

**SISTEM PENDETEKSI KEMIRIPAN JAWABAN ESSAY
MENGUNAKAN ALGORITMA
RABIN KARP**

TUGAS AKHIR



OLEH:

OMMA SARIANI SIREGAR

NIM. 171221134

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS MUHAMMADIYAH PONTIANAK
2021**

PERNYATAAN KEASLIAN

SISTEM PENDETEKSI KEMIRIPAN JAWABAN ESSAY MENGUNAKAN ALGORITMA RABIN KARP

TUGAS AKHIR

Saya mengakui skripsi ini hasil kerja dari saya sendiri kecuali kutipan dan ringkasan yang sudah dicantumkan masing-masing sumbernya.

Pontianak, 1 September 2021

Omma Sariani Siregar

NIM. 171221134

LEMBAR PERSETUJUAN

Yang bertanda tangan dibawah ini Dosen Pembimbing Tugas Akhir, menerangkan bahwa:

Nama	: Omma Sariani Siregar
NIM	: 171221134
Judul	: Sistem Pendeteksi Kemiripan Jawaban Essay Menggunakan Algoritma Rabin Karp

Diperiksa dan Disetujui

Dosen Pembimbing I



Alda Cendekia Siregar,
S.Kom, MCs
NIDN. 1113098502

Dosen Pembimbing II



Syarifah Putri Agustini
Alkadri, S.T, M.Kom
NIDN. 1111088803

Mengetujui
Dekan Fakultas Teknik dan Ilmu Komputer



Fuazen, S.T., M.T
NIDN. 1122087301

PENGESAHAN SIDANG TUGAS AKHIR

Tugas akhir ini telah disidangkan dan dipertahankan di depan tim penguji pada hari Rabu, 1 September 2021 dan dapat diterima sebagai salah satu syarat akhir studi pada Program Studi Teknik Informatika Fakultas Teknik Dan Ilmu Komputer Universitas Muhammadiyah Pontianak.

TIM PEMBIMBING

Dosen Pembimbing I



Alda Cendekia Siregar,
S.Kom, MCs
NIDN. 1113098502

Dosen Pembimbing II



Syarifah Putri Agustini
Alkadri, S.T, M.Kom
NIDN. 1111088803

TIM PENGUJI

Dosen Penguji I



Asrul Abdullah, S.Kom.,
M.Cs
NIDN. 1128059002

Dosen Penguji II



Sucipto, S.Kom/M.Kom
NIDN. 1130038301

Mengetahui
Dekan Fakultas Teknik dan Ilmu Komputer



Fuazen, S.T., M.T
NIDN. 1122087301

ABSTRAK

Konsep kemiripan (*similarity*) sudah menjadi isu yang sangat penting hampir di setiap bidang ilmu pengetahuan. Pendeteksian menggunakan konsep *similarity* atau kemiripan yang merupakan salah satu cara untuk mendeteksi kemiripan jawaban *essay*. Banyaknya jawaban penalaran mahasiswa yang sama menyebabkan kurangnya kreatifitas pemikiran dari mahasiswa itu sendiri dan dosen memerlukan waktu pengoreksian yang cukup lama dan kurang efisien karena memeriksa jawaban mahasiswa dengan satu persatu.

Pengimplementasian algoritma *rabin karp* pada sistem pendeteksi kemiripan jawaban *essay* berhasil mendeteksi kemiripan jawaban mahasiswa dan membantu Dosen dalam memeriksa jawaban ujian *essay* mahasiswa satu dengan mahasiswa yang lainnya, dan ini sangat penting dilakukan untuk menemukan tingkat kemiripan dari jawaban masing-masing mahasiswa, sehingga meminimalisir untuk tidak saling mencontoh satu sama lain.

Berdasarkan tabel pengujian k-gram nilai persentase antara k-gram 4 dengan k-gram 5 mempunyai perbedaan yaitu pada k-gram 4 nilai kemiripan antara dokumen1 & dokumen2 senilai 0,4594, sedangkan pada k-gram 5 nilai dokumen1 & dokumen2 senilai 0,3698 dengan demikian selisih diantara keduanya adalah 8,96% . Dapat disimpulkan bahwa semakin besar nilai k-gram maka nilai persentase semakin kecil dan semakin kecil nilai k-gram maka nilai persentase kemiripan semakin besar. Akan tetapi semakin kecil nilai *k-gram*, semakin akurat pula nilai persentase kemiripan dokumen tersebut.

Kata Kunci: *Similarity, Rabin Karp, Essay, K-gram*

ABSTRACT

The concept of similarity has become a very important issue in almost every field of science. Detection uses the concept of similarity which is one way to detect the similarity of essay answers. The number of answers to the same student reasoning causes a lack of creative thinking on the part of the students themselves and the lecturers require a long and inefficient correction time because they check student answers one by one.

The implementation of the rabin karp algorithm in the essay answer similarity detection system has succeeded in detecting the similarity of student answers and assisting lecturers in checking students' essay exam answers with other students, and this is very important to do to find the level of similarity of each student's answers, so therefore minimizing imitate each other.

Based on the k-gram test table, the percentage value between k-gram 4 and k-gram 5 has a difference, namely in k-gram 4 the similarity value between document1 & document2 is 0.4594, while in k-gram 5 the value of document1 & document2 is 0.3698, thus the difference between the two is 8.96%. It can be concluded that the greater k-gram value, the smaller of percentage value and the smaller k-gram value, the greater of percentage similarity value. However, the smaller of k-gram value, the more accurate of percentage value the document similarity.

Keyword: Similarity, Rabin Karp, Essay, K-gram

KATA PENGANTAR

Alhamdulillah segala puji dan syukur kepada Allah SWT, penulis dapat menyelesaikan Tugas Akhir ini yang berjudul “SISTEM PENDETEKSI KEMIRIPAN JAWABAN ESSAY MENGGUNAKAN ALGORITMA RABIN KARP” atas motivasi yang telah diberikan kepada penulis, oleh karena itu penulis banyak mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Kedua Orang Tua, Ayahku Muhammad Nasir Siregar dan Ibuku Siti Derhinun Harahap yang menjadi dasar motivasi dalam menyelesaikan Tugas Akhir ini. Banyak sekali dukungan yang telah diberikan kepada penulis baik secara moril maupun materi, serta doa-doa yang selalu dipanjatkan kepada Allah SWT untuk penulis.
2. Tulang Mara Latong Harahap dan Nantulang Dra.Sa’adiah Bainuri, banyak sekali dukungan yang telah diberikan kepada penulis baik secara moril maupun materi, serta doa-doa yang selalu dipanjatkan kepada Allah SWT untuk penulis.
3. Ibu Alda Cendekia Siregar, S.Kom.,M.Cs., sebagai pembimbing utama yang telah memberikan bimbingan, arahan, ide dan masukan yang sangat berguna dalam menyelesaikan Tugas Akhir ini.
4. Ibu Syarifah Putri Agustini Alkadri, S.T, M.Kom., selaku pembimbing kedua yang telah memberikan bimbingan, arahan, dan masukan kepada penulis dalam menyelesaikan Tugas Akhir ini.
5. Ibu Menur Wahyu Pangestika, S.Kom., M.Kom., selaku dosen pembimbing akademik yang selalu mengawasi, menasehati dan mengarahkan selama perkuliahan.
6. Bapak Yulrio Brianorman, S.Si., M.T., selaku dosen pembimbing akademik yang selalu mengawasi, menasehati dan mengarahkan selama perkuliahan.
7. Seluruh Dosen dan tenaga Dosen yang pernah mengajar di Program Studi Teknik Informatika Fakultas Teknik Dan Ilmu Komputer yang sudah memberikan ilmu dari awal perkuliahan hingga sekarang.
8. Seluruh pengurus Fakultas Teknik Dan Ilmu Komputer yang sudah memberikan pelayanan kepada penulis dalam menyelesaikan Tugas Akhir ini.

9. Kepada sahabat khususnya Teknik Informatika angkatan 2016, banyak suka duka yang telah dilalui semasa perkuliahan hingga sampai penulis menyelesaikan tugas akhir ini.
10. Teman-teman seperjuangan Ciwi-Ciwi kelas 02 Fauseh, Ida Ayu R.N, Rizka Amalia, Dwi Putri RG, Nayunda Permatasari, Tika Indah P, Nur Sri Utami, Marlia Wulandari, V. Vibiola di Fakultas Teknik Dan Ilmu Komputer Universitas Muhammadiyah Pontianak.
11. Oppung menekku tercinta, banyak sekali dukungan yang telah diberikan kepada penulis baik secara moril serta doa-doa yang selalu dipanjatkan kepada Allah SWT untuk penulis.
12. Abang Mara Hombang Siregar, yang telah banyak membantu penulis dengan dukungan secara moril maupun materi secara langsung.
13. Tulang menek Muhammad Juaro Harahap, yang telah banyak membantu penulis dengan dukungan secara moril maupun materi secara langsung.
14. Ante Nur Sonja harahap S.Pd, terima kasih atas motivasi dan sarannya kepada penulis hingga bisa menyelesaikan penelitian Tugas akhir ini.
15. Dan semua pihak yang turut membantu dan mendukung penulis dalam menyelesaikan tugas akhir ini yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa penyusunan tugas akhir ini jauh dari sempurna, semoga Allah SWT memberikan balasan yang berlipat ganda kepada semua pihak yang telah turut membantu penulis dalam menyelesaikan penulisan tugas akhir ini. Oleh karena itu, penulis berharap atas saran dan kritik yang bersifat membangun dari pembaca.

Semoga tugas akhir ini bermanfaat bagi pengembangan teknologi informasi dimasa depan.

Pontianak, 1 September 2021

Omma Sariyani Siregar
NIM. 171221134

DAFTAR ISI

PERNYATAAN KEASLIAN.....	i
LEMBAR PERSETUJUAN.....	ii
PENGESAHAN SIDANG TUGAS AKHIR.....	Error! Bookmark not defined.
ABSTRAK.....	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Metodologi	4
1.7 Sistematika Penulisan.....	5
BAB II LANDASAN TEORI.....	6
2.1 Tinjauan Pustaka	6
2.2 Algoritma Rabin Karp	7
2.2.1 K-gram	9
2.2.2 Preprocessing	9
2.3 Perancangan Sistem.....	10
2.3.1 Use Case Diagram.....	10
2.3.2 Activity Diagram.....	11
2.3.3 Sequence Diagram	11
2.3.4 Class Diagram	12
2.4 Perancangan User Interface.....	13
2.5 Metode Waterfall.....	14
2.6 Website.....	15

2.6.1	Pengertian Website.....	15
2.6.2	Jenis-Jenis <i>Website</i>	16
2.7	MySQL.....	16
2.7.1	Pengenalan Database.....	16
2.7.2	Tipe data pada MySQL	17
2.8	PHP.....	19
2.9	Metode Pengujian Black Box.....	19
2.10	Metode Pengujian White Box	20
BAB III METODE PENELITIAN.....		21
3.1	Identifikasi Masalah	21
3.2	Pengumpulan Data	21
3.3	Analisis Kebutuhan	22
3.4	Analisis Dan Perancangan Sistem.....	22
3.5	Implementasi	23
3.6	Pengolahan Data.....	23
3.7	Pengujian	24
BAB IV ANALISIS DAN PERANCANGAN SISTEM.....		26
4.1	Analisis Sistem.....	26
4.1.1	Kebutuhan Fungsional	26
4.1.2	Kebutuhan Non Fungsional.....	27
4.2	Analisis Perancangan	28
4.2.1	Use Case Diagram.....	28
4.2.2	Activity Diagram.....	29
4.2.3	Sequance Diagram	36
4.2.4	Class Diagram	41
4.3	Perancangan Sistem.....	42
BAB V HASIL DAN PEMBAHASAN.....		46
5.1	Implementasi Sistem	46
5.1.1	Tampilan Halaman Login	46
5.1.2	Tampilan Halaman Data Mahasiswa	46
5.1.3	Tampilan Halaman Data Stopword.....	47

5.1.4	Tampilan Halaman <i>Preprocessing</i>	48
5.1.5	Tampilan Halaman Hasil <i>Preprocessing</i>	48
5.1.6	Tampilan Halaman Jawab Soal Mahasiswa.....	49
5.1.7	Tampilan Halaman Input Data Soal.....	49
5.1.8	Tampilan Halaman Data Jawaban Mahasiswa.....	50
5.1.9	Tampilan Halaman Cek Kemiripan/ <i>Similarity</i> Jawaban.....	50
5.1.10	Hasil <i>Similarity</i> Rabin Karp.....	51
5.1.11	Tampilan Rekap Hasil <i>Similarity</i>	51
5.1.12	Hasil Rekap <i>Similarity</i>	52
5.2	Hasil Perhitungan Manual	52
5.3	Pengujian Sistem	58
5.3.1	Pengujian Black Box.....	58
5.3.2	Pengujian White Box	60
5.3.3	Pengujian K-gram	64
BAB VI KESIMPULAN DAN SARAN		65
6.1	Kesimpulan.....	65
6.2	Saran.....	65
DAFTAR PUSTAKA		66
LAMPIRAN I. CONTOH KODE PROGRAM.....		
LAMPIRAN II. BIOGRAFI PENULIS.....		

DAFTAR GAMBAR

Gambar 1.1 Diagram Alur Penelitian.....	4
Gambar 2.1 Metode <i>Waterfall</i>	14
Gambar 3.1 Diagram alur algoritma <i>rabin karp</i>	23
Gambar 4.1 <i>Use case diagram</i>	28
Gambar 4.2 <i>Activity diagram login</i>	29
Gambar 4.3 <i>Activity diagram</i> mengelola data mahasiswa	30
Gambar 4.4 <i>Activity diagram</i> mengelola data stopword.....	31
Gambar 4.5 <i>Activity diagram</i> mengelola <i>preprocessing</i>	32
Gambar 4.6 <i>Activity diagram</i> mahasiswa menjawab soal.....	32
Gambar 4.7 <i>Activity diagram</i> mengelola soal	33
Gambar 4.8 <i>Activity diagram</i> mengelola jawaban	34
Gambar 4.9 <i>Activity diagram</i> mengelola hitung <i>rabin karp</i>	34
Gambar 4.10 <i>Activity diagram</i> rekap <i>similarity/kemiripan</i>	35
Gambar 4.11 <i>Sequance diagram</i> mengelola data mahasiswa	36
Gambar 4.12 <i>Sequance diagram</i> mengelola data <i>stopword</i>	37
Gambar 4.13 <i>Sequance diagram</i> mengelola <i>preprocessing</i>	38
Gambar 4.14 <i>Sequance diagram</i> mahasiswa menjawab soal.....	39
Gambar 4.15 <i>Sequance diagram</i> mengelola data soal	39
Gambar 4.16 <i>Sequance diagram</i> Mengelola <i>hitung rabin karp</i>	40
Gambar 4.17 <i>Sequance diagram</i> mengelola rekap kemiripan	41
Gambar 4.18 <i>Class Diagram</i>	42
Gambar 4.19 Perancangan halaman <i>login</i>	42
Gambar 4.20 Perancangan halaman data mahasiswa.....	43
Gambar 4.21 Perancangan halaman input jawaban	43
Gambar 4.22 Perancangan halaman input data soal.....	44
Gambar 4.23 Perancangan halaman data jawaban mahasiswa	44
Gambar 4.24 Perancangan halaman data <i>stopword</i>	45
Gambar 4.25 Perancangan halaman <i>preprocessing</i>	45
Gambar 4.26 Perancangan halaman <i>similarity rabin karp</i>	46
Gambar 4.27 Perancangan halaman rekap <i>similarity</i>	46

Gambar 5.1 Tampilan halaman <i>login</i>	46
Gambar 5.2 Tampilan halaman data mahasiswa.....	47
Gambar 5.3 Tampilan halaman Data <i>Stopword</i>	47
Gambar 5.4 Tampilan halaman <i>preprocessing</i>	48
Gambar 5.5 Tampilan halaman hasil <i>preprocessing</i>	48
Gambar 5.6 Tampilan halaman input data jawaban.....	49
Gambar 5.7 Tampilan halaman input data soal.....	49
Gambar 5.8 Tampilan halaman data jawaban	50
Gambar 5.9 Tampilan halaman <i>similarity rabin karp</i>	50
Gambar 5.10 Tampilan halaman <i>similarity rabin karp</i>	51
Gambar 5.11 Tampilan halaman rekap <i>similarity</i>	51
Gambar 5.12 Tampilan hasil rekap <i>similarity</i>	52
Gambar 5.13 <i>Flowchart</i> perhitungan nilai akhir.....	60
Gambar 5.14 <i>Flowgraph</i> perhitungan nilai akhir.....	61

DAFTAR TABEL

Tabel 2.1	Komponen <i>use case diagram</i>	10
Tabel 2.2	Komponen <i>activity diagram</i>	11
Tabel 2.3	Komponen <i>sequence diagram</i>	12
Tabel 2.4	Komponen <i>Class Diagram</i>	13
Tabel 2.5	Kelompok tipe data numerik.....	18
Tabel 2.6	Kelompok tipe data waktu dan tanggal.....	18
Tabel 2.7	Kelompok tipe data string.....	18
Tabel 2.8	Hubungan <i>cyclomatic complexity</i> dengan resiko	20
Tabel 5.1	Data dan hasil <i>preprocessing</i>	52
Tabel 5.2	Hasil k-gram 4.....	53
Tabel 5.3	Hasil k-gram 5.....	54
Tabel 5.4	Nilai <i>hash</i> k-gram 4.....	55
Tabel 5.5	Nilai <i>hash</i> k-gram 5.....	56
Tabel 5.6	Rekap nilai <i>similarity</i> untuk fauseh	57
Tabel 5.7	Rekap nilai <i>similarity</i> untuk ida ayu	57
Tabel 5.8	Rekap nilai <i>similarity</i> untuk nur sri	58
Tabel 5.9	Pengujian <i>black box</i> sistem pendeteksi kemiripan jawaban <i>essay</i>	58
Tabel 5.10	<i>Source code</i> program fungsi nilai akhir kemiripan jawaban.....	61
Tabel 5.11	Jalur Independen	62
Tabel 5.12	<i>Tes case flowgraph</i>	63
Tabel 5.13	Pengujian k-gram	64

BAB I

PENDAHULUAN

1.1 Latar Belakang

Konsep kemiripan (*similarity*) sudah menjadi isu yang sangat penting hampir di setiap bidang ilmu pengetahuan[1]. Kemiripan yang dimaksud disini adalah mengambil atau menjiplak hasil pemikiran orang lain, tanpa mengubah sedikitpun dengan penalaran sendiri. Akan tetapi hal ini belum termasuk kategori plagiarisme sebagaimana yang dimaksud oleh Sulianta[2]. Pendeteksian menggunakan konsep *similarity* atau kemiripan yang merupakan salah satu cara untuk mendeteksi kemiripan jawaban *essay*. Ujian *essay* merupakan salah satu proses evaluasi yang digunakan untuk mengetahui kemampuan mahasiswa dalam menjawab soal[3].

Dengan dilaksanakannya ujian berbasis *essay*, mahasiswa dituntut untuk menjawab soal dengan pemahaman yang mereka miliki akan tetapi banyaknya tugas yang menumpuk mengakibatkan mahasiswa melakukan tindakan saling mencontoh satu sama lain, hal ini menyebabkan kurangnya kreatifitas pemikiran dari mahasiswa itu sendiri. Soal *essay* akan menghasilkan berbagai kemungkinan jawaban sesuai dengan pemahaman masing-masing mahasiswa. Ada beberapa masalah saat dosen memeriksa jawaban mahasiswa yaitu menyebabkan banyaknya jawaban penalaran mahasiswa yang sama, dan dosen memerlukan waktu pengoreksian yang cukup lama dan kurang efisien karena memeriksa jawaban mahasiswa dengan satu persatu. Oleh karena itu dibutuhkan lah sistem pendeteksi kemiripan jawaban *essay*. Untuk membantu Dosen dalam memeriksa jawaban ujian *essay* mahasiswa satu dengan mahasiswa yang lainnya, dan ini sangat penting dilakukan untuk menemukan tingkat kemiripan dari jawaban masing-masing mahasiswa, sehingga meminimalisir untuk tidak saling mencontoh satu sama lain.

Adapun algoritma yang terdapat di beberapa penelitian terdahulu yang memiliki kesamaan permasalahan dengan topik penelitian ini. Penelitian pertama yang berhasil ditemukan adalah Maskur dkk, dari Universitas Muhammadiyah Malang, pada tahun 2020 yang berjudul “Deteksi Kemiripan Dokumen Pengajuan

Proposal Menggunakan Algoritma Biword Wining Pada Sistem Informasi Penelitian Dan Pengabdian” hasil penelitian ini adalah mengetahui apakah proposal yang diajukan itu sudah atau pernah diajukan. Kelemahan dari metode ini adalah tidak melakukan proses pembuangan kata awal dan akhiran sehingga terjadinya pemborosan kata[4].

Penelitian ini juga dilakukan oleh Latus Hermawan , Maria Bellanier Ismiati, dari Universitas Katolik Musi Charitas, pada tahun 2020 yang berjudul “Aplikasi Pengecekan Dokumen Digital Tugas Mahasiswa Berbasis Website” dalam penelitian ini menerapkan algoritma pencocokan string yang dapat dihasilkan suatu keluaran yang akan memberikan informasi seberapa dekat antar dokumen.[5]. Kelemahan dari metode ini adalah semua teks yang berisi *term* tertentu diperlakukan sama, sehingga IDF tidak memperhitungkan jumlah kemunculan suatu *term* pada suatu dokumen[6].

Penelitian ini juga dilakukan oleh Dharmawan dkk, dari Universitas Halu Oleo Kendari, pada tahun 2020 yang berjudul “Implementasi Latent Semantic Analysis Method Untuk Mendeteksi Kemiripan Kalimat Esai Pada Ujian Online Teknik Informatika Universitas Halu Oleo” Penelitian ini menunjukkan lama kerja sistem yang diuji dengan jumlah peserta ujian dan jumlah kata yang berbeda [7].

Berdasarkan beberapa penelitian sebelumnya penulis menerapkan algoritma *Rabin Karp*. Algoritma *rabin karp* dibuat oleh *Michael O. Rabin dan Richard M. Karp*. Algoritma *Rabin Karp* adalah metode *hash* dalam mencari suatu kata[8]. Algoritma *Rabin Karp* memiliki beberapa kelebihan yaitu, memiliki proses perhitungan yang relatif mudah dan dapat digunakan dalam kasus pencarian string dengan pola yang panjang. Sedangkan kekurangan Algoritma *Rabin Karp* adalah memiliki tahapan *processing* yang panjang sehingga membutuhkan waktu yang agak lama dan keakuratan pendeteksian algoritma ini sangat tergantung dari posisi kalimat[9].

Dari latar belakang tersebut, penulis tertarik untuk membangun sebuah sistem yaitu “Sistem Pendeteksi Kemiripan Jawaban Essay Menggunakan Algoritma Rabin Karp” yang dapat melakukan proses pendeteksian kemiripan lebih dari dua dokumen.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka rumusan masalah yang didapatkan adalah.

1. Bagaimana algoritma *rabin karp* mendeteksi kemiripan jawaban ujian *essay* untuk membantu dalam memeriksa kemiripan jawaban?
2. Bagaimana tingkat keakurasian algoritma *rabin karp* dalam mendeteksi kemiripan jawaban *essay* ?

1.3 Batasan Masalah

Terdapat beberapa batasan masalah pada penelitian ini , yaitu:

1. Sistem yang di rancang berbasis *website*.
2. Sistem pendeteksi kemiripan jawaban ujian berbasis *essay* diperuntukkan untuk *staff* sebagai admin, dosen dan mahasiswa sebagai user.
3. Sistem menunjukkan jumlah persentase similaritasnya.
4. Deteksi kemiripan tidak memperhatikan parafrase atau kutipan terkait sumber.
5. Data yang dibandingkan adalah data dokumen satu dengan dokumen lainnya.
6. Data yang dinilai adalah format teks yang di input langsung dari *keyboard*.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Membuat sebuah sistem pendeteksi kemiripan jawaban ujian *essay* dengan algoritma *rabin karp*.
2. Dilakukan pengujian keakurasian algoritma *rabin karp* dengan menggunakan *k-gram*.

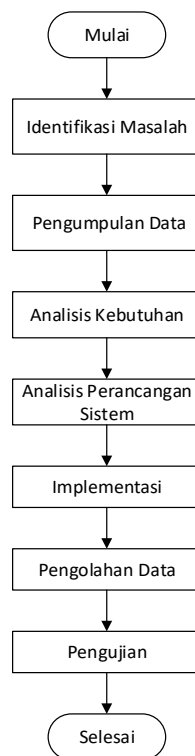
1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Membantu Dosen dalam memeriksa jawaban ujian *essay* untuk menemukan tingkat kemiripan dari jawaban masing-masing mahasiswa.
2. Mencegah Mahasiswa dari terjadinya saling mencontoh satu sama lain.

1.6 Metodologi

Rancangan metodologi penelitian ini disajikan berdasarkan dalam tahap-tahap dibawah ini agar lebih mudah untuk dimengerti dan dipahami. Metodologi ini disusun secara terstruktur untuk menyusun proses penelitian.



Gambar 1.1 Diagram Alur Penelitian

Pada Gambar 1.1 diatas, dapat dilihat proses penelitian yang dimulai dari mengidentifikasi masalah, dari hasil identifikasi masalah dapat disimpulkan topik penelitian, rumusan masalah, batasan masalah, tujuan penelitian dan manfaat penelitian yang akan diambil. Kemudian lanjut proses pengumpulan data yaitu observasi, wawancara, dan studi pustaka, kemudian analisis kebutuhan untuk mengetahui apa saja yang akan dibutuhkan dalam melaksanakan penelitian yang dilakukan. Setelah analisis kebutuhan selesai maka lanjut ke analisis dan perancangan sistem dengan menerapkan algoritma *rabin karp*, kemudian implementasi pengkodean dan penerapan antarmuka aplikasi sesuai dengan analisa dan perancangan, lanjut ke tahap pengolahan data diambil berdasarkan data yang dikumpulkan, terakhir pengujian dilakukan untuk menguji sistem, apakah sistem sudah layak atau belum serta menyesuaikan sistem telah sesuai rancangan.

1.7 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari enam bab, dengan sistematika penulisan sebagai berikut:

BAB I Pendahuluan

Bab ini mencakup latar belakang penelitian yang berjudul "Sistem Pendeteksi Kemiripan Jawaban Essay Menggunakan Algoritma Rabin Karp" rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II Landasan Teori

Bab ini membahas konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang di lakukan serta tinjauan pustaka dari penelitian-penelitian sebelumnya.

BAB III Metode Penelitian

Bab ini berisi tentang metode penelitian yang digunakan dalam penelitian ini. Bagian ini menyajikan secara lengkap seluruh langkah-langkah prosedur penelitian yang dilakukan di bab ini.

BAB IV Analisis dan Perancangan Sistem

Bab ini berisi tentang pemaparan analisis sistem yang terdiri dari analisis hasil, dari analisis tersebut kemudian digunakan untuk melakukan perancangan perangkat lunak yang terdiri dari, *use case*, *activity diagram*, *sequence diagram*, *class diagram*, dan *User Interface*.

BAB V Hasil dan Pengujian

Bab ini menjelaskan implementasi dari bab III, tampilan antarmuka, pengujian sistem, dan pengujian terhadap metode *similarity*.

BAB VI Kesimpulan Dan Saran

Bab penutup berisi hal hal penting yang telah dibahas pada bab sebelumnya, kemudian dibuatlah kesimpulan dan saran yang diberikan untuk pengembangan pada penelitian selanjutnya.

BAB V

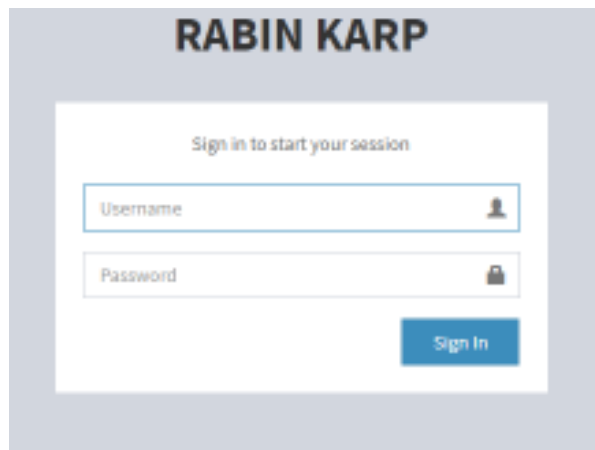
HASIL DAN PENGUJIAN

5.1 Implementasi Sistem

Setelah sistem dirancang, maka selanjutnya akan dilakukan tahap implementasi. Tahapan memenuhi kebutuhan pengguna dalam berinteraksi dengan sistem yang dibangun. Penelitian ini melakukan analisis dan pengujian terhadap Sistem Pendeteksi Kemiripan Jawaban *Essay* Menggunakan Algoritma *Rabin Karp*. Berikut ini adalah implemetasi hasil perancangan tampilan dari Sistem Pendeteksi Kemiripan Jawaban *essay*.

5.1.1 Tampilan Halaman Login

Halaman *login* adalah bentuk autentikasi *user* untuk masuk kedalam sistem. Terdapat 2 kolom inputan yaitu *username* dan *password* dan tombol *sign in* untuk masuk, seperti terlihat pada gambar 5.1.



Gambar 5.1 Tampilan halaman *login*

5.1.2 Tampilan Halaman Data Mahasiswa

Halaman ini berfungsi untuk menyimpan data mahasiswa yang akan melaksanakan ujian. Pada halaman ini terdapat tabel untuk menampilkan data data mahasiswa, pada kolom aksi *icon* berwarna kuning berfungsi untuk mengubah data mahasiswa dan *icon* merah untuk menghapus data mahasiswa. Terdapat juga tombol tambah untuk menambah mahasiswa dan tombol perbarui untuk menyegarkan tabel, kolom *search* untuk pencarian mahasiswa, seperti gambar 5.2.

No	NIM	Nama Mahasiswa	Username	Password	Aksi
1	2002001	Fauzan	fff	fff	[Yellow] [Red]
2	2002002	Rida Ayu Alhasan Alhasani	ayu	fff	[Yellow] [Red]
3	2002003	Fitri Siti Utami	fit	fff	[Yellow] [Red]
4	2002004	Dzohna Nurafiqi Wajidi	afiqah	fff	[Yellow] [Red]

Gambar 5.2 Tampilan halaman data mahasiswa

5.1.3 Tampilan Halaman Data Stopword

Pada halaman ini terdapat tabel untuk menampilkan data kata *stopword*, pada kolom aksi *icon* berwarna kuning berfungsi untuk mengubah data kata *stopword* dan *icon* merah untuk menghapus data kata *stopword*. Terdapat juga tombol tambah untuk menambah kata *stopword* dan tombol perbarui untuk menyegarkan tabel, kolom *search* untuk pencarian kata *stopword*. Halaman ini berfungsi menyimpan daftar kata *stopword* atau kata umum untuk dihilangkan jika terdapat pada jawaban mahasiswa karena kata tersebut tidak mengandung arti atau makna apapun, seperti terlihat pada gambar 5.3.

No	Nama Stopword	Aksi
1	ada	[Yellow] [Red]
2	adapun	[Yellow] [Red]
3	adanya	[Yellow] [Red]
4	adanya	[Yellow] [Red]
5	ada	[Yellow] [Red]
6	adanya	[Yellow] [Red]
7	ada	[Yellow] [Red]
8	ada	[Yellow] [Red]
9	ada	[Yellow] [Red]

Gambar 5.3 Tampilan halaman Data *Stopword*

5.1.4 Tampilan Halaman *Preprocessing*

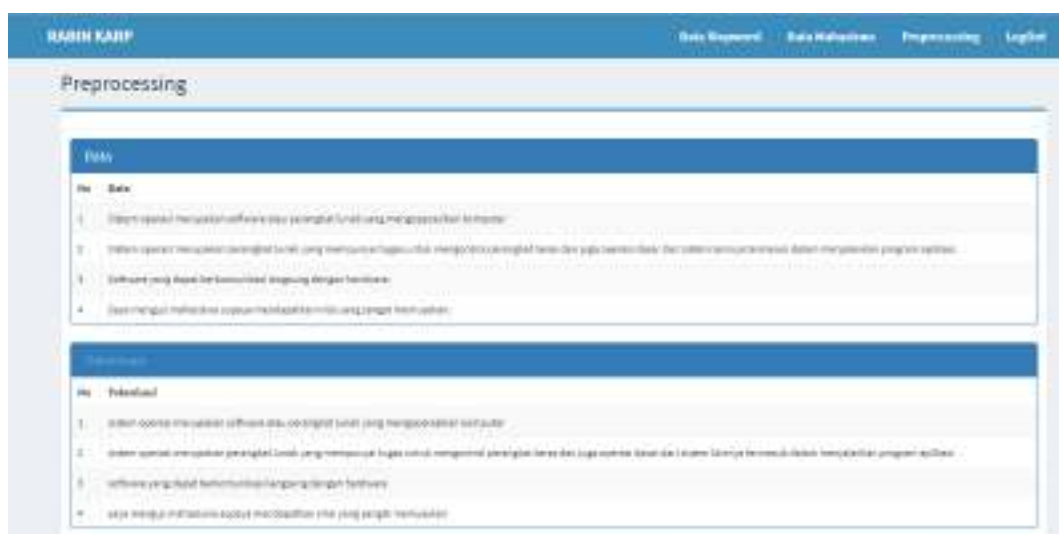
Halaman *preprocessing* ini akan menampilkan kalimat yaitu data *training* jawaban dari seluruh mahasiswa yang telah mengisi ujian esai seperti terlihat pada gambar 5.4.



Gambar 5.4 Tampilan halaman *preprocessing*

5.1.5 Tampilan Halaman Hasil *Preprocessing*

Tampilan halaman *preprocessing* ini terdapat tabel data untuk menampilkan data data jawaban mahasiswa, tabel tokenisasi untuk menampilkan jawaban yang sudah ditokenisasi, tabel *stopword* untuk jawaban mahasiswa yang sudah dibuang kata kata yang tidak mengandung makna, dan tabel *stemming* untuk menampilkan jawaban yang dikembalikan menjadi kata dasar dengan membuang kata imbuhan, seperti terlihat pada gambar 5.5.



Gambar 5.5 Tampilan halaman hasil *preprocessing*

5.1.6 Tampilan Halaman Jawab Soal Mahasiswa

Halaman ini berfungsi untuk mahasiswa mengisi jawaban sesuai dengan nomor soal, setelah terisi semua mahasiswa menekan tombol simpan dan jawaban akan tersimpan di *database* data jawaban yang nantinya akan di lakukan *text preprocessing*. Pada halaman ini terdapat *textbox* untuk mengisi jawaban, tombol simpan untuk menyimpan jawaban di *database* dan tombol kembali jika ingin membatalkan mengisi jawaban, seperti terlihat pada gambar 5.6.

Gambar 5.6 Tampilan halaman input data jawaban

5.1.7 Tampilan Halaman Input Data Soal

Halaman soal berfungsi untuk mengisi data soal ujian mahasiswa. Pada halaman ini terdapat tabel untuk menampilkan soal soal, pada kolom aksi *icon* berwarna kuning berfungsi untuk mengubah soal dan *icon* merah untuk menghapus soal. Terdapat tombol tambah untuk membuat soal baru, dan tombol perbarui untuk menyegarkan kembali tabel, kolom *search* untuk pencarian soal, seperti terlihat pada gambar 5.7.

Gambar 5.7 Tampilan halaman input data soal

5.1.8 Tampilan Halaman Data Jawaban Mahasiswa

Pada halaman ini terdapat tabel untuk menampilkan jawaban yang sudah diisi mahasiswa, pada kolom aksi *icon* berwarna kuning berfungsi untuk mengubah jawaban dan *icon* merah untuk menghapus jawaban. Terdapat juga tombol tambah jika ingin memasukan jawaban, kolom *search* untuk pencarian jawaban. Halaman jawaban merupakan isi jawaban ujian yang telah di inputkan oleh mahasiswa, seperti terlihat pada gambar 5.8.

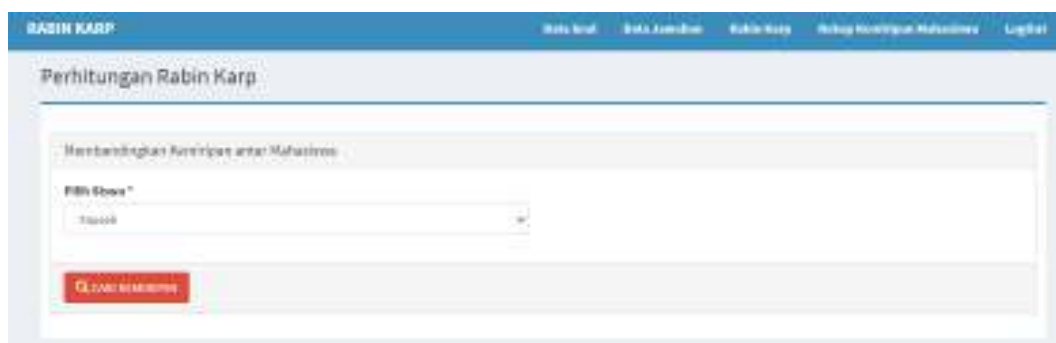


No	NIM	Jawab	Jawaban	Tipe Pengesahan
1	2000000	Meneret penduaan [aman.aps.tu.001]	Sistem operasi merupakan software atau perangkat lunak yang mengoperasikan komputer.	sisitem operasi tipe software angket bentuk operasi komputer
2	2000000	Meneret penduaan [aman.aps.tu.001]	Sistem operasi merupakan perangkat lunak yang mempunyai tugas untuk mengontrol perangkat keras dan juga mengatur data dan sistem lainnya ke dalam sistem manajemen program aplikasi.	sisitem operasi tipe angket bentuk penyaji tugas kontrol angket bentuk operasi data dan sistem manaj program aplikasi
3	2000000	Meneret penduaan [aman.aps.tu.001]	Software yang dibuat berdasarkan kegunaan angket hardware.	software dapat kontrol data kegunaan hardware

Gambar 5.8 Tampilan halaman data jawaban

5.1.9 Tampilan Halaman Cek Kemiripan/Similarity Jawaban

Pada halaman ini terdapat halaman untuk membandingkan similarity/kemiripan dari jawaban mahasiswa, dengan memilih mahasiswa pada *icon* pilih mahasiswa kemudian tekan tombol cari kemiripan untuk memproses hitung rabin karp, seperti terlihat pada gambar 5.9.



Gambar 5.9 Tampilan halaman *similarity rabin karp*

5.1.10 Hasil Similarity Rabin Karp

Pada halaman ini terdapat proses perhitungan *rabin karp*, dengan menampilkan nama mahasiswa, k-gram, untuk proses pemotongan jawaban yang dilakukan dengan cara membagi rata teks jawaban ke dalam k-gram sesuai dengan ketentuan awal yaitu k-gram 4. Lalu *hash* diubah ke jenis data yang ada menjadi bilangan bulat yang sederhana, *fingerpint* yaitu nilai hasil *hash* yang sama dari dua dokumen. Kemudian menampilkan kemiripan dari dua dokumen antar mahasiswa, seperti terlihat pada gambar 5.10.

MATA KULIAH	Nama	Nilai Keakuratan
RPL 2018	[Student Name]	[Similarity Value]
RPL 2019	[Student Name]	[Similarity Value]
RPL 2020	[Student Name]	[Similarity Value]
RPL 2021	[Student Name]	[Similarity Value]
RPL 2018	[Student Name]	[Similarity Value]
RPL 2019	[Student Name]	[Similarity Value]
RPL 2020	[Student Name]	[Similarity Value]
RPL 2021	[Student Name]	[Similarity Value]

Gambar 5.10 Tampilan halaman *similarity rabin karp*

5.1.11 Tampilan Rekap Hasil Similarity

Pada halaman ini terdapat hasil rekap kemiripan , dengan cara memilih nama mahasiswa kemudian tekan tombol perbarui untuk memunculkan hasil rekap kemiripan sebagaimana terlihat pada gambar 5.11.

Gambar 5.11 Tampilan halaman rekap *similarity*

5.1.12 Hasil Rekap Similarity

Pada halaman ini terlihat hasil rekap kemiripan yang sudah di perbarui, lalu memunculkan hasil perbandingan *similarity*/kemiripan dari masing-masing jawaban mahasiswa, seperti terlihat pada gambar 5.12.

No	Nama Mahasiswa	Nama Soal Perbandingan	Kemiripan
1	Fitriah	Makna Esoknya Manusia	82.81%
2	Fitriah	Korosi	8.74%

Gambar 5.12 Tampilan hasil rekap *similarity*

5.2 Hasil Perhitungan Manual

5.2.1 Data Preprocessing

Data disini adalah jawaban *essay* mahasiswa kemudian dilakukan *preprocessing* yaitu tabel tokenisasi dengan mengubah huruf menjadi huruf kecil serta menghapus tanda baca seluruhnya, tabel *stopword* untuk jawaban mahasiswa yang sudah dibuang kata kata yang tidak mengandung makna, dan tabel *stemming* untuk menampilkan jawaban yang dikembalikan menjadi kata dasar dengan membuang kata imbuhan, sebagaimana terlihat pada tabel 5.1.

Tabel 5.1 Data dan hasil *preprocessing*

	Dokumen
Data	<ol style="list-style-type: none"> 1. Sistem operasi merupakan <i>software</i> atau perangkat lunak yang mengoperasikan komputer. 2. Sistem operasi merupakan perangkat lunak yang mempunyai tugas untuk mengontrol perangkat keras dan juga operasi dasar dari sistem lainnya termasuk dalam menjalankan program aplikasi. 3. <i>Software</i> yang dapat berkomunikasi langsung dengan <i>hardware</i>.
<i>Tokenisasi</i>	<ol style="list-style-type: none"> 1. sistem operasi merupakan software atau perangkat lunak yang mengoperasikan komputer 2. sistem operasi merupakan perangkat lunak yang mempunyai tugas untuk mengontrol perangkat keras dan juga operasi dasar dari sistem lainnya termasuk dalam menjalankan program aplikasi 3. software yang dapat berkomunikasi langsung dengan hardware
<i>Stopword</i>	<ol style="list-style-type: none"> 1. sistem operasi merupakan software perangkat lunak mengoperasikan komputer 2. sistem operasi merupakan perangkat lunak mempunyai tugas mengontrol perangkat keras operasi dasar dari sistem lainnya termasuk menjalankan program aplikasi 3. software dapat berkomunikasi langsung hardware

<i>Stemming</i>	<ol style="list-style-type: none"> 1. sistem operasi rupa software angkat lunak operasi komputer 2. sistem operasi rupa angkat lunak punya tugas kontrol angkat keras operasi dasar dari sistem lain masuk jalan program aplikasi 3. software dapat komunikasi langsung hardware
-----------------	---

5.2.2 Data K-gram

Data k-gram disini adalah data yang sudah diproses dari jawaban mahasiswa setelah melakukan preprocessing, nilai k-gram disini menggunakan k-gram 4 seperti terlihat pada tabel 5.2 dan nilai k-gram 5 dapat dilihat pada tabel 5.3.

Tabel 5.2 Hasil k-gram 4

	Dokumen1	Dokumen2
<i>K-Gram 4</i>	{sist} {iste} {stem} {temo} {emop} {mope} {oper} {pera} {eras} {rasi} {asir} {siru} {irup} {rupa} {upas} {paso} {asof} {soft} {oftw} {ftwa} {twar} {ware} {area} {rean} {eang} {angk} {ngka} {gkat} {katl} {atlu} {tlun} {luna} {unak} {nako} {akop} {kope} {oper} {pera} {eras} {rasi} {asik} {siko} {ikom} {komp} {ompu} {mput} {pute} {uter}	{sist} {iste} {stem} {temo} {emop} {mope} {oper} {pera} {eras} {rasi} {asir} {siru} {irup} {rupa} {upaa} {paan} {aang} {angk} {ngka} {gkat} {katl} {atlu} {tlun} {luna} {unak} {nakp} {akpu} {kpun} {puny} {unya} {nyat} {yatu} {atug} {tuga} {ugas} {gask} {asko} {skon} {kont} {ontr} {ntro} {trol} {rola} {olan} {lang} {angk} {ngka} {gkat} {katk} {atke} {tker} {kera} {eras} {raso} {asop} {sope} {oper} {pera} {eras} {rasi} {asid} {sida} {idas} {dasa} {asar} {sard} {arda} {rdar} {dari} {aris} {risi} {isis} {sist} {iste} {stem} {temm} {emma} {mmas} {masu} {asuk} {sukj} {ukja} {kjal} {jala} {alan} {lanp} {anpr} {npro} {prog} {rogr} {ogra} {gram} {rama} {amap} {mapl} {apli} {plik} {lika} {ikas} {kasi}
		Dokumen3 {soft} {oftw} {ftwa} {twar} {ware} {ared} {reda} {edap} {dapa} {apat} {patk} {atko} {tkom} {komu} {omun} {muni} {unik} {nika} {ikas} {kasi} {asil} {sila} {ilan} {lang} {angs} {ngsu} {gsun} {sung} {ungh} {ngha} {ghar} {hard} {ardw} {rdwa} {dwar} {ware}

Tabel 5.3 Hasil k-gram 5

	Dokumen1	Dokumen2
<i>K-Gram 5</i>	{siste} {istem} {stemo} {temop} {emope} {moper} {opera} {peras} {erasi} {rasir} {asiru} {sirup} {irupa} {rupas} {upaso} {pasof} {asoft} {softw} {oftwa} {ftwar} {tware} {warea} {arean} {reang} {eangk} {angka} {ngkat} {gkatl} {katlu} {atlun} {tluna} {lunak} {unako} {nakop} {akope} {koper} {opera} {peras} {erasi} {rasik} {asiko} {sikom} {ikomp} {kompu} {omput} {mpute} {puter}	{siste} {istem} {stemo} {temop} {emope} {moper} {opera} {peras} {erasi} {rasir} {asiru} {sirup} {irupa} {rupaa} {upaan} {paang} {aangk} {angka} {ngkat} {gkatl} {katlu} {atlun} {tluna} {lunak} {unakp} {nakpu} {akpun} {kpuny} {punya} {unyat} {nyatu} {yatug} {atuga} {tugas} {ugask} {gasko} {askon} {skont} {kontr} {ontro} {ntrol} {trola} {rolan} {olang} {langk} {angka} {ngkat} {gkatk} {katke} {atker} {tkera} {keras} {eraso} {rasop} {asope} {soper} {opera} {peras} {erasi} {rasid} {asida} {sidas} {idasa} {dasar} {asard} {sarda} {ardar} {rdari} {daris} {arisi} {rasis} {isist} {siste} {istem} {stemm} {temma} {emmas} {mmasu} {masuk} {asukj} {sukja} {ukjal} {kjala} {jalan} {alanp} {lanpr} {anpro} {nprog} {progr} {rogra} {ogram} {grama} {ramap} {amapl} {mapli} {aplik} {plika} {likas} {ikasi}
		Dokumen3 {softw} {oftwa} {ftwar} {tware} {ward} {areda} {redap} {edapa} {dapat} {apatk} {patko} {atkom} {tkomu} {komun} {omuni} {munik} {unika} {nikas} {ikasi} {kasil} {asila} {silan} {ilang} {langs} {angsu} {ngsun} {gsung} {sungh} {ungha} {nghar} {ghard} {hardw} {ardwa} {rdwar} {dware}

5.2.3 Nilai Hash

Nilai *hash* disini adalah nilai yang diproses dari data k-gram kemudian di ubah menjadi nilai bilangan desimal dengan menggunakan aturan *Horner's* pada persamaan (2.1), kemudian mencari nilai hash dengan menggunakan persamaan (2.2). Nilai *hash* yang sudah di proses dapat dilihat pada tabel 5.4 untuk k-gram 4 dan tabel 5.5 untuk k-gram 5.

Tabel 5.4 Nilai *hash* k-gram 4

	Dokumen1	Dokumen2
<i>Hash</i>	18999 9994 19952 19534 5355 13554 15557 15570 5718 17188 1897 18990 9915 19150 21518 15194 1945 19469 14712 7120 21217 22174 1740 17413 4136 1370 13700 7019 10201 2030 20313 13130 21310 13114 1155 11554 15557 15570 5718 17188 1890 18914 9152 11535 15370 13719 17194 21957	18999 9994 19952 19534 5355 13554 15557 15570 5718 17188 1897 18990 9915 19150 21500 15013 136 1370 13700 7019 10201 2030 20313 13130 21310 13115 1170 11713 17154 21540 15419 24210 2106 21060 20618 6190 1914 19153 11549 15507 15084 20851 18510 15113 11136 1370 13700 7019 10200 2004 20057 10570 5718 17194 1955 19554 15557 15570 5718 17188 1883 18830 8318 3180 1817 18173 1730 17317 3178 1798 17988 9898 18999 9994 19952 19532 5320 13218 12200 2010 20109 21090 10911 9110 1113 11145 1467 14684 16846 18477 14770 7712 17120 1215 12161 1618 16190 11900 9018 10188
Jumlah <i>Hash</i>	48	100
<i>Hash</i> yang sama	18999 9994 19952 19534 5355 13554 15557 15570 5718 17188 1897 18990 9915 19150 1370 13700 7019 10201 2030 20313 13130 21310 1370 13700 7019 5718 17194 15557 15570 5718 17188 18999 9994 19952	
Jumlah	34	
		Dokumen3
		19469 14712 7120 21217 22174 1743 17430 4315 3150 1519 15200 2014 20152 11540 15413 14138 21390 13900 9018 10188 1891 18910 9113 11136 1378 13800 8013 20136 21367 13670 6717 7173 1752 17520 5217 22174
Jumlah <i>Hash</i>		36
<i>Hash</i> yang sama	19469 14712 7120 21217 22174	
Jumlah	5	

Menghitung nilai *hash* misalnya {Sist} dengan menggunakan rumus persamaan (2.2) untuk lebih jelasnya dapat dilihat sebagai berikut:

$$\begin{aligned}
 \{sist\} &= (s \cdot 10^3) + (i \cdot 10^2) + (s \cdot 10^1) + (t \cdot 10^0) \\
 &= (18 \cdot 1000) + (8 \cdot 100) + (18 \cdot 10) + (19 \cdot 1) \\
 &= (18000) + (800) + (180) + (19) \\
 &= 18999
 \end{aligned}$$

Tabel 5.5 Nilai *hash* k-gram 5

	Dokumen1	Dokumen2
<i>Hash</i>	189994 99952 199534 195355 53554 135557 155570 155718 57188 171897 18990 189915 99150 191518 215194 151945 19469 194712 147120 71217 212174 221740 17413 174136 41370 13700 137019 70201 102030 20313 203130 131310 213114 131155 11554 115557 155570 155718 57188 171890 18914 189152 91535 115370 153719 137194 171957	189994 99952 199534 195355 53554 135557 155570 155718 57188 171897 18990 189915 99150 191500 215013 150136 1370 13700 137019 70201 102030 20313 203130 131310 213115 131170 11713 117154 171540 215419 154210 242106 21060 210618 206190 61914 19153 191549 115507 155084 150851 208510 185113 151136 111370 13700 137019 70200 102004 20057 200570 105718 57194 171955 19554 195557 155570 155718 57188 171883 18830 188318 83180 31817 18173 181730 17317 173178 31798 17988 179898 98999 189994 99952 199532 195320 53218 132200 122010 20109 201090 210911 109110 91113 11145 111467 14684 146846 168477 184770 147712 77120 171215 12161 121618 16190 161900 119018 90188
Jumlah <i>Hash</i>	47	99
<i>Hash</i> yang sama	189994 99952 199534 195355 53554 135557 155570 155718 57188 171897 18990 189915 99150 13700 137019 70201 102030 20313 203130 131310 13700 137019 155570 155718 57188 189994 99952	
Jumlah	27	
		Dokumen3
		194712 147120 71217 212174 221743 17430 174315 43150 31519 15200 152014 20152 201540 115413 154138 141390 213900 139018 90188 101891 18910 189113 91136 111378 13800 138013 80136 201367 213670 136717 67173 71752 17520 175217 52174
Jumlah <i>Hash</i>		35
<i>Hash</i> yang sama	194712 147120 71217 212174	
Jumlah	4	

5.2.4 Nilai Similarity/Kemiripan

Untuk mencari nilai kemiripan/*similarity* dengan melakukan rumus persamaan (2.3) berikut ini.

1. Kemiripan Menggunakan K-gram 4

$$\text{Dok1\&dok2} \quad S = \frac{2*34}{(48+100)} = \frac{68}{148} = 0,4594 = 45,94\% \quad (2.3)$$

$$\text{Dok1\&dok3} \quad S = \frac{2*5}{(48+36)} = \frac{10}{84} = 0,1190 = 11,90\% \quad (2.3)$$

$$\text{Dok2\&dok3} \quad S = \frac{2*3}{(100+36)} = \frac{6}{136} = 0,0441 = 4,41\% \quad (2.3)$$

2. Kemiripan menggunakan K-gram 5

$$\text{Dok1\&dok2} \quad S = \frac{2*27}{(47+99)} = \frac{54}{146} = 0,3698 = 36,98\% \quad (2.3)$$

$$\text{Dok1\&dok3} \quad S = \frac{2*4}{(47+35)} = \frac{8}{82} = 0,0975 = 9,75\% \quad (2.3)$$

$$\text{Dok2\&dok3} \quad S = \frac{2*1}{(99+35)} = \frac{2}{134} = 0,0149 = 1,49\% \quad (2.3)$$

5.2.5 Rekap Perbandingan Similarity

Rekap perbandingan *similarity* yaitu nilai *similarity*/kemiripan jawaban mahasiswa dengan menggunakan rumus persamaan 2.3. Kemudian dari nilai perbandingan tersebut dibuat lah rekap nilai antar mahasiswa, sebagaimana terlihat pada tabel 5.6, tabel 5.7, dan tabel 5.8.

Tabel 5.6 Rekap nilai *similarity* untuk fauseh

K-gram 4	Nama Mahasiswa	Nama Mahasiswa Pemanding	Kemiripan
	Fauseh	Ida Ayu Rahayu Nirahim	36,48%
	Fauseh	Nur Sri Utami	11,90%
K-gram 5	Nama Mahasiswa	Nama Mahasiswa Pemanding	Kemiripan
	Fauseh	Ida Ayu Rahayu Nirahim	36,98%
	Fauseh	Nur Sri Utami	9,75%

Pada tabel 5.6 dapat dilihat nilai perbandingan kemiripan mahasiswa yaitu berdasarkan k-gram 4 dan k-gram 5.

Tabel 5.7 Rekap nilai *similarity* untuk ida ayu

K-gram 4	Nama Mahasiswa	Nama mahasiswa Pemanding	Kemiripan
	Ida Ayu Rahayu Nirahim	Fauseh	36,48%
	Ida Ayu Rahayu Nirahim	Nur Sri Utami	4,41%
K-gram 5	Nama Mahasiswa	Nama Mahasiswa Pemanding	Kemiripan
	Ida Ayu Rahayu Nirahim	Fauseh	36,98%
	Ida Ayu Rahayu Nirahim	Nur Sri Utami	1,49%

Pada tabel 5.7 dapat dilihat nilai perbandingan kemiripan mahasiswa yaitu berdasarkan k-gram 4 dan k-gram 5.

Tabel 5.8 Rekap nilai similarity untuk nur sri

K-gram 4	Nama Mahasiswa	Nama Mahasiswa Pemanding	Kemiripan
	Nur Sri Utami	Fauseh	11,90%
	Nur Sri Utami	Ida Ayu Rahayu Nirahim	4,41%
K-gram 5	Nama Mahasiswa	Nama mahasiswa Pemanding	Kemiripan
	Nur Sri Utami	Fauseh	9,75%
	Nur Sri Utami	Ida Ayu Rahayu Nirahim	1,49%

Pada tabel 5.8 dapat dilihat nilai perbandingan kemiripan mahasiswa yaitu berdasarkan k-gram 4 dan k-gram 5.

5.3 Pengujian Sistem

5.3.1 Pengujian Black Box

Pada tahap ini akan membahas mengenai hasil pengujian dari sistem yang telah dibuat. Tahap validasi uji coba tersebut menggunakan uji coba perangkat lunak (*Black Box*) sebagaimana terlihat pada tabel 5.9.

Tabel 5.9 Pengujian *black box* sistem pendeteksi kemiripan jawaban *essay*

No	Skenario pengujian	Kasus pengujian	Hasil yang di harapkan	Hasil pengujian	Status
Pengujian login					
1	Login pengguna sesuai pilihan	Masuk pada <i>form menu login</i> dan menginputkan <i>username</i> dan <i>password</i> kemudian klik tombol <i>login</i>	Pengguna dapat masuk kedalam sistem	Sesuai	Normal
2	Login pengguna menggunakan kombinasi yang salah	Masuk pada <i>form menu login</i> dan menginputkan <i>username</i> dan <i>password</i> Menggunakan kombinasi yang salah kemudian klik tombol <i>login</i>	Pengguna tidak dapat masuk kedalam sistem	Sesuai	Error
Pengujian data soal					
1	Tambah Soal	Masukan semua soal kemudian klik tombol <i>simpan</i>	Data soal berhasil ditambahkan	Sesuai	Normal

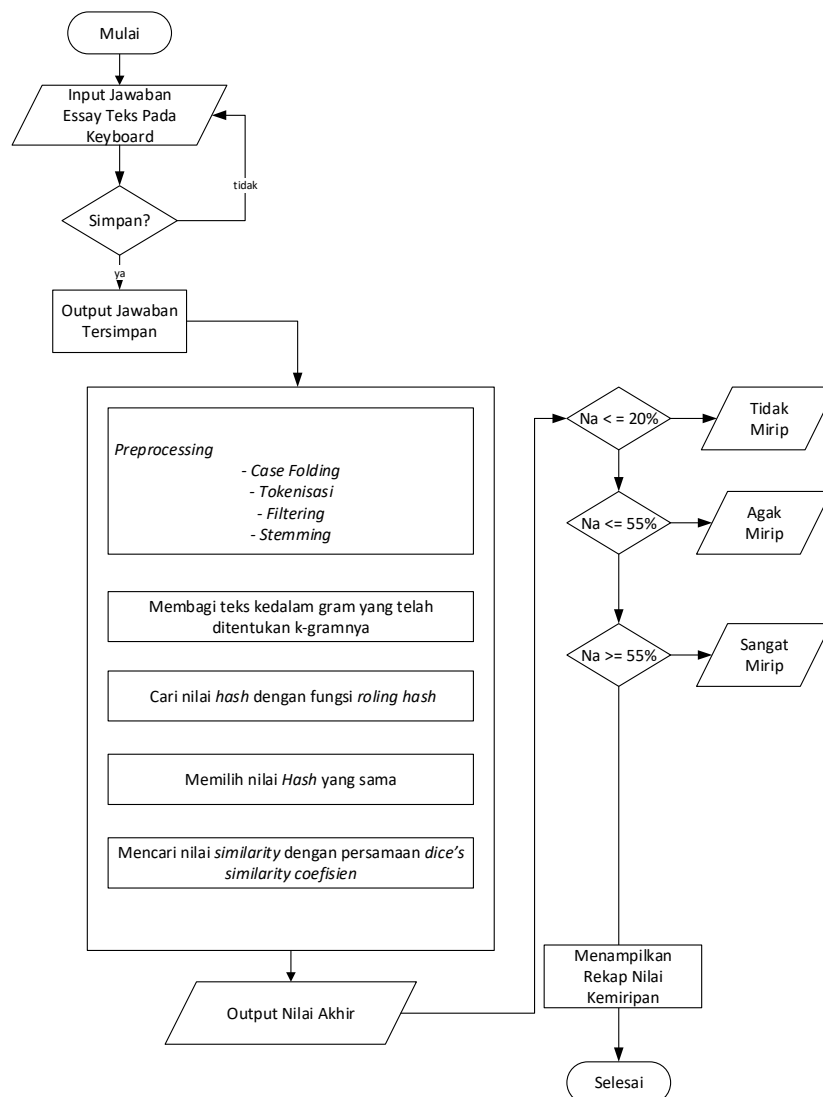
2	Tambah Soal	Masukkan soal yang sama	Data soal muncul error soal sudah ada	Sesuai	Error
2	Ubah data soal	Klik data soal yang mau diubah kemudian klik tombol ubah	Data soal berhasil diubah	Sesuai	Normal
3	Hapus data soal	Klik data soal yang mau dihapus kemudian klik tombol hapus	Data soal berhasil dihapus	Sesuai	Normal
Pengujian ujian mahasiswa					
1	Mengisi jawaban pada menu jawab soal	Mahasiswa menjawab soal pada kolom yang sesuai dengan nomor soal kemudian klik tombol simpan soal	Data jawaban mahasiswa tersimpan ke <i>database</i>	Sesuai	Normal
Pengujian pengolahan Preprocessing dan rabin karp					
1	Memilih <i>menu preprocessing</i>	Memilih menu <i>preprocessing</i> untuk melakukan proses <i>preprocessing</i> pada data <i>training</i> jawaban mahasiswa	Sistem menampilkan data <i>training</i> jawaban mahasiswa yang sudah diproses oleh <i>preprocessing</i>	Sesuai	Normal
2	Memilih <i>menu</i> hitung rabin karp	Memilih mahasiswa yang akan dihitung nilai kemiripannya, kemudian memilih <i>button</i> cari kemiripan	Sistem menampilkan hasil <i>preprocessing</i> dan perhitungan <i>rabin karp</i> serta similaritas persentasenya	Sesuai	Normal
3	Memilih menu rekap kemiripan	Memilih mahasiswa kemudian tekan <i>button</i> perbarui	Sistem menampilkan rekap kemiripan jawaban mahasiswa	Sesuai	Normal

5.3.2 Pengujian White Box

Pada tahap ini akan membahas mengenai hasil pengujian dari sistem perhitungan nilai akhir dari algoritma rabin karp. Pengujian dilakukan dengan membuat *test case* yang di dapatkan dari jalur independen. Dan jalur independen itu sendiri didapat dengan menghitung nilai *cyclomatic complexity* dari *flowgraph*, *flowgraph* dibuat berdasarkan dari *flowchart* sistem penilaian.

5.3.2.1 Flowchart

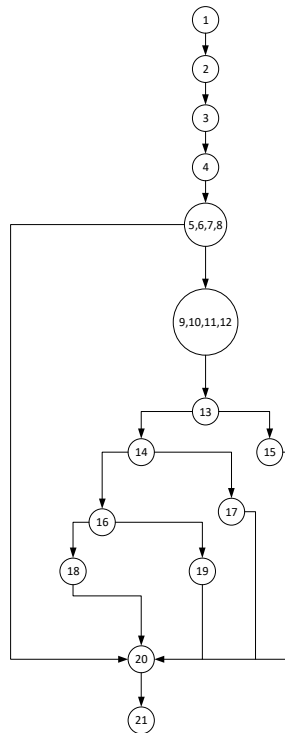
Perhitungan nilai akhir ditentukan dari nilai kemiripan jawaban essay dari masing-masing mahasiswa. *Flowchart* dari perhitungan nilai akhir dapat dilihat pada gambar 5.13 berikut ini.



Gambar 5.13 *Flowchart* perhitungan nilai akhir

5.3.2.2 Flowgraph

Dari *flowgraph* yang ada dalam sistem penilaian pada gambar 5.14 akan dihitung nilai *cyclomatic complexity*-nya. Nilai *cyclomatic complexity* dapat dihitung dengan 3.1.



Gambar 5.14 *Flowgraph* perhitungan nilai akhir
Dari *flowgraph* sistem penilaian pada gambar 5.14 dapat dihitung *cyclomatic complexity*:

$$V(G) = 18 \text{ edge} - 15 \text{ node} + 2 = 5$$

Jadi *cyclomatic complexity* untuk *flowgraph* pada gambar 5.14 adalah 5.

Tabel 5.10 *Source code* program fungsi nilai akhir kemiripan jawaban

```
function alphabet_to_number($string)
{
    $pangkat = array( '0' => 4, '1' => 3, '2' => 2, '3' => 1, '4' => 0,
);
    $leng = strlen($string);
    $total=0;
    for ($i=0; $i <$leng ; $i++) {
        $pot_kat = substr($string, $i,1);
        $nilai_kat = get_number_alpha($pot_kat);
        $akhir=$nilai_kat*pow(10,$pangkat[$i]);
        $total+=$akhir;
    }
}
```

```

return $total;
}

<th>KEMIRIPAN</th>
  <td colspan="2">
    <?php
      $akhir =
(2*$total_kemiripan) / (count($kat)+count($kat_b));
      $persen = round($akhir*100,2);
    ?>
    <?=$persen?> %
  </td>
</tr>
<tr>
  <th>KEMIRIPAN</th>
  <td colspan="2">
    <?php
      // print_r($total_kemiripan);
      $akhir = $akhir * 100;
      if($akhir>=56):?>
        Sangat Mirip
      <?php elseif($akhir>=21):?>
        Agak Mirip
      <?php else:?>
        Tidak Mirip
      <?php endif;?>
    </td>
  </tr>

```

5.3.2.3 Jalur Independen

Tabel 5.11 Jalur Independen

No	Jalur Independen	Keterangan
1	Jalur 1 = 1-2-3-4-5-6-7-8-20-21	Mahasiswa input jawaban, hasil nilai kemiripan ada direkap kemiripan.
2	Jalur 2 = 1-2-3-4-5-6-7-8-9-10-11-12-13-15-20-21	Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 20\%$ Tidak Mirip, nilai akhir ada di rekap kemiripan.
3	Jalur 3 = 1-2-3-4-5-6-7-8-9-10-11-12-13-14-17-20-21	Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 55\%$ Agak Mirip, nilai akhir ada di rekap kemiripan.
4	Jalur 4 = 1-2-3-4-5-6-7-8-9-10-11-12-13-14-16-19-10-21	Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \geq 55\%$ Sangat Mirip, nilai akhir ada di rekap kemiripan.
5	Jalur 5 = 1-2-3-4-5-6-7-8-9-10-11-12-13-14-16-18-10-21	Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 20\%$ Tidak Mirip, $Na \leq 55\%$, $Na \geq 55\%$ Sangat Mirip, nilai akhir ada di rekap kemiripan.

Sistem penilaian pada tabel 5.11 menurut hubungan nilai *cyclomatic complexity* dengan resiko pada penjelasan tabel 2.8 dengan lima jalur termasuk dalam prosedur yang cukup kompleks dengan resiko yang sedang.

5.3.2.4 Test Case Flowgraph Perhitungan Nilai Akhir

Dalam pembuatan test case dilakukan validasi dengan menguji coba dengan program. Sehingga diketahui apakah hasil yang diharapkan sesuai dengan hasil dari aplikasi sebagaimana terlihat pada tabel 5.12.

Tabel 5.12 Tes case flowgraph

1.Skenario Uji				
Skenario		Kondisi		
Input jawaban hitung preprocessing		Mahasiswa input jawaban, hasil nilai kemiripan ada direkap kemiripan.		
Input jawaban hitung nilai akhir		Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 20\%$ Tidak Mirip, nilai akhir ada di rekap kemiripan.		
Input jawaban hitung nilai akhir		Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 55\%$ Agak Mirip, nilai akhir ada di rekap kemiripan.		
Input jawaban hitung nilai akhir		Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \geq 55\%$ Sangat Mirip, nilai akhir ada di rekap kemiripan.		
Input jawaban hitung nilai akhir		Mahasiswa input jawaban kemudian di <i>processing</i> lalu hitung algoritma $Na \leq 20\%$ Tidak Mirip, $Na \leq 55\%$, $Na \geq 55\%$ Sangat Mirip, nilai akhir ada di rekap kemiripan.		
2.Uji Coba Dengan Data				
Skenario	Data	Hasil Yang Diharapkan	Hasil	Validasi
Input jawaban hitung preprocessing	Data dipreprocessing	Menampilkan data yang dipreprocessing	Menampilkan data yang dipreprocessing	Valid
Input jawaban hitung nilai akhir	$Na \leq 20\%$	Laporan menampilkan $Na \leq 20\%$ Tidak Mirip	Menampilkan $Na \leq 20\%$ Tidak Mirip	Valid
Input jawaban hitung nilai akhir	$Na \leq 55\%$	Laporan menampilkan $Na \leq 55\%$ Agak Mirip	Menampilkan $Na \leq 55\%$ Agak Mirip	Valid
Input jawaban hitung nilai akhir	$Na \geq 55\%$	Laporan menampilkan $Na \geq 55\%$ Sangat Mirip	Menampilkan $Na \geq 55\%$ Sangat Mirip	Valid
Input jawaban hitung nilai akhir	$Na \leq 20\%$	Laporan menampilkan $Na \leq 20\%$ Tidak Mirip	Menampilkan $Na \leq 20\%$ Tidak Mirip	Invalid

	Na<=55%	Laporan menampilkan Na<=55%	Menampilkan Na<=55% Agak Mirip	
	Na>=55%	Laporan menampilkan Na>=55% Sangat Mirip	Menampilkan Na>=55% Sangat Mirip	

3. Hasil Tes Case

Skenario	Hasil Pengujian	Sesuai
Input jawaban hitung preprocessing	Sama	Valid
Input jawaban hitung nilai akhir	Sama	Valid
Input jawaban hitung nilai akhir	Sama	Valid
Input jawaban hitung nilai akhir	Sama	Valid
Input jawaban hitung nilai akhir	Tidak	Invalid

5.3.3 Pengujian K-gram

Pengujian k-gram digunakan untuk melihat nilai selisih persentase antara k-gram 4 dengan k-gram 5, untuk lebih jelasnya dapat dilihat pada tabel 5.13.

Tabel 5.13 Pengujian k-gram

Nama Dokumen	K-gram 4	K-gram 5	Selisih (Kgram4 – Kgram 5)	Selisih %
Dok1&dok2	0,4594	0,3698	0,0896	8,96 %
Dok1&dok3	0,1190	0,0975	0,0215	2,15 %
Dok2&dok1	0,4594	0,3698	0,0896	8,96 %
Dok2&dok3	0,0441	0,0149	0,0292	2,92 %
Dok3&dok1	0,1190	0,0975	0,0215	2,15 %
Dok3&dok2	0,0441	0,0149	0,0292	2,92 %

Berdasarkan tabel 5.13 dapat dilihat bahwa nilai persentase antara k-gram 4 dengan k-gram 5 mempunyai perbedaan yaitu pada k-gram 4 nilai kemiripan antara dokumen1&dokumen2 senilai 0,4594, sedangkan pada k-gram 5 nilai dokumen1&dokumen2 senilai 0,3698 dengan demikian selisih diantara keduanya adalah 8,96% . Dapat disimpulkan bahwa semakin besar nilai k-gram maka nilai persentase semakin kecil dan semakin kecil nilai k-gram maka nilai persentase kemiripan semakin besar. Akan tetapi semakin kecil nilai *k-gram*, semakin akurat pula nilai persentase kemiripan dokumen tersebut.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian dengan judul Sistem Pendeteksi Kemiripan Jawaban Essay Menggunakan algoritma Rabin Karp, maka dapat ditarik kesimpulan sebagai berikut :

1. Sistem pendeteksi kemiripan jawaban *essay* dengan menggunakan algoritma *Rabin karp* sudah sesuai yang diharapkan, yaitu menampilkan nilai persentase kemiripan jawaban.
2. Hasil pengujian keakurasian algoritma *rabin karp* dengan menggunakan *k-gram* nilai persentase antara k-gram 4 dengan k-gram 5 mempunyai perbedaan yaitu pada k-gram 4 nilai kemiripan antara dokumen1 & dokumen2 senilai 0,4594, sedangkan pada k-gram 5 nilai dokumen1 & dokumen2 senilai 0,3698 dengan demikian selisih diantara keduanya adalah 8,96% . Jadi dapat disimpulkan bahwa semakin besar nilai k-gram maka nilai persentase semakin kecil dan semakin kecil nilai k-gram maka nilai persentase kemiripan semakin besar. Akan tetapi semakin kecil nilai *k-gram*, semakin akurat pula nilai persentase kemiripan dokumen tersebut.

6.2 Saran

Berdasarkan hasil penelitian skripsi mengenai “Sistem Pendeteksi Kemiripan Jawaban Essay Menggunakan Algoritma Rabin Karp”, masih terdapat banyak kekurangan dan kelemahan pada sistem, mengingat keterbatasan pengetahuan, pengalaman dan waktu, maka peneliti memberikan beberapa saran untuk pengembangan sistem selanjutnya :

1. Pengembangan sistem selanjutnya disarankan untuk mencoba melakukan penelitian dengan membandingkan metode yang lain dalam melakukan penilaian *Similarity*.
2. Diharapkan dapat dikembangkan dalam penggunaan *time stamp* waktu *login* tercatat, dan waktu mundur pengerjaan bisa diatur lebih dari 60 menit dan kurang dari 60 menit.

DAFTAR PUSTAKA

- [1] S. Adi, “Penerapan Algoritma Rabin Karp Untuk Mendeteksi Kemiripan Judul Skripsi,” *J. Mantik Penuasa*, vol. 22, no. 1, pp. 125–130, 2018.
- [2] F. Sulianta, *Seri Referensi Praktis: Konten Internet*. Elex Media Komputindo, 2007.
- [3] N. Hidayat and L. Afuan, “Penilaian Ujian Otomatis untuk Soal Bertipe Essay pada PJJ APTIKOM menggunakan Cosine Similarity,” in *Seminar Nasional Aptikom (SEMNASITIK) 2019*, 2019, pp. 259–271.
- [4] M. Maskur, D. Q. Putra, and N. Hayatin, “Deteksi Kemiripan Dokumen Pengajuan Proposal Menggunakan Algoritma Biword Winnowing Pada Sistem Informasi Penelitian Dan Pengabdian,” *J. Repos.*, 2020, doi: 10.22219/repositor.v2i5.266.
- [5] L. Hermawan and M. B. Ismiati, “Aplikasi Pengecekan Dokumen Digital Tugas Mahasiswa Berbasis Website,” *J. Buana Inform.*, 2020, doi: 10.24002/jbi.v11i2.3706.
- [6] W. Abdul Gani, “Klasifikasi Emosi Pada Teks Bahasa Indonesia Menggunakan Metode K-Nearest Neighbor Dengan Pembobotan Widf.” Universitas Komputer Indonesia, 2016.
- [7] M. B. Dharmawan, B. Pramono, S. Stasiswaty, and M. I. Sarita, “Implementasi Latent Semantic Analysis Method Untuk Mendeteksi Kemiripan Kalimat Esai Pada Ujian Online Teknik Informatika Universitas Halu Oleo,” *semanTIK*, vol. 6, no. 1, pp. 39–48.
- [8] Atmopawiro Alsasian, “Pengkajian Dan Analisis Tiga Algoritma Efisien Rabin-Karp, Knuth-Morris-Pratt, Dan Boyer-Moore Dalam Pencarian Pola Dalam Suatu Teks.”
- [9] A. Sunyoto, “Implementasi Algoritma Rabin Karp untuk Pendeteksian Plagiat Dokumen Teks Menggunakan Konsep Similarity,” in *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, 2013, vol. 1, no. 1.

- [10] E. I. S. Azhar Azziz Affi, "Sistem Ujian Susun Kata Dan Susun Kalimat Online Menggunakan Metode Rabin-Karp Sistem Ujian Susun Kata Dan Susun Kalimat Online Menggunakan Metode Rabin-Karp," vol. 1, no. 1, 2019.
- [11] L. J. Purba and L. Sitorus, "Perancangan Aplikasi Untuk Menghitung Persentase Kemiripan Proposal Dan Isi Skripsi Dengan Algoritma Rabin-Karp," *J. Tek. Inform. UNIKA St. Thomas*, vol. 3, no. 1, pp. 17–25, 2018.
- [12] R. Novia, Y. Prastyaningsih, and H. Rhomadhona, "Implementasi Metode Rabin Karp Pada Aplikasi Pengecekan Kemiripan Judul Tugas Akhir (Studi Kasus : Jurusan Teknik Informatika Politala) Implementation of The Rabin Karp Method in Final Project Title Checking Application," vol. 4, no. 1, pp. 69–81, 2021.
- [13] I. Widaningrum, D. Mustikasari, R. Arifin, and E. Dyah Cahyani, "Analisa Penggunaan K-Gram pada Karakter, Kata dan Kalimat untuk Mendeteksi Kesamaan Dokumen," *Pros. Semin. Nas. Teknoka*, vol. 5, no. 2502, pp. 59–64, 2020, doi: 10.22236/teknoka.v5i.333.
- [14] A. Putera Utama Siahaan and Sugianto, "Analisis k-gram, basis dan modulo rabin-karp sebagai penentu akurasi persentase kemiripan dokumen," *SENASPRO 2017 | Semin. Nas. dan Gelar Prod.*, pp. 198–206, 2017.
- [15] A. Filcha and M. Hayaty, "Implementasi Algoritma Rabin-Karp untuk Pendeteksi Plagiarisme pada Dokumen Tugas Mahasiswa," *JUITA J. Inform.*, vol. 7, no. 1, p. 25, 2019, doi: 10.30595/juita.v7i1.4063.
- [16] A. Sukmana, Kusri, and A. Sunyoto, "Perbandingan Penggunaan Stemming Pada Deteksi Kemiripan Dokumen Menggunakan Metode Rabin Karp Dan Jaccard Similarity," *Semin. Nas. Teknol. Inf. dan Multimed.*, pp. 67–72, 2018, [Online]. Available: <https://ojs.amikom.ac.id/index.php/semnasteknomedia/article/viewFile/2120/1925>.
- [17] A. S. Rossa and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. 2014.

- [18] S. Lestiansyah, *Pengertian User Interface*. Jakarta: PT. Elex Media Komputindo, 2014.
- [19] R. Abdulloh, *7 in 1 Pemrograman Web Untuk Pemula*. Elex Media Komputindo, 2018.
- [20] A. F. K. Sibero, "Web programming power pack," 2013.
- [21] D. Suprapti *et al.*, "Analisa Pengujian Sistem Informasi Penjualan," *Anal. Penguji. Sist. Inf. Penjualan Menggunakan Metod. White Box*, pp. 1–12, 2017.
- [22] Prof.Dr.Sugiyono, "Metode Penelitian Kualitatif, kuantitatif dan R&D," *Bandung CV Alf.*, 2013.
- [23] N. Adi, "Rekayasa Perangkat Lunak Menggunakan UML dan Java," *Andi, Yogyakarta*, 2009.

LAMPIRAN I. CONTOH KODE PROGRAM

Kode Program rabin_karp_hasil.php

```
<?php
function get_number_alpha($string) {

    $string = strtoupper($string);
    $length = strlen($string);
    $number = 0;
    $level = 1;
    while ($length >= $level ) {
        $char = $string[$length - $level];
        $c = ord($char) - 65;
        $number += $c * (26 ** ($level-1));
        $level++;
    }
    return $number;
}

function alphabet_to_number($string)
{
    $pangkat = array( '0' => 4, '1' => 3, '2' => 2, '3' => 1, '4' => 0,
);
    $len = strlen($string);
    $total=0;
    for ($i=0; $i <$len ; $i++) {
        $pot_kat = substr($string, $i,1);
        $nilai_kat = get_number_alpha($pot_kat);
        $akhir=$nilai_kat*pow(10,$pangkat[$i]);
        $total+=$akhir;
    }
    return $total;
}

$time_start = microtime(true);
$TRAINING = $DATA;
$TERMS = get_all_terms( $TRAINING);
$str = $kata;

$siswa_awal = $db->get_results("SELECT * FROM tb_data WHERE
id_mahasiswa='$_POST[siswa]'");

$rows = $db->get_results("SELECT * FROM tb_data");
$GET_NAMA= array();
foreach ($rows as $row) {
    $stm[$_POST[id_mahasiswa] .= " ".$row->data;
}

$arr_token = get_tokenisasi($stm);
$arr_stopword = get_stopword($arr_token);
$arr_stemming = get_stemming($arr_stopword);
$arr_akhir = get_stopword($arr_stemming);
```

```

$rows = $db->get_results("SELECT * FROM tb_mahasiswa group by
id_mahasiswa");
foreach ($rows as $row) {
    $GET_NAMA[$row->id_mahasiswa] = $row->nama_mahasiswa;
}

?>

<div class="panel panel-primary">
    <div class="panel-heading">
        <h3 class="panel-title"><a data-toggle=""
href="#c2">Tokenisasi</a></h3>
    </div>
    <div class="table-responsive " id="c2">
        <table class="table table-bordered table-hover table-striped
small">
            <thead><tr>
                <th>No</th>
                <th>Tokenisasi</th>
            </tr></thead>
            <?php
            $no = 1;
            foreach($arr_token as $key => $val):?>
                <tr>
                    <td><?=$no++?></td>
                    <td><?=$val?></td>
                </tr>
            <?php endforeach?>
        </table>
    </div>
</div>

<div class="panel panel-primary">
    <div class="panel-heading">
        <h3 class="panel-title"><a data-toggle=""
href="#c3">Stopword</a></h3>
    </div>
    <div class="table-responsive " id="c3">
        <table class="table table-bordered table-hover table-striped
small">
            <thead><tr>
                <th>No</th>
                <th>Stopword</th>
            </tr></thead>
            <?php
            $no = 1;
            foreach($arr_stopword as $key => $val):?>
                <tr>
                    <td><?=$no++?></td>
                    <td><?=$val?></td>
                </tr>
            <?php endforeach?>
        </table>
    </div>
</div>

```

```

<div class="panel panel-primary">
  <div class="panel-heading">
    <h3 class="panel-title"><a data-toggle=""
href="#c4">Stemming</a></h3>
  </div>
  <div class="table-responsive " id="c4">
    <table class="table table-bordered table-hover table-striped
small">
      <thead><tr>
        <th>No</th>
        <th>Stemming</th>
      </tr></thead>
      <?php
      $no = 1;
      foreach($arr_stemming as $key => $val):?>
        <tr>
          <td><?=$no++?></td>
          <td><?=$val?></td>
        </tr>
      <?php endforeach?>
    </table>
  </div>
</div>
<?php
$rows = $db->get_results("select * from tb_data WHERE
id_mahasiswa!='$_POST[siswa]' GROUP BY id_mahasiswa");

$jawaban_siswa_a = str_replace(' ',
'', $arr_akhir[$_POST['siswa']]);
$siswa_a = strlen($jawaban_siswa_a)-5;
$kat = array();
$cek_finger=array();
for ($i=0; $i <= $siswa_a; $i++) {
  $kat[$i]['k_gram'] = substr($jawaban_siswa_a, $i,5);
  $kat[$i]['hash']=alphabet_to_number($kat[$i]['k_gram']);

  $kat[$i]['fingerprint']=alphabet_to_number($kat[$i]['k_gram']);
  $cek_finger[$kat[$i]['fingerprint']]=$kat[$i]['fingerprint'];
}

foreach ($rows as $row):
$total_finger=0;
$jawaban_siswa_b = str_replace(' ', '', $arr_akhir[$row-
>id_mahasiswa]);
$siswa_b = strlen($jawaban_siswa_b)-5;
$kat_b = array();
$cek_finger_b=array();

for ($i=0; $i <= $siswa_b; $i++) {
  $kat_b[$i]['k_gram'] = substr($jawaban_siswa_b, $i,5);
  $kat_b[$i]['hash']=alphabet_to_number($kat_b[$i]['k_gram']);

  $kat_b[$i]['fingerprint']=alphabet_to_number($kat_b[$i]['k_gram'])
;

```

```

$cek_finger_b[$kat_b[$i]['fingerprint']]=$kat_b[$i]['fingerprint']
;
}

?>
<div class="panel panel-primary">
  <div class="panel-heading">
    <h3 class="panel-title"><a data-toggle=""
href="#c4">Perbandingan Mahasiswa NIM <?=$_POST['siswa']?> dengan
NIM <?=$row->id_mahasiswa?></a></h3>
  </div>
  <div class="table-responsive " id="c4">
    <table class="table table-bordered table-hover table-striped
small">
      <thead>
        <tr>
          <th>NAMA MAHASISWA</th>
          <th><?=$_GET_NAMA[$_POST['siswa']]?></th>
          <th><?=$_GET_NAMA[$row->id_mahasiswa]?></th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th>K-GRAM</th>
          <td>
            <?php foreach ($kat as $key => $value):?>
              {<?=$value['k_gram']?>}
            <?php endforeach;?>
          </td>
          <td>
            <?php foreach ($kat_b as $key => $value):?>
              {<?=$value['k_gram']?>}
            <?php endforeach;?>
          </td>
        </tr>
        <tr>
          <th>HASH</th>
          <td>
            <?php foreach ($kat as $key => $value):?>
              <?=$value['hash']?>
            <?php endforeach;?>
          </td>
          <td>
            <?php foreach ($kat_b as $key => $value):?>
              <?=$value['hash']?>
            <?php endforeach;?>
          </td>
        </tr>
        <tr>
          <th>FINGERPRINT</th>
          <td>
            <?php

// print_r(count($kat));
foreach ($kat as $key => $value):

```

```

        $fingerprint1+=$value['fingerprint'];
        $total_finger+=$value['fingerprint'];?>
        <?=$value['fingerprint']?>
    <?php endforeach;?>
</td>
<td>
    <?php

        foreach ($kat_b as $key => $value):
            $fingerprint2=0;
            $fingerprint2+=$value['fingerprint'];
            $total_finger+=$value['fingerprint'];?>
            <?=$value['fingerprint']?>
        <?php endforeach;?>
    </td>
</tr>
<tr>
    <th>FINGERPRINT 1 DAN 2</th>
    <td>

        <?php
        if(count($kat)<=count($kat_b)):
            $total_kemiripan=0;
            foreach ($kat_b as $key => $value):?>
                <?php
if(!empty($cek_finger[$value['fingerprint']])):$total_kemiripan++?
>
                    <?=$cek_finger[$value['fingerprint']]?>
                <?php endif;?>
            <?php endforeach;
            else:?>
            <?php $total_kemiripan=0;
            foreach ($kat as $key => $value):?>
                <?php
if(!empty($cek_finger_b[$value['fingerprint']])):$total_kemiripan+
+?>
                    <?=$cek_finger_b[$value['fingerprint']]?>
                <?php endif;?>
            <?php endforeach;?>
        <?php endif;?>
    </td>
</tr>
<tr>
    <th>KEMIRIPAN</th>
    <td colspan="2">
        <?php

            $akhir =
(2*$total_kemiripan)/(count($kat)+count($kat_b));
            $persen = round($akhir*100,2);
            ?>
            <?=$persen?> %
        </td>
</tr>
<tr>

```

```

        <th>KEMIRIPAN</th>
        <td colspan="2">
            <?php
                // print_r($total_kemiripan);
                $akhir = $akhir * 100;
                if($akhir>=56):?>
                    Sangat Mirip
                <?php elseif($akhir>=21):?>
                    Agak Mirip
                <?php else:?>
                    Tidak Mirip
                <?php endif;?>
            </td>
        </tr>
        <?php
            $cek = $db->get_row("SELECT * FROM tb_hasil WHERE
            id_mahasiswa_a='$_POST[siswa]' AND id_mahasiswa_b='$_row-
            >id_mahasiswa'");
            if($cek)
            {
                $db->query("UPDATE tb_hasil set
            kemiripan='$_persen' WHERE id_mahasiswa_a='$_POST[siswa]' AND
            id_mahasiswa_b='$_row->id_mahasiswa'");
            }else{
                $db->query("INSERT INTO tb_hasil
            (id_mahasiswa_a,id_mahasiswa_b,kemiripan) VALUES
            ('$_POST[siswa]','$_row->id_mahasiswa','$_persen')");
            }
        ?>
    </tbody>
</table>
</div>
</div>
<?php endforeach;?>

```


LAMPIRAN II. BIOGRAFI PENULIS

Nama : Omma Sariani Siregar
Tempat Tanggal Lahir : Batang Baruhar Julu, 27 July 1997
Jenis Kelamin : Perempuan
Agama : Islam
Status : Belum Menikah
Alamat : Jl. Parit H.Husin 1 Gg Al-Qadar No.1 Pontianak
Email : omma.siregar@unmuhpnk.ac.id

PENDIDIKAN FORMAL

Tahun 2004-2010 : SD Negeri 101340 Batang Baruhar Julu
Tahun 2010-2013 : MTsN Padang Bolak
Tahun 2013-2016 : SMK Negeri 1 Padang Sidempuan
Tahun 2016-2021 : Universitas Muhammadiyah Pontianak

Pontianak, 1 September 2021
Yang Menyatakan,

Omma Sariani Siregar
NIM. 171221134