

PENCARIAN DATA TUGAS AKHIR MENGGUNAKAN METODE CASE BASED REASONING

Diajukan Sebagai Syarat Untuk Menyelesaikan
Pendidikan Program Strata- 1 Pada
Jurusan Teknik Informatika



Oleh:

YUWAN SAMEGA
NIM: 09021281823072

Jurusan teknik informatika

FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA

2021

LEMBAR PENGESAHAN SKRIPSI

PENCARIAN DATA TUGAS AKHIR MENGGUNAKAN METODE CASE
BASED REASONING

Oleh:

Yuwan Samega

NIM: 09021281823072

Palembang, 30 Desember 2021

Pembimbing I



Dian Palupi Rini, M.Kom., Ph.D.

NIP. 197802232006042002

Pembimbing II



Osvari Arsalan, S.Kom., M.T.

NIP. 198806282018031001

Mengetahui,

Ketua Jurusan Teknik Informatika



Alvi Syahrini Utami, M.Kom.

NIP. 197812222006042003

 Dipindai dengan CamScanner

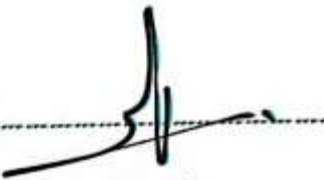
TANDA LULUS UJIAN SIDANG SKRIPSI

Pada hari Kamis tanggal 30 Desember 2021 telah dilaksanakan ujian sidang skripsi oleh Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya

Nama : Yuwan Samega
NIM : 09021281823072
Judul : Pencarian Data Tugas Akhir menggunakan Metode Case Based Reasoning

1. Ketua

Abdiansah, M.Cs.
NIP. 198410012009121005



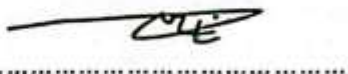
2. Pembimbing I

Dian Palupi Rini, M.Kom., Ph.D.
NIP. 197802232006042002



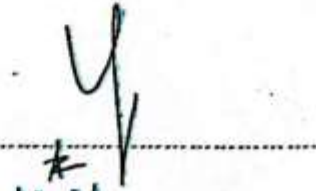
3. Pembimbing II

Osvari Arsalan, S.Kom., M.T.
NIP. 198806282018031001



4. Penguji I

Yunita, M.Cs.
NIP. 198306062015042002



5. Penguji II

Hadipurnawan Satria, Ph.D.
NIPUS. 198004182015109101



Mengetahui,
Ketua Jurusan Teknik Informatika

Alvi Syahrini Utami, M.Kom.
NIP. 19781222006042003



HALAMAN PERNYATAAN BEBAS PLAGIAT

Yang bertanda tangan dibawah ini:

Nama : Yuwan Samega

NIM : 09021281823072

Program Studi : Teknik Informatika

Judul Skripsi : Pencarian Data Tugas Akhir menggunakan Metode Case Based Reasoning

Hasil Pengecekan Software (Thenticate/Turnitin) : 17%

Menyatakan bahwa laporan penelitian saya merupakan hasil karya saya sendiri dan bukan hasil penjiplakan/plagiat. Apabila ditemukan unsur penjiplakan/plagiat dalam laporan penelitian ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.



Palembang, 30 Desember 2021



Yuwan Samega

NIM. 09021281823072

KATA PENGANTAR

Puji syukur kepada Allah atas berkat dan rahmat-Nya yang telah diberikan kepada Penulis sehingga dapat menyelesaikan Tugas Akhir ini dengan baik. Tugas akhir ini disusun untuk memenuhi salah satu syarat guna menyelesaikan pendidikan program Strata-1 pada Fakultas Ilmu Komputer Program Studi Teknik Informatika di Universitas Sriwijaya. Dalam menyelesaikan Tugas Akhir ini banyak pihak yang telah memberikan bantuan dan dukungan baik secara langsung maupun secara tidak langsung. Penulis ingin menyampaikan rasa terima kasih kepada:

1. Ayah dan ibuku tercinta, Enda Jaya dan Latifah H. yang selalu mendoakan serta memberikan dukungan.
2. Saudaraku Septian Kusuma yang telah banyak memberi dukungan dan doa.
3. Bapak Jaidan Jauhari, M.T selaku Dekan Fakultas Ilmu Komputer Universitas Sriwijaya.
4. Ibu Alvi Syahrini Utami, M.Kom. selaku Ketua Jurusan Teknik Informatika,
5. Ibu Dian Palupi Rini, M.Kom., Ph.D. selaku dosen pembimbing I serta Bapak Osvari Arsalan, S.Kom., M.T. selaku pembimbing II yang telah membimbing, mengarahkan, dan memberikan motivasi penulis dalam proses perkuliahan

dan pengerjaan Tugas Akhir.

6. Seluruh dosen Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.
7. Mbak Wiwin dan seluruh staf tata usaha yang telah membantu dalam kelancaran proses administrasi dan akademik selama masa perkuliahan.
8. Teman-teman Teknik Informatika Regular kelas C yang telah mewarnai hari-hari selama masa perkuliahan
9. Serta pihak-pihak lainnya yang terlibat selama pelaksanaan Tugas Akhir ini yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari dalam penyusunan Tugas Akhir ini masih terdapat banyak kekurangan disebabkan keterbatasan pengetahuan dan pengalaman, oleh karena itu kritik dan saran yang membangun sangat diharapkan untuk kemajuan penelitian selanjutnya. Akhir kata semoga Tugas Akhir ini dapat berguna dan bermanfaat bagi kita semua.

Palembang, 30 Desember 2021

Yuwan Samega

NIM. 09021281823072

PENCARIAN DATA TUGAS AKHIR MENGGUNAKAN METODE CASE BASED REASONING

YUWAN SAMEGA
09021281823072

ABSTRACT

Students can search for final project information through various applications, one of which is the SIPETA application which is owned by the Department of Information Engineering, Faculty of Computer Science, Sriwijaya University. However, the search on the application has a drawback that is only able to search for information with the exact same keywords and also sequentially based on the keywords entered by the user. This is because the system has not implemented *information retrieval*. Therefore, the problem formulated in this research is how to implement and evaluate the CBR method with Cosine Similarity to the student's final task search. Based on these problems, the purpose of this research is to implement and find out the evaluation of the CBR method with Cosine Similarity to the student's final task search. The method used in accordance with the problem and also the purpose of this study is to use Case Based Reasoning where the method consists of 4 phases including retrieve, reuse, revise and retain combined with the algorithm *cosine similarity* for the calculation process to find the similarity value between data. The evaluation obtained from the research results shows that the highest and lowest similarity values that can be achieved by the system are 0.628961 and 0.171224 with a threshold value of 0.1 from the amount of data stored as much as 143 data where the acquisition of the value is influenced by the large number of recommended data. Evaluation Comparison with the SIPETA application shows that the system has the ability to provide data recommendation results with keywords even though the keywords are reversed or changed the order and is able to display data containing keywords even though only part of the keywords entered by the user and display data sorted according to order of data that is most relevant to the keywords by looking at the similarity value.

Keyword: *Information Retrieval, Case Based Reasoning, Cosine Similarity*, final task data search

PENCARIAN DATA TUGAS AKHIR MENGGUNAKAN METODE CASE BASED REASONING

YUWAN SAMEGA
09021281823072

ABSTRAK

Pencarian informasi tugas akhir dapat dilakukan mahasiswa melalui berbagai aplikasi salah satunya aplikasi SIPETA yang dimiliki oleh jurusan teknik informatika fakultas ilmu komputer universitas sriwijaya. Namun pencarian pada aplikasi tersebut memiliki kekurangan yakni hanya mampu mencari informasi dengan kata kunci yang sama persis dan juga secara berurutan berdasarkan kata kunci yang dimasukkan oleh pengguna. Hal tersebut dikarenakan sistem belum menerapkan *information retrieval*. Oleh karena itu permasalahan yang dirumuskan pada penelitian ini adalah bagaimana mengimplementasikan dan evaluasi metode CBR dengan Cosine Similarity terhadap pencarian tugas akhir mahasiswa. Berdasarkan permasalahan tersebut maka tujuan dari penelitian ini adalah mengimplementasikan dan mengetahui evaluasi metode CBR dengan Cosine Similarity terhadap pencarian tugas akhir mahasiswa. Metode yang digunakan sesuai dengan permasalahan dan juga tujuan pada penelitian ini ialah menggunakan Case Based Reasoning dimana metode tersebut terdiri dari 4 fase diantaranya retrieve, reuse, revise dan retain dipadukan dengan algoritma *cosine similarity* untuk proses perhitungan mencari nilai kemiripan antar data. Evaluasi yang diperoleh dari hasil penelitian menunjukkan nilai similaritas tertinggi dan terendah yang dapat dicapai oleh sistem adalah bernilai 0.628961 dan 0.171224 dengan nilai ambang batas adalah 0.1 dari jumlah data yang tersimpan sebanyak 143 data yang mana perolehan nilai tersebut dipengaruhi oleh banyaknya jumlah data hasil rekomendasi. Evaluasi Perbandingan dengan aplikasi SIPETA menunjukkan bahwa sistem memiliki kemampuan untuk memberikan hasil rekomendasi data dengan kata kunci meskipun kata kunci dibalik atau diubah urutannya dan mampu menampilkan data yang mengandung kata kunci walaupun hanya sebagian dari kata kunci yang dimasukkan oleh pengguna serta menampilkan data yang terurut sesuai dengan urutan data yang paling relevan terhadap kata kunci dengan melihat nilai similaritasnya.

Kata Kunci: *Information Retrieval, Case Based Reasoning, Cosine Similarity, Pencarian Data Tugas Akhir*

DAFTAR ISI

LEMBAR PENGESAHAN PROPOSAL SKRIPSI	ii
TANDA LULUS UJIAN SIDANG.....	iii
HALAMAN PERNYATAAN BEBAS PLAGIAT	iv
KATA PENGANTAR.....	v
ABSTRACT	vii
ABSTRAK	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL.....	xvi
DAFTAR LAMPIRAN	xvii
BAB I	1
PENDAHULUAN	1
1.1 Pendahuluan	1
1.2 Latar Belakang	1
1.3 Rumusan Masalah	4
1.4 Tujuan Penelitian.....	4
1.5 Batasan Masalah.....	4
1.6 Manfaat.....	5
1.7 Sistematika Penulisan.....	5
1.8 Kesimpulan.....	7
BAB II.....	8
KAJIAN LITERATUR	8
2.1 Pendahuluan	8
2.2 Landasan Teori	8
2.2.1 Case Base Reasoning (CBR)	8
2.2.2 Siklus Case Base Reasoning.....	8
2.2.2.1 Retrieve	10
2.2.2.2 Reuse	11

2.2.2.3	Revisi.....	12
2.2.2.4	Retain.....	12
2.2.3	Rational Unified Model (RUP).....	13
2.2.3.1	Inception.....	14
2.2.3.2	Elaboration	14
2.2.3.3	Construction	14
2.2.3.4	Transition.....	14
2.2.4	Unified Modelling Language (UML).....	15
2.2.5	Text Mining.....	15
2.2.6	Text Preprocessing.....	16
2.2.6.1	Case Folding.....	16
2.2.6.2	Tokenizing	16
2.2.6.3	Stopword Removal	17
2.2.6.4	Stemming.....	17
2.2.7	Term Frequency Inverse Document (TF-IDF).....	17
2.2.8	Cosine Similarity	19
2.3	Penelitian Lain yang Relevan.....	20
2.4	Kesimpulan.....	21
BAB III		22
METODE PENELITIAN.....		22
3.1	Pendahuluan	22
3.2	Pengumpulan Data	22
3.2.1	Jenis Data.....	22
3.2.1.1	Data Sekunder	22
3.2.2	Sumber Data.....	23
3.2.3	Metode Pengumpulan Data.....	23
3.2.3.1	Dokumentasi.....	24
3.3	Tahapan Penelitian.....	24
3.3.1	Mengumpulkan Data Penelitian.....	25
3.3.2	Kerangka Kerja Penelitian	25
3.3.2.1	Preprocessing.....	28

3.3.2.2	Data direpresentasikan	29
3.3.2.3	Retrieval	29
3.3.2.4	Reuse	30
3.3.2.5	Revise dan Retain	30
3.3.3	Kriteria Data Penelitian.....	31
3.3.4	Format Data Penelitian.....	31
3.3.5	Alat Penelitian.....	32
3.3.6	Pengujian Penelitian.....	32
3.3.7	Analisis Hasil Pengujian	33
3.4	Metode Pengembangan Perangkat Lunak	33
3.5	Manajemen Proyek Penelitian.....	33
3.6	Kesimpulan.....	34
BAB IV		35
PENGEMBANGAN PERANGKAT LUNAK		35
4.1	Pendahuluan	35
4.2	Fase Insepsi	35
4.2.1	Kebutuhan Fungsional	35
4.2.2	Kebutuhan Non Fungsional	42
4.3	Fase Elaborasi.....	43
4.3.1	Use Case	43
4.3.1.1	Use Case Diagram	44
4.3.1.2	Definisi Aktor	44
4.3.1.3	Definisi Use Case	45
4.3.1.4	Use Case Scenario	46
4.3.2	Activity Diagram	52
4.3.3	Sequence Diagram	54
4.3.4	Data Model	55
4.3.5	Perancangan Antar Muka.....	56
4.4	Fase Kontruksi.....	58
4.4.1	Implementasi Sistem.....	58
4.4.1	Implementasi Antarmuka.....	58

4.5	Fase Transisi	60
4.5.1	Pengujian Perangkat Lunak	60
4.5.1.1	Pengujian Use Case Melakukan Pencarian Data.....	60
4.5.1.2	Pengujian Use Case Melakukan Preprocessing Data	61
4.5.1.3	Pengujian Use Case Melakukan Pembobotan Data.....	62
4.5.1.4	Pengujian Use Case Melakukan Perhitungan Similaritas	63
4.6	Kesimpulan.....	64
BAB V.....		66
HASIL DAN ANALISIS PENELITIAN		66
5.1	Pendahuluan	66
5.2.2	Hasil Penelitian.....	66
5.3	Analisis Hasil Penelitian	72
5.4	Kesimpulan.....	76
BAB VI		77
KESIMPULAN DAN SARAN.....		77
6.1	Pendahuluan	77
6.2	Kesimpulan.....	77
6.3	Saran.....	78
DAFTAR PUSTAKA		79
LAMPIRAN		83

DAFTAR GAMBAR

Gambar II – 1 Siklus Case Based Reasoning (Mathew et al., 2006)	9
Gambar II – 2 RUP (Andrian et al., 2014)	12
Gambar III – 1 Data Tugas Akhir Mahasiswa.....	21
Gambar III – 2 Tahapan Penelitian	22
Gambar III – 3 Kerangka Kerja Penelitian	24
Gambar III – 4 Manajemen Proyek Penelitian.....	30
Gambar IV – 1 Use Case Diagram.....	41
Gambar IV – 2 Activity Diagram.....	50
Gambar IV – 3 Sequence Diagram	51
Gambar IV – 4 Data Model.....	52
Gambar IV – 5 Rancangan Antar Muka Home	53
Gambar IV – 6 Rancangan Antar Muka Halaman Result	54
Gambar IV – 7 Antar Muka Halaman Home	56
Gambar IV – 8 Antar Muka Halaman Result.....	56
Gambar V – 1 Data Hasil Rekomendasi “Levenshtein Distance” 1	68
Gambar V – 2 Data Hasil Rekomendasi “Levenshtein Distance” 2	68
Gambar V – 3 Data Hasil Rekomendasi “Levenshtein Distance” 3	69
Gambar V – 4 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 1	70
Gambar V – 5 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 2	70
Gambar V – 6 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 3	71
Gambar V – 7 Histogram Similaritas.....	73

DAFTAR TABEL

Tabel III-1 Format Data Penelitian	28
Tabel IV-1 Kebutuhan Fungsional	33
Tabel IV-2 Indexing Keywords	34
Tabel IV-3 IDF Keywords.....	36
Tabel IV-4 TF-IDF Keywords.....	36
Tabel IV-5 Panjang Vektor Dokumen dan Query.....	37
Tabel IV-6 $Wqk * Wdk$	38
Tabel IV-7 Cosine Similarity Antar Dokumen	39
Tabel IV-8 Non Fungsional	40
Tabel IV-9 Definisi Aktor.....	42
Tabel IV-10 Definisi Use Case	42
Tabel IV-11 Use Case Scenario 01	44
Tabel IV-12 Use Case Scenario 02.....	45
Tabel IV-13 Use Case Scenario 03.....	46
Tabel IV-14 Use Case Scenario 04.....	47
Tabel IV-15 Pengujian Use Case Melakukan Pencarian Data	57
Tabel IV-16 Pengujian Use Case Melakukan Preprocessing Data.....	58
Tabel IV-17 Pengujian Use Case Melakukan Pembobotan Data	60
Tabel IV-18 Pengujian Use Case Melakukan Perhitungan Similaritas	61
Tabel V-1 Data Hasil Pengujian Penelitian	64

DAFTAR LAMPIRAN

Lampiran 1. Analisis Perhitungan Similaritas.....	83
Lampiran 2. Source Code Program.....	104

BAB I

PENDAHULUAN

1.1 Pendahuluan

Pada bab pendahuluan akan membahas latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan serta kesimpulan yang ada pada bab 1. Bab ini juga berisikan penjelasan mengenai gambaran umum dari keseluruhan penelitian yang dilakukan bersamaan dengan penyusunan skripsi

1.2 Latar Belakang

Penyusunan tugas akhir atau skripsi merupakan persyaratan kelulusan yang harus mahasiswa penuhi dalam jenjang pendidikan prodi sarjana di Universitas Sriwijaya. Mencari referensi judul dari jurnal ataupun tugas akhir alumni yang berkaitan dengan ide yang akan diajukan biasa dilakukan mahasiswa sebagai langkah awal persiapan penyusunan tugas akhir. Dari pencarian tersebut informasi yang diperoleh selanjutnya dapat dijadikan pedoman ataupun acuan kedepannya oleh mahasiswa yang akan menyusun tugas akhir.

Pencarian informasi atau data tugas akhir saat ini dapat dilakukan mahasiswa secara online melalui berbagai aplikasi salah satunya aplikasi yang dimiliki oleh

jurusan teknik informatika yaitu SIPETA (Sistem Pencarian Tugas Akhir). Namun, pencarian pada aplikasi tersebut memiliki kekurangan yakni hanya mampu mencari informasi dengan kata kunci yang sama persis dan juga secara berurutan berdasarkan kata kunci yang dimasukkan oleh pengguna. misal, jika kata kunci yang dimasukkan adalah “sistem informasi” maka akan ditampilkan seluruh data yang mengandung kata “sistem informasi” sedangkan apabila kata kunci dibalik menjadi “informasi sistem” maka data yang ditampilkan akan berbeda dari kata kunci yang dimasukkan sebelumnya. Hal tersebut dikarenakan sistem tidak menerapkan *information text retrieval* yang bertujuan untuk mengukur kemiripan antar data satu sama lain dengan menghitung nilai similaritas. *Information retrieval* berfokus pada proses yang terlibat di dalam representasi, media penyimpanan, mencari dan menemukan informasi yang relevan dari informasi yang diinginkan oleh user (Ingwersen, 1992), (A. Mubarak, 2020).

Case-Based Reasoning (CBR) menjadi salah satu model pendekatan yang mampu memberikan solusi dengan cara menghitung tingkat kemiripan antara kasus yang lampau dengan kasus yang baru (M. Salmin & S. Hartati, 2018). Metode Case Based Reasoning (CBR) dapat digunakan sebagai sistem rekomendasi topik skripsi bagi mahasiswa (Merawati & Hartati, 2018). Telah banyak dilakukan penelitian yang menggunakan pendekatan CBR dan menghasilkan hasil penelitian yang akurat pada berbagai bidang (E. Faizal, 2014).

Penelitian terkait perhitungan similaritas atau nilai kedekatan dilakukan dengan berbagai metode, diantaranya berbasis *distance* dengan algoritma *Jaccard Similarity* dan *Cosine Similarity* (Nurdiana et al., 2016), serta metode *hybrid* (Riyani et al., 2019), (Melita et al., 2018)

Cosine Similarity lebih diunggulkan dari segi nilai kemiripan karena metode tersebut mempunyai konsep normalisasi panjang vektor data dengan membandingkan N-gram yang sejajar satu sama lain dari 2 pembanding. Sedangkan pada metode *Jaccard* hanya membandingkan isi N-gram dengan eksak dan hanya melihat apakah ada suatu N-gram tertentu pada pembanding tanpa melihat posisi penulisan yang berbeda (Nurdiana et al., 2016). Algoritma *Cosine Similarity* memiliki kelebihan utama yakni algoritma tersebut tidak memiliki pengaruh terhadap panjang pendeknya dari suatu dokumen serta memiliki tingkat akurasi yang akurat. (Riyani et al., 2019). *Cosine Similarity* digabungkan dengan pembobotan TF-IDF akan memberikan hasil berupa nilai kemiripan dari masing-masing dokumen pembanding secara akurat. (Melita et al., 2018).

Berdasarkan permasalahan dan penelitian-penelitian yang telah dipaparkan pada penelitian ini akan diterapkan metode *Case Base Reasoning* dengan *Cosine Similarity* sebagai *Information Retrieval* untuk pencarian data tugas akhir mahasiswa serta membandingkan keluaran yang dihasilkan oleh sistem dengan aplikasi serupa bernama SIPETA.

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan metode CBR dengan Cosine Similarity terhadap pencarian tugas akhir mahasiswa
2. Bagaimana evaluasi metode CBR dengan Cosine Similarity pada masalah pencarian tugas akhir mahasiswa.

1.4 Tujuan Penelitian

Berdasarkan paparan rumusan masalah tersebut maka tujuan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasikan metode CBR dengan Cosine Similarity terhadap pencarian tugas akhir mahasiswa
2. Mengetahui evaluasi metode CBR dengan Cosine Similarity terhadap permasalahan pencarian tugas akhir mahasiswa

1.5 Batasan Masalah

Agar pembahasan lebih terarah dan terfokus penulis menetapkan batasan masalah sebagai berikut:

1. Data yang digunakan berasal dari data tugas akhir mahasiswa reguler tahun 2012-2016 dan bilingual tahun 2012-2016 jurusan teknik informatika fakultas ilmu komputer universitas sriwijaya.
2. Sistem yang akan dibuat berbasis pada website.

1.6 Manfaat

Manfaat yang diharapkan pada penelitian penerapan metode CBR terhadap aplikasi pencarian tugas akhir adalah sebagai berikut:

1. Memberikan kemudahan bagi mahasiswa dalam mendapatkan informasi terkait tugas akhir
2. Memberikan pengetahuan bagaimana metode CBR jika diterapkan terhadap suatu aplikasi pencarian tugas akhir.
3. Aplikasi yang dihasilkan dapat digunakan oleh masyarakat lingkungan universitas
4. Hasil penelitian dapat dijadikan rujukan oleh peneliti lain dimasa yang akan datang.

1.7 Sistematika Penulisan

Terbagi dalam 5 BAB, yakni:

BAB I. PENDAHULUAN

Pada bab I diuraikan mengenai latar belakang, perumusan masalah, tujuan dan manfaat penelitian, batasan masalah serta sistematika penulisan.

BAB II. STUDI PUSTAKA

Pada bab II akan dibahas dasar-dasar teori yang digunakan dalam penelitian, seperti definisi CBR, pre-processing, Cosine Similarity, dan TF-IDF

BAB III. METODOLOGI PENELITIAN

Pada bab III akan dibahas mengenai pengumpulan data, tahapan penelitian, Di akhir bab ini berisi perancangan manajemen proyek pada pelaksanaan penelitian.

BAB IV. PENGEMBANGAN PERANGKAT LUNAK

Pada bab IV akan dibahas mengenai proses pengembangan perangkat lunak yang dilakukan sesuai dengan fase-fase pada metode RUP

BAB V. METODOLOGI PENELITIAN

Pada bab V berisi hasil pengujian berdasarkan langkah-langkah yang telah direncanakan. Analisis diberikan sebagai basis dari kesimpulan yang diambil dalam penelitian ini.

BAB VI. KESIMPULAN DAN SARAN

Bab VI berisi kesimpulan hasil pengujian perangkat lunak yang telah didapat serta saran-saran untuk penelitian selanjutnya.

Pada bab ini akan dibahas mengenai pengumpulan data, tahapan penelitian, Di akhir bab ini berisi perancangan manajemen proyek pada pelaksanaan penelitian.

1.8 Kesimpulan

Berdasarkan uraian yang telah dipaparkan dapat disimpulkan bahwa akan dibuat pencarian data tugas akhir mahasiswa menggunakan metode CBR dengan Cosine Similarity. Metode tersebut diharapkan mampu memberikan hasil yang terbaik ke dalam sistem.

BAB II

KAJIAN LITERATUR

2.1 Pendahuluan

Pada bab kajian literatur akan dipaparkan teori dalam penyelesaian masalah untuk bab metodologi penelitian. Adapun struktur penulisan pada bab ini diantaranya landasan teori, penelitian yang relevan, dan kesimpulan.

2.2 Landasan Teori

Landasan teori menjadi bagian penting dalam penelitian ilmiah. Dalam landasan teori ini akan dikemukakan tentang teori-teori atau konsep-konsep yang erat kaitannya dengan permasalahan penelitian.

2.2.1 Case Base Reasoning (CBR)

CBR dapat disebut juga dengan Knowledge Based Recommendation System yang dimana merupakan sistem rekomendasi berbasis pengetahuan. Pada umumnya metode KBRS menggunakan pengetahuan sebagai pola untuk memperoleh hasil yang menjadi rekomendasi. KBRS memiliki peran penting dimana sistem dibangun dengan berbasis pada pengetahuan serta pengetahuan tersebut direpresentasikan untuk dapat dipahami oleh sistem (Isinkaye et al., 2015).

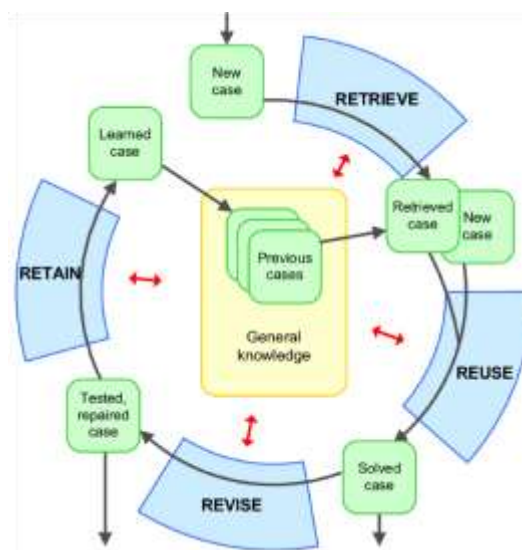
Metode CBR sendiri memiliki 4 fase diantaranya retrieve, reuse, revise, dan retain. Retrieve merupakan proses pencocokan kasus-kasus yang lama dengan kasus baru yang akan digunakan untuk dapat dijadikan sebagai sebuah solusi, reuse merupakan proses dimana sistem menggunakan kembali kasus-kasus sebelumnya atau informasi dari permasalahan sebelumnya yang memiliki kesamaan dengan kasus yang baru atau permasalahan yang baru, revise merupakan proses dimana sistem mengevaluasi, memperbaiki, dan mengkalkulasi berdasarkan kasus yang lama. Sedangkan pada fase retain, sistem akan melakukan proses mengintegrasikan, mengekstrak, dan mengindeks solusi-solusi baru dari permasalahan sebelumnya (Pal & Shiu, 2004).

Inti dari keempat fase CBR adalah untuk dapat memecahkan masalah atau memberikan solusi terhadap kasus-kasus yang telah lampau. Definisi dari kasus itu sendiri adalah suatu pengetahuan berdasarkan permasalahan yang telah diperoleh solusinya dan disimpan di dalam basis kasus. Setiap kasus berisi permasalahan dan juga jawaban, sehingga kasus yang lebih mirip memiliki suatu pola tertentu. CBR memiliki sifat yang dinamis seiring dengan bertambahnya pengetahuan, Pengetahuan biasanya disajikan dalam bentuk berupa kasus-kasus (Richter & Rosina, 2013). Representasi dari kasus yang ada pada sistem CBR memiliki 3 permasalahan utama yakni berupa mendefinisikan dari sebuah atribut yang menggambarkan suatu kasus dalam sistem, menentukan struktur suatu kasus yang ada untuk menggambarkan isi dari

kasus dan bagaimana cara untuk mengatur kasus-kasus yang ada pada basis kasus (Sappagh & Elmogy, 2015).

2.2.2 Siklus Case Base Reasoning

CBR terdiri dari empat bagian proses yang direpresentasikan sebagai sebuah siklus seperti yang ditunjukkan pada gambar II-1 (Aamodt & Plaza, 1994), (Hidayat et al., 2020).



Gambar II-1 Siklus Case Based Reasoning (Mathew et al., 2006)

2.2.2.1 Retrieve

Retrieve adalah proses untuk mencocokkan atau menemukan suatu kasus antara kasus baru dengan kasus sebelumnya yang berada di dalam basis

kasus kemudian akan digunakan kembali untuk dijadikan solusi dari kasus baru. Proses retrieve memiliki tahapan-tahapan diantaranya mengidentifikasi masalah, melakukan pencocokan masalah dan melakukan seleksi pada masalah. Langkah yang paling penting dalam tahapan pada sistem yang dibangun menggunakan CBR adalah menetapkan tingkat kemiripan atau *similarity* antar kasus. Dengan diperolehnya nilai dari perhitungan kemiripan antar kasus, maka dapat dibuat peringkat berdasarkan kasus-kasus yang mempunyai kemiripan (Pal & Shiu, 2004).

2.2.2.2 Reuse

Kasus-kasus berdasarkan hasil dari tahapan retrieve merupakan kasus lama yang serupa dengan permasalahan pada kasus baru, maka hasil yang diperoleh dari kasus lama dapat digunakan kembali oleh permasalahan baru apabila semua ketentuan pada masalah baru dengan kasus lama adalah sama, oleh karena itu hasil yang didapatkan berdasar dari kasus baru membutuhkan penyesuaian dengan kondisinya, proses ini dapat disebut dengan proses adaptasi. Reuse merupakan suatu kasus dalam artian sebuah konteks kasus baru yang difokuskan pada dua aspek yaitu (Watson, 1997), (Merawati & Hartati, 2018):

1. Perbedaan yang dimiliki antar kasus yang ada dengan kasus yang baru.

2. Pencarian pada kasus lama yang dapat digunakan kembali pada kasus yang baru.

Terdapat dua opsi yang digunakan untuk dapat menggunakan kembali sebuah kasus yang telah ada yaitu:

1. Menggunakan ulang dari kasus yang telah dijadikan sebagai solusi (transformational reuse).
2. Menggunakan ulang metode dari kasus yang telah ada untuk dapat membuat sebuah solusi (derivational reuse).

2.2.2.3 Revisi

Proses pada revise memiliki dua tahapan, tahapan yang pertama yaitu melakukan evaluasi dari hasil berupa solusi yang diperoleh serta tahapan yang kedua yakni melakukan pengkajian dan pembaruan pada solusi apabila diperlukan. Terdapat beberapa cara yang dapat dilakukan pada tahapan evaluasi yakni dengan menerima simpulan dari seorang ahli atau dengan melakukan uji coba terhadap solusi tersebut di dunia nyata. tahapan evaluasi juga dapat dilakukan berdasarkan simulasi dari solusi yang diterapkan (Smyth dan Cunningham, 1996), (Merawati & Hartati, 2018).

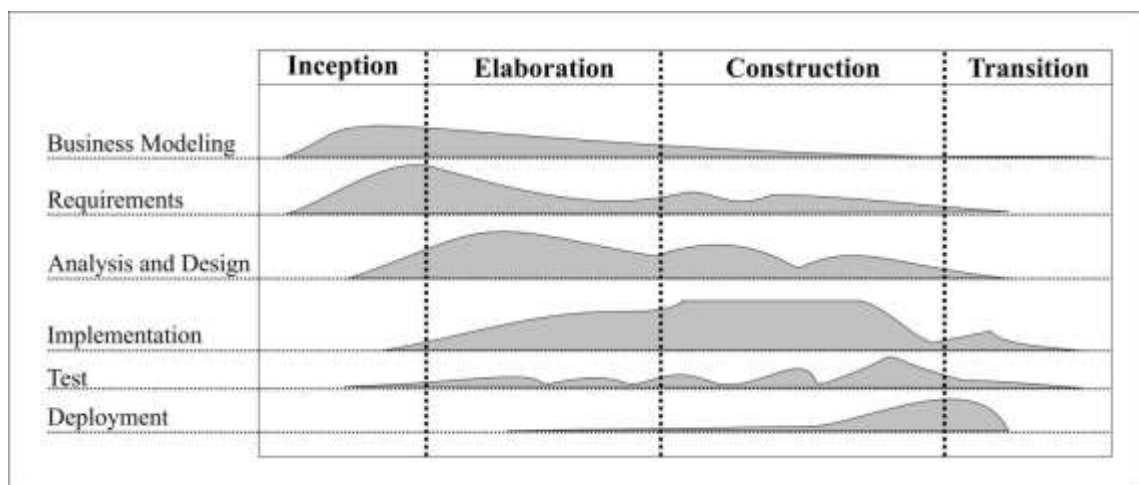
2.2.2.4 Retain

Retain merupakan proses bertambahnya suatu kasus yang baru setelah melalui tahapan revise ke dalam basis kasus. Semakin banyak bertambahnya

jenis kasus pada basis kasus, maka semakin bagus juga solusi yang dihasilkan oleh sistem CBR. Namun, efisiensi CBR dapat menurun disebabkan oleh hal tersebut. Pada proses retain terjadi penggabungan solusi dari kasus yang baru ke dalam basis kasus yang ada (Smyth dan Cunningham, 1996), (Merawati & Hartati, 2018).

2.2.3 Rational Unified Model (RUP)

RUP (Rational Unified Process) merupakan kerangka kerja dengan pendekatan pengembangan perangkat lunak yang dilakukan secara berulang kali (iterative), arsitektur terpusat (architecture-centric), penggunaan kasus yang lebih terarah (use case driven). RUP merupakan proses yang dilakukan untuk merencanakan perangkat lunak yang didasarkan pada definisi yang baik (well defined) dan juga struktur yang baik (well structured) (Rossa et al., 2014).



Gambar II-2 RUP (Andrian et al., 2014)

Metode RUP mempunyai empat fase, yaitu:

2.2.3.1 Inception

Tahap dimana kita memodelkan proses bisnis yang dibutuhkan (business modeling) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (requirements).

2.2.3.2 Elaboration

Tahap dimana perencanaan pada arsitektur sistem difokuskan. Tahap ini lebih pada analisis dan desain sistem.

2.2.3.3 Construction

Tahapan untuk mengembangkan suatu komponen serta fitur-fitur yang ada di dalam sistem. Pengujian sistem yang berfokus pada implementasi perangkat lunak berupa kode program.

2.2.3.4 Transition

Tahap dimana kita deployment atau Instalasi sistem agar dapat dimengerti oleh user. Aktivitas pada tahap ini termasuk pada pelatihan user dan pemeliharaan.

2.2.4 Unified Modelling Language (UML)

UML (Unified Modelling Language) merupakan seperangkat konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak dalam hal objek (Whitten & Lonnie, 2007).

2.2.5 Text Mining

Text mining merupakan proses pengkajian data pada teks yang berasal dari sumber data yang diperoleh dari sebuah dokumen (Ronen Feldman, 2007). Text mining biasa digunakan untuk mengklasifikasikan suatu dokumen berupa text sesuai dengan topik dokumen tersebut. Dengan adanya text mining melakukan pengelompokan suatu dokumen akan menjadi mudah dan pemanfaatan waktu menjadi efisien.

Tahapan kajian pada text mining yakni dengan melakukan pengumpulan data terlebih dahulu kemudian dilakukan ekstraksi terhadap fitur pada data yang akan digunakan (Ronen Feldman, 2007). Teknik yang digunakan untuk mengekstraksi fitur pada text mining diantaranya case folding, tokenizing, stop words removal, dan stemming.

2.2.6 Text Preprocessing

Text Preprocessing adalah tahapan dari proses awal terhadap teks untuk mempersiapkan teks menjadi data yang akan diolah lebih lanjut. algoritma pencarian tidak dapat memproses suatu teks secara langsung, oleh karena itu diperlukannya preprocessing text untuk mengubah teks menjadi data numeric (Feldman et al., 2007). Berikut merupakan tahap preprocessing:

2.2.6.1 Case Folding

Case folding adalah proses pertama dari rangkaian preprocessing teks terhadap dokumen. Dalam proses ini semua huruf yang ada di dalam dokumen diubah menjadi huruf kecil. Hanya huruf a sampai dengan z yang diterima (Feldman et al., 2007).

2.2.6.2 Tokenizing

Tahap Tokenizing adalah tahap pemotongan terhadap string input berdasarkan tiap kata yang menyusunnya (Francis & Flynn 2010). Karakter selain huruf akan dianggap sebagai delimiter dan akan dihilangkan atau dihapus untuk mendapatkan kata-kata penyusun teks (Riyani 2019).

2.2.6.3 Stopword Removal

Tahap stopword removal merupakan tahap menghilangkan kata-kata yang tidak penting dari suatu teks. Stopword adalah kata-kata yang tidak deskriptif yang dapat dihilangkan dalam pendekatan bag-of-words (Feldman et al., 2007).

2.2.6.4 Stemming

Tahap stemming merupakan tahap untuk mencari root kata dari tiap kata hasil proses filtering (Francis & Flynn 2010). Pada tahap ini dilakukan proses pengembalian berbagai bentukan kata ke dalam suatu representasi yang sama. Pencarian kata dasar dilakukan dengan cara menghilangkan semua imbuhan dari kata, awalan kata, kata sisipan, dan akhiran kata (Francis & Flynn 2010).

2.2.7 Term Frequency Inverse Document (TF-IDF)

Metode TF-IDF merupakan metode yang digunakan untuk menghitung bobot setiap kata yang sering digunakan pada teknologi information retrieval atau IR. Metode IR dikenal sebagai metode yang efisien, mudah digunakan dan memberikan hasil yang akurat (Ma'arif, 2015).

Metode TF-IDF merupakan cara untuk memberikan bobot hubungan dari suatu kata (term) terhadap dokumen. TF-IDF juga sebagai sebuah ukuran statistik data yang

digunakan untuk melakukan evaluasi seberapa pentingnya sebuah kata yang berada di dalam sebuah dokumen atau sebuah kelompok kata. Untuk setiap kalimat pada dokumen tunggal didefinisikan sebagai dokumen. Frekuensi dari berapa banyak kemunculan kata di dalam dokumen yang diberikan menyatakan kepentingan dari kata itu di dalam dokumen tersebut. Frekuensi dari dokumen yang memuat kata tersebut juga memperlihatkan seberapa umumnya kata tersebut. Bobot kata akan semakin besar jika kata tersebut sering muncul dalam suatu dokumen dan akan semakin kecil jika muncul di dalam banyak dokumen (Putra, 2016).

Rumus yang digunakan untuk menghitung bobot (W) masing-masing dokumen terhadap kata kunci pada algoritma TF-IDF adalah sebagai berikut:

$$W_{TF.IDF}(t_i.d_j) = f(t_i.d_j) \times (\log (D/ t_i)) \quad (II-1)$$

Keterangan:

$W_{TF.IDF}(t_i.d_j)$: pembobotan kata i pada dokumen j

$f(t_i.d_j)$: banyak kata atau term i pada dokumen j

D : total dokumen dalam dataset

t_i : total dokumen yang memunculkan kata i

2.2.8 Cosine Similarity

Cosine Similarity merupakan ukuran kesamaan antara dua buah vektor di dalam sebuah ruang dimensi yang berasal dari nilai berupa sudut cosinus dari perkalian dua buah vektor yang dibandingkan. Karena nilai cosinus dari nol adalah satu dan untuk sudut yang lain adalah kurang dari satu, maka apabila nilai similarity mendekati satu atau adalah satu nilai tersebut dapat dikatakan mirip (Dewa, 2016). Rumus dari cosine similarity adalah sebagai berikut:

$$Sim(\vec{q}, \vec{d}) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}| |\vec{d}|} = \frac{\sum_{k=1}^t w_{qk} \times w_{dk}}{\sqrt{\sum_{k=1}^t (w_{qk})^2} \cdot \sqrt{\sum_{k=1}^t (w_{dk})^2}} \quad (II - 2)$$

Keterangan:

\vec{q} : Vektor q

\vec{d} : Vektor d

w_{qk} : bobot term q dalam blok w_{qk}

w_{dk} : bobot term d dalam blok w_{dk}

k : Jumlah term dalam kalimat

t : Jumlah vector

2.3 Penelitian Lain yang Relevan

Penelitian terkait pernah dilakukan oleh (Merawati & Hartati, 2018). Pada penelitian tersebut CBR digunakan sebagai metode dalam sistem rekomendasi topik skripsi dengan inputan nilai mata kuliah dan output berupa dosen pembimbing dan topik skripsi. Hasil implementasi metode CBR menunjukkan nilai akurasi yang tinggi sehingga mampu memberikan rekomendasi topik skripsi bagi mahasiswa.

(Hidayat et al., 2020) pernah melakukan penelitian menggunakan metode CBR untuk diterapkan pada sistem rekomendasi topik skripsi dengan inputan berupa transkrip nilai mata kuliah mahasiswa. Hasil dari penelitian tersebut menunjukkan nilai akurasi pada sistem dari metode yang diterapkan memiliki nilai akurasi sistem hingga sampai lebih dari 80%. Hal tersebut menunjukkan metode CBR dapat diterapkan pada sistem rekomendasi topik skripsi secara akurat.

Penerapan metode CBR dalam rekomendasi topik skripsi juga dilakukan oleh (Utami et al., 2021). Sistem yang dikembangkan pada penelitian ini adalah sistem CBR untuk melakukan perolehan hasil rekomendasi terhadap topik konsentrasi skripsi mahasiswa. Penelitian tersebut menggunakan data yang didapat dari mahasiswa S1 Teknik Informatika IST AKPRIND Yogyakarta berjumlah sebanyak 115 data yang terdiri dari 80 data sebagai data latih dan 35 data sebagai data uji. Fitur dalam penelitian menggunakan nilai IPK dan nilai mata kuliah. Proses rekomendasi dilakukan dengan menggunakan Metode Nearest Neighbor dan Manhattan Distance. Penelitian mendapat

hasil nilai akurasi 97.14% pada metode KNN dan 94.29% pada metode *manhattan distance*.

2.4 Kesimpulan

Bab ini berisikan tentang metode-metode yang dinilai cocok untuk diterapkan dalam penyelesaian masalah terhadap penelitian yakni dengan menggunakan metode CBR dan cosine similarity measure sebagai tahap pencarian data yang relevan.

BAB III

METODOLOGI PENELITIAN

3.1 Pendahuluan

Pada bab ini membahas tentang rangkaian tahapan-tahapan dalam penelitian dan simulasi perhitungan similaritas serta metode yang digunakan sesuai dengan rumusan masalah agar sistem dapat berfungsi dengan baik mengacu pada tujuan penelitian.

3.2 Pengumpulan Data

Pada subbab pengumpulan data ini berisikan tentang metode akuisisi data yang meliputi jenis data, sumber data, dan metode pengumpulan data.

3.2.1 Jenis Data

Jenis data yang digunakan peneliti dalam melakukan penelitian ini adalah:

3.2.1.1 Data Sekunder

Data yang dikumpulkan berjenis data sekunder dimana data sebelumnya telah dimiliki oleh admin jurusan teknik informatika fakultas ilmu komputer universitas sriwijaya yang berisikan informasi tentang tugas akhir mahasiswa

dengan format excel yang selanjutnya data tersebut akan diseleksi dan ditransformasikan oleh peneliti melalui tahapan pre-processing pada subab tahapan penelitian. Gambar III-1 menunjukkan contoh data yang telah dikumpulkan.

Ind_Aksh	Ind	Ind1	Ind2	Ind3	Ind4	Ind5	Ind6	Ind7	Ind8	Ind9	Ind10	Ind11	Ind12
0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001	0001000001
0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002	0001000002

Gambar III-1 Data Tugas Akhir Mahasiswa

3.2.2 Sumber Data

Data bersumber dari dokumen kearsipan mahasiswa jurusan teknik informatika universitas sriwijaya yang diambil secara langsung.

3.2.3 Metode Pengumpulan Data

Data yang digunakan dikumpulkan dengan menggunakan metode:

3.2.3.1 Dokumentasi

Metode yang digunakan berupa dokumentasi dimana data yang dikumpulkan telah ada pada dokumen sebelumnya yakni dokumen kerasipan mahasiswa yang dimiliki oleh jurusan teknik informatika dan telah disetujui oleh admin jurusan untuk digunakan dalam penelitian.

3.3 Tahapan Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini adalah:



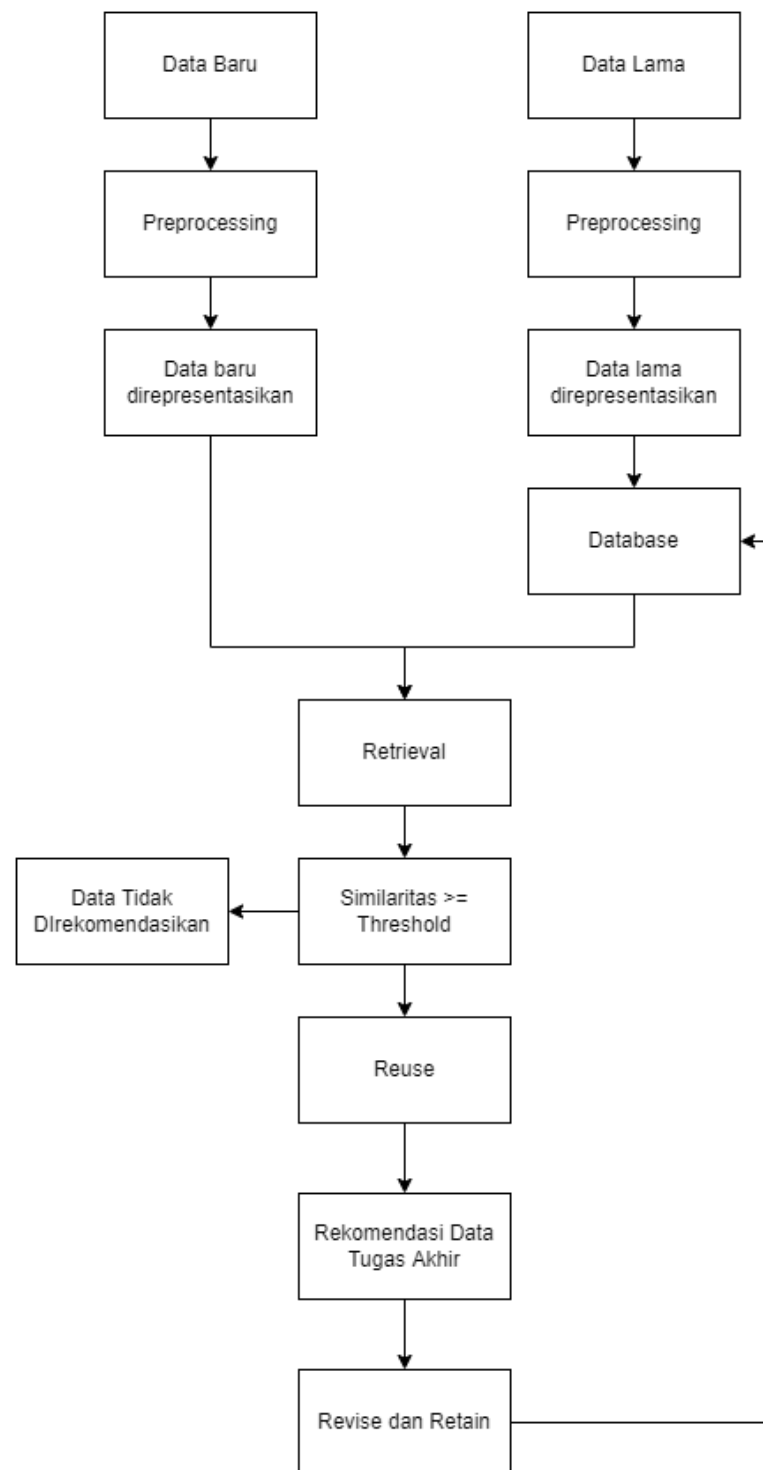
Gambar III-2 Tahapan Penelitian

3.3.1 Mengumpulkan Data Penelitian

Data tugas akhir mahasiswa berformat excel dikumpulkan dari staff admin jurusan teknik informatika fakultas ilmu komputer universitas sriwijaya dengan atribut judul, abstrak, bidang riset, nama, nim, kelas, angkatan, status, dosen pembimbing 1 dan dosen pembimbing 2. Data yang telah dikumpulkan sebanyak 249 data untuk bilingual angkatan 2012-2016, 280 data untuk regular angkatan 2012-2016. Jika dijumlahkan maka data yang telah dikumpulkan sebanyak 526 data.

3.3.2 Kerangka Kerja Penelitian

Kerangka kerja dari penelitian ini adalah sebagai berikut:



Gambar III-3 Kerangka Kerja Penelitian

Berdasarkan diagram alur pada kerangka kerja penelitian tersebut terdapat perbedaan pada proses sebelum memasuki tahapan retrieval dimana pada proses pertama data baru tidak dimasukkan ke dalam case base melainkan dijadikan data yang akan dibandingkan dengan data yang ada di dalam case base. Sebaliknya data lama dimasukkan ke dalam case base setelah memasuki tahapan perepresentasian. Baik data baru maupun data lama sama-sama memasuki tahapan preprocessing dan tahapan perepresentasian data dimana pada tahap preprocessing proses yang dilakukan meliputi case folding, tokenizing, stopword removal dan stemming. Tujuan dilakukannya proses tersebut adalah untuk mempersiapkan data sehingga dapat diolah lebih lanjut. Selanjutnya untuk perepresentasian data data diberikan bobot pada setiap katanya yang berguna untuk perhitungan similaritas nantinya yang mana penetapan bobot pada tahapan tersebut adalah menggunakan TF-IDF. Lalu setelah itu baru lah masing-masing data baru dan data lama akan dicocokkan atau dicari kemiripannya melalui tahapan retrieval. Pada tahapan retrieval diterapkan cosine similarity untuk proses perhitungan mencari nilai kemiripannya. Apabila perhitungan tersebut melewati nilai threshold maka akan dijadikan sebagai data hasil rekomendasi yang paling relevan berdasarkan kata kunci yang dimasukkan melalui tahapan reuse. Apabila tidak melewati nilai threshold maka data tersebut tidak dapat dijadikan sebagai data hasil rekomendasi. Data hasil rekomendasi akan disimpan kedalam database melalui tahapan

revise & retain dengan mempertimbangkan kesesuaian data dengan query dan dilakukan perbaikan apabila diperlukan.

3.3.2.1 Preprocessing

Preprocessing merupakan proses awal terhadap suatu teks untuk mempersiapkan teks dari data yang tidak terstruktur menjadi data yang terstruktur. Tujuan pra-pengolahan adalah untuk penyeragaman dan memudahkan untuk membaca data masukan. Tahap pertama dalam preprocessing adalah cleaning. Pada tahap cleaning, data akan dibersihkan dengan menghilangkan karakter dan tanda baca sehingga dapat mengurangi noise dalam proses klasifikasi. Kata atau karakter yang dihilangkan pada komentar adalah angka, dan tanda baca.

Tahap kedua adalah case folding. Pada tahap ini, semua huruf pada dokumen akan diseragamkan menjadi huruf kecil, dan hanya karakter 'a' hingga 'z' yang diterima. Tahap ketiga adalah normalisasi yakni proses pengubahan kata yang tidak baku menjadi kata yang lebih baku. Pada tahap ini akan digunakan kamus bahasa Indonesia. Selanjutnya tahap keempat tokenizing yakni memecah komentar menjadi beberapa token. Selanjutnya adalah tahap stopwords removal yang berfungsi menghilangkan kata-kata yang tidak memiliki makna dan sering muncul dalam kumpulan kata. Tahap terakhir

adalah proses stemming, tahap mencari root atau kata dasar dari hasil tahap filtering.

Data yang dilakukan proses preprocessing disini adalah data dengan atribut judul, dan abstrak.

3.3.2.2 Data direpresentasikan

Setelah melakukan preprocessing, tahap selanjutnya adalah perepresentasian data yakni dengan memberikan bobot untuk setiap kata pada dokumen yang berguna untuk proses perhitungan similaritas pada tahapan retrieval. Pembobotan dilakukan dengan menggunakan algoritma gabungan term frequency dan inverse document frequency yang dikenal dengan istilah TF-IDF. Penggunaan algoritma tersebut adalah untuk menganalisa hubungan antara sebuah frase/kalimat dengan sekumpulan dokumen yakni sebagai penentuan urutan peringkat data berdasarkan query yang digunakan.

3.3.2.3 Retrieval

Setelah pembobotan selesai dilakukan maka akan dilanjutkan dengan tahapan retrieval. Pada tahapan ini terdapat proses menentukan tingkat kemiripan antar data. Data baru dan data lama akan dihitung nilai kemiripannya guna mengetahui apakah data tersebut memiliki keterkaitan dengan query yang

diinputkan. Perhitungan similaritas pada tahapan retrieval akan dihitung dengan menggunakan algoritma cosine similarity.

3.3.2.4 Reuse

Pada tahapan *reuse* hasil perhitungan similaritas berupa nilai cosine similarity akan dijadikan sebagai rekomendasi data tugas akhir dan digunakan kembali apabila nilai tersebut melebihi atau sama dengan threshold yang telah ditentukan. Data yang memiliki nilai similaritas tidak melebihi atau melawati nilai threshold maka data tersebut tidak ditampilkan sebagai data hasil rekomendasi atau data yang relevan berdasarkan kata kunci yang dimasukkan oleh pengguna.

3.3.2.5 Revise dan Retain

Setelah data dijadikan sebagai data hasil rekomendasi dan dapat dilihat oleh pengguna sesuai dengan urutan nilai similaritasnya dari yang bernilai tinggi sampai ke yang bernilai rendah. Selanjutnya data masuk ke tahapan revise & retain dimana data akan dievaluasi dengan mempertimbangkan nilai kemiripan dari hasil rekomendasi dengan query. Evaluasi data dapat dilakukan dengan cara menambah, mengubah atau menghapus data tersebut apabila evaluasi diperlukan. Bersamaan dengan dievaluasinya data, data hasil rekomendasi ditambahkan atau disimpan kedalam database sebagai rekam data

atau sebagai bank data untuk proses pencocokan atau pencarian kemiripan lainnya dengan data baru nantinya.

3.3.3 Kriteria Data Penelitian

Kriteria data pada penelitian ini adalah:

1. Data yang digunakan dalam pengujian penelitian adalah data tugas akhir yang memiliki atribut judul dan abstrak.
2. Sebelum pengujian data penelitian terlebih dahulu melewati proses preprocessing dan juga perepresentasian terhadap data
3. Data penelitian menghasilkan nilai similaritas yang paling tinggi.

3.3.4 Format Data Penelitian

Format data penelitian yang digunakan untuk mengevaluasi hasil data penelitian ditunjukkan pada table III-2

Tabel III-1 Format Data Penelitian

No	Keywords	Similarity
1		
2		
3		
4		

5		
...	...	
50		

3.3.5 Alat Penelitian

Alat yang digunakan peneliti dalam melakukan penelitian adalah sebagai berikut:

1. Hardware (Perangkat Keras):

Laptop Vivobook X430FN_S430FN dengan spesifikasi sebagai berikut:

- Processor: Intel Core i5-8265U
- Memory: 8192MB RAM
- Operating System: Windows 10 64-bit

2. Software (Perangkat Lunak)

- Visual Studio Code
- Xampp, PHP, Mysql
- Google Chrome

3.3.6 Pengujian Penelitian

Pengujian penelitian dilakukan berdasarkan alur kerangka kerja penelitian yang telah dipaparkan sebelumnya. Penelitian diuji dengan memasukkan keyword ke dalam sistem berupa teks. Agar pencarian data terfokuskan maka teks yang dimasukkan

adalah teks berupa metode-metode yang sering digunakan dalam penelitian tugas akhir mahasiswa teknik informatika contohnya naïve bayes, fuzzy, algoritma greedy dan yang lainnya. Data dengan nilai similaritas tertinggi berdasarkan hasil pencarian akan direkap dan dimasukkan ke dalam tabel III-2. Proses pengujian dilakukan secara berulang dengan keyword yang berbeda sebanyak 50 kali untuk memperoleh data dengan nilai *cosine similarity* tertinggi.

3.3.7 Analisis Hasil Pengujian

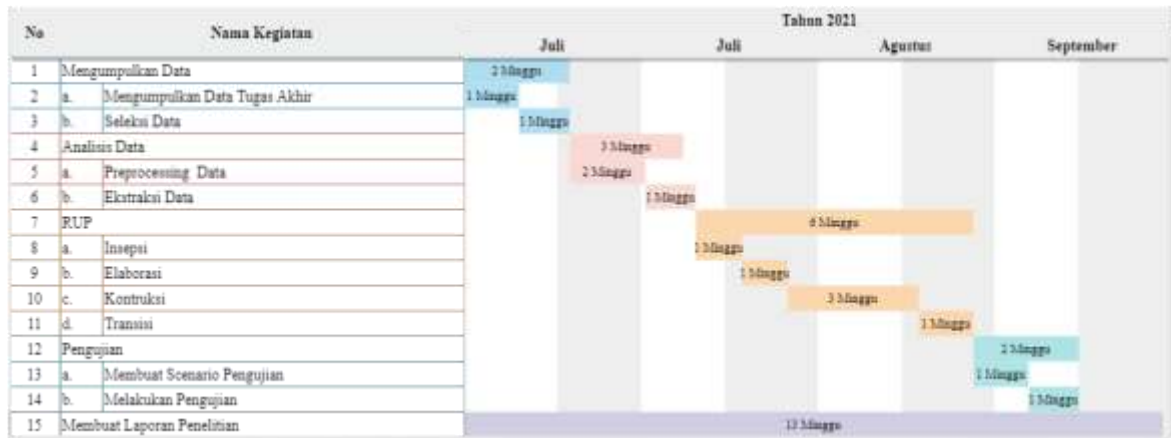
Hasil pengujian akan dianalisis dengan menghitung nilai rata-rata similaritas berdasarkan nilai-nilai yang diperoleh dari tabel III-2 lalu nilai rata-rata dibandingkan terhadap data yang memiliki nilai similaritas tertinggi dan terendah dari keseluruhan data yang ada di dalam tabel.

3.4 Metode Pengembangan Perangkat Lunak

Pengembangan sistem ini akan menggunakan metode RUP dimana fase-fasenya telah dijelaskan pada bab II kajian literatur subbab *rational unified model*.

3.5 Manajemen Proyek Penelitian

Manajemen proyek dari penelitian yang akan dilakukan ditunjukkan pada gambar III-4



Gambar III-4 Manajemen Proyek Penelitian

3.6 Kesimpulan

Bab ini telah membahas mengenai tahapan penelitian, kerangka kerja penelitian yang akan dilakukan, perolehan data, proses pengujian penelitian serta metode pengembangan perangkat lunak yang akan digunakan. Selain itu juga menjelaskan mengenai teknik pengumpulan data sebagai bahan uji dalam penelitian

BAB IV

PENGEMBANGAN PERANGKAT LUNAK

4.1 Pendahuluan

Telah disebutkan sebelumnya pada bab 3 diperlukan sistem yang dapat digunakan dalam melakukan penelitian, oleh karena itu penulis menggunakan metode RUP sebagai pemrograman berorientasi objek untuk pengembangan perangkat lunak. Pada bab ini akan membahas mengenai proses pengembangan perangkat lunak yang akan digunakan dalam penelitian.

4.2 Fase Insepsi

Untuk mendefinisikan kebutuhan sistem yang bertujuan untuk mengatasi masalah yang dihadapi, pada fase ini akan dibahas mengenai kebutuhan fungsional dan kebutuhan non fungsional dari pada sistem yang akan dikembangkan.

4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional dari suatu sistem berisikan fungsi-fungsi yang harus dilakukan oleh sebuah sistem untuk dapat membuat sistem tersebut bekerja dengan baik agar terfokus pada tujuan yang ingin dicapai. Sistem tersebut harus memiliki

kemampuan yang sesuai dengan masalah yang ingin diatasi, oleh karena itu sistem yang akan dikembangkan memiliki kemampuan sebagai berikut:

Tabel IV-1 Kebutuhan Fungsional

No.	Fungsional
1	Melakukan proses pencarian data tugas akhir terhadap <i>keyword</i> yang diinputkan.
2	Melakukan proses preprocessing data.
3	Melakukan proses pembobotan kata.
4	Melakukan proses perhitungan untuk memperoleh nilai similaritas antar data.

4.2.1.1 Analisis Proses Perhitungan Similaritas

Sebelum sistem menampilkan nilai similaritas dari masing-masing data yang relevan terhadap kata kunci secara terurut, terdapat suatu proses yang dilakukan oleh sistem yakni perhitungan similaritas untuk menghasilkan nilai tersebut yang dimana sesuai dengan penerapan perhitungan menggunakan *cosine similarity*. Langkah awal sistem dalam melakukan proses perhitungan similaritas adalah dengan melakukan indexing terlebih dahulu pada setiap data seperti yang tertera pada tabel IV-2

Pada tabel IV-2 terdapat 5 data atau dokumen ditambah 1 data berupa query. Setiap kata pada data dilakukan ekstraksi untuk dihitung frekuensi kemunculan kata yang sama di dalam data tersebut.

Tabel IV-2 Indexing Keywords

Document	Keywords	research field
Query	kelompok:1; clustering:1; balita:1; gizi:1; metode:1; fuzzy:1; klasifikasi:1; c-means:1; algoritma:1; perangkat:1 lunak:1;	Soft Computing
Document 1	kelompok:4; status:6; gizi:7; balita:5; metode:4; fuzzy:5; c-means:4; algoritma:4; genetika:4; deteksi:1; teliti:2; kembang:1; hasil:2; perangkat:1; lunak:1; bangkit:1; bilang:1; random:1; hitung:4; pusat:1; cluster:4; fungsi:2; obyektif:1; fitness:1; seleksi:1; crossover:3; mutasi:3; parameter:1; badan:2; berat:1; umur:1; uji:2; ubah:2; matriks:1; partisi:1; inisial:1; populasi:3; iterasi:2; peluang:4; ukur:2; validasi:1; davies:1; boulding:1; index:1; dapat:2; nilai:3;	Soft Computing
Document 2	banding:3; kelompok:7; status:6; gizi:6; hamil:7; fuzzy:10; c-means:3; subtractive:6; clustering:6; guna:1; mudah:1; teliti:3; metode:5; variabel:1; umur:1; berat:2; badan:3; lingkaran:1; lengan:1; usia:1; tambah:1; hasil:3; validasi:5; davies:1; bouldin:1; index:2; dbi:2; sum:1; xbi:2; sesuai:2; variasi:1; data:5; rendah:3; dapat:3; sehingga:1 pilih:1; sebagai:1;	Soft Computing

Document 3	aman:4; integritas:6; alamat:9; bitcoin:12; karyawan:6; algoritma:5; md:5; fokus:1; teliti:3; otentikasi:4; message:1; digest:1; kembang:2; aplikasi:2; gaji:3; lengkap:3; application:1; programming:1; interface:1; sedia:1; indodax:1; data:5; pribadi:1; salah:1; satu:1; isi:1; simpan:4; basis:4; bangkit:2; nilai:4; hash:4; pisah:2; proses:1; banding:1; dasar:1; uji:1; deteksi:1; ms:2; rata:1;	Kriptografi
Document 4	klasifikasi:2; sakit:3; kanker:3; payudara:3; metode:5; backpropagation:3; adaptive:3; learning:3; rate:3; iring:1; maju:1; teknologi:1; deteksi:1; ahli:1; medis:1; salah:1; proses:3; ajar:1; lambat:1; optimasi:1; tahan:1; update:1; bobot:2; teliti:2; data:5; total:2; bagi:1; latih:1; persentase:1; dasar:1; training:2; nilai:3; iterasi:2; beda:1; hasil:2; rata:1; akurasi:2; baik:1; oleh:1;	Data Mining
Document 5	implementasi:1; metode:6; analitic:1; network:1; process:1; anp:5; technique:1; preference:1; similarity:1; ideal:1; solution:1; topsis:4; sistem:4; dukung:3; putus:3; distribusi:3; beras:3; masyarakat:1; miskin:3; keluarga:2; raskin:3; salah:1; program:1; perintah:1; upaya:1; tingkat:1; tahan:1; pangan:1; lindung:1; proses:1; alami:1; hambat:1; pilih:1; terima:2; bantu:3; tentu:1; urut:1; sesuai:1; kriteria:3; butuh:1; milik:1; jenis:1; rangking:4; tulis:1; gabung:1; awal:1; hitung:1; bobot:1; uji:2; teliti:1; banding:1; hasil:3; manual:1; staff:1; camat:1; data:1; persentase:1; cocok:1;	Soft Computing

Selanjutnya, dari angka yang diperoleh dihitung DF (Document Frequency) dan IDF (Inverse Document Frequency) nya. DF merupakan jumlah dokumen yang memunculkan masing-masing kata dari hasil ekstraksi. IDF dihitung dengan menerapkan rumus $(\log(D/t_i))$ sehingga menghasilkan nilai seperti yang tertera pada tabel IV-3

Tabel IV-3 IDF Keywords

Term	TF						DF	IDF
	Q	D1	D2	D3	D4	D5		
adaptive					3		1	0.77815125
ahli					1		1	0.77815125
ajar					1		1	0.77815125
akurasi					2		1	0.77815125
alamat				9			1	0.77815125
alami						1	1	0.77815125
algoritma	1	4		5			3	0.301029996
aman				4			1	0.77815125
anality						1	1	0.77815125
anp						5	1	0.77815125
...
xbi			2				1	0.77815125

TF-IDF didapat dengan menerapkan rumus $TF \cdot IDF$ yang ditunjukkan pada tabel IV-4

Tabel IV-4 TF-IDF Keywords

Term	TF-IDF					
	Q	D1	D2	D3	D4	D5

adaptive	0	0	0	0	2.334453751	0
ahli	0	0	0	0	0.77815125	0
ajar	0	0	0	0	0.77815125	0
akurasi	0	0	0	0	1.556302501	0
alamat	0	0	0	7.003361253	0	0
alami	0	0	0	0	0	0.77815125
algoritma	0.301029996	1.204119983	0	1.505149978	0	0
aman	0	0	0	3.112605002	0	0
anality	0	0	0	0	0	0.77815125
anp	0	0	0	0	0	3.890756252
...
xbi	0	0	1.556302501	0	0	0

Setelah nilai TF-IDF diperoleh maka dilanjutkan dengan melakukan perhitungan untuk mendapatkan nilai panjang vektor dari tiap-tiap dokumen. Pada tabel IV-5 nilai panjang vektor tiap dokumen termasuk kata kunci didapatkan melalui penjumlahan akar kuadrat dari TF-IDF setiap di dalam dokumen lalu diakarkan.

Tabel IV-5 Panjang Vektor Dokumen dan Query

Term	Panjang vektor q dan d1,d2,d3,d4,d5					
	q	d1	d2	d3	d4	d5
adaptive	0	0	0	0	5.449674316	0
ahli	0	0	0	0	0.605519368	0
ajar	0	0	0	0	0.605519368	0
akurasi	0	0	0	0	2.422077474	0
alamat	0	0	0	49.04706885	0	0
alami	0	0	0	0	0	0.605519368
algoritma	0.090619058	1.449904933	0	2.265476457	0	0
aman	0	0	0	9.688309896	0	0
anality	0	0	0	0	0	0.605519368
anp	0	0	0	0	0	15.13798421

...
xbi	0	0	2.422077474	0	0	0
Total Jumlah	1.460562786	86.93960065	112.8716957	278.6685272	60.17047916	108.9703742
Akar	1.208537458	9.324140746	10.62410917	16.69336776	7.756963269	10.43888759

Terakhir, sistem melakukan perhitungan $Wqk * Wdk$ yang dimana perhitungan tersebut ditunjukkan pada tabel IV-6 $Wqk * Wdk$ didapatkan melalui mengkalikan TF-IDF setiap kata antara dokumen kata kunci dengan dokumen yang memiliki kata kunci yang sama dan dijumlahkan per dokumen.

*Tabel IV-6 $Wqk * Wdk$*

Term	$Wqk * Wdk$				
	Wqd1	Wqd2	Wqd3	Wqd4	Wqd5
adaptive	0	0	0	0	0
ahli	0	0	0	0	0
ajar	0	0	0	0	0
akurasi	0	0	0	0	0
alamat	0	0	0	0	0
alami	0	0	0	0	0
algoritma	0.362476233	0	0.453095291	0	0
aman	0	0	0	0	0
anality	0	0	0	0	0
anp	0	0	0	0	0
...
xbi	0	0	0	0	0
Jumlah Total	4.155925153	2.931158214	0.939733023	0	0.037618018

Dengan membagi nilai $Wqk * Wdk$ dengan perkalian panjang vektor antara dokumen kata kunci dan dokumen pembanding maka didapatkan hasil perhitungan similaritas seperti yang ditunjukkan pada

tabel IV-7 Dari hasil simulasi perhitungan similaritas seperti yang tertera pada tabel urutan data paling relevan terhadap dokumen kata kunci adalah D1,D2, D3,D5 dan D4.

Tabel IV-7 Cosine Similarity Antar Dokumen

Document	Cosine Similarity	Research Field
D1	0.368806682	Soft Computing
D2	0.228289859	Soft Computing
D3	0.0465801	Kriptografi
D4	0	Data Mining
D5	0.002981821	Soft Computing

4.2.2 Kebutuhan Non Fungsional

Dalam mencapai tujuan kebutuhan sistem tidak luput dari kebutuhan terkait dengan dukungan terhadap sistem yang akan dikembangkan. Maka, untuk dapat mendukung kebutuhan fungsional kebutuhan non fungsional diperlukan dalam hal pengembangan sistem yang dimana kebutuhan non fungsional tersebut terdiri dari sebagai berikut:

Tabel IV-8 Non Fungsional

No.	Non Fungsional
1	Sistem yang dikembangkan mudah digunakan
2	Sistem yang dikembangkan dapat menampilkan seluruh data yang disimpan
3	Sistem yang dikembangkan memiliki tampilan menarik
4	Sistem yang dikembangkan berbasis website

4.3 Fase Elaborasi

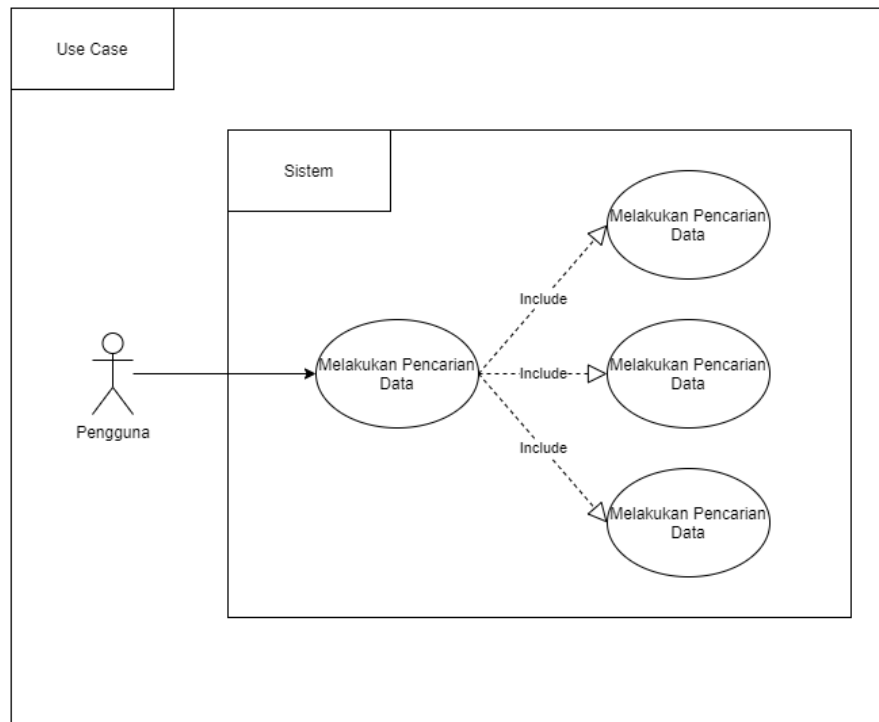
Memasuki fase elaborasi dimana pada fase tersebut akan difokuskan membahas mengenai perancangan arsitektur dari pada sistem yang dikembangkan diantaranya use case diagram, activity diagram, sequence diagram, entity relationship diagram, dan perancangan antar muka

4.3.1 Use Case

Pada subsubab *use case* akan dijelaskan rancangan desain perangkat lunak melalui pemodel *use case* diantaranya *use case diagram*, definisi aktor, definisi *usecase*, dan *use case scenario*.

4.3.1.1 Use Case Diagram

Ilustrasi mengenai interaksi antar aktor dan sistem digambarkan melalui diagram use case sebagai berikut:



Gambar IV-1 Use Case Diagram

4.3.1.2 Definisi Aktor

Definisi aktor pada use case diagram dijelaskan pada tabel IV-10

Tabel IV-9 Definisi Aktor

No	Aktor	Definisi
1	Pengguna	Actor pada use case diagram merupakan orang yang dapat mengoperasikan atau menggunakan keseluruhan fitur yang ada pada sistem perangkat lunak.

4.3.1.3 Definisi Use Case

Definisi *use case* pada *use case* diagram akan dijelaskan pada tabel IV-

11

Tabel IV-10 Definisi Use Case

No	Use Case	Definisi
1	Melakukan Pencarian Data	<i>Use case</i> melakukan pencarian data merupakan proses untuk melakukan pencarian data berupa data tugas akhir mahasiswa yang relevan terhadap <i>keyword</i> yang dimasukkan ke dalam sistem perangkat lunak.

2	Melakukan Preprocessing Data	<i>Use case</i> melakukan preprocessing data merupakan suatu proses untuk mempersiapkan teks menjadi yang dapat sebelum data tersebut memasuki proses pembobotan.
3	Melakukan Pembobotan Kata	<i>Use case</i> melakukan pembobotan kata merupakan suatu proses untuk membobotkan setiap kata dalam data sebelum dapat diproses ke perhitungan similaritas.
4	Melakukan Perhitungan Similaritas	<i>Use case</i> melakukan perhitungan similaritas merupakan suatu proses untuk memperoleh nilai similaritas antar data untuk mencari data yang paling relevan terhadap <i>keyword</i> dan ditampilkan.

4.3.1.4 Use Case Scenario

Use case scenario pada *use case* diagram akan dijelaskan pada tabel IV-12 – IV-15

Tabel IV-11 Use Case Scenarion 01

Identifikasi			
No	01		
Nama Use Case	Melakukan Pencarian Data		
Aktor	Actor		
Tujuan	Melihat data hasil pencarian		
Deskripsi	melakukan pencarian data berupa data tugas akhir mahasiswa yang relevan terhadap <i>keyword</i> yang dimasukkan ke dalam sistem perangkat lunak.		
Skenario Normal			
Aksi Aktor		Reaksi Sistem	
1	Memasukkan kata kunci		
2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		
		3	Menampilkan data yang relevan terhadap kata kunci
4	Melihat data dari hasil pencarian data		
Skenario Alternatif			
1	Memasukkan kata kunci		

2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		
		3	Tidak menampilkan data yang relevan terhadap kata kunci
4	Kembali Memasukkan kata kunci dan mengklik tombol “Submit”		

Tabel IV-12 Use Case Scenarion 02

Identifikasi			
No	02		
Nama Use Case	Melakukan Preprocessing Data		
Aktor	Actor		
Tujuan	Mempersiapkan teks menjadi data yang dapat diolah		
Deskripsi	<i>Use case</i> melakukan preprocessing data merupakan suatu proses untuk mempersiapkan teks menjadi yang dapat sebelum data tersebut memasuki proses pembobotan.		
Skenario Normal			
Aksi Aktor		Reaksi Sistem	
1	Memasukkan kata kunci		
2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		

		3	Melakukan proses preprocessing data
4	Mengklik menu “Term” pada navbar		
		5	Menampilkan teks hasil preprocessing
6	Melihat teks hasil dari preprocessing data		

Tabel IV-13 Use Case Scenarion 03

Identifikasi	
No	03
Nama Use Case	Melakukan Pembobotan Kata
Aktor	Actor
Tujuan	Memperoleh nilai pembobotan setiap kata pada setiap data
Deskripsi	<i>Use case</i> melakukan pembobotan kata merupakan suatu proses untuk membobotkan setiap kata dalam data sebelum dapat diproses ke perhitungan similaritas.
Skenario Normal	
Aksi Aktor	Reaksi Sistem
1	Memasukkan kata kunci

2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		
		3	Melakukan proses pembobotan data
4	Mengklik menu “Tf-Idf” pada navbar		
		5	Menampilkan nilai hasil pembobotan kata
6	Melihat nilai hasil pembobotan kata		

Tabel IV-14 Use Case Scenarion 04

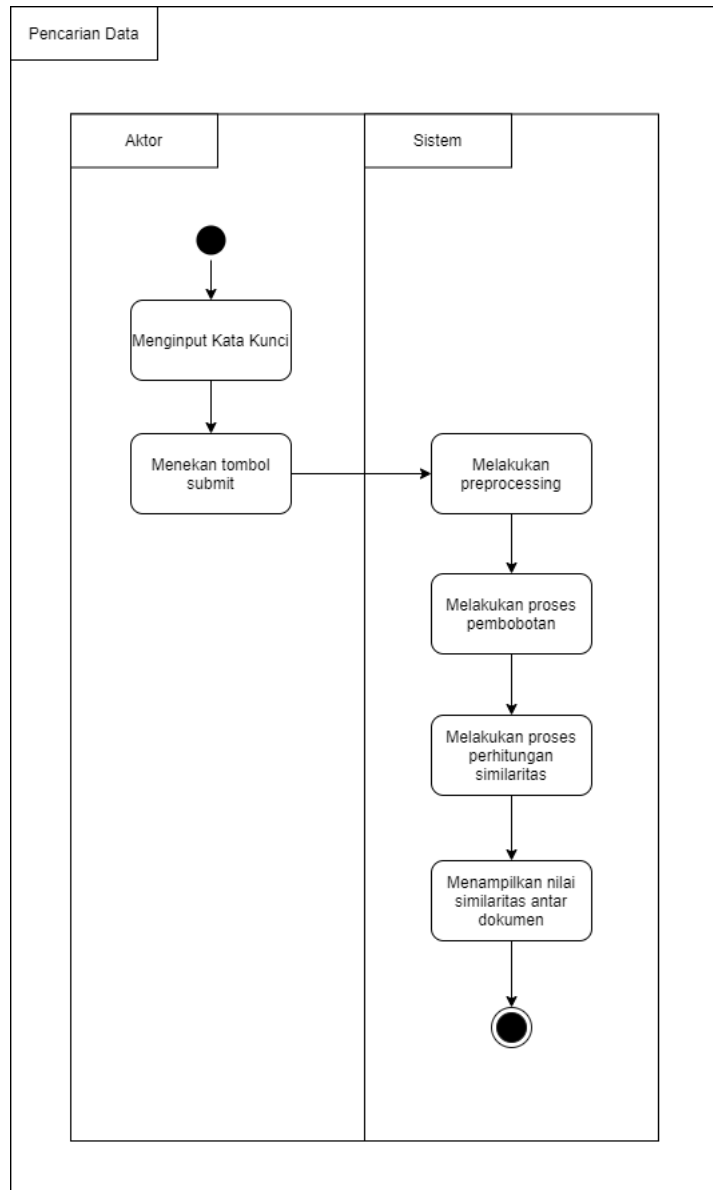
Identifikasi	
No	04
Nama Use Case	Melakukan Perhitungan Similaritas
Aktor	Actor
Tujuan	Memperoleh nilai similaritas antar data
Deskripsi	<i>Use case</i> melakukan perhitungan similaritas merupakan suatu proses untuk memperoleh nilai similaritas antar data untuk mencari data yang paling relevan terhadap <i>keyword</i> dan ditampilkan.

Skenario Normal			
Aksi Aktor		Reaksi Sistem	
1	Memasukkan kata kunci		
2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		
		3	Melakukan proses perhitungan similaritas
		4	Menampilkan nilai similaritas antar data tertinggi sampai terendah
5	Melihat nilai hasil perhitungan similaritas		
Skenario Alternatif			
1	Memasukkan kata kunci		
2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard		
		3	Melakukan proses perhitungan similaritas
		4	Tidak menampilkan nilai similaritas data

5	Kembali Memasukkan kata kunci dan mengklik tombol "Submit"		
---	---	--	--

4.3.2 Activity Diagram

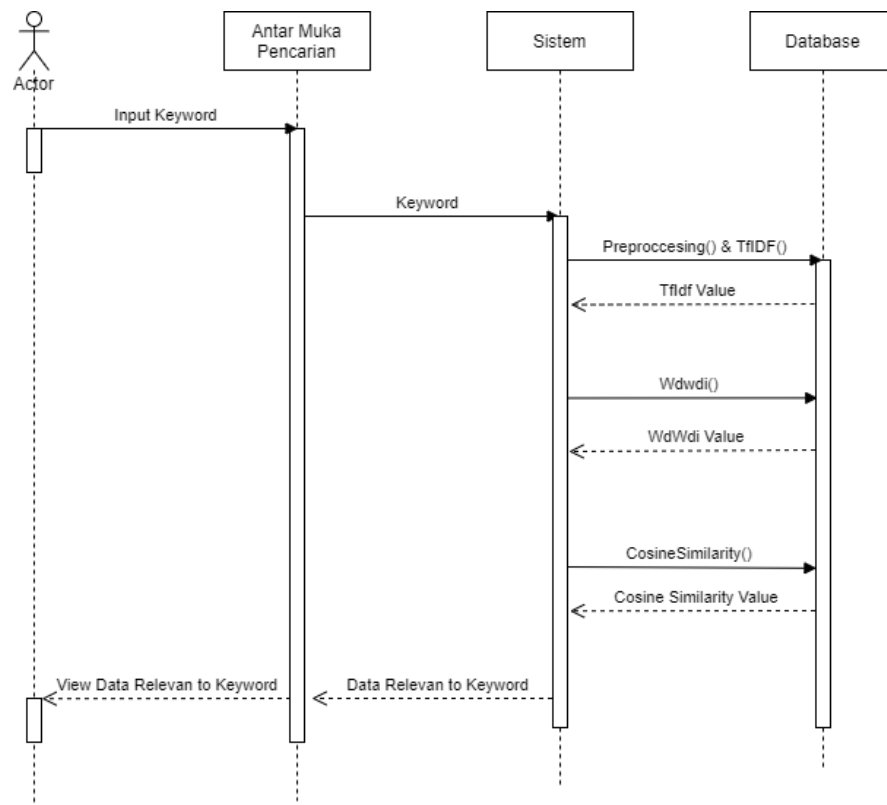
Aliran kerja atau aktivitas dari suatu sistem digambarkan melalui diagram aktivitas. Diagram aktivitas dari sistem pencarian data tugas akhir adalah sebagai berikut:



Gambar IV-2 Activity Diagram

4.3.3 Sequence Diagram

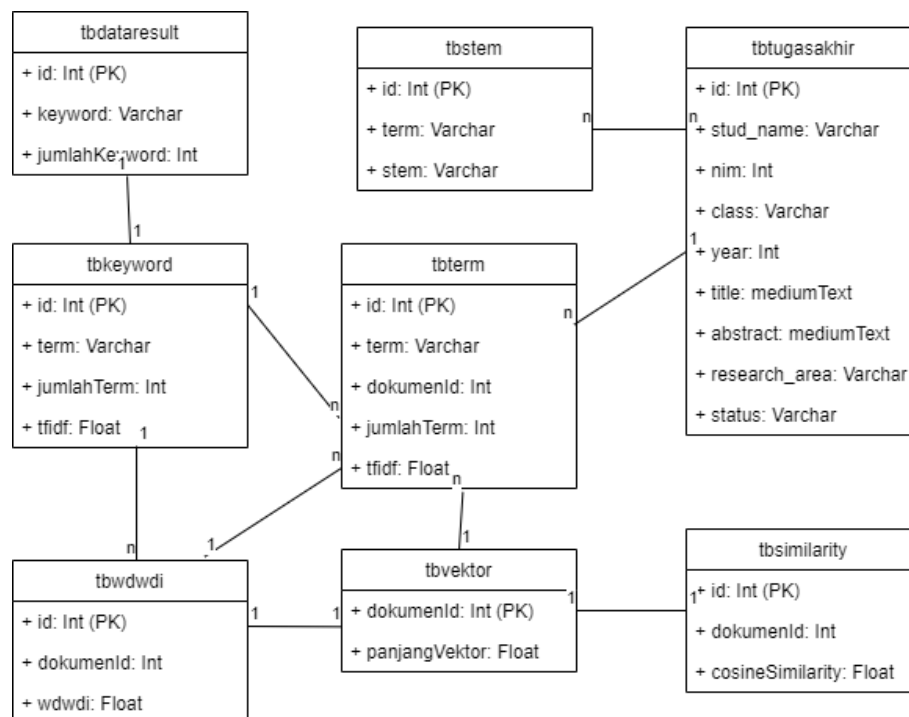
Untuk dapat mengetahui interaksi antar object, sequence diagram dirancang untuk mengetahui hal tersebut secara terperinci. Tujuan dibuatnya sequence diagram adalah untuk mengetahui urutan kejadian yang dapat menghasilkan output yang diinginkan. Berikut merupakan rancangan sequence diagram pada sistem yang dikembangkan.



Gambar IV-3 Sequence Diagram

4.3.4 Data Model

Dalam mempresentasikan hubungan antar data penggunaan data model dalam proses pengembangan perangkat lunak sangat penting untuk diterapkan. Dengan menerapkan konsep tersebut dapat lebih mudah untuk mengidentifikasi serta membatasi ruang lingkup dalam sistem yang akan dikembangkan. Berikut merupakan presentasi dari data model pada sistem pencarian data tugas akhir:



Gambar IV-4 Data Model

4.3.5 Perancangan Antar Muka

Pada subbab ini akan membahas mengenai perancangan antar muka dari sistem yang dibangun. Adapun rancangan antar muka yang dibangun meliputi halaman *home* dan halaman *result* adalah sebagai berikut:



Gambar IV-5 Rancangan Antar Muka Halaman Home

Halaman homepage merupakan halaman utama yang memiliki fungsi untuk melakukan pencarian data berdasarkan keyword yang diinputkan. Selain itu dihalaman home juga dapat melihat data yang ada di dalam database ketika tombol lihat diklik.

Navbar Term Vektor Tf-idf WdWdi

Pencarian Data Tugas Akhir

Hasil Pencarian

No	Judul TA	Abstract	Research Area	Nim	Nama	Kelas	Angkatan	Similarity

Gambar IV-6 Rancangan Antar Muka Halaman Result

Halaman result merupakan halaman yang selanjutnya muncul apabila pengguna telah menginputkan keyword dan menekan tombol submit atau enter di keyboard. Pada halaman tersebut menampilkan hasil pencarian berupa data tugas akhir yang telah melewati proses perhitungan similaritas pada setiap dokumen. Data yang muncul adalah data yang relevan terhadap keyword secara terurut dari nilai similaritas yang tertinggi.

4.4 Fase Kontruksi

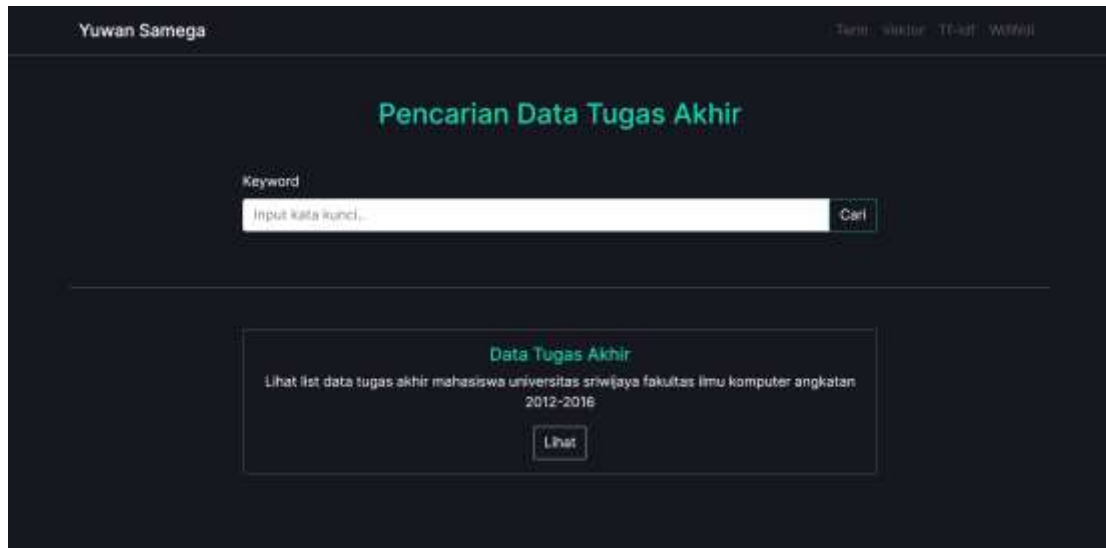
Fase kontruksi pada subab ini membahas mengenai hasil implementasi dari apa yang telah dirancang pada fase elaborasi subab 4.3 sebelumnya dengan melakukan sederet iterasi pengembangan sistem. Hasil yang diharapkan berupa produk yang siap digunakan dalam melakukan penelitian. Adapun subsubab yang akan dibahas adalah implementasi antar muka dari pada sistem yang dibangun.

4.4.1 Implementasi Sistem

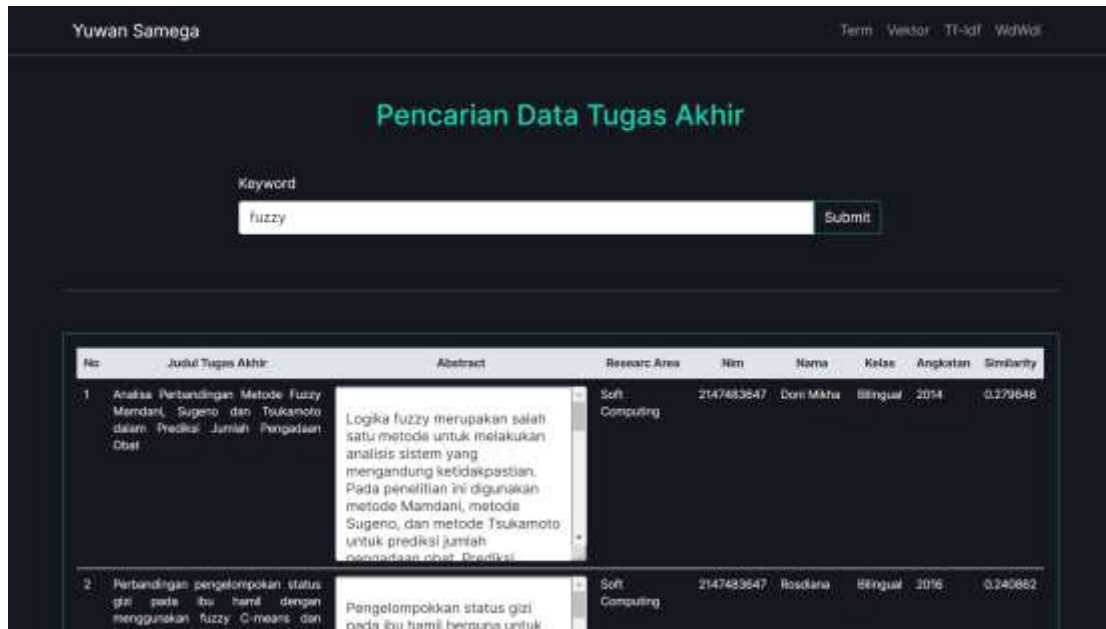
Perangkat lunak yang dikembangkan dengan berbasis pada website yakni menggunakan HTML dan CSS serta bahasa yang digunakan adalah bahasa PHP. Database yang digunakan pada perangkat lunak adalah MySql dan server APACHE.

4.4.1 Implementasi Antarmuka

Implementasi antar muka digambarkan berdasarkan rancangan antarmuka pada fase elaborasi. Antarmuka dari rancangan yang telah dibuat ditunjukkan pada gambar IV-7 dan IV-8



Gambar IV-7 Antar Muka Halaman Home



Gambar IV-8 Antar Muka Halaman Result

4.5 Fase Transisi

Pada fase transisi akan membahas mengenai pengujian sistem pencarian data tugas akhir berdasarkan keyword yang dimasukkan. Pembahasan tersebut diantaranya data pengujian dan rencana pengujian.

4.5.1 Pengujian Perangkat Lunak

Pengujian perangkat lunak akan dilakukan dengan menggunakan pengujian *black box testing*. Pengujian *black box testing* digunakan untuk memeriksa fungsionalitas suatu perangkat lunak yang telah didefinisikan pada fase inepsi. Pengujian perangkat lunak dengan *black box testing* berdasarkan *use case* ditunjukkan pada tabel IV-16 – IV-19

4.5.1.1 Pengujian Use Case Melakukan Pencarian Data

Pengujian untuk *use case* melakukan pencarian data ditunjukkan pada tabel IV-16

Tabel IV-15 Pengujian Use Case Melakukan Pencarian Data

Identifikasi	Deskripsi	Jenis Pengujian	Prosedur Pengujian	Keluaran yang Diharapkan	Hasil yang Dikeluarkan	Tingkat Pengujian	Kesimpulan
--------------	-----------	-----------------	--------------------	--------------------------	------------------------	-------------------	------------

UC-01-1	Memasukkan kata kunci	<i>Black Box Testing</i>	Memasukkan kata kunci ke dalam kotak pencarian	Keyword masuk di kotak pencarian	Keyword masuk di kotak pencarian	Pengujian Antarmuka	Terpenuhi
UC-01-2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard	<i>Black Box Testing</i>	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard setelah keyword dimasukkan	Menampilkan hasil berupa data hasil pencarian berdasarkan keyword masukan	Menampilkan hasil berupa data hasil pencarian berdasarkan keyword masukan	Pengujian Unit	Terpenuhi

4.5.1.2 Pengujian Use Case Melakukan Preprocessing Data

Pengujian untuk *use case* melakukan preprocessing data ditunjukkan pada tabel IV-17

Tabel IV-16 Pengujian Use Case Melakukan Preprocessing Data

Identifikasi	Deskripsi	Jenis Pengujian	Prosedur Pengujian	Keluaran yang Diharapkan	Hasil yang Dikeluarkan	Tingkat Pengujian	Kesimpulan
--------------	-----------	-----------------	--------------------	--------------------------	------------------------	-------------------	------------

UC-02-1	Memasukkan kata kunci	<i>Black Box Testing</i>	Memasukkan kata kunci ke dalam kotak pencarian	Keyword masuk di kotak pencarian	Keyword masuk di kotak pencarian	Pengujian Antarmuka	Terpenuhi
UC-02-2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard & Mengklik menu “Term” pada navbar	<i>Black Box Testing</i>	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard setelah keyword dimasukkan lalu mengklik menu "Term" pada navbar	Menampilkan hasil berupa teks hasil proses processing	Menampilkan hasil berupa teks hasil proses processing	Pengujian Unit	Terpenuhi

4.5.1.3 Pengujian Use Case Melakukan Pembobotan Data

Pengujian untuk *use case* melakukan pembobotan data ditunjukkan pada tabel IV-18

Tabel IV-17 Pengujian Use Case Melakukan Pembobotan Data

Identifikasi	Deskripsi	Jenis Pengujian	Prosedur Pengujian	Keluaran yang Diharapkan	Hasil yang Dikeluarkan	Tingkat Pengujian	Kesimpulan
UC-03-1	Memasukkan kata kunci	<i>Black Box Testing</i>	Memasukkan kata kunci ke dalam kotak pencarian	Keyword masuk di kotak pencarian	Keyword masuk di kotak pencarian	Pengujian Antarmuka	Terpenuhi
UC-03-2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard & Mengklik menu “Tf-Idf” pada navbar	<i>Black Box Testing</i>	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard setelah keyword dimasukkan lalu mengklik menu "Tf-Idf" pada navbar	Menampilkan hasil berupa nilai bobot per kata dari hasil proses pembobotan kata	Menampilkan hasil berupa nilai bobot per kata dari hasil proses pembobotan kata	Pengujian Unit	Terpenuhi

4.5.1.4 Pengujian Use Case Melakukan Perhitungan Similaritas

Pengujian untuk *use case* melakukan perhitungan similaritas ditunjukkan pada tabel IV-19

Tabel IV-18 Pengujian Use Case Melakukan Perhitungan Similaritas

Identifikasi	Deskripsi	Jenis Pengujian	Prosedur Pengujian	Keluaran yang Diharapkan	Hasil yang Dikeluarkan	Tingkat Pengujian	Kesimpulan
UC-04-1	Memasukkan kata kunci	<i>Black Box Testing</i>	Memasukkan kata kunci ke dalam kotak pencarian	Keyword masuk di kotak pencarian	Keyword masuk di kotak pencarian	Pengujian Antarmuka	Terpenuhi
UC-04-2	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard	<i>Black Box Testing</i>	Mengklik tombol “Submit” atau menekan “Enter” pada keyboard setelah keyword dimasukkan	Menampilkan hasil berupa nilai bobot similaritas antar data dari mulai yang tertinggi sampai yang terendah dari hasil proses perhitungan similaritas	Menampilkan hasil berupa nilai bobot per kata dari hasil proses pembobotan kata	Pengujian Unit	Terpenuhi

4.6 Kesimpulan

Pada bab IV telah dijelaskan mengenai pengembangan perangkat lunak yang terdiri dari 4 fase yakni fase inepsi (kebutuhan fungsional dan kebutuhan non

fungsional), fase elaborasi (use case diagram, activity diagram, sequence diagram, data model, dan perancangan antar muka), fase konstruksi (implementasi antar muka), dan fase transisi (pengujian perangkat lunak). Pembahasan mengenai hasil dan analisis penelitian akan dijelaskan pada bab berikutnya yakni bab V.

BAB V

HASIL DAN ANALISIS PENELITIAN

5.1 Pendahuluan

Pada bab IV telah dilakukan pengembangan perangkat lunak yang digunakan sebagai alat penelitian dalam melakukan penelitian pencarian data tugas akhir dengan menggunakan metode CBR. Pada bab V akan menjelaskan proses dan tahapan dalam pengujian penelitian serta hasil dan analisis penelitian.

5.2 Hasil Penelitian

Berdasarkan alur pengujian penelitian yang telah dibahas, sesuai dengan format data penelitian didapatkan hasil penelitian seperti pada tabel V-1. Pada tabel V-1 dilakukan proses pengujian memasukkan kata kunci sebanyak 50 kali dengan kata kunci yang berbeda yang mana kata tersebut merupakan metode-metode yang sering digunakan pada penelitian tugas akhir mahasiswa teknik informatika. Setiap kata kunci yang dimasukkan menghasilkan data dengan nilai similaritas tertinggi. Nilai similaritas tertinggi ada pada data dengan keyword "*Levenshtein Distance*" bernilai 0.628961. Sedangkan nilai similaritas terendah ada pada data dengan keyword "*Lexical Database*" bernilai 0.171224.

Tabel V-1 Data Hasil Pengujian Penelitian

No	Keywords	Similarity
1	Naive Bayes	0.467975
2	Fuzzy	0.28518
3	Particle Swarm Optimization	0.388253
4	Levenshtein Distance	0.628961
5	Lexcical Database	0.171224
6	Backpropagation	0.299574
7	Algoritma Harmony Search	0.242955
8	Euclidean Distance	0.388552
9	K-Nearest Neighbor	0.604087
10	Algoritma Double Metaphone	0.60547
11	Learning Vector Quantization	0.386055
12	Maximum Marginal Relevance	0.269295
13	Dempster-Shafer	0.549625
14	Reccurent Neural Network	0.333681
15	Algoritma Artificial Bee Colony	0.611358
16	Algoritma Greedy	0.437703
17	Strukture Feature Extraction	0.422364
18	Metode Tabu Search	0.281089
19	Self-Organizing Maps	0.515038
20	Improved Swarm Optimization	0.225083
21	Learning Vector Quantization	0.386055
22	Least Significant Bit	0.34948
23	Algoritma Particle Swarm Optimization	0.393675
24	Ant Colony Optimization	0.329155
25	Latent Semantic Analysis	0.269734
26	Template Matching	0.235346
27	Local Binary Pattern	0.349794

28	Case Based Reasoning	0.329593
29	Algoritma N-Gram	0.588613
30	Algoritma Dijkstra	0.387778
31	Algoritma Classification And Regression Tree	0.305415
32	Randomized Hough Transform	0.262102
33	Support Vector Machine	0.433195
34	Simple Multi Attribute Rating Technique	0.309337
35	Algoritma Winnowing	0.226592
36	Simple Additive Weighting	0.60795
37	Random Forest	0.285221
38	Top Down Parsing	0.464045
39	Hybrid Clarke Wright Saving Algorithm	0.440469
40	Algoritma Local Binary Pattern	0.355118
41	Biomimetic Pattern Recognition	0.25648
42	Convolutional Neural Network	0.600735
43	Advanced Encryption Standard	0.192943
44	Structured Support Vector Machine	0.40999
45	Principal Component Analysis	0.311801
46	Interpolasi Bicubic	0.546486
47	Algoritma Boyer-Moore	0.52769
48	Algoritma Jaro-Wrinkler Distance	0.537628
49	Geometric Moment Invariants	0.201234
50	Chinese Remainder Theorem	0.243803

Hasil keluaran yang ditampilkan oleh sistem jika kata kunci yang memiliki nilai similaritas tertinggi berdasarkan data tersebut yakni kata “*Levenshtein Distance*” ditunjukkan pada gambar V-1 – V-3.

Yuwana Samega Tamu - Vektor - TF-IDF - Word2Vec

Pencarian Data Tugas Akhir

Keyword

Levenshtein Distance Submit

No	Judul Tugas Akhir	Abstrak	Resorc Area	Nim	Nama	Kelas	Angkatan	Similarity
1	Perbandingan Algoritma Jaro-Minkow Distance dan Levenshtein Distance Dalam Mendeteksi Kemiripan Dokumen Bahasa Indonesia	Deteksi kemiripan dokumen digunakan untuk mengetahui kemiripan antara dua buah dokumen yang biasanya diukur berdasarkan kesamaan semantik atau kesamaan leksikal. Pada penelitian ini dilakukan deteksi kemiripan secara leksikal dengan menggunakan algoritma Levenshtein Distance.	Natural Language Processing	2147483647	Riva Syahbana	Reguler	2013	0.629811
2	Penggunaan Algoritma Damerau-Levenshtein Distance Untuk Mengoreksi Kesalahan Ejaan	Kesalahan penulisan atau ejaan	Natural Language Processing	2147483647	Fad Wafid	Bilingual	2013	0.621335

Gambar V-1 Data Hasil Rekomendasi “Levenshtein Distance” 1

2	Penggunaan Algoritma Damerau-Levenshtein Distance Untuk Mengoreksi Ejaan Kata Bertabasa Indonesia	Salah satu permasalahan yang dihadapi algoritma Damerau-Levenshtein distance dan algoritma levenshtein distance. Hasil penelitian ini menghasilkan nilai optimisasi sebesar 22,04 hal ini menunjukkan optimasi yang dilakukan oleh algoritma Damerau-Levenshtein distance terhadap algoritma Levenshtein distance.	Natural Language Processing	2147483647	Fad Wafid	Bilingual	2013	0.621335
3	Perbandingan Algoritma Jaro-Winkler Distance dan Levenshtein Distance pada Sistem Pengoreksi Ejaan Kata Bertabasa Indonesia	Ejaan merupakan salah satu bagian terpenting dalam sebuah teks tulisan. Kirap dijumpai kesalahan ejaan pada kalimat yang memengaruhi makna yang terkandung dan mengakibatkan pembaca kesulitan dalam mengartikannya. Pengoreksian kata yang dilakukan secara	Natural Language Processing	2147483647	Achsa Daviana	Reguler	2015	0.396451
4	Analisa Perbandingan Algoritma Damerau-Levenshtein Distance dan Soundex Similarity pada Pengoreksian Ejaan Kata Chromatis	Damerau-Levenshtein Distance dan Soundex Similarity merupakan dua metode yang dapat digunakan dalam mengoreksi kesalahan ejaan kata secara otomatis. Perbedaan kedua algoritma terletak pada dasar pengoreksian dimana algoritma Damerau-Levenshtein Distance mengoreksi kesalahan	Natural Language Processing	2147483647	Abson Hadi	Bilingual	2012	0.381664
5	Algoritma Spellchecking Terasy Search Tree dan Levenshtein Distance dalam Pengoreksian Kata Bertabasa	Kesalahan dalam ejaan atau	Natural Language Processing	2147483647	Vekry Medina Rizky	Bilingual	2013	0.2887

Gambar V-2 Data Hasil Rekomendasi “Levenshtein Distance” 2

algoritma Levenshtein							
3	Algoritma Spellchecking Ternary Search Tree dan Levenshtein Distance dalam Pengoreksian Kata Bertahap Inggris	Kesalahan dalam ejaan atau penulisan kata dalam sebuah perangkat lunak pengisian teks adalah hal yang seringkali terjadi, biasanya disebabkan oleh kelelahan pengguna. Maka dari itu diperlukan sebuah perangkat lunak yang dapat	Natural Language Processing	2147483547	Valery Medzina Riky	Sibregal 2013	0,2857
6	PENGARUH PERHITUNGAN RENTANG DATA TERHADAP PERFORMANSI ALGORITMA CLUSTERING SELF ORGANIZING MAP	Clustering merupakan bidang penelitian yang sangat penting dalam data mining. Clustering mengelompokkan data berdasarkan kedekatannya atau kemiripannya. Komponen yang sangat penting dalam clustering adalah pengukuran rentang antar data. Berbeda metode	Data Mining	2147483547	Tri Kurnia Sari	Reguler 2015	0,203478
7	Pertandingan Metode Euclidean Distance dengan Coefficient Correlation pada Klasifikasi Penyakit Multiple Sclerosis Lesion Brain (MSLB)	"Pendeteksian penyakit Multiple Sclerosis Lesion Brain (MSLB) pada segmentasi citra otak masih belum begitu banyak diteliti. Flower Pollination Algorithm digunakan untuk mengoptimasi hasil segmentasi sehingga nilainya dapat digunakan untuk proses klasifikasi penyakit. Data yang digunakan	Data Mining	2147483547	Sarah Aprilia	Sibregal 2012	0,145469

Gambar V-3 Data Hasil Rekomendasi "Levenshtein Distance" 3

Apabila kata kunci yang dimasukkan dibalik atau ditukar urutannya menjadi "Distance Levenshtein", maka data hasil rekomendasi yang ditampilkan akan tetap sama.

Dan jika hasil keluaran yang ditampilkan oleh sistem dibandingkan dengan aplikasi SIPETA yang mana aplikasi tersebut belum menerapkan metode CBR dalam proses pencariannya ditunjukkan pada gambar V-4 dan gambar V-5

SIPETA

Silahkan input kata kunci pencarian.

Kata Kunci

Carilah berdasarkan judul, abstrak, NIM, nama, pembimbing...

Car

- Angkatan - - Kelas - - Status - - Bidang Riset -

Hasil pencarian : 9
keyword : %Levenshtein Distance%

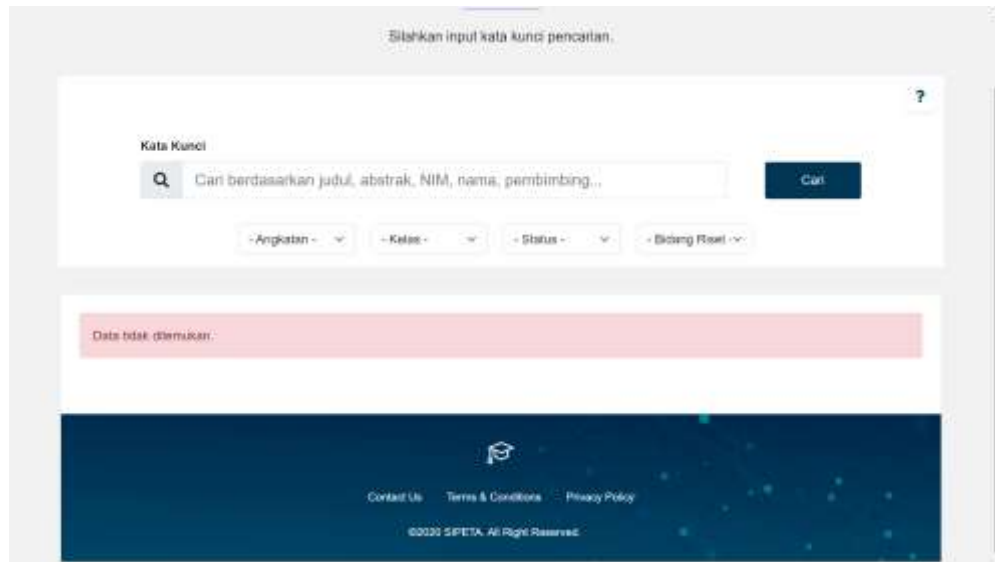
No	Judul Tugas Akhir	NIM - Nama	Kelas - Angkatan	Dosen Pembimbing 1	Dosen Pembimbing 2
1	Algoritma Spellchecking Ternary Search Tree dan Levenshtein Distance dalam Pengoreksian Kata Berbahasa Inggris	0981320022 - Vallery Medista Rizky	Bilingual - 2013	Rifka Primartha, M.T.	Yopyy Sazaki, M.T.

Gambar V-4 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 1

No	Judul Tugas Akhir	NIM - Nama	Kelas - Angkatan	Dosen Pembimbing 1	Dosen Pembimbing 2
1	Algoritma Spellchecking Ternary Search Tree dan Levenshtein Distance dalam Pengoreksian Kata Berbahasa Inggris	0981320022 - Vallery Medista Rizky	Bilingual - 2013	Rifka Primartha, M.T.	Yopyy Sazaki, M.T.
2	Analisa Perbandingan Algoritma Cosine-Levenshtein Distance dan Soudex Similarity pada Pengoreksian Ejaan Kata Otomatis	09121002015 - Aboon Hadi	Bilingual - 2012	M. Fachrudzi, M.T.	Novi Yuliani, M.T.
3	Penerapan Algoritma Rabin-Karp dan Levenshtein Distance pada Deteksi Kemiripan Dokumen Berbahasa Indonesia	09121002053 - Dana D Kamindo	Reguler - 2012	Novi Yuliani, M.T.	
4	Penggunaan Algoritma Cosine-Levenshtein Distance Untuk Mengoreksi Ejaan Kata Berbahasa Indonesia	0981320008 - Fuzul Wahid	Bilingual - 2013	Rifka Primartha, M.T.	Kendra Januar Mrazawati, M.T.
5	Pengoreksian Ejaan Kata Berbahasa Indonesia Menggunakan Algoritma Levenshtein Distance	0981320050 - Muhammad Omar Bradley	Bilingual - 2013	M. Fachrudzi, M.T.	Novi Yuliani, M.T.
6	Perbandingan Algoritma Jaro-Winkler Distance dan Levenshtein Distance Dalam Mendeteksi Kemiripan Dokumen Bahasa Indonesia	09021181320023 - Niwa Syakhanisa	Reguler - 2013	Novi Yuliani, M.T.	Oscar Aruban, M.T.
7	Perbandingan Algoritma Jaro-Winkler Distance dan Levenshtein Distance pada Sistem Pengoreksi Ejaan Kata Berbahasa Indonesia	09021181520009 - Achma Deswaria	Reguler - 2015	Novi Yuliani, M.T.	Desty Rodiah, M.T.
8	Perbandingan Algoritma Levenshtein Distance dan Babai Karp dalam Pendeteksian Kemiripan Teks	0981320002 - Muhammad Rizki Aman	Bilingual - 2013	Samsuryadi, M.Kom., Ph.D	Novi Yuliani, M.T.
9	Spelling Checker Menggunakan Algoritma Cosine Levenshtein Distance dan Cosine Similarity	09021181520125 - Nur Hamidah	Reguler - 2015	Novi Yuliani, M.T.	

Gambar V-5 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 1

Kemudian jika kata kunci pada aplikasi SIPETA dibalik atau ditukar urutannya menjadi “Distance Levenshtein”, maka keluaran yang diberikan oleh aplikasi ditunjukkan pada gambar V-6



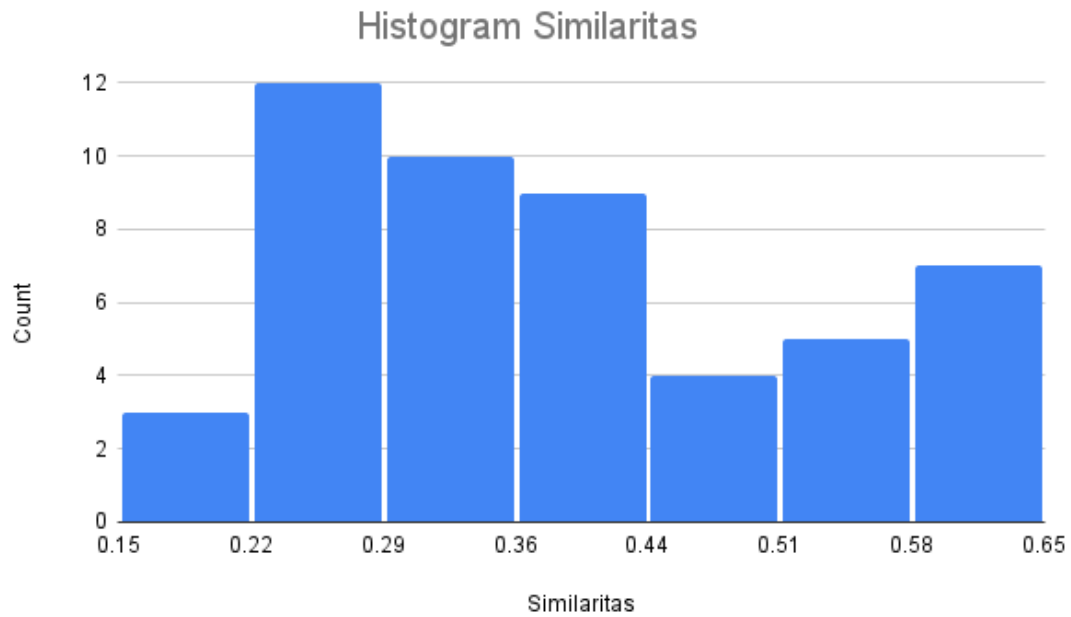
Gambar V-6 Data Hasil Rekomendasi “Levenshtein Distance” SIPETA 3

5.3 Analisis Hasil Penelitian

Berdasarkan tabel V-1 diperoleh 50 nilai similaritas sesuai dengan kata kunci yang diuji. Nilai similaritas tertinggi yang dapat dicapai oleh sistem adalah bernilai 0.628961 dari pencarian berdasarkan kata kunci “Lavenshtein Distance”. Dan nilai terendah yang dapat dicapai oleh sistem bernilai 0.171224 dengan kata kunci pencarian “Lexcical Database”. Data dengan pencapaian nilai similaritas tertinggi dapat dicapai dikarenakan data hasil rekomendasi berdasarkan kata kunci memiliki jumlah yang lebih dari satu data. Sebaliknya, Data dengan pencapaian nilai similaritas terendah

dapat dicapai karena data hasil rekomendasi berdasarkan kata kunci memiliki jumlah yang sedikit atau hanya satu data. Hal tersebut membuktikan banyaknya data baik yang ada dalam database maupun hasil rekomendasi saat melakukan pencarian mempengaruhi pencapaian nilai similaritas yang dihasilkan. Semakin banyak data hasil rekomendasi maka nilai similaritas yang dapat dicapai akan semakin tinggi, sebaliknya jika data hasil rekomendasi berjumlah sedikit atau satu data maka nilai similaritas yang dicapai tidak melebihi nilai dari data hasil rekomendasi yang berjumlah lebih dari satu.

Data dengan rentang nilai 0.15 sampai 0.22 memiliki jumlah data sebanyak 3, Data dengan rentang nilai 0.22 sampai 0.29 memiliki jumlah data sebanyak 12, Data dengan rentang nilai 0.29 sampai 0.36 memiliki jumlah data sebanyak 10, Data dengan rentang nilai 0.36 sampai 0.44 memiliki jumlah data sebanyak 9, Data dengan rentang nilai 0.44 sampai 0.51 memiliki jumlah data sebanyak 4, Data dengan rentang nilai 0.51 sampai 0.58 memiliki jumlah data sebanyak 5, Data dengan rentang nilai 0.58 sampai 0.65 memiliki jumlah data sebanyak 7. Data dengan rentang nilai 0.22 sampai 0.29 adalah data dengan jumlah terbanyak.



Gambar V-7 Histogram Similaritas

Berdasarkan perolehan data hasil penelitian tersebut juga memberikan hasil berupa nilai similaritas yang dapat dicapai oleh sistem memiliki rata-rata sebagai berikut:

$$\text{Rata-rata} = \frac{\sum_{i=1}^n \text{Similaritas}_i}{n}$$

$$\text{Rata-rata} = \frac{0.457452 + 0.285729 + \dots + 0.243705}{50} = 0.38501958$$

Berdasarkan hasil penelitian pada subab 5.2 hasil data rekomendasi yang dihasilkan melalui kedua aplikasi dimana aplikasi yang satu telah menerapkan metode CBR dan aplikasi yang lain belum menerapkan metode CBR memiliki perbedaan.

Perbedaan yang pertama yakni terletak pada jumlah data yang ditampilkan sistem, jika mengesampingkan atau mengeliminasi data dari hasil rekomendasi sistem pada aplikasi SIPETA yang tidak memiliki abstrak maka jumlah data rekomendasi yang dihasilkan pada sistem yang menerapkan CBR adalah sebanyak 7 Data dan untuk aplikasi SIPETA jumlah data yang dihasilkan berjumlah 5 Data. Keseluruhan data yang dihasilkan pada aplikasi SIPETA dimiliki oleh sistem yang menerapkan CBR. Dua data yang dimiliki pada sistem CBR tidak ditampilkan pada aplikasi SIPETA dikarenakan data tersebut hanya mengandung sebagian dari kata kunci yang dimasukkan yang menunjukkan bahwa data tersebut masih relevan terhadap kata kunci. Dari hal tersebut sistem yang menerapkan CBR mampu menampilkan data dengan kata kunci walaupun kata kunci tersebut hanya sebagian yang mana hal tersebut tidak berlaku pada aplikasi SIPETA. Perbedaan kedua yakni terletak pada urutan data yang ditampilkan dimana data pada sistem CBR ditampilkan berdasarkan urutan nilai similaritas dari yang tertinggi hingga yang terendah sehingga membuat data tersebut memiliki prioritas untuk dapat dilihat oleh pengguna pertama kali dibanding data yang lain, sedangkan pada aplikasi SIPETA data yang ditampilkan tidak diurutkan berdasarkan nilai similaritas melainkan diurutkan berdasarkan abjad dari judul tugas akhir yang mana hal tersebut menunjukkan aplikasi tidak memiliki data prioritas untuk dapat dilihat oleh pengguna pertama kali. Dan perbedaan terakhir ada pada saat kata kunci yang dimasukkan dibalik atau ditukar urutannya sehingga membuat aplikasi SIPETA tidak dapat menampilkan data apapun dari hasil pencarian, namun tidak demikian dengan

sistem CBR dimana data yang dihasilkan tetap sama dari kata kunci yang dimasukkan sebelumnya. Hal tersebut membuktikan bahwa sistem yang menerapkan CBR mampu menampilkan hasil rekomendasi data walaupun kata kunci yang dimasukkan tidak secara terurut.

5.4 Kesimpulan

Dari keseluruhan pembahasan pada bab V, Telah dibahas mengenai hasil penelitian dan juga analisis hasil penelitian dimana diperoleh nilai rata-rata pada data hasil pengujian penelitian. Kesimpulan dan juga saran penelitian yang telah dilakukan akan dibahas pada bab berikutnya.

BAB VI

KESIMPULAN DAN SARAN

6.1 Pendahuluan

Pada bab VI akan membahas mengenai kesimpulan dan saran dari penelitian pencarian data tugas akhir menggunakan metode case base reasoning yang telah peneliti lakukan yang diharapkan mampu memberikan edukasi bagi peneliti selanjutnya apabila ingin meneliti dengan penelitian yang mirip seperti penelitian tersebut.

6.2 Kesimpulan

Bedasarkan paparan analisis hasil penelitian yang telah dipaparkan pada subab 5.3 maka penelitian pencarian data tugas akhir menggunakan metode CBR memiliki kesimpulan berdasarkan poin-poin berikut yang mana poin-poin tersebut adalah untuk menjawab rumusan masalah yang ada pada penelitian:

1. Proses implementasi metode CBR dapat diterapkan terhadap pencarian tugas akhir mahasiswa dimana proses tersebut diawali dengan pembuatan sistem pencarian berbasis website yang dijelaskan pada bab IV lalu dipadukan dengan algoritma perhitungan similaritas menggunakan cosine similarity yang

disimulasikan pada subsubsubbab 4.2.1.1 untuk menghasilkan nilai similaritas pada data yang relevan terhadap kata kunci.

2. Evaluasi metode CBR dengan cosine similarity terhadap pencarian tugas akhir menghasilkan nilai pencapaian tertinggi oleh sistem adalah 0.628961 dan terendah adalah 0.171224 dimana hasil perolehan nilai tersebut dipengaruhi oleh jumlah data hasil rekomendasi dan untuk perbandingan terhadap aplikasi SIPETA sistem yang menerapkan CBR mampu menghasilkan rekomendasi data tugas akhir yang mengandung kata kunci hanya sebagian atau salah satunya serta mampu memberikan data hasil rekomendasi walaupun kata kunci dibalik atau diubah urutannya secara terurut berdasarkan nilai similaritas dengan mengutamakan data yang paling relevan terhadap kata kunci.

6.3 Saran

Sistem yang telah dibuat belum dapat dikatakan sempurna masih terdapat hal yang perlu diterapkan kedepannya untuk menyempurnakan sistem tersebut. Dalam proses pembuatan sistem peneliti menemui kendala dimana proses tersebut membutuhkan waktu yang cukup lama karena peneliti hanya menggunakan bahasa php murni. Maka, saran dari peneliti sebagai dasar referensi untuk penelitian selanjutnya yang lebih baik agar dapat mempersingkat waktu proses pembuatan sistem peneliti menyarankan untuk dapat memanfaatkan atau menerapkan *library* atau *framework* dalam proses pembuatan sistem tersebut.

DAFTAR PUSTAKA

- Andrian, R., Sakethi, D., & Chairuddin, M. (2016). Pengembangan Sistem Informasi Penelitian dan Pengabdian Dosen Jurusan Ilmu Komputer Menggunakan Metode Rational Unified Process (RUP). *Jurnal Komputasi*, 2(2).
- Dewa, Arie, dan Agustinus. 2016. Pengukuran Kemiripan Dokumen Teks Bahasa Indonesia Menggunakan Metode Cosine Similarity. *Jurnal. Teknik Informatika Universitas Sam Ratulangi Manado*.
- E. Faizal, “Case Based Reasoning Diagnosis Penyakit Cardiovascular Dengan Metode Simple Matching Coefficient Similarity,” *J. Teknol. Inf. dan Ilmu Komput.*, 2014, doi: 10.25126/jtiik.201412116.
- Feldman, Ronen , Sanger, dkk. (2007). *The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, New York.
- Feldman, Ronen, dan Sanger, James. 2007. *The Text Mining Handbook Advanced Approaches in Analyzing Unstructured Data*. New York: Cambridge University Press.
- Ferdinandus, Subari. 2015. Sistem Information Retrieval Layanan Kesehatan Untuk Berobat dengan Metode Vector Space Model berbasis WebGis. *Jurnal. Teknik Informatika. STIKI Malang*.
- Hidayat, M. S., Puspaningrum, E. Y., & Swari, M. H. P. (2020). PENERAPAN CASE BASED REASONING PADA SISTEM REKOMENDASI TOPIK SKRIPSI PROGRAM STUDI

- INFORMATIKA UPN "VETERAN" JAWA TIMUR. *Jurnal Informatika dan Sistem Informasi (JIFoSI)*, 1(2), 272-281.
- Ingwersen, P. (1992). *Information retrieval interaction* (Vol. 246). London: Taylor Graham.
- Isinkaye, F.O. Folajimi, Y.O. & Ojokoh, B.A. 2015. Recommendation Systems: Principles, Methods and Evaluation. *Egyptian Informatics Journal*. vol. 16. no. 3. pp 261-273.
- L. Francis dan M. Flynn, —Text Mining Handbook, I Casualty Actuarial Society E Forum. 2010.
- Ma'arif, Abdul Aziz. 2015. Penerapan Algoritma Tf-Idf Untuk Pencarian Karya Ilmiah. *Jurnal Jurusan Teknik Informatika. Fakultas Ilmu Komputer. Universitas Dian Nuswantoro Semarang*.
- Mathew, A. D., Ma, L., & Narasimhan, V. L. 2006. Case-based reasoning for data warehouse schema design.
- Melita, R., Amrizal, V., Suseno, H. B., & Dirjam, T. (2018). Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Syarah Umdatil Ahkam). *Jurnal Teknik Informatika*, 11(2), 149-164.
- Merawati, N. L. P., & Hartati, S. (2018). Sistem Rekomendasi Topik Skripsi Menggunakan Metode Case Based Reasoning. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 4(3).

- Nurdiana, O., Jumadi, J., & Nursantika, D. (2016). Perbandingan metode Cosine Similarity dengan metode Jaccard Similarity pada aplikasi pencarian terjemah Al-Qur'an dalam Bahasa Indonesia. *Jurnal Online Informatika*, 1(1), 59-63.
- Pal, S.K. & Shiu, S.C.K. 2004. *Foundations of Soft Case-Based Reasoning*. John Willey and Sons. New Jersey.
- Putra, Agung Auliaguntary Arif. 2016. *Implementasi Text Summarization Menggunakan Metode Vector Space Model pada Artikel Berita Bahasa Indonesia*. Skripsi. Jurusan Teknik Informatika. Fakultas Teknik dan Ilmu Komputer. Universitas Komputer Indonesia.
- Richter, M.M & Rosina, O.W. 2013. *Case Based Reasoning: A Textbook*. Springer-Verlag Berlin Heidelberg.
- Riyani, A., Naf'an, M. Z., & Burhanuddin, A. (2019). Penerapan Cosine Similarity dan Pembobotan TF-IDF untuk Mendeteksi Kemiripan Dokumen. *Jurnal Linguistik Komputasional (JLK)*, 2(1), 23-27.
- Rossa, A S dan M. Shalahuddin. 2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- S. Mulyana; and S. Hartati, "Tinjauan Singkat Perkembangan Case- Based Reasoning," in *Seminar Nasional Informatika 2009 (Semnasif 2009) Upn "Veteran" Yogyakarta*, 2009.

- Salmin, M., & Hartati, S. (2019). Case Based Reasoning untuk Diagnosis Penyakit Infeksi Saluran Pernapasan Akut. *JIKO (Jurnal Informatika dan Komputer)*, 1(1), 21-26.
- Sappagh, S.H. & Elmogy, M. 2015. Case Based Reasoning: Case Representation Methodologies. *International Journal of Advanced Computer Science and Applications (IJACSA)*. Vol. 6. No. 11. pp 192-208.
- Sunyoto, A. (2013, June). Implementasi Algoritma Rabin Karp untuk Pendeteksian Plagiat Dokumen Teks Menggunakan Konsep Similarity. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)* (Vol. 1, No. 1).
- Utami, A., Suyanto, Y., & Sihabuddin, A. Recommendations on Selecting The Topic of Student Thesis Concentration using Case Based Reasoning. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 15(1), 1-10.
- Whitten, J.L. & Lonnie, D.B. (2007). *Systems Analysis and Design Methods*. Seventh Edition. McGraw-Hill.

LAMPIRAN

Lampiran 1. Analisis proses perhitungan similaritas

Document	Keywords	research field
Query	kelompok:1; clustering:1; balita:1; gizi:1; metode:1; fuzzy:1; klasifikasi:1; c-means:1; algoritma:1; perangkat:1 lunak:1;	Soft Computing
Document 1	kelompok:4; status:6; gizi:7; balita:5; metode:4; fuzzy:5; c-means:4; algoritma:4; genetika:4; deteksi:1; teliti:2; kembang:1; hasil:2; perangkat:1; lunak:1; bangkit:1; bilang:1; random:1; hitung:4; pusat:1; cluster:4; fungsi:2; obyektif:1; fitness:1; seleksi:1; crossover:3; mutasi:3; parameter:1; badan:2; berat:1; umur:1; uji:2; ubah:2; matriks:1; partisi:1; inisial:1; populasi:3; iterasi:2; peluang:4; ukur:2; validasi:1; davies:1; boulding:1; index:1; dapat:2; nilai:3;	Soft Computing
Document 2	banding:3; kelompok:7; status:6; gizi:6; hamil:7; fuzzy:10; c-means:3; subtractive:6; clustering:6; guna:1; mudah:1; teliti:3; metode:5; variabel:1; umur:1; berat:2; badan:3; lingkak:1; lengan:1; usia:1; tambah:1; hasil:3; validasi:5; davies:1; bouldin:1; index:2; dbi:2; sum:1; xbi:2; sesuai:2; variasi:1; data:5; rendah:3; dapat:3; sehingga:1 pilih:1; sebagai:1;	Soft Computing

Document 3	aman:4; integritas:6; alamat:9; bitcoin:12; karyawan:6; algoritma:5; md:5; fokus:1; teliti:3; otentikasi:4; message:1; digest:1; kembang:2; aplikasi:2; gaji:3; lengkap:3; application:1; programming:1; interface:1; sedia:1; indodax:1; data:5; pribadi:1; salah:1; satu:1; isi:1; simpan:4; basis:4; bangkit:2; nilai:4; hash:4; pisah:2; proses:1; banding:1; dasar:1; uji:1; deteksi:1; ms:2; rata:1;	Kriptografi
Document 4	klasifikasi:2; sakit:3; kanker:3; payudara:3; metode:5; backpropagation:3; adaptive:3; learning:3; rate:3; iring:1; maju:1; teknologi:1; deteksi:1; ahli:1; medis:1; salah:1; proses:3; ajar:1; lambat:1; optimasi:1; tahan:1; update:1; bobot:2; teliti:2; data:5; total:2; bagi:1; latih:1; persentase:1; dasar:1; training:2; nilai:3; iterasi:2; beda:1; hasil:2; rata:1; akurasi:2; baik:1; oleh:1;	Data Mining
Document 5	implementasi:1; metode:6; analitic:1; network:1; process:1; anp:5; technique:1; preference:1; similarity:1; ideal:1; solution:1; topsis:4; sistem:4; dukung:3; putus:3; distribusi:3; beras:3; masyarakat:1; miskin:3; keluarga:2; raskin:3; salah:1; program:1; perintah:1; upaya:1; tingkat:1; tahan:1; pangan:1; lindung:1; proses:1; alami:1; hambat:1; pilih:1; terima:2; bantu:3; tentu:1;urut:1; sesuai:1; kriteria:3; butuh:1; milik:1; jenis:1; rangking:4; tulis:1; gabung:1; awal:1; hitung:1; bobot:1; uji:2; teliti:1; banding:1; hasil:3; manual:1; staff:1; camat:1; data:1; persentase:1; cocok:1;	Soft Computing

Term	TF					DF	IDF
	Q	D1	D2	D3	D4		

adaptive					3		1	0.77815125
ahli					1		1	0.77815125
ajar					1		1	0.77815125
akurasi					2		1	0.77815125
alamat				9			1	0.77815125
alami						1	1	0.77815125
algoritma	1	4		5			3	0.301029996
aman				4			1	0.77815125
anality						1	1	0.77815125
anp						5	1	0.77815125
aplikasi				2			1	0.77815125
application				1			1	0.77815125
awal						1	1	0.77815125
backpropagation					3		1	0.77815125
badan		2	3				2	0.477121255
bagi					1		1	0.77815125
baik					1		1	0.77815125
balita	1	5					2	0.477121255
banding			3	1		1	3	0.301029996
bangkit		1		2			2	0.477121255
bantu						3	1	0.77815125
basis				4			1	0.77815125
beda					1		1	0.77815125
beras						3	1	0.77815125
berat		1	2				2	0.477121255
bilang		1					1	0.77815125
bitcoin				12			1	0.77815125
bobot					2	1	2	0.477121255
boulding		1	1				2	0.477121255
butuh						1	1	0.77815125
camat						1	1	0.77815125
clustering	1	4	6				3	0.301029996
c-means	1	4	3				3	0.301029996
cocok						1	1	0.77815125
crossover		3					1	0.77815125
dapat		2	3				2	0.477121255
dasar				1	1		2	0.477121255
data			5	5		1	3	0.301029996
davies		1	1				2	0.477121255

dbi			2				1	0.77815125
deteksi		1		1	1		3	0.301029996
digest				1			1	0.77815125
distribusi						3	1	0.77815125
dukung						3	1	0.77815125
fitness		1					1	0.77815125
fokus				1			1	0.77815125
fungsi		2					1	0.77815125
fuzzy	1	5	10				3	0.301029996
gabung						1	1	0.77815125
gaji				3			1	0.77815125
genetika		4					1	0.77815125
gizi	1	7	6				3	0.301029996
guna				1			1	0.77815125
hambat						1	1	0.77815125
hamil			7				1	0.77815125
hash				4			1	0.77815125
hasil		2	3		2	3	4	0.176091259
hitung		4				1	2	0.477121255
ideal						1	1	0.77815125
implementasi						1	1	0.77815125
index		1	2				2	0.477121255
indodax				1			1	0.77815125
inisial		1					1	0.77815125
integritas				6			1	0.77815125
interface				1			1	0.77815125
iring					1		1	0.77815125
isi				1			1	0.77815125
iterasi		2			2		2	0.477121255
jenis						1	1	0.77815125
kanker					3		1	0.77815125
karyawan				6			1	0.77815125
kelompok	1	4	7				3	0.301029996
keluarga						2	1	0.77815125
kembang		1		2			2	0.477121255
klasifikasi	1			2			2	0.477121255
kriteria						3	1	0.77815125
lambat					1		1	0.77815125
latih					1		1	0.77815125

learning					3		1	0.77815125
lengan			1				1	0.77815125
lengkap				3			1	0.77815125
lindung						1	1	0.77815125
lingkar			1				1	0.77815125
lunak	1	1					2	0.477121255
maju					1		1	0.77815125
manual						1	1	0.77815125
masyarakat						1	1	0.77815125
matriks		1					1	0.77815125
md				5			1	0.77815125
medis					1		1	0.77815125
message				1			1	0.77815125
metode	1	4	5	5		6	5	0.079181246
milik						1	1	0.77815125
misikin						3	1	0.77815125
ms				2			1	0.77815125
mudah				1			1	0.77815125
mutasi		3					1	0.77815125
network						1	1	0.77815125
nilai		3			3		2	0.477121255
obyektif		1					1	0.77815125
oleh					1		1	0.77815125
optimasi					1		1	0.77815125
otentikasi				4			1	0.77815125
pangan						1	1	0.77815125
parameter		1					1	0.77815125
partisi		1					1	0.77815125
payudara					3		1	0.77815125
peluang		4					1	0.77815125
perangkat	1	1					2	0.477121255
perintah						1	1	0.77815125
persentase					1	1	2	0.477121255
pilih			1			1	2	0.477121255
pisah				2			1	0.77815125
populasi		3					1	0.77815125
preference						1	1	0.77815125
pribadi				1			1	0.77815125
process						1	1	0.77815125

program					1	1	0.77815125
programing			1			1	0.77815125
proses			1	3	1	3	0.301029996
pusat	1					1	0.77815125
putus					3	1	0.77815125
random	1					1	0.77815125
rangking					4	1	0.77815125
raskin					1	1	0.77815125
rata			1	1		2	0.477121255
rate				3		1	0.77815125
rendah		3				1	0.77815125
sakit				3		1	0.77815125
salah			1	1	1	3	0.301029996
satu			1			1	0.77815125
sebagai		1				1	0.77815125
sedia			1			1	0.77815125
sehingga		1				1	0.77815125
seleksi	1					1	0.77815125
sesuai		2			1	2	0.477121255
similarity					1	1	0.77815125
simpan			4			1	0.77815125
sistem					4	1	0.77815125
solution					1	1	0.77815125
staff					1	1	0.77815125
status	6	6				2	0.477121255
subtractive		6				1	0.77815125
sum		1				1	0.77815125
tahan				1		1	0.77815125
tahan					1	1	0.77815125
tambah		1				1	0.77815125
technique					1	1	0.77815125
teknologi				1		1	0.77815125
teliti	2	3	3		1	4	0.176091259
tentu					1	1	0.77815125
terima					2	1	0.77815125
tingkat					1	1	0.77815125
topsis					4	1	0.77815125
total				2		1	0.77815125
training				2		1	0.77815125

tulis					1	1	0.77815125
ubah		2				1	0.77815125
uji		2		1		2	0.301029996
ukur		2				1	0.77815125
umur		1	1			2	0.477121255
upaya					1	1	0.77815125
update					1	1	0.77815125
urut					1	1	0.77815125
usia			1			1	0.77815125
validasi		1	5			2	0.477121255
variable				1		1	0.77815125
variasi			1			1	0.77815125
xbi			2			1	0.77815125

Term	TF-IDF					
	Q	D1	D2	D3	D4	D5
adaptive	0	0	0	0	2.334453751	0
ahli	0	0	0	0	0.77815125	0
ajar	0	0	0	0	0.77815125	0
akurasi	0	0	0	0	1.556302501	0
alamat	0	0	0	7.003361253	0	0
alami	0	0	0	0	0	0.77815125
algoritma	0.301029996	1.204119983	0	1.505149978	0	0
aman	0	0	0	3.112605002	0	0
anality	0	0	0	0	0	0.77815125
anp	0	0	0	0	0	3.890756252
aplikasi	0	0	0	1.556302501	0	0
application	0	0	0	0.77815125	0	0
awal	0	0	0	0	0	0.77815125
backpropagation	0	0	0	0	2.334453751	0
badan	0	0.954242509	1.431363764	0	0	0
bagi	0	0	0	0	0.77815125	0
baik	0	0	0	0	0.77815125	0
balita	0.477121255	2.385606274	0	0	0	0
banding	0	0	0.903089987	0.301029996	0	0.301029996
bangkit	0	0.477121255	0	0.954242509	0	0
bantu	0	0	0	0	0	2.334453751

basis	0	0	0	3.112605002	0	0
beda	0	0	0	0	0.77815125	0
beras	0	0	0	0	0	2.334453751
berat	0	0.477121255	0.954242509	0	0	0
bilang	0	0.77815125	0	0	0	0
bitcoin	0	0	0	9.337815005	0	0
bobot	0	0	0	0	0.954242509	0.477121255
boulding	0	0.477121255	0.477121255	0	0	0
butuh	0	0	0	0	0	0.77815125
camat	0	0	0	0	0	0.77815125
clustering	0.301029996	1.204119983	1.806179974	0	0	0
c-means	0.301029996	1.204119983	0.903089987	0	0	0
cocok	0	0	0	0	0	0.77815125
crossover	0	2.334453751	0	0	0	0
dapat	0	0.954242509	1.431363764	0	0	0
dasar	0	0	0	0.477121255	0.477121255	0
data	0	0	1.505149978	1.505149978	0	0.301029996
davies	0	0.477121255	0.477121255	0	0	0
dbi	0	0	1.556302501	0	0	0
deteksi	0	0.301029996	0	0.301029996	0.301029996	0
digest	0	0	0	0.77815125	0	0
distribusi	0	0	0	0	0	2.334453751
dukung	0	0	0	0	0	2.334453751
fitness	0	0.77815125	0	0	0	0
fokus	0	0	0	0.77815125	0	0
fungsi	0	1.556302501	0	0	0	0
fuzzy	0.301029996	1.505149978	3.010299957	0	0	0
gabung	0	0	0	0	0	0.77815125
gaji	0	0	0	2.334453751	0	0
genetika	0	3.112605002	0	0	0	0
gizi	0.301029996	2.10720997	1.806179974	0	0	0
guna	0	0	0	0.77815125	0	0
hambat	0	0	0	0	0	0.77815125
hamil	0	0	5.447058753	0	0	0
hash	0	0	0	3.112605002	0	0
hasil	0	0.352182518	0.528273777	0	0.352182518	0.528273777
hitung	0	1.908485019	0	0	0	0.477121255
ideal	0	0	0	0	0	0.77815125
implementasi	0	0	0	0	0	0.77815125

index	0	0.477121255	0.954242509	0	0	0
indodax	0	0	0	0.77815125	0	0
inisial	0	0.77815125	0	0	0	0
integritas	0	0	0	4.668907502	0	0
interface	0	0	0	0.77815125	0	0
iring	0	0	0	0	0.77815125	0
isi	0	0	0	0.77815125	0	0
iterasi	0	0.954242509	0	0	0.954242509	0
jenis	0	0	0	0	0	0.77815125
kanker	0	0	0	0	2.334453751	0
karyawan	0	0	0	4.668907502	0	0
kelompok	0.301029996	1.204119983	2.10720997	0	0	0
keluarga	0	0	0	0	0	1.556302501
kembang	0	0.477121255	0	0.954242509	0	0
klasifikasi	0.477121255	0	0	0.954242509	0	0
kriteria	0	0	0	0	0	2.334453751
lambat	0	0	0	0	0.77815125	0
latih	0	0	0	0	0.77815125	0
learning	0	0	0	0	2.334453751	0
lengan	0	0	0.77815125	0	0	0
lengkap	0	0	0	2.334453751	0	0
lindung	0	0	0	0	0	0.77815125
lingkar	0	0	0.77815125	0	0	0
lunak	0.477121255	0.477121255	0	0	0	0
maju	0	0	0	0	0.77815125	0
manual	0	0	0	0	0	0.77815125
masyarakat	0	0	0	0	0	0.77815125
matriks	0	0.77815125	0	0	0	0
md	0	0	0	3.890756252	0	0
medis	0	0	0	0	0.77815125	0
message	0	0	0	0.77815125	0	0
metode	0.079181246	0.316724984	0.39590623	0.39590623	0	0.475087476
milik	0	0	0	0	0	0.77815125
misikin	0	0	0	0	0	2.334453751
ms	0	0	0	1.556302501	0	0
mudah	0	0	0	0.77815125	0	0
mutasi	0	2.334453751	0	0	0	0
network	0	0	0	0	0	0.77815125
nilai	0	1.431363764	0	0	1.431363764	0

obyektif	0	0.77815125	0	0	0	0
oleh	0	0	0	0	0.77815125	0
optimasi	0	0	0	0	0.77815125	0
otentikasi	0	0	0	3.112605002	0	0
pangan	0	0	0	0	0	0.77815125
parameter	0	0.77815125	0	0	0	0
partisi	0	0.77815125	0	0	0	0
payudara	0	0	0	0	2.334453751	0
peluang	0	3.112605002	0	0	0	0
perangkat	0.477121255	0.477121255	0	0	0	0
perintah	0	0	0	0	0	0.77815125
persentase	0	0	0	0	0.477121255	0.477121255
pilih	0	0	0.477121255	0	0	0.477121255
pisah	0	0	0	1.556302501	0	0
populasi	0	2.334453751	0	0	0	0
preference	0	0	0	0	0	0.77815125
pribadi	0	0	0	0.77815125	0	0
process	0	0	0	0	0	0.77815125
program	0	0	0	0	0	0.77815125
programing	0	0	0	0.77815125	0	0
proses	0	0	0	0.301029996	0.903089987	0.301029996
pusat	0	0.77815125	0	0	0	0
putus	0	0	0	0	0	2.334453751
random	0	0.77815125	0	0	0	0
rangking	0	0	0	0	0	3.112605002
raskin	0	0	0	0	0	0.77815125
rata	0	0	0	0.477121255	0.477121255	0
rate	0	0	0	0	2.334453751	0
rendah	0	0	2.334453751	0	0	0
sakit	0	0	0	0	2.334453751	0
salah	0	0	0	0.301029996	0.301029996	0.301029996
satu	0	0	0	0.77815125	0	0
sebagai	0	0	0.77815125	0	0	0
sedia	0	0	0	0.77815125	0	0
sehingga	0	0	0.77815125	0	0	0
seleksi	0	0.77815125	0	0	0	0
sesuai	0	0	0.954242509	0	0	0.477121255
similarity	0	0	0	0	0	0.77815125
simpan	0	0	0	3.112605002	0	0

sistem	0	0	0	0	0	3.112605002
solution	0	0	0	0	0	0.77815125
staff	0	0	0	0	0	0.77815125
status	0	2.862727528	2.862727528	0	0	0
subtractive	0	0	4.668907502	0	0	0
sum	0	0	0.77815125	0	0	0
tahan	0	0	0	0	0.77815125	0
tahan	0	0	0	0	0	0.77815125
tambah	0	0	0.77815125	0	0	0
technique	0	0	0	0	0	0.77815125
teknologi	0	0	0	0	0.77815125	0
teliti	0	0.352182518	0.528273777	0.528273777	0	0.176091259
tentu	0	0	0	0	0	0.77815125
terima	0	0	0	0	0	1.556302501
tingkat	0	0	0	0	0	0.77815125
topsis	0	0	0	0	0	3.112605002
total	0	0	0	0	1.556302501	0
training	0	0	0	0	1.556302501	0
tulis	0	0	0	0	0	0.77815125
ubah	0	1.556302501	0	0	0	0
uji	0	0.602059991	0	0.301029996	0	0.602059991
ukur	0	1.556302501	0	0	0	0
umur	0	0.477121255	0.477121255	0	0	0
upaya	0	0	0	0	0	0.77815125
update	0	0	0	0	0.77815125	0
urut	0	0	0	0	0	0.77815125
usia	0	0	0.77815125	0	0	0
validasi	0	0.477121255	2.385606274	0	0	0
variable	0	0	0	0.77815125	0	0
variasi	0	0	0.77815125	0	0	0
xbi	0	0	1.556302501	0	0	0

Term	Panjang vektor q dan d1,d2,d3,d4,d5					
	q	d1	d2	d3	d4	d5
adaptive	0	0	0	0	5.449674316	0
ahli	0	0	0	0	0.605519368	0
ajar	0	0	0	0	0.605519368	0

akurasi	0	0	0	0	2.422077474	0
alamat	0	0	0	49.04706885	0	0
alami	0	0	0	0	0	0.605519368
algoritma	0.090619058	1.449904933	0	2.265476457	0	0
aman	0	0	0	9.688309896	0	0
anality	0	0	0	0	0	0.605519368
anp	0	0	0	0	0	15.13798421
aplikasi	0	0	0	2.422077474	0	0
application	0	0	0	0.605519368	0	0
awal	0	0	0	0	0	0.605519368
backpropagation	0	0	0	0	5.449674316	0
badan	0	0.910578767	2.048802225	0	0	0
bagi	0	0	0	0	0.605519368	0
baik	0	0	0	0	0.605519368	0
balita	0.227644692	5.691117293	0	0	0	0
banding	0	0	0.815571525	0.090619058	0	0.090619058
bangkit	0	0.227644692	0	0.910578767	0	0
bantu	0	0	0	0	0	5.449674316
basis	0	0	0	9.688309896	0	0
beda	0	0	0	0	0.605519368	0
beras	0	0	0	0	0	5.449674316
berat	0	0.227644692	0.910578767	0	0	0
bilang	0	0.605519368	0	0	0	0
bitcoin	0	0	0	87.19478906	0	0
bobot	0	0	0	0	0.910578767	0.227644692
boulding	0	0.227644692	0.227644692	0	0	0
butuh	0	0	0	0	0	0.605519368
camat	0	0	0	0	0	0.605519368
clustering	0.090619058	1.449904933	3.262286098	0	0	0
c-means	0.090619058	1.449904933	0.815571525	0	0	0
cocok	0	0	0	0	0	0.605519368
crossover	0	5.449674316	0	0	0	0
dapat	0	0.910578767	2.048802225	0	0	0
dasar	0	0	0	0.227644692	0.227644692	0
data	0	0	2.265476457	2.265476457	0	0.090619058
davies	0	0.227644692	0.227644692	0	0	0
dbi	0	0	2.422077474	0	0	0
deteksi	0	0.090619058	0	0.090619058	0.090619058	0
digest	0	0	0	0.605519368	0	0

distribusi	0	0	0	0	0	5.449674316
dukung	0	0	0	0	0	5.449674316
fitness	0	0.605519368	0	0	0	0
fokus	0	0	0	0.605519368	0	0
fungsi	0	2.422077474	0	0	0	0
fuzzy	0.090619058	2.265476457	9.061905829	0	0	0
gabung	0	0	0	0	0	0.605519368
gaji	0	0	0	5.449674316	0	0
genetika	0	9.688309896	0	0	0	0
gizi	0.090619058	4.440333856	3.262286098	0	0	0
guna	0	0	0	0.605519368	0	0
hambat	0	0	0	0	0	0.605519368
hamil	0	0	29.67044906	0	0	0
hash	0	0	0	9.688309896	0	0
hasil	0	0.124032526	0.279073184	0	0.124032526	0.279073184
hitung	0	3.642315067	0	0	0	0.227644692
ideal	0	0	0	0	0	0.605519368
implementasi	0	0	0	0	0	0.605519368
index	0	0.227644692	0.910578767	0	0	0
indodax	0	0	0	0.605519368	0	0
inisial	0	0.605519368	0	0	0	0
integritas	0	0	0	21.79869727	0	0
interface	0	0	0	0.605519368	0	0
iring	0	0	0	0	0.605519368	0
isi	0	0	0	0.605519368	0	0
iterasi	0	0.910578767	0	0	0.910578767	0
jenis	0	0	0	0	0	0.605519368
kanker	0	0	0	0	5.449674316	0
karyawan	0	0	0	21.79869727	0	0
kelompok	0.090619058	1.449904933	4.440333856	0	0	0
keluarga	0	0	0	0	0	2.422077474
kembang	0	0.227644692	0	0.910578767	0	0
klasifikasi	0.227644692	0	0	0.910578767	0	0
kriteria	0	0	0	0	0	5.449674316
lambat	0	0	0	0	0.605519368	0
latih	0	0	0	0	0.605519368	0
learning	0	0	0	0	5.449674316	0
lengan	0	0	0.605519368	0	0	0
lengkap	0	0	0	5.449674316	0	0

lindung	0	0	0	0	0	0.605519368
lingkar	0	0	0.605519368	0	0	0
lunak	0.227644692	0.227644692	0	0	0	0
maju	0	0	0	0	0.605519368	0
manual	0	0	0	0	0	0.605519368
masyarakat	0	0	0	0	0	0.605519368
matriks	0	0.605519368	0	0	0	0
md	0	0	0	15.13798421	0	0
medis	0	0	0	0	0.605519368	0
message	0	0	0	0.605519368	0	0
metode	0.00626967	0.100314716	0.156741743	0.156741743	0	0.22570811
milik	0	0	0	0	0	0.605519368
miskin	0	0	0	0	0	5.449674316
ms	0	0	0	2.422077474	0	0
mudah	0	0	0	0.605519368	0	0
mutasi	0	5.449674316	0	0	0	0
network	0	0	0	0	0	0.605519368
nilai	0	2.048802225	0	0	2.048802225	0
obyektif	0	0.605519368	0	0	0	0
oleh	0	0	0	0	0.605519368	0
optimasi	0	0	0	0	0.605519368	0
otentikasi	0	0	0	9.688309896	0	0
pangan	0	0	0	0	0	0.605519368
parameter	0	0.605519368	0	0	0	0
partisi	0	0.605519368	0	0	0	0
payudara	0	0	0	0	5.449674316	0
peluang	0	9.688309896	0	0	0	0
perangkat	0.227644692	0.227644692	0	0	0	0
perintah	0	0	0	0	0	0.605519368
persentase	0	0	0	0	0.227644692	0.227644692
pilih	0	0	0.227644692	0	0	0.227644692
pisah	0	0	0	2.422077474	0	0
populasi	0	5.449674316	0	0	0	0
preference	0	0	0	0	0	0.605519368
pribadi	0	0	0	0.605519368	0	0
process	0	0	0	0	0	0.605519368
program	0	0	0	0	0	0.605519368
programing	0	0	0	0.605519368	0	0
proses	0	0	0	0.090619058	0.815571525	0.090619058

pusat	0	0.605519368	0	0	0	0
putus	0	0	0	0	0	5.449674316
random	0	0.605519368	0	0	0	0
rangking	0	0	0	0	0	9.688309896
raskin	0	0	0	0	0	0.605519368
rata	0	0	0	0.227644692	0.227644692	0
rate	0	0	0	0	5.449674316	0
rendah	0	0	5.449674316	0	0	0
sakit	0	0	0	0	5.449674316	0
salah	0	0	0	0.090619058	0.090619058	0.090619058
satu	0	0	0	0.605519368	0	0
sebagai	0	0	0.605519368	0	0	0
sedia	0	0	0	0.605519368	0	0
sehingga	0	0	0.605519368	0	0	0
seleksi	0	0.605519368	0	0	0	0
sesuai	0	0	0.910578767	0	0	0.227644692
similarity	0	0	0	0	0	0.605519368
simpan	0	0	0	9.688309896	0	0
sistem	0	0	0	0	0	9.688309896
solution	0	0	0	0	0	0.605519368
staff	0	0	0	0	0	0.605519368
status	0	8.195208901	8.195208901	0	0	0
subtractive	0	0	21.79869727	0	0	0
sum	0	0	0.605519368	0	0	0
tahan	0	0	0	0	0.605519368	0
tahan	0	0	0	0	0	0.605519368
tambah	0	0	0.605519368	0	0	0
technique	0	0	0	0	0	0.605519368
teknologi	0	0	0	0	0.605519368	0
teliti	0	0.124032526	0.279073184	0.279073184	0	0.031008132
tentu	0	0	0	0	0	0.605519368
terima	0	0	0	0	0	2.422077474
tingkat	0	0	0	0	0	0.605519368
topsis	0	0	0	0	0	9.688309896
total	0	0	0	0	2.422077474	0
training	0	0	0	0	2.422077474	0
tulis	0	0	0	0	0	0.605519368
ubah	0	2.422077474	0	0	0	0
uji	0	0.362476233	0	0.090619058	0	0.362476233

ukur	0	2.422077474	0	0	0	0
umur	0	0.227644692	0.227644692	0	0	0
upaya	0	0	0	0	0	0.605519368
update	0	0	0	0	0.605519368	0
urut	0	0	0	0	0	0.605519368
usia	0	0	0.605519368	0	0	0
validasi	0	0.227644692	5.691117293	0	0	0
variable	0	0	0	0.605519368	0	0
variasi	0	0	0.605519368	0	0	0
xbi	0	0	2.422077474	0	0	0
Total Jumlah	1.460562786	86.93960065	112.8716957	278.6685272	60.17047916	108.9703742
Akar	1.208537458	9.324140746	10.62410917	16.69336776	7.756963269	10.43888759

Term	Wqk * Wdk				
	Wqd1	Wqd2	Wqd3	Wqd4	Wqd5
adaptive	0	0	0	0	0
ahli	0	0	0	0	0
ajar	0	0	0	0	0
akurasi	0	0	0	0	0
alamat	0	0	0	0	0
alami	0	0	0	0	0
algoritma	0.362476233	0	0.453095291	0	0
aman	0	0	0	0	0
anality	0	0	0	0	0
anp	0	0	0	0	0
aplikasi	0	0	0	0	0
application	0	0	0	0	0
awal	0	0	0	0	0
backpropagation	0	0	0	0	0
badan	0	0	0	0	0
bagi	0	0	0	0	0
baik	0	0	0	0	0
balita	1.138223459	0	0	0	0
banding	0	0	0	0	0
bangkit	0	0	0	0	0
bantu	0	0	0	0	0
basis	0	0	0	0	0

beda	0	0	0	0	0
beras	0	0	0	0	0
berat	0	0	0	0	0
bilang	0	0	0	0	0
bitcoin	0	0	0	0	0
bobot	0	0	0	0	0
boulding	0	0	0	0	0
butuh	0	0	0	0	0
camat	0	0	0	0	0
clustering	0.362476233	0.54371435	0	0	0
c-means	0.362476233	0.271857175	0	0	0
cocok	0	0	0	0	0
crossover	0	0	0	0	0
dapat	0	0	0	0	0
dasar	0	0	0	0	0
data	0	0	0	0	0
davies	0	0	0	0	0
dbi	0	0	0	0	0
deteksi	0	0	0	0	0
digest	0	0	0	0	0
distribusi	0	0	0	0	0
dukung	0	0	0	0	0
fitness	0	0	0	0	0
fokus	0	0	0	0	0
fungsi	0	0	0	0	0
fuzzy	0.453095291	0.906190583	0	0	0
gabung	0	0	0	0	0
gaji	0	0	0	0	0
genetika	0	0	0	0	0
gizi	0.634333408	0.54371435	0	0	0
guna	0	0	0	0	0
hambat	0	0	0	0	0
hamil	0	0	0	0	0
hash	0	0	0	0	0
hasil	0	0	0	0	0
hitung	0	0	0	0	0
ideal	0	0	0	0	0
implementasi	0	0	0	0	0
index	0	0	0	0	0

indodax	0	0	0	0	0
inisial	0	0	0	0	0
integritas	0	0	0	0	0
interface	0	0	0	0	0
iring	0	0	0	0	0
isi	0	0	0	0	0
iterasi	0	0	0	0	0
jenis	0	0	0	0	0
kanker	0	0	0	0	0
karyawan	0	0	0	0	0
kelompok	0.362476233	0.634333408	0	0	0
keluarga	0	0	0	0	0
kembang	0	0	0	0	0
klasifikasi	0	0	0.455289383	0	0
kriteria	0	0	0	0	0
lambat	0	0	0	0	0
latih	0	0	0	0	0
learning	0	0	0	0	0
lengan	0	0	0	0	0
lengkap	0	0	0	0	0
lindung	0	0	0	0	0
lingkar	0	0	0	0	0
lunak	0.227644692	0	0	0	0
maju	0	0	0	0	0
manual	0	0	0	0	0
masyarakat	0	0	0	0	0
matriks	0	0	0	0	0
md	0	0	0	0	0
medis	0	0	0	0	0
message	0	0	0	0	0
metode	0.025078679	0.031348349	0.031348349	0	0.037618018
milik	0	0	0	0	0
miskin	0	0	0	0	0
ms	0	0	0	0	0
mudah	0	0	0	0	0
mutasi	0	0	0	0	0
network	0	0	0	0	0
nilai	0	0	0	0	0
obyektif	0	0	0	0	0

oleh	0	0	0	0	0
optimasi	0	0	0	0	0
otentikasi	0	0	0	0	0
pangan	0	0	0	0	0
parameter	0	0	0	0	0
partisi	0	0	0	0	0
payudara	0	0	0	0	0
peluang	0	0	0	0	0
perangkat	0.227644692	0	0	0	0
perintah	0	0	0	0	0
persentase	0	0	0	0	0
pilih	0	0	0	0	0
pisah	0	0	0	0	0
populasi	0	0	0	0	0
preference	0	0	0	0	0
pribadi	0	0	0	0	0
process	0	0	0	0	0
program	0	0	0	0	0
programing	0	0	0	0	0
proses	0	0	0	0	0
pusat	0	0	0	0	0
putus	0	0	0	0	0
random	0	0	0	0	0
rangking	0	0	0	0	0
raskin	0	0	0	0	0
rata	0	0	0	0	0
rate	0	0	0	0	0
rendah	0	0	0	0	0
sakit	0	0	0	0	0
salah	0	0	0	0	0
satu	0	0	0	0	0
sebagai	0	0	0	0	0
sedia	0	0	0	0	0
sehingga	0	0	0	0	0
seleksi	0	0	0	0	0
sesuai	0	0	0	0	0
similarity	0	0	0	0	0
simpan	0	0	0	0	0
sistem	0	0	0	0	0

solution	0	0	0	0	0
staff	0	0	0	0	0
status	0	0	0	0	0
subtractive	0	0	0	0	0
sum	0	0	0	0	0
tahan	0	0	0	0	0
tahan	0	0	0	0	0
tambah	0	0	0	0	0
technique	0	0	0	0	0
teknologi	0	0	0	0	0
teliti	0	0	0	0	0
tentu	0	0	0	0	0
terima	0	0	0	0	0
tingkat	0	0	0	0	0
topsis	0	0	0	0	0
total	0	0	0	0	0
training	0	0	0	0	0
tulis	0	0	0	0	0
ubah	0	0	0	0	0
uji	0	0	0	0	0
ukur	0	0	0	0	0
umur	0	0	0	0	0
upaya	0	0	0	0	0
update	0	0	0	0	0
urut	0	0	0	0	0
usia	0	0	0	0	0
validasi	0	0	0	0	0
variable	0	0	0	0	0
variasi	0	0	0	0	0
xbi	0	0	0	0	0
Jumlah Total	4.155925153	2.931158214	0.939733023	0	0.037618018

Document	Cosine Similarity	Research Field
D1	0.368806682	Soft Computing
D2	0.228289859	Soft Computing
D3	0.0465801	Kriptografi
D4	0	Data Mining

D5	0.002981821	Soft Computing
----	-------------	----------------

Lampiran 2. Source Code Program

File: fungsi.php

```
<?php
//===== Kumpulan fungsi =====
//fungsi untuk melakukan preprocessing terhadap teks
//tokenizing, stopword/stoplist, stemming, dan casefolding.
//-----
function preprocessing($teks) {
    //import php koneksi
    include "koneksi.php";
    //Tokenizing (hilangkan tanda baca)
    $teks = str_replace(" ", " ", $teks);
    $teks = str_replace("-", " ", $teks);
    $teks = str_replace(")", " ", $teks);
    $teks = str_replace("(", " ", $teks);
    $teks = str_replace("\"", " ", $teks);
    $teks = str_replace("/", " ", $teks);
    $teks = str_replace("=", " ", $teks);
    $teks = str_replace(".", " ", $teks);
    $teks = str_replace(",", " ", $teks);
    $teks = str_replace(":", " ", $teks);
    $teks = str_replace("; ", " ", $teks);
    $teks = str_replace("!", " ", $teks);
    $teks = str_replace("?", " ", $teks);
    $teks = str_replace("%", " ", $teks);
    //remove numbers
    $teks = preg_replace('/0/', " ", $teks ); // remove numbers
    $teks = preg_replace('/1/', " ", $teks ); // remove numbers
    $teks = preg_replace('/2/', " ", $teks ); // remove numbers
    $teks = preg_replace('/3/', " ", $teks ); // remove numbers
    $teks = preg_replace('/4/', " ", $teks ); // remove numbers
    $teks = preg_replace('/5/', " ", $teks ); // remove numbers
    $teks = preg_replace('/6/', " ", $teks ); // remove numbers
    $teks = preg_replace('/7/', " ", $teks ); // remove numbers
    $teks = preg_replace('/8/', " ", $teks ); // remove numbers
    $teks = preg_replace('/9/', " ", $teks ); // remove numbers
    //casefolding
    $teks = strtolower(trim($teks));
    //stopword removal
```

\$bstoplist=array

("ada","adalah","adanya","adapun","agak","agaknya","agar","akan","akankah","akhir","akhiri","akhirnya","aku","akulah","amat","amatlah","anda","andalah","antar","antara","antaranya","apa","apaan","apabila","apakah","apalagi","apatah","artinya","asal","asalkan","atas","atau","ataukah","ataupun","awal","awalnya","bagai","bagaikan","bagaimana","bagaimanakah","bagaimanapun","bagi","bagian","bahkan","bahwa","bahwasanya","baik","bakal","bakalan","balik","banyak","bapak","baru","bawah","beberapa","begini","beginian","beginikah","beginilah","begitu","begitukah","begitulah","begitupun","bekerja","belakang","belakangan","belum","belumlah","benar","benarkah","benarlah","berada","berakhir","berakhirlah","berakhirnya","berapa","berapakah","berapalah","berapapun","berarti","berawal","berbagai","berdatangan","beri","berikan","berikut","berikutnya","berjumlah","berkalikali","berkata","berkehendak","berkeinginan","berkenaan","berlainan","berlalu","berlangsung","berlebihan","bermacam","bermacam-macam","bermaksud","bermula","bersama","bersamasama","bersiap","bersiap-siap","bertanya","bertanyanya","berturut","berturut-turut","bertutur","berujar","berupa","besar","betul","betulkah","biasa","biasanya","bila","bilakah","bisa","bisakah","boleh","bolehkah","bolehlah","buat","bukan","bukankah","bukanlah","bukannya","bulan","bung","cara","caranya","cukup","cukupkah","cukuplah","cuma","dahulu","dalam","dan","dapat","dari","daripada","datang","dekat","demi","demikian","demikianlah","dengan","depan","di","dia","diakhiri","diakhirinya","dialah","diantara","diantaranya","diberi","diberikan","diberikannya","dibuat","dibuatnya","didapat","didatangkan","digunakan","diibaratkan","diibaratkannya","diingat","diingatkan","diinginkan","dijawab","dijelaskan","dijelaskannya","dikarenakan","dikatakan","dikatakannya","dikerjakan","diketahui","diketahuinya","dikira","dilakukan","dilalui","dilihat","dimaksud","dimaksudkan","dimaksudkannya","dimaksudnya","diminta","dimintai","dimisalkan","dimulai","dimulailah","dimulainya","dimungkinkan","dini","dipastikan","diperbuat","diperbuatnya","dipergunakan","diperkirakan","diperlihatkan","diperlukan","diperlukannya","dipersoalkan","dipertanyakan","dipunyai","diri","dirinya","disampaikan","disebut","disebutkan","disebutkannya","disini","disinilah","ditambahkan","ditandaskan","ditanya","ditanyai","ditanyakan","ditegaskan","ditujukan","ditunjuk","ditunjuki","ditunjukkan","ditunjukkannya","ditunjuknya","dituturkan","dituturkannya","diucapkan","diucapkannya","diungkapkan","dong","dua","dulu","empat","enggak","enggaknya","entah","entahlah","guna","gunakan","hal","hampir","hanya","hanyalah","hari","harus","haruslah","harusnya","hendak","hendaklah","hendaknya","hingga","ia","ialah","ibarat","ibaratkan","ibaratnya","ibu","ikut","ingat","ingatan","ingin","inginkah","inginkan","ini","inikah","inilah","itu","itukah","itulah","jadi","jadilah","jadinya","jangan","janganlah","jauh","jawab","jawaban","jawabnya","jelas","jelaskan","jelaslah","jelasnya","jika","jikalau","juga","jumlah","jumlahnya","justru","kala","kalaulah","kalaupun","kalian","kami","kamilah","kamu","kamulah","kan","kapan","kapankah","kapanpun","karena","karenanya","kasus","kata","katakan","katakanlah","katanya","ke","keadaan","kebetulan","kecil","kedua","keduanya","keinginan","kelamaan","kelihatan","kelihatannya","kelima","keluar","kembali","kemudian","kemung

kinan", "kemungkinannya", "kenapa", "kepada", "kepadanya", "kesampaian", "keseluruhan", "keseluruhannya", "keterlaluhan", "ketika", "khususnya", "kini", "kinilah", "kira", "kira-kira", "kiranya", "kita", "kitalah", "kok", "kurang", "lagi", "lagian", "lah", "lain", "lainnya", "lalu", "lama", "lamanya", "lanjut", "lanjutnya", "lebih", "lewat", "lima", "luar", "macam", "maka", "makanya", "makin", "malah", "malahan", "mampu", "mampukah", "mana", "mana-kala", "manalagi", "masa", "masalah", "masalahnya", "masih", "masihkah", "masing", "masing-masing", "mau", "maupun", "melainkan", "melakukan", "melalui", "melihat", "melihatnya", "memang", "memastikan", "memberi", "memberikan", "membuat", "memerlukan", "memihak", "meminta", "memintakan", "memisalkan", "memperbuat", "mempergunakan", "memperkirakan", "memperlihatkan", "mempersiapkan", "mempersoalkan", "mempertanyakan", "mempunyai", "memulai", "memungkinkan", "menaiki", "menambahkan", "menandaskan", "menanti", "menantianti", "menantikan", "menanya", "menanyai", "menanyakan", "mendapat", "mendapatkan", "mendatang", "mendatangi", "mendatangkan", "mendegaskan", "mengakhiri", "mengapa", "mengatakan", "mengatakannya", "mengenai", "mengerjakan", "mengetahui", "mengggunakan", "menghendaki", "mengibaratkan", "mengibaratkannya", "mengingat", "mengingatkan", "menginginkan", "mengira", "mengucapkan", "mengucapkannya", "mengungkapkan", "menjadi", "menjawab", "menjelaskan", "menuju", "menunjuk", "menunjuki", "menunjukkan", "menunjuknya", "menurut", "menuturkan", "menyampaikan", "menyangkut", "menyatakan", "menyebutkan", "menyeluruh", "menyiapkan", "merasa", "mereka", "merekalah", "merupakan", "meski", "meskipun", "meyakini", "meyakinkan", "minta", "mirip", "misal", "misalkan", "misalnya", "mula", "mulai", "mula-ailah", "mulanya", "mungkin", "mungkinkah", "nah", "naik", "namun", "nanti", "nantinya", "nyaris", "nyatanya", "oleh", "olehnya", "pada", "padahal", "padanya", "pak", "paling", "panjang", "pantas", "para", "pasti", "pastilah", "penting", "pentingnya", "per", "percuma", "perlu", "perluakah", "perlunya", "pernah", "persoalan", "pertama", "pertama-tama", "pertanyaan", "pertanyakan", "pihak", "pihaknya", "pukul", "pula", "pun", "punya", "rasa", "rasanya", "rata", "rupanya", "saat", "saatnya", "saja", "sajalah", "saling", "sama", "samasama", "sambil", "sampai", "sampaisampai", "sampaikan", "sana", "sangat", "sangatlah", "satu", "saya", "sayalah", "se", "sebab", "sebabnya", "sebagai", "sebagaimana", "sebagainya", "sebagian", "sebaik", "sebaikbaiknya", "sebaiknya", "sebaliknya", "sebanyak", "sebegini", "sebegitu", "sebelum", "sebelumnya", "sebenarnya", "seberapa", "sebesar", "sebetulnya", "sebisanya", "sebuah", "sebut", "sebutlah", "sebutnya", "secara", "secukupnya", "sedang", "sedangkan", "sedemikian", "sedikit", "sedikitnya", "seenaknya", "segala", "segalanya", "segera", "seharusnya", "sehingga", "seingat", "sejak", "sejauh", "sejenak", "sejumlah", "sekadar", "sekadarnya", "sekali", "sekaligali", "sekalian", "sekaligus", "sekalipun", "sekarang", "sekarang", "sekecil", "seketika", "sekiranya", "sekitar", "sekitarnya", "sekurang-kurangnya", "sekurangnya", "sela", "selain", "selaku", "selalu", "selama", "selama-lamanya", "selamanya", "selanjutnya", "seluruh", "seluruhnya", "semacam", "semakin", "semampu", "semampunya", "semasa", "semasih", "semata", "sematamata", "semaunya", "sementara", "semisal", "semisalnya", "sempat", "semua", "semuanya", "semula", "sendiri", "sendirian", "sendirinya", "seolah", "seolaholah", "seorang", "sepanjang", "sepantasnya", "sepantasnyalah", "seperlunya", "seperti", "sepertinya", "sepihak", "sering", "seringnya", "

```
serta","serupa","sesaat","sesama","sesampai","sesegera","sesekali","seseorang","sesu
atu","sesuatunya","sesudah","sesudahnya","setelah","setempat","setengah","seterusny
a","setiap","setiba","setibanya","setidaktidaknya","setidaknya","setinggi","seusai","se
waktu","siap","siapa","siapakah","siapapun","sini","sinilah","soal","soalnya","suatu",
"sudah","sudahkah","sudahlah","supaya","tadi","tadinya","tahu","tahun","tak","tamb
ah","tambahnya","tampak","tampaknya","tandas","tandasnya","tanpa","tanya","tanya
kan","tanyanya","tapi","tegas","tegasnya","telah","tempat","tengah","tentang","tentu"
,"tentulah","tentunya","tepat","terakhir","terasa","terbanyak","terdahulu","terdapat","t
erdiri","terhadap","terhadapnya","teringat","teringatingat","terjadi","terjadilah","terja
dinya","terkira","terlalu","terlebih","terlihat","termasuk","ternyata","tersampaikan","t
ersebut","tersebutlah","tertentu","tertuju","terus","terutama","tetap","tetapi","tiap","ti
ba","tibatiba","tidak","tidakkah","tidaklah","tiga","tinggi","toh","tunjuk","turut","tutu
r","tuturnya","ucap","ucapnya","ujar","ujarnya","umum","umumnya","ungkap","ung
kapnya","untuk","usah","usai","waduh","wah","wahai","waktu","waktunya","walau",
"walaupun","wong","yaitu","yakin","yakni","yang","nya");
```

```
    foreach ($bstoplist as $i => $value) {
        $teks = str_replace($bstoplist[$i], "", $teks);
    }
    //stemming
    //membuat query berdasarkan tabel tbstem dan dibandingkan dengan dokumen
diurutkan berdasarkan id dari kecil ke besar.
    $query_tbstem = mysqli_query($conn,"SELECT * FROM tbstem ORDER BY
Id");
    while($row_tbstem = mysqli_fetch_array($query_tbstem)) {
        $teks = str_replace($row_tbstem['Term'], $row_tbstem['Stem'], $teks);
    }
    //kembalikan teks yang telah dipreproses
    $teks = strtolower(trim($teks));
    return $teks;
}
//end function preproses
//-----
```

```
//Fungsi menyimpan dataresult.
```

```
//-----
function saveDataResult(){
    include "koneksi.php";
    $get_keyword = $_GET['search'];
    $user_keyword = $get_keyword;
    $query_tbdataresult = mysqli_query($conn, "SELECT jumlahKeyword FROM
tbdataresult WHERE keyword = '$user_keyword'");
```

```

    $query_tbsimilarity = mysqli_query($conn, "SELECT * FROM tbsimilarity
WHERE cosineSimilarity = (SELECT MAX(cosineSimilarity) FROM tbsimilarity)");
    $row_q_tbsimilarity = mysqli_fetch_array($query_tbsimilarity);
    $dokumenId = $row_q_tbsimilarity['dokumenId'];
    @$amount_perkeyword = mysqli_num_rows($query_tldataresult);
    if ($amount_perkeyword > 0){
        $row_amount_perkeyword = mysqli_fetch_array($query_tldataresult);
        $jumlahKeyword = $row_amount_perkeyword['jumlahKeyword'];
        $jumlahKeyword++;
        mysqli_query($conn, "UPDATE tldataresult SET jumlahKeyword =
$jumlahKeyword WHERE keyword = '$user_keyword'");
    }
    else{
        mysqli_query($conn, "INSERT INTO tldataresult (keyword,
dokumenId, jumlahKeyword) VALUES ('$user_keyword', $dokumenId, 1)");
    }
}
//-----

//Fungsi membuat term atau term.
//-----
function makeTerm(){
    include "koneksi.php";
    //Hapus tabel tbterm sebelumnya.
    mysqli_query($conn, "TRUNCATE TABLE tbterm");
    //membuat query berdasarkan tabel tbtugasakhir (title dan abstract=TA).
    $query_tbtugasakhir = mysqli_query($conn, "SELECT * FROM tbtugasakhir
ORDER BY id");
    //Melakukan simpan ke tbterm pada setiap baris dokumen
    while ($row_tbtugasakhir = mysqli_fetch_array($query_tbtugasakhir)){
        //mengambil kolom 'id' pada tabel tbtugasakhir setiap baris dokumen.
        $dokumenId = $row_tbtugasakhir['id'];
        //mengambil/menyatukan kolom 'title dan abstract' pada tabel
tbtugasakhir setiap baris dokumen
        $term_title_abstract = $row_tbtugasakhir['title']."
".$row_tbtugasakhir['abstract'];
        //melakukan preprocess terhadap term title dan abstract setiap baris
dokumen.
        $term_title_abstract = preprocessing($term_title_abstract);
        // //memisahkan string term setiap baris dokumen
        $sexterm_title_abstract = explode(" ", trim($term_title_abstract));
        // //melakukan perulangan setiap term pada setiap baris dokumen.

```

```

        foreach ($sexterm_title_abstract as $i => $value){
            //membuat kondisi jika kolom $term_title_abstract tidak null
atau kosong.
            if($sexterm_title_abstract[$i] != ""){
                // mengambil query kolom jumlah term
                $query_tborderm = mysqli_query($conn, "SELECT
jumlahTerm FROM tborderm WHERE term = '$sexterm_title_abstract[$i]' AND
dokumenId = $dokumenId");
                //mengambil baris kolom jumlah term
                @$amount_perterm =
mysqli_num_rows($query_tborderm);
                //membuat kondisi jika baris term sudah ada maka
jumlah term + 1
                if($amount_perterm > 0){
                    $row_amount_perterm =
mysqli_fetch_array($query_tborderm);
                    $amount_term =
$row_amount_perterm['jumlahTerm'];
                    $amount_term++;
                    mysqli_query($conn, "UPDATE tborderm SET
jumlahTerm = $amount_term WHERE term = '$sexterm_title_abstract[$i]' AND
dokumenId = $dokumenId");
                }
                else {
                    //Insert term ke tborderm per dokumenId dengan
setiap term berjumlah 1
                    mysqli_query($conn, "INSERT INTO tborderm
(term, dokumenId, jumlahTerm) VALUES ('$sexterm_title_abstract[$i]', $dokumenId,
1)");
                }//end else
            }//end if
        }//end foreach
    }//end while
}//end function
//-----

//-----
//fungsi hitungbobot, menggunakan perhitungan tf.idf
function tfidfTerm(){
    include "koneksi.php";
    // mengambil nilai yang berbeda pada kolom dokumenId untuk setiap baris di
tabel tborderm.

```

```

$query_tbterm = mysqli_query($conn, "SELECT DISTINCT dokumenId from
tbterm");
// mengambil nilai total jumlah dokumen, n (ada berapa dokumen.)
$n_doc = mysqli_num_rows($query_tbterm);
//mengambil query kolom term pada tabel tbterm
$query_tbterm_id = mysqli_query($conn, "SELECT * FROM tbterm ORDER
BY id");

while($row_query_tbterm_id = mysqli_fetch_array($query_tbterm_id)){
    $term = $row_query_tbterm_id['term'];
    //term frequency
    $tf = $row_query_tbterm_id['jumlahTerm'];
    $id = $row_query_tbterm_id['id'];
    //Mengambil query jumlah dokumen yang mewakili term. N term
    (Berapa dokumen yang mewakili term)
    $query_tbterm_N_term = mysqli_query($conn, "SELECT COUNT(*)
AS N FROM tbterm WHERE term = '$term'");
    $row_query_tbterm_N_term =
mysqli_fetch_array($query_tbterm_N_term);
    //Menambah jumlah dokumen yang mewakili term dari keyword
    //mengambil query term dari tabel tbkeyword yang sama dengan term
    tbterm.
    $query_tbkeyword_term = mysqli_query($conn, "SELECT * FROM
tbkeyword WHERE term = '$term'");
    $row_query_tbkeyword_term =
mysqli_fetch_array($query_tbkeyword_term);
    $amount_term_keyword =
mysqli_num_rows($query_tbkeyword_term);
    if($amount_term_keyword > 0){
        $N_term =
$row_query_tbterm_N_term['N']+$row_query_tbkeyword_term['jumlahTerm'];
    }else{
        $N_term = $row_query_tbterm_N_term['N'];
    }
    $n_doc_after = $n_doc+1;
    //idf = jumlah total dokumen / jumlah dokumen yang mewakili term
    $idf = log10($n_doc_after/$N_term);
    $tfidf = $tf * $idf;
    mysqli_query($conn, "UPDATE tbterm SET tfidf = $tfidf WHERE id
= $id");
} //end while
} //end function

```

```

//-----

//-----
//Fungsi hitung panjang vektor setiap dokumen.
//Akar dari penjumlahan kuadrat setiap term per dokumen.
function vectorLengthDoc(){
    include "koneksi.php";
    //hapus isi tabel tbvektor sebelumnya.
    mysqli_query($conn, "TRUNCATE TABLE tbvektor");
    //mengambil query nilai kolom dokumenId yang berbeda
    //hitung panjang vektor setiap dokumenId yang berbeda
    //simpan kedalam tabel tbvektor.
    $query_tborderm = mysqli_query($conn, "SELECT DISTINCT dokumenId
FROM tborderm");
    while($row_query_tborderm = mysqli_fetch_array($query_tborderm)){
        $doc_id = $row_query_tborderm['dokumenId'];
        $query_tborderm_docid = mysqli_query($conn, "SELECT tfidf FROM
tborderm WHERE dokumenId = $doc_id");
        //panjang vektor = akar penjumlahan kuadrat setiap term per dokumen
        $vector_length = 0;
        while($row_query_tborderm_docid =
mysqli_fetch_array($query_tborderm_docid)){
            $vector_length = sqrt($vector_length +
$row_query_tborderm_docid['tfidf'] * $row_query_tborderm_docid['tfidf']);
        }
    }
    //end while baris vektor

    $vector_length = sqrt($vector_length);

    mysqli_query($conn, "INSERT INTO tbvektor (dokumenId,
panjangVektor) VALUES ($doc_id, $vector_length)");
}
//end while baris dokumen id
}
//end function
//-----

//-----
//Fungsi hitung bobot keyword (tfidf)
function tfidfKeyword(){
    include "koneksi.php";
    //mengambil nilai yang berbeda dari dokumen id pada tborderm dan jumlah total
dokumen

```

```

        $query_tborderm_docid = mysqli_query($conn, "SELECT DISTINCT
dokumenId FROM tborderm");
        $n_doc = mysqli_num_rows($query_tborderm_docid);
        $query_tborderm_keyword = mysqli_query($conn, "SELECT * FROM tborderm_keyword");
        while($row_query_tborderm_keyword = mysqli_fetch_array($query_tborderm_keyword)){
            $id_keyword = $row_query_tborderm_keyword['id'];
            $tf_keyword = $row_query_tborderm_keyword['jumlahTerm'];
            $term_keyword = $row_query_tborderm_keyword['term'];
            //mencari dokumen yang mewakili term tborderm_keyword yang sama dengan
term tborderm
            $query_tborderm_totalterm = mysqli_query($conn, "SELECT
COUNT(*) AS totalTerm FROM tborderm WHERE term = '$term_keyword'");
            $row_query_tborderm_totalterm =
mysqli_fetch_array($query_tborderm_totalterm);
            //jumlah dokumen tborderm_keyword + dokumen keyword.
            $n_doc_after = $n_doc + 1;
            $N_term = $row_query_tborderm_totalterm['totalTerm']+1;
            //rumus tfidf keyword
            $idf = log10($n_doc_after/$N_term);
            $tfidf = $tf_keyword * $idf;
            mysqli_query($conn, "UPDATE tborderm_keyword SET tfidf = $tfidf WHERE
id = $id_keyword");
        }//end while
    }//end function tfidfKeyword.
//-----

//-----
//Fungsi hitung 2 vektor yang sama (wdwdi).
function wdwdi(){
    include "koneksi.php";
    //hapus perhitungan wdwdi sebelumnya dan kosongkan.
    mysqli_query($conn, "DELETE FROM tborderm_wdwdi");
    mysqli_query($conn, "TRUNCATE TABLE tborderm_wdwdi");
    $get_tfidf_keyword = mysqli_query($conn, "SELECT * FROM tborderm_keyword
ORDER BY id");
    while($row_get_tfidf_keyword = mysqli_fetch_array($get_tfidf_keyword)){
        $term_keyword = $row_get_tfidf_keyword['term'];
        $tfidf_keyword = $row_get_tfidf_keyword['tfidf'];
        $query_tborderm = mysqli_query($conn, "SELECT * FROM tborderm
WHERE term = '$term_keyword'");
        $wdwdi = 0;
        while($row_query_tborderm = mysqli_fetch_array($query_tborderm)){

```

```

        $tbterm_doc_id = $row_query_tbterm['dokumenId'];
        $wdwdi = $tfidf_keyword * $row_query_tbterm['tfidf'];
        mysqli_query($conn, "INSERT INTO tbwdwdi
(dokumenId,wdwdi) VALUES ($tbterm_doc_id,$wdwdi)");
    }//end while
} //end while
} //end function
//-----

//-----
//Fungsi menghitung kemiripan.
function cosineSimilarity(){
    include "koneksi.php";
    mysqli_query($conn, "DELETE FROM tbsimilarity");
    mysqli_query($conn, "TRUNCATE TABLE tbsimilarity");
    $query_tbkeyword_tfidf = mysqli_query($conn, "SELECT tfidf FROM
tbkeyword");
    $vector_length_keyword = 0;
    while($row_query_tbkeyword_tfidf =
mysqli_fetch_array($query_tbkeyword_tfidf)){
        $tfidf_keyword = $row_query_tbkeyword_tfidf['tfidf'];
        $vector_length_keyword = $vector_length_keyword + ($tfidf_keyword
* $tfidf_keyword);
    }
    $vector_length_keyword = sqrt($vector_length_keyword);
    $query_tbwdwdi = mysqli_query($conn, "SELECT dokumenId, SUM(wdwdi)
as total_wdwdi FROM tbwdwdi GROUP BY dokumenId");
    while($row_query_tbwdwdi = mysqli_fetch_array($query_tbwdwdi)){
        $total_wdwdi = $row_query_tbwdwdi['total_wdwdi'];
        $dokumen_id = $row_query_tbwdwdi['dokumenId'];
        $query_tbvektor = mysqli_query($conn, "SELECT * FROM tbvektor
WHERE dokumenID = $dokumen_id");
        while($row_query_tbvektor = mysqli_fetch_array($query_tbvektor)){
            $vector_doc_id = $row_query_tbvektor['dokumenId'];
            $vector_length_doc = $row_query_tbvektor['panjangVektor'];
            $cosine_similarity =
$total_wdwdi/($vector_length_keyword*$vector_length_doc);
            if($cosine_similarity > 0.1){
                mysqli_query($conn, "INSERT INTO tbsimilarity(dokumenId,
cosineSimilarity) VALUES ($vector_doc_id, $cosine_similarity)");
            }
        }
    }
}

```



```
    }  
  }  
?>
```

File: index.php

```
<?php  
include "koneksi.php";  
$query_tbtugasakhir = mysqli_query($conn, "SELECT * FROM tbtugasakhir");  
$total_result_query = mysqli_num_rows($query_tbtugasakhir);  
$limit_data = 10;  
$total_page = ceil($total_result_query / $limit_data);  
$active_page = (isset($_GET["page"]) ) ? $_GET["page"] : 1;  
  
$data_index = ($limit_data * $active_page) - $limit_data;  
$query = mysqli_query($conn, "SELECT * FROM tbtugasakhir LIMIT $data_index,  
$limit_data");  
  
?>
```

```
<!DOCTYPE html>  
<html lang="en">
```

```
<head>
```

```
  <!-- Required meta tags -->  
  <meta charset="utf-8" />  
  <meta name="viewport" content="width=device-width, initial-scale=1" />  
  <!-- Bootstrap CSS -->  
  <link  
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"  
rel="stylesheet"  
  integrity="sha384-  
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLA  
SjC" crossorigin="anonymous" />  
  <!-- Custom CSS -->  
  <link rel="stylesheet" href="assets/styleIndex.css" />  
  <link rel="preconnect" href="https://fonts.googleapis.com">  
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>  
  <link  
href="https://fonts.googleapis.com/css2?family=Inter:wght@100;200;300;400;500;6  
00;700;800;900&display=swap"  
rel="stylesheet">  
  <title>Pencarian Data Tugas Akhir</title>  
</head>
```

```

<body>
  <!-- Navbar -->
  <section>
    <nav class="navbar navbar-expand-lg navbar-dark border-bottom border-secondary">
      <div class="container">
        <a class="navbar-brand active fs-5" href="index.php" style="font-weight: 500;">Yuwan Samega</a>
        <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
          <div class="navbar-nav ms-auto">
            <a class="nav-link disabled" aria-current="page" href="term.php">Term</a>
            <a class="nav-link disabled" href="vektor.php">Vektor</a>
            <a class="nav-link disabled" href="tfidf.php">Tf-Idf</a>
            <a class="nav-link disabled">WdWdi</a>
          </div>
        </div>
      </div>
    </nav>
  </section>
  <!-- Title -->
  <section name="title">
    <div class="container text-center mt-5 mb-5">
      <h2 style="color: #14d9b6;">Pencarian Data Tugas Akhir</h2>
    </div>
  </section>
  <!-- End of Title -->
  <!-- Forms -->
  <section>
    <div class="container mb-5">
      <div class="row justify-content-center">
        <div class="col-md-8">
          <form action="searchProcess.php" method="GET">
            <div class="mb-3">
              <label for="InputData" class="form-label text-light">Keyword</label>
              <div class="input-group">
                <input type="text" class="form-control" id="searchData" aria-describedby="searchData" name="search" placeholder="Input kata kunci.." />
                <a id="cari">

```

```

                <button type="submit" class="btn text-light"
                    style="border: 1px solid; border-
color:#14d9b6;">Cari</button>
            </a>
        </div>
    </div>
</form>
</div>
</div>
</div>
</div>
</section>
<!-- End of Forms -->
<div class=" container mb-5 border-bottom border-secondary">
</div>
<!-- Data List -->
<section name="data-list">
    <div class="container mb-5">
        <div class="row justify-content-center">
            <div class="col-8">
                <div class="card text-center border border-secondary">
                    <div class="card-body">
                        <h5 class="card-title" style="color: #14d9b6;">Data Tugas
Akhir</h5>
                        <p class="card-text text-light">Lihat list data tugas akhir
mahasiswa universitas sriwijaya
fakultas
ilmu komputer angkatan 2012-2016
                        </p>
                        <a href="listdata.php">
                            <button class="btn btn-outline-light">Lihat</button>
                        </a>
                    </div>
                </div>
            </div>
        </div>
    </div>
</section>
<!-- End of Data List -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"

```

```

        integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIax
VXM" crossorigin="anonymous">
    </script>
</body>

</html>

```

File:koneksi.php

```

<?php
$conn = mysqli_connect("localhost", "root", "", "dbdatatugasakhir");
?>

```

File:listdata.php

```

<?php
include "koneksi.php";
$query_tbtugasakhir = mysqli_query($conn, "SELECT * FROM tbtugasakhir");
$total_result_query = mysqli_num_rows($query_tbtugasakhir);
$limit_data = 10;
$total_page = ceil($total_result_query / $limit_data);
$active_page = (isset($_GET["page"]) ) ? $_GET["page"] : 1;
$data_index = ($limit_data * $active_page) - $limit_data;
$query = mysqli_query($conn, "SELECT * FROM tbtugasakhir LIMIT $data_index,
$limit_data");
?>
<!DOCTYPE html>
<html lang="en">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <!-- Bootstrap CSS -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet"
        integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLA
SjC" crossorigin="anonymous" />
    <!-- Custom CSS -->
    <link rel="stylesheet" href="css/custom.css" />
    <title>List Data Tugas Akhir</title>

```

```

</head>

<body>
  <!-- Navbar -->
  <section>
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container">
        <a class="navbar-brand active" href="index.php">Tugas Akhir</a>
        <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
          <div class="navbar-nav ms-auto">
            <a class="nav-link disabled" href="term.php">Term</a>
            <a class="nav-link disabled" href="vektor.php">Vektor</a>
            <a class="nav-link disabled" href="tfidf.php">Tf-Idf</a>
            <a class="nav-link disabled" href="wdwdi.php">WdWdi</a>
          </div>
        </div>
      </div>
    </nav>
  </section>
  <!-- Title -->
  <section name="title">
    <div class="container text-center mt-5 mb-5">
      <h2>Tabel Data Tugas Akhir</h2>
    </div>
  </section>
  <!-- End of Title -->
  <!-- Table -->
  <section>
    <div class="container">
      <div class="row justify-content-center">
        <div class="col">
          <table class="table">
            <thead class="table-dark">
              <tr style="font-size: small">
                <th scope="col">No</th>
                <th scope="col" style="width: 40%;">Judul Tugas Akhir</th>
                <th scope="col">Category</th>
                <th scope="col">Nim</th>
                <th scope="col">Nama</th>
                <th scope="col">Kelas</th>
                <th scope="col">Angkatan</th>
              </tr>
            </thead>
          </table>
        </div>
      </div>
    </div>
  </section>

```

```

        </tr>
    </thead>
    <tbody>
        <?php $i=$data_index+1; ?>
        <?php while ($row = mysqli_fetch_array($query)) : ?>
        <tr style="font-size: small">
            <td scope="row"><?= $i++; ?></td>
            <td><?= $row["title"]; ?></td>
            <td><?= $row["research_area"]; ?></td>
            <td><?= $row["nim"]; ?></td>
            <td><?= $row["stud_name"]; ?></td>
            <td><?= $row["class"]; ?></td>
            <td><?= $row["year"]; ?></td>
        </tr>
        <?php endwhile; ?>
    </tbody>

</table>

</div>
</div>

<div class="row justify-content-center">
    <div class="col">
        <nav aria-label="Page navigation example">
            <ul class="pagination justify-content-center">

                <?php if ($active_page > 1) : ?>
                <li class="page-item">
                    <a class="page-link" href="?page=<?= 1 ?>" aria-
label="Previous">
                        <span aria-hidden="true">&laquo;</span>
                    </a>
                </li>
                <li class="page-item"><a class="page-link"
                    href="?page=<?= $active_page-1 ?>"><?= $active_page-
1 ?></a></li>
                <?php endif; ?>
                <li class="page-item"><a class="page-link"

```

```

                href="?page=<?= $active_page ?>"><?=
$active_page ?></a></li>
                <?php if ($active_page < $total_page) : ?>
                <li class="page-item"><a class="page-link"
                href="?page=<?= $active_page+1 ?>"><?=
$active_page+1 ?></a></li>
                <li class="page-item">
                <a class="page-link" href="?page=<?= $total_page ?>" aria-
label="Next">
                <span aria-hidden="true">&raquo;</span>
                </a>
                </li>
                <?php endif; ?>
            </ul>
        </nav>
    </div>
</div>
</div>
</section>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
MrcW6ZMFY1zclA8NI+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtflax
VXM" crossorigin="anonymous">
</script>
</body>

</html>

```

File:searchProcess.php

```

<?php
include "koneksi.php";
include "fungsi.php";

$get_keyword = $_GET['search'];
$user_keyword = $get_keyword;

// mysqli_query($conn, "TRUNCATE TABLE tbdataresult");
// mysqli_query($conn, "INSERT INTO tbdataresult (keyword, jumlahKeyword)
VALUES ('$user_keyword', 1)");

```

```

mysqli_query($conn, "DELETE FROM tbkeyword");
// mysqli_query($conn, "TRUNCATE TABLE tbterm");

$user_keyword = $get_keyword;
$user_keyword = preprocessing($user_keyword);

$words = str_word_count($user_keyword, 1);
$words = array_count_values($words);
$count_words = count($words);

if($count_words > 0){

    foreach($words as $w => $value){

        mysqli_query($conn, "INSERT INTO tbkeyword (term, jumlahTerm) VALUES
('$w', $value)");
    }
    header('location:result.php?search='.$get_keyword);
}

?>

```

File:result.php

```

<?php
include "koneksi.php";
include "fungsi.php";

$getKeyword = $_GET['search'];

// $user_keyword = $get_keyword;

// mysqli_query($conn, "TRUNCATE TABLE tbdataresult");
// mysqli_query($conn, "INSERT INTO tbdataresult (keyword, jumlahKeyword)
VALUES ('$user_keyword', 1)");

```



```

// makeTerm();
// tfidfTerm(); /*
// vectorLengthDoc();
tfidfKeyword(); /*
wdwdi(); /*
cosineSimilarity(); /*
saveDataResult();

$query = mysql_query($conn, "SELECT
tbsimilarity.dokumenId,tbsimilarity.cosineSimilarity,tbtugasakhir.id,tbtugasakhir.title,
tbtugasakhir.abstract,tbtugasakhir.nim,tbtugasakhir.stud_name,tbtugasakhir.class,tbtu
gasakhir.year,tbtugasakhir.status,tbtugasakhir.research_area FROM tbsimilarity
INNER JOIN tbtugasakhir ON tbsimilarity.dokumenId = tbtugasakhir.id ORDER BY
cosineSimilarity DESC");
$total_result_query = mysql_num_rows($query);

```

?>

```

<!DOCTYPE html>
<html lang="en">

```

```

<head>

```

```

  <!-- Required meta tags -->

```

```

  <meta charset="utf-8" />

```

```

  <meta name="viewport" content="width=device-width, initial-scale=1" />

```

```

  <!-- Bootstrap CSS -->

```

```

  <link

```

```

    href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"

```

```

    rel="stylesheet"

```

```

      integrity="sha384-

```

```

      EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWFspd3yD65VohhpuuCOmLA

```

```

      SjC" crossorigin="anonymous" />

```

```

  <!-- Custom CSS -->

```

```

  <link rel="stylesheet" href="assets/styleResult.css" />

```

```

  <link rel="preconnect" href="https://fonts.googleapis.com">

```

```

  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

```

```

<link
href="https://fonts.googleapis.com/css2?family=Inter:wght@100;200;300;400;500;6
00;700;800;900&display=swap"
rel="stylesheet">

<title>Pencarian Data Tugas Akhir</title>
</head>

<body>
  <!-- Navbar -->
  <section>

    <nav class="navbar navbar-expand-lg navbar-dark border-bottom border-
secondary">
      <div class="container">
        <a class="navbar-brand" href="index.php">Yuwan Samega</a>

        <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
          <div class="navbar-nav ms-auto">
            <a class="nav-link" aria-current="page" href="term.php">Term</a>
            <a class="nav-link" href="vektor.php">Vektor</a>
            <a class="nav-link" href="tfidf.php">Tf-Idf</a>
            <a class="nav-link" href="wdwdi.php">WdWdi</a>
          </div>
        </div>
      </nav>

    </section>
    <!-- Title -->
    <section name="title">
      <div class="container text-center mt-5 mb-5">
        <a href="index.php" style="text-decoration: none; color: black;">
          <h2 style="color: #14d9b6;">Pencarian Data Tugas Akhir</h2>
        </a>
      </div>
    </section>
    <!-- End of Title -->

    <!-- Forms -->
    <section>
      <div class="container mb-5">

```

```

    <div class="row justify-content-center">
      <div class="col-md-8">
        <form action="searchProcess.php" method="GET">
          <div class="mb-3">
            <label for="InputData" class="form-label text-
light">Keyword</label>
            <div class="input-group">
              <input type="text" class="form-control" id="searchData" aria-
describedby="searchData"
              name="search" value="<?= $getKeyword ?>" />
              <a id="cari">
                <button type="submit" class="btn text-light"
                style="border: 1px solid; border-
color:#14d9b6;">Submit</button>
              </a>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
</section>
<!-- End of Forms -->

<div class=" container mb-5 border-bottom border-secondary">
</div>

<!-- Table -->
<section>
  <div class="container border p-3" id="t-cont">
    <div class="row justify-content-center">
      <div class="col">
        <table class="table">
          <thead class="text-dark " style="text-align: center; background-color:
#dfe4eb">
            <tr style="font-size: small">
              <th scope="col">No</th>
              <th scope="col" style="width: 300px;">Judul Tugas Akhir</th>
              <th scope="col" style="width: 400px;">Abstract</th>
              <th scope="col">Researc Area</th>

```

```

        <th scope="col">Nim</th>
        <th scope="col">Nama</th>
        <th scope="col">Kelas</th>
        <th scope="col">Angkatan</th>
        <th scope="col">Similarity</th>

    </tr>
</thead>
<tbody class="text-light">

    <?php $i = 1; ?>
    <?php while ($row_query = mysqli_fetch_array($query)) :
// if ($row_query['cosineSimilarity' == 0]) {
//     continue;
// }

?>

<tr style="font-size: small">
    <td scope="row"><?=$i ?></td>
    <td style="text-align: justify;">
        <?=$row_query["title"]; ?>
    </td>
    <td style="text-align: justify;">
        <div class="form-floating">
            <textarea class="form-control" placeholder="Leave a
comment here"
                id="floatingTextarea2"
                style="height: 200px"><?=$row_query["abstract"]; ?></textarea>
        </div>

    </td>
    <td><?=$row_query["research_area"]; ?></td>
    <td><?=$row_query["nim"]; ?></td>
    <td><?=$row_query["stud_name"]; ?></td>
    <td><?=$row_query["class"]; ?></td>
    <td><?=$row_query["year"]; ?></td>
    <td><?=$row_query["cosineSimilarity"]; ?></td>
</tr>
<?php $i++; ?>

```

```

        <?php endwhile; ?>
    </tbody>

</table>
</div>

</div>
</div>
</section>
<!-- End of Table -->

<!-- Optional JavaScript; choose one of the two! -->

<!-- Option 1: Bootstrap Bundle with Popper -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIax
VXM" crossorigin="anonymous">
</script>

<!-- Option 2: Separate Popper and Bootstrap JS -->
<!--
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"
integrity="sha384-
IQsoLXl5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R
8p" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.min.js"
integrity="sha384-
cVKIPhGWic2A14u+LWgxfKTRlcfu0JTxR+EQDz/bglDoEyl4H0zUF0QKbrJ0EcQF
" crossorigin="anonymous"></script>
-->
</body>

</html>

```

