->
        <div class="span6">
            <h2>Hasil Ringkasan</h2>
            <textarea name="txthasil" id="txthasil" cols="60" rows="12" class="input-xxlarge"><?php echo !empty($output['summary'])? $output['summary'] : "";"?></textarea>
        </div><!--/span-->
        Rasio Kompresi (%)
        <table border="0">
            <tr>
                <td><input type="range" id="slider" min="1" max="100" value=<?php echo !empty($compression)? $compression : "50";?> step="1" style="width:100%;" onChange="printValue('slider','rangeValue')"></td>
                <td><input type="text" id="rangeValue" name="compression" value=<?php echo !empty($compression)? $compression : '50';?>" style="width:35px;" /></td>
            </tr>
        </table>
        <h3>Ringkas</h3>
        <input class="btn btn-primary" type="submit" value="GO" style="float: left;" />
    </form>
</div><!--/row-->
</div><!--/span-->
</div><!--/row-->

<hr>
<footer>
</footer>
</div><!--/.fluid-container-->

<!--<script src=".js/latest.js"></script>-->
<!-- <script src=".js/bootstrap.min.js"></script>-->
</body>
</html>

```

The diagram illustrates a web page structure with various code snippets highlighted in different colors:

- Yellow highlights:** `<!--<script src=".js/latest.js"></script>-->`, `<!-- <script src=".js/bootstrap.min.js"></script>-->`, `</body>`, and `</html>`.
- Blue highlights:** `<?php`, `function indexer() {`, `// scan nama file korpus`, `$dir_corpus = "./corpus";`, `$files = scandir($dir_corpus);`, `$files = array_slice($files, 2);`, `$N = sizeof($files); // total dokumen`, `// load daftar stopwords`, `$stopwords = file_get_contents("./stopwords.txt");`, `$stopwords = preg_split("/[\s]+/", $stopwords);`, `// inisialisasi`, `$all_token = array();`, `$freq_token = array();`, `$df_token = array();`, `$inv_index = array();`, `// load isi file korpus`, `foreach ($files as $key => $filename) {`, `$load_file = file_get_contents("./corpus/".$filename);`, `// tokenisasi dengan membuang stopwords`, `$token = preg_split("/[\d\W\s]+/", strtolower($load_file));`, `$token = array_diff($token, $stopwords);`, `$token = array_values($token); // perbaiki indeks`, `// menyimpan nilai df tiap token`, `$token = array_count_values($token); // hilangkan`, and `redundansi token`.
- Green highlights:** `<footer>`, `<p align="center">Salamah (2014) ©</p>`, `</footer>`, and `</div><!--/.fluid-container-->`.

```
<?php
function indexer() {
    // scan nama file korpus
    $dir_corpus = "./corpus";
    $files      = scandir($dir_corpus);
    $files      = array_slice($files, 2);
    $N          = sizeof($files); // total dokumen

    // load daftar stopwords
    $stopwords  = file_get_contents("./stopwords.txt");
    $stopwords  = preg_split("/[\s]+/", $stopwords);

    // inisialisasi
    $all_token  = array();
    $freq_token = array();
    $df_token   = array();
    $inv_index  = array();

    // load isi file korpus
    foreach ($files as $key => $filename) {
        $load_file = file_get_contents("./corpus/".$filename);

        // tokenisasi dengan membuang stopwords
        $token = preg_split("/[\d\W\s]+/", strtolower($load_file));
        $token = array_diff($token, $stopwords);
        $token = array_values($token); // perbaiki indeks

        // menyimpan nilai df tiap token
        $token = array_count_values($token); // hilangkan
        redundansi token

    }
}

// footer
<?php
function footer() {
    <p align="center">Salamah (2014) &copy;</p>
}</?php>
</div><!--/.fluid-container-->

<!--<script src=".js/latest.js"></script>-->
<!-- <script src=".js/bootstrap.min.js"></script>-->
</body>
</html>
```

```
foreach ($token as $key => $value) {
    // jika token sudah ada
    if (array_key_exists($key, $df_token))
        $df_token[$key]++;
    else if (!isset($df_token[$key]))
        $df_token[$key] = 1;
}

// membuat array inverted index
foreach ($df_token as $key => $value) {
    $df      = $value;
    $inv_index[$key]
    $inv_index[$key]['idf'] = array();
    $inv_index[$key]['idf'] = log10($N/$df); // simpan
}

//print_r($inv_index);
//save_xml();
return $inv_index;
}
```

koneksi.php

```
<?php
$user = "root";
$pass = "";
$db = "kamus";
$host = "localhost";
$konek = mysql_connect($host, $user, $pass);
mysql_select_db($db, $konek);
?>
```

proses.php

```
<html>
  <head>
    <title>Hasil Tokenisasi</title>
    <!-- Bootstrap -->
    <link href=".//css/bootstrap.min.css" rel="stylesheet" media="screen">
    <link href=".//css/bootstrap-responsive.min.css" rel="stylesheet" media="screen">
  </head>
  <body>
    <div class="navbar navbar-inverse navbar-fixed-top">
      <div class="navbar-inner">
        <div class="container-fluid">
          <a class="brand" href="index.php">Auto
          <div class="nav-collapse collapse">
            <ul class="nav">
              <li><a class="brand" href="#" style="margin-left:85px;"><?php// echo $title;?></a></li>
            </ul>
          </div>
        </div>
      </div>
    </div>
    <?php
      error_reporting(0);
    ?>
    <?php
      $string = $_POST['text'];
      $banyak = $_POST['banyak'];
      $word = str_word_count(strtolower($string),1);
      $jumlah = count($word);
    ?>
    <div class="container-fluid">
      <div class="row-fluid">
        <div class="span11">
          <div class="row-fluid" style="margin-top:0px;">
            <?php
```

```

echo "<b><h3>Teks
Awal:</h3></b>";
```

Tokenisasi</h3>;

```

echo $string;
echo "<br>";
echo "<b><h3>Hasil
-----";
array_count_values($word);
echo "<b><br>";
```

Tokenisasi</h3>;

```

echo "<table border='1'>";
echo "<tr><th>Kata</th><th>Frekuensi</th></tr>";
foreach ($word as $key=>$val) {
echo "<tr><td>$key</td><td>$val</td></tr>";
}
echo "</table>";
echo "<b>Jumlah Kata : " .jumlah.
// echo "-----";
echo "<hr>";
echo "<br>";
$word_count
=
```

```

arsort($word_count);
$jumlah2 = count($word_count);
echo "<b><h3>Frekuensi
-----";
echo "<table border='1'>";
echo
foreach ($word_count as
echo
mysql_query("INSERT INTO
                ")
            INTO
                )
echo "</table>";
echo "<b>Jumlah Kata : " .jumlah2.
// echo "-----";
echo "<br>";
echo "<b><h3>Hasil " .banyak. "
-----";
echo "<table border='1'>";
echo
"<tr><th>Kata</th><th>Frekuensi</th></tr>";
```

Frekuensi Teratas</h3>;

```
$word_count2 = array_slice($word_count, 0, $banyak);
foreach ($word_count2 as $key2=>$val2) {
    echo "<tr><td>$key2</td><td>$val2</td></tr>";
}
echo "</table>";
// echo "-----";
};

class Keywords
{
    $words2 = array_diff($words2, $this->stopwords2);
    $wordCount = array_count_values($words2);
    arsort($wordCount);
    echo "<b><h2>Hasil Tokenisasi Setelah Difilter dengan Stopword</h2></b>";
    echo "<b>Stopword bahasa indonesia:</b><br>";
    echo "<b><h3>Hasil Tokenisasi</h3></b>";
    echo "<table border='1'>";
    echo "<tr><th>Kata</th><th>Frekuensi</th></tr>";
    $jumlah3 = count($wordCount);
    foreach ($wordCount as $key3=>$val3) {
        echo "<tr><td>$key3</td><td>$val3</td></tr>";
    }
    echo "</table>";
    echo "<b>Jumlah Kata : " . $jumlah3. "</b><br>";
    // echo "-----";
    $wordCount = array_slice($wordCount, 0, $nbrwords2);
    return array_keys($wordCount);
}
private function filter(&$val3, $key3)
{
    $val3 = strtolower($val3);
}
private function setStopwords2()
{
    $this->stopwords2 = array();
}
}

$test = new Keywords();
$keywords2 = $test->getKeywords($string, 9);
echo "<br>";
echo "<b><h3>Hasil " . $banyak. " Frekuensi Teratas</h3></b>";
echo "<table border='1'>";
```

```
echo "<tr><th>Kata</th></tr>";
$wordCount2 = (array_slice($keywords2, 0, $banyak));
foreach ($wordCount2 as $key4=>$val4) {
echo "<tr><td>$val4</td></tr>";
}
echo "</table>";
// echo "-----";
?>
<?php
error_reporting(0);
?>
</div><!--/row-->
</div><!--/span-->
</div><!--row-->
<hr>
<footer>
</footer>
</div><!--/.fluid-container-->
</body>
</html>
```

```
<!--<html>
<head>
<title>Hasil Tokenisasi</title>
</head>
<body>
<?php
//error_reporting(0);
?>
<?php
// $string = $_POST['text'];
// $banyak = $_POST['banyak'];
// $word = str_word_count(strtolower($string ),1);
// $jumlah = count($word);
// echo "<b><h3>Teks Awal:</h3></b>";
// echo $string;
// echo "<br>";

// echo "<b><h3>Hasil Tokenisasi</h3></b>";
// echo "<table border='1'>";
// echo "<tr><th>Kata</th></tr>";
// foreach ($word as $key=>$val) {
// echo "<tr><td>$val</td></tr>";
```

```
// }
// echo "</table>";
// echo "<b>Jumlah Kata : " . $jumlah. "</b><br>";
// echo "-----";
// echo "<br>";
// $word_count = array_count_values($word);
// arsort($word_count);
// $jumlah2 = count($word_count);
// echo "<b><h3>Frekuensi Tokenisasi</h3></b>";
// echo "<table border='1'>";
// echo "<tr><th>Kata</th><th>Frekuensi</th></tr>";
// foreach ($word_count as $key=>$val) {
// echo "<tr><td>$key</td><td>$val</td></tr>";
// mysql_query("INSERT INTO tokenisasi VALUES ('bla','$key','$val')");
// }
// echo "</table>";
// echo "<b>Jumlah Kata : " . $jumlah2. "</b><br>";
// echo "-----";
// echo "<br>";
// echo "<b><h3>Hasil " . $banyak. " Frekuensi Teratas</h3></b>";
// echo "<table border='1'>";
// echo "<tr><th>Kata</th><th>Frekuensi</th></tr>";
// $word_count2 = (array_slice($word_count, 0, $banyak));
// foreach ($word_count2 as $key2=>$val2) {
// echo "<tr><td>$key2</td><td>$val2</td></tr>";
// }
// echo "</table>";
// echo "-----";
?>
<?php
// class Keywords
// echo "<b><h3>Hasil Tokenisasi</h3></b>";
// echo "<table border='1'>";
// echo "<tr><th>Kata</th><th>Frekuensi</th></tr>";
// $jumlah3 = count($wordCount);
// foreach ($wordCount as $key3=>$val3) {
// echo "<tr><td>$key3</td><td>$val3</td></tr>";
// }
// echo "</table>";
// echo "<b>Jumlah Kata : " . $jumlah3. "</b><br>";
// echo "-----";
// $wordCount = array_slice($wordCount, 0, $nbrwords2);
// return array_keys($wordCount);
// }
// private function filter(&$val3, $key3)
```

The diagram illustrates the execution flow of a PHP script. It starts with a green-bordered box containing the PHP code for 'indexer.php'. A yellow curved arrow points from this box to a blue-bordered box containing the code for 'summarize.php'. From 'summarize.php', another yellow curved arrow points to a white box containing the final HTML output. Below the boxes, there is a large watermark-like graphic of a classical building with four columns and a pediment.

```
// {
// $val3 = strtolower($val3);
// }
// private function setStopwords2()
// {
// $this->stopwords2 = array();
// }
// }
// $test = new Keywords();
// $keywords2 = $test->getKeywords($string, 9);
// echo "<br>";
// echo "<b><h3>Hasil " . $banyak. " Frekuensi Teratas</h3></b>";
// echo "<table border='1'>";
// echo "<tr><th>Kata</th></tr>";
// $wordCount2 = (array_slice($keywords2, 0, $banyak));
// foreach ($wordCount2 as $key4=>$val4) {
// echo "<tr><td>$val4</td></tr>";
// }
// echo "</table>";
// echo "-----";
?>
<?php
//error_reporting(0);
?>
</body>
</html>
```

summarize.php

```
<?php
include "./indexer.php";

function summarize ($filename, $compress) {
    // proses indexing
    $inv_index = indexer();

    // load file dan daftar stopwords
    $load_file = $filename;
    $sentence = preg_split("/[.]+/", $load_file);
    $sentence = array_slice($sentence, 0, sizeof($sentence)-1); // buang array terakhir (kosong)

    $stopwords = file_get_contents("./stopwords.txt");
    $stopwords = preg_split("/[\s]+/", $stopwords);
```

```
// jumlah kalimat yang diringkas
$compression_rate = $compress/100;
$max_sentence =
floor(sizeof($sentence)*$compression_rate);

// inisialisasi
$sentence_weight = array();

// menghitung bobot tf.idf tiap kalimat
foreach ($sentence as $key => $value) {
    // tokenisasi dengan membuang stopwords
    $word = preg_split("/[\d\W\s]+/", strtolower($value));
    $word = array_diff($word, $stopwords);
    $word = array_values($word); // perbaiki indeks

    // inisialisasi bobot dan hitung frekuensi token
    $tf_idf = 0;
    $freq_word = array_count_values($word);

    // hitung bobot tf.idf
    foreach ($freq_word as $token => $tf)
        $tf_idf += $tf * $inv_index[$token]['idf'];

    // simpan nilai bobot kalimat
    array_push($sentence_weight, $tf_idf);
}

// sorting bobot tertinggi -> potong array -> sorting urutan kalimat
arsort($sentence_weight);
$sorted = array_slice($sentence_weight, 0, $max_sentence, true);
ksort($sorted);

// gabungkan ringkasan
$summary = "";
foreach ($sorted as $key => $value)
    $summary = $summary.$sentence[$key]. " ";

// return teks asli dan hasil ringkasan
$output = array();
$output['original'] = $load_file;
$output['summary'] = $summary;

return $output;
}
```