

RANCANG BANGUN SISTEM PENDETEKSI PLAGIARISME TUGAS AKHIR DENGAN MENGGUNAKAN ALGORITMA RABIN-KARP

Siti Lailiyah, S.Kom., M.Kom.¹, Amelia Yusnita, S.Kom., M.Kom.², Daniel Evan Rory³

Program Studi Teknik Informatika, STMIK Widya Cipta Dharma
Jl. M. Yamin No.25, Samarinda, 75123
danielrory6@gmail.com

ABSTRAK

Tugas akhir, yang berupa skripsi, merupakan karya ilmiah yang dihasilkan oleh Mahasiswa sebagai syarat untuk memperoleh gelar sarjana sesuai dengan jenjang pendidikan yang dijalani. Permasalahannya adalah dalam dunia pendidikan tinggi saat ini, plagiarisme di kalangan mahasiswa telah menjadi suatu masalah dengan seiring berkembangnya kemajuan teknologi informasi sehingga memungkinkan mahasiswa dengan cepat menyalin atau mengcopy-paste karya orang lain.

Dalam konteks penelitian ini, kekhawatiran terhadap kasus plagiat dalam dokumen semakin bertambah. Untuk memahami sejauh mana tingkat plagiarisme dalam suatu dokumen, penggunaan sistem pendeteksi plagiarisme berbasis algoritma Rabin-Karp menjadi penting. Algoritma ini digunakan untuk memeriksa kemiripan antara dokumen berupa PDF dengan sumber-sumber yang telah ada, membantu mengidentifikasi pola yang mirip dengan sumber aslinya.

Sistem pendeteksi plagiarisme ini memberikan informasi yang penting untuk mengukur tingkat keaslian suatu karya dan memahami sejauh mana kebutuhan untuk melakukan tindakan pencegahan atau koreksi yang diperlukan. Metode pengumpulan data menggunakan studi literatur, wawancara dan metode pengembangan sistem menggunakan prototype. Karena metode ini mempunyai tahapan – tahapan yang jelas nyata dan praktis. Kemudian pengujian sistem yaitu Black Box dan Beta Testing, sehingga sistem yang didapat sesuai dengan keinginan para calon pengguna.

Kata Kunci : *Sistem Pendeteksi, Plagiarisme, Algoritma Rabin-Karp.*

1. PENDAHULUAN

Tugas akhir, yang berupa skripsi, merupakan karya ilmiah yang dihasilkan oleh Mahasiswa sebagai syarat untuk memperoleh gelar sarjana sesuai dengan jenjang pendidikan yang dijalani. Saat ini, pengelolaan data tugas akhir masih bersifat terbatas dan belum terintegrasi dengan baik, dengan data yang sering dikelola dalam format PDF. Masalah utama lainnya adalah dalam dunia pendidikan tinggi saat ini, plagiarisme di kalangan mahasiswa telah menjadi suatu masalah dengan seiring berkembangnya kemajuan teknologi informasi sehingga memudahkan dalam mengakses berbagai sumber informasi, dengan demikian memungkinkan mahasiswa dengan cepat menyalin atau mengcopy-paste karya orang lain.

Dalam konteks penelitian yang akan dilakukan, kekhawatiran terhadap kasus plagiat dalam dokumen semakin bertambah. Untuk memahami sejauh mana tingkat plagiarisme dalam suatu dokumen, penggunaan sistem pendeteksi plagiarisme berbasis algoritma Rabin-Karp menjadi penting. Algoritma ini digunakan untuk memeriksa kemiripan antara dokumen berupa PDF dengan sumber-sumber yang telah ada, membantu mengidentifikasi pola yang mirip dengan sumber aslinya. Sistem pendeteksi plagiarisme ini memberikan informasi yang penting untuk mengukur tingkat keaslian suatu karya

dan memahami sejauh mana kebutuhan untuk melakukan tindakan pencegahan atau koreksi yang diperlukan.

Dengan demikian, penyusunan sebuah sistem pendeteksi plagiarisme pada skripsi, dengan memanfaatkan algoritma Rabin-Karp, menjadi langkah penting untuk meningkatkan efektivitas dan kehandalan manajemen tugas akhir. Sistem ini diharapkan dapat mengurangi kasus plagiat pada skripsi bagi mahasiswa.

2. RUANG LINGKUP PENELITIAN

2.1 RUMUSAN MASALAH

Berdasarkan latar belakang yang telah dikemukakan di atas, maka rumusan masalah dalam penelitian ini adalah “Bagaimana merancang dan mengimplementasikan sebuah sistem pendeteksi plagiarisme pada skripsi dengan menggunakan algoritma Rabin-Karp sebagai solusi untuk meningkatkan efektivitas pengelolaan tugas akhir?”

2.2 BATASAN MASALAH

Dalam membuat sistem pendeteksi plagiarisme tugas akhir dengan menggunakan algoritma rabin-karp, terdapat beberapa batasan masalah yang perlu diperhatikan. Beberapa batasan masalah tersebut antara lain:

1. Penelitian ini fokus pada perbandingan antara dua dokumen yang diperoleh dari data skripsi mahasiswa

STMIK Wicida dengan dokumen yang ingin dibandingkan, dan dokumen akan dibandingkan secara terpisah.

2. Sistem tidak memperhatikan kesalahan ejaan / penulisan pada dokumen.
3. Sistem tidak dapat melakukan login.
4. Hanya menguji data berupa teks, tidak menguji data berupa gambar maupun suara.
5. Penelitian tidak mencakup implementasi sistem pada aspek selain deteksi plagiarisme, seperti manajemen kehadiran atau progress pengerjaan tugas akhir mahasiswa.

2.3 TUJUAN PENELITIAN

Tujuan dari penelitian ini adalah merancang dan mengimplementasikan sebuah sistem pendeteksi plagiarisme pada skripsi dengan menggunakan algoritma Rabin-Karp.

3. BAHAN DAN METODE

Adapun bahan dan metode yang digunakan dalam membangun penelitian ini yaitu:

3.1 Rancang Bangun

Rancang adalah urutan prosedur untuk menafsirkan hasil analisa dari suatu sistem ke dalam bahasa pemrograman untuk menjelaskan dengan rinci bagaimana komponen-komponen sistem diterapkankan. Rancang bangun merupakan aktivitas menafsirkan hasil analisa ke dalam bentuk kemasan perangkat lunak (software) lalu membuat sistem tersebut ataupun merenovasi sistem yang telah ada (Surahman et al., 2022).

3.2 Sistem

Sistem menurut (Hidayah et al., 2020) adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasanya terbagi dalam sub system yang lebih kecil yang mendukung system yang lebih besar.

3.3 Website

Menurut (Nurjani & Inda Aditya, 2023) website merupakan sebuah media yang memiliki banyak halaman yang saling terhubung (hyperlink), dimana website memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara dan animasi atau penggabungan dari semuanya

3.4 Plagiarisme

Pengertian plagiarisme menurut Kamus Besar Bahasa Indonesia (KBBI) merupakan tindakan penjiplakan yang melanggar hak cipta. Sedangkan plagiat atau jiplakan berarti “pengambilan karangan (pendapat dan sebagainya) orang lain dan menjadikannya seolah-olah karangan (pendapat dan sebagainya) sendiri, misalnya menerbitkan karya tulis orang lain atas nama dirinya sendiri” (Lubis et al., 2023).

3.5 Bahasa Pemrograman

Bahasa pemrograman, juga disebut bahasa komputer atau bahasa pemrograman komputer, adalah seperangkat instruksi standar untuk mengendalikan komputer. Bahasa pemrograman ini adalah seperangkat aturan sintaksis dan semantik yang digunakan untuk mendefinisikan program komputer (Syam et al., 2024).

3.6 Visual Studio Code

Visual Studio Code adalah editor kode sumber yang ringan namun kuat yang berjalan didesktop dan tersedia untuk Windows, macOS, dan Linux. Muncul dengan dukungan built-in untuk JavaScript, TypeScript dan Node.js dan memiliki ekosistem ekstensi yang kaya untuk bahasa lain (seperti C++, C#, Java, Python, PHP, Go) dan runtime (seperti .NET dan Unity) (Sofi & Dharmawan, 2022)

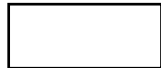
3.7 Flowchart


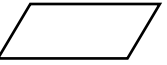
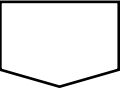
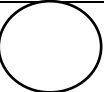
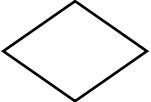


Menurut (Pemrograman, 2020) Flowchart dapat diartikan sebagai langkah-langkah penyelesaian masalah yang dituliskan dalam suatu simbol. Candra, B., Arfyanti, I., & Harianto, K. (2020). Sistem Informasi Manajemen Turnamen Futsal Berbasis Web Pada Rumah Futsal Melak. *Jurnal Informatika Wicida*, 9(2), 47–53. <https://doi.org/10.46984/inf-wcd.1237bol-simbol> tertentu. Diagram alir ini akan menunjukkan alur di dalam program secara logika. Flowchart ini diperlukan tidak Masukan Proses Keluaran hanya sebagai alat komunikasi tetapi juga sebagai pedoman, dan sebelum komponen-komponennya dapat lebih dipahami, perlu dikomunikasikan aturan-aturan desain org chart, yaitu:

1. Flowchart digambarkan dengan top-orientasi ke bawah dan kiri ke kanan.
2. Setiap aktivitas atau proses dalam bagan organisasi harus dinyatakan dengan jelas atau tidak ambigu.
3. Setiap diagram alur harus dimulai dari awalan atau status awal dan diakhiri dengan satu atau lebih status terminal/akhir/hals.
4. Gunakan konektor Status Halaman dan konektor Keluar halaman dengan label yang sama untuk menunjukkan bahwa koneksi antar algoritme misalnya karena terputus, perpindahan/perubahan halaman.

Untuk menggambarkan langkah atau pemecahan masalah secara sederhana, dapat dimengerti, rapi dan tidak ambigu dengan menggunakan beberapa simbol-simbol yang bisa dibilang standart merupakan tujuan dari flowchart pada tabel 1:

Tabel 1 Simbol-simbol flowchart


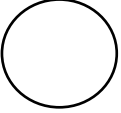
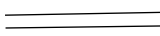
Simbol	Nama Simbol	Fungsi Simbol
	Proses	Untuk menyatakan proses yang dilakukan komputer. Bila ada proses yang dilakukan oleh komputer secara otomatis, maka kalian bisa mempergunakan simbol ini.

	Terminator	Untuk menyatakan awal dan juga akhir dari sebuah sistem atau proses.
	Input - Output	Untuk mempresentasikan fungsi I/O yang membuat data dapat diproses (input) atau ditampilkan (output).
	Konektor Off Page	Untuk menyambungkan proses bila tidak pada satu lembar kerja yang sama. Jadi kalau kalian ingin meneruskan proses tapi berganti halaman.
	Preparation	Untuk menghubungkan atau menyambungkan proses di dalam satu lembar kerja yang sama.
	Percabangan/Keputusan	Untuk melakukan percabangan, yaitu pemeriksaan terhadap suatu kondisi yang akan menghasilkan dua jawaban dari satu proses yang sudah terjadi. Kemungkinan jawaban yang ada dengan menggunakan simbol flowchart ini adalah jawaban iya dan juga tidak.
	Konektor On Page	Untuk menghubungkan atau menyambungkan proses dalam satu lembar kerja yang sama.
	Arah Aliran	Untuk menghubungkan antara satu simbol dengan simbol lainnya di flowchart.

3.8 Data Flow Diagram

Data flow diagram (DFD) merupakan suatu diagram yang menggunakan notasi-notasi dalam menggambarkan alur data sistem. DFD ini sangat membantu untuk memahami proses kerja sistem secara logika, terstruktur dan jelas. Dengan menggunakan DFD, permasalahan atau kelemahan dalam proses bisnis dapat teridentifikasi, sehingga dapat dilakukan perbaikan atau perbaikan yang diperlukan (Satyaningrat et al., 2023). Dapat dilihat pada tabel 2:





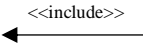
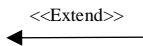
Tabel 2 Simbol-simbol DFD

Notasi	Nama	Keterangan
	Entitas Eksternal	Sistem yang menghasilkan informasi bagi transformasi oleh perangkat lunak atau menerima informasi yang dihasilkan oleh perangkat lunak.
	Proses	Menunjukkan transformasi yang diaplikasikan ke data (kontrol) dan mengubahnya dengan berbagai macam cara.
	Data Store	Tempat penyimpanan data dalam sistem

3.9 Use Case

Menurut (Rudianto & Achyani, 2022) "Diagram use case menyajikan interaksi antara use case dan actor. Dimana aktor dapat berupa orang, peralatan atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. Use case menggambarkan fungsionalitas sistem atau persyaratan yang harus dipenuhi sistem dari pandangan". Dapat dilihat pada tabel 3:

Tabel 3 Simbol-simbol Use Case

Simbol	Keterangan
	Aktor: Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
	Use case: Abstraksi dan interaksi antara sistem dan aktor
	Association: Abstraksi dari penghubung antara aktor dengan use case
	Generalisasi: Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

3.10 Sequence Diagram

Menurut (Syukron, 2019) Sequence Diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang

dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Dapat dilihat pada tabel 4:

Tabel 4 Simbol-simbol Sequence Diagram

Gambar	Nama	Keterangan
	Entity Class	Gambaran sistem sebagai landasan dalam menyusun basis data
	Boundary Class	Menangani komunikasi antar lingkungan sistem
	Control Class	Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika
	Recursive	Pesan untuk dirinya
	Activation	Mewakili proses durasi aktivasi sebuah operasi
	Life Line	Komponen yang digambarkan garis putus terhubung dengan objek

3.11 Algoritma Rabin-Karp

Algoritma Rabin-Karp adalah suatu algoritma pencarian string yang ditemukan oleh Michael Rabin dan Richard Karp. Algoritma ini menggunakan hashing untuk menemukan sebuah substring dalam sebuah teks. Hashing adalah metode yang menggunakan fungsi hash untuk mengubah suatu jenis data menjadi beberapa bilangan bulat sederhana. Disebut algoritma “pencarian string” dan bukan “pencocokan string” seperti Knuth-Morris-Pratt atau Boyer Moore karena memang algoritma Rabin-Karp tidak bertujuan menemukan string yang cocok dengan string masukan, melainkan menemukan pola (pattern) yang sekiranya sesuai dengan teks masukan (Priambodo, 2018).

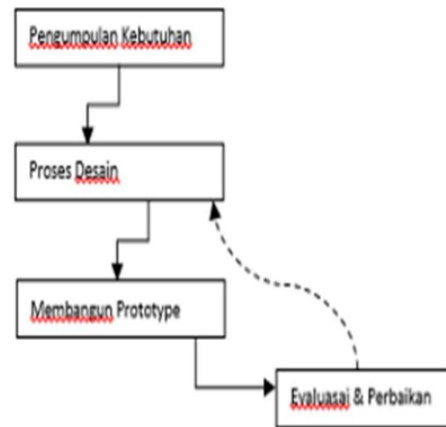


Gambar 1 Model Proses Rabin Karp

3.12 Model Prototype

Menurut Purnomo (2017), Prototyping ialah metode pengembangan perangkat lunak, yang berupa model fisik kerja sistem dan berfungsi menjadi versi awal yang berasal dari sistem. Dengan metode prototyping ini akan didapatkan prototype sistem sebagai perantara pengembang dan pengguna agar bisa berinteraksi pada proses aktivitas pengembangan sistem informasi. Agar proses pembuatan prototype ini berhasil dengan baik artinya dengan mendefinisikan aturan-

aturan di tahap awal, yaitu pengembang serta pengguna harus satu pemahaman bahwa prototype dibangun buat mendefinisikan kebutuhan awal.



Gambar 2 Langkah metode pengembangan sistem berbasis Prototype

3.13 Black Box

Pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Metode Blackbox Testing merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi (Cholifah et al., 2018).

3.14 Beta Testing

Menurut (Candra et al., 2020) Pengujian beta merupakan pengujian yang dilakukan secara objektif, Dimana pengujian dilakukan secara langsung terhadap pengguna, biasanya menggunakan kuisisioner mengenai tanggapan pengguna atas perangkat lunak yang telah dibangun. Metode penilaian pengujian yang digunakan adalah metode kuantitatif berdasarkan data dari pengguna.

$$Y = \frac{P}{Q} \times 100\%$$

Y = Nilai Persentase

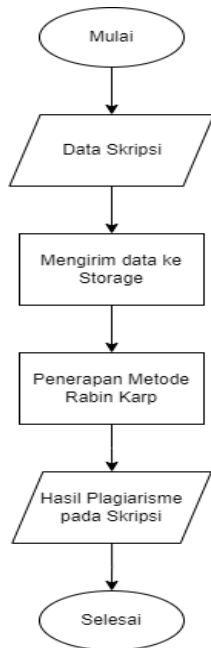
P = Banyaknya Jawaban Responden Tiap Soal

Q = Jumlah Responden

4. RANCANGAN SISTEM ATAU WEBSITE

Pada tahap rancangan sistem / website ini bertujuan untuk memberikan suatu gambaran umum tentang alur sistematis dari sistem tersebut.

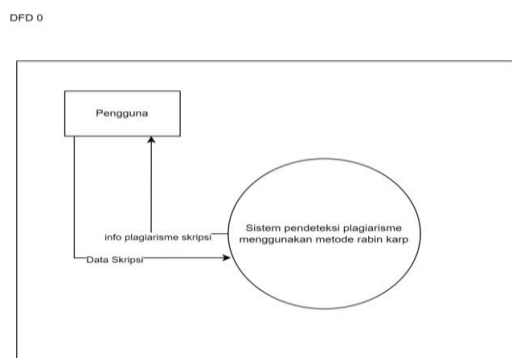
4.1 Flowchat Sistem pendeteksi plagiarisme



Gambar 3 Flowchart Sistem Pendeteksi Plagiarisme

Pada gambar 3 terdapat flowchart sistem pendeteksi plagiarisme yang dimana akan menginputkan data skripsi yang diperoleh dari internet, kemudian akan dikirimkan ke storage laptop yang digunakan, setelah itu akan diterapkan metode rabin-karp untuk memproses file skripsi tersebut. Dan setelah selesai diproses akan muncul hasil plagiarisme yang terdeteksi dalam bentuk tampilan persentase.

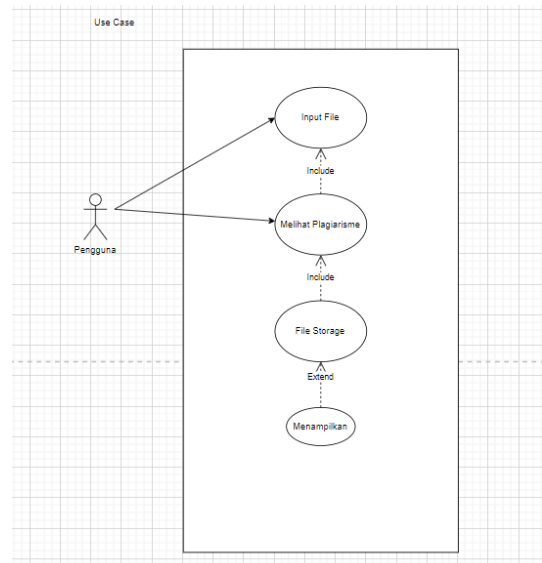
4.2 Data Flow Diagram Sistem Pendeteksi Plagiarisme



Gambar 4 DFD Sistem Pendeteksi Plagiarisme

Pada gambar 4, terdapat data flow diagram DFD 0 dari sistem pendeteksi plagiarisme yang dimana pertama pengguna akan menginputkan data skripsi ke dalam sistem tersebut, kemudian akan diproses oleh sistem pendeteksi plagiarisme menggunakan metode rabin-karp. Setelah proses selesai akan menampilkan info plagiarisme skripsi yang terdeteksi untuk pengguna.

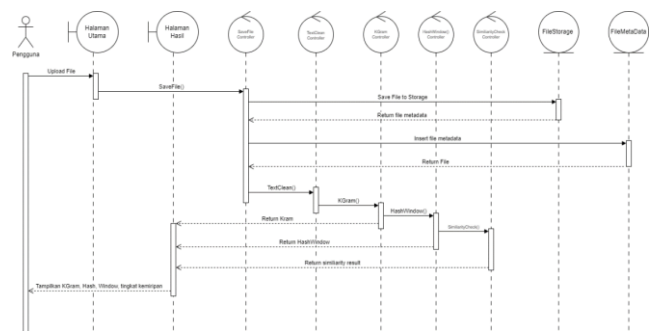
4.3 Use case Diagram Sistem Pendeteksi Plagiarisme



Gambar 5 Use Case Sistem Pendeteksi Plagiarisme

Pada gambar 5, terdapat use case diagram dari sistem pendeteksi plagiarisme, yang dimana alur kerjanya dimulai dengan adanya suatu Input File yang kemudian akan diproses melalui sistem plagiarisme. Proses ini bertujuan untuk menganalisis isi file tersebut dan mendeteksi kemungkinan adanya plagiarisme. Setelah proses analisis selesai, file akan disimpan dalam komponen File Storage. Selanjutnya, hasil analisis plagiarisme akan ditampilkan melalui komponen menampilkan.

4.4 Sequence Diagram Sistem Pendeteksi Plagiarisme



Gambar 6 Sequence Diagram Sistem Pendeteksi Plagiarisme

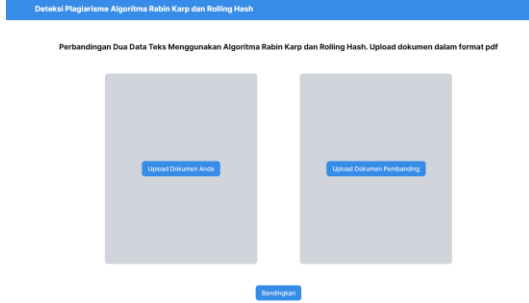
Pada gambar 6, terdapat sequence diagram yang dimana pengguna akan mengunggah file yang kemudian disimpan di penyimpanan. Setelah file disimpan, file tersebut akan diambil untuk dilakukan pembersihan teks dari karakter atau elemen yang tidak diperlukan. Setelah teks dibersihkan, teks tersebut akan diubah menjadi hash, yaitu representasi unik dari teks asli dalam bentuk kode. Selanjutnya, hash tersebut akan dibandingkan dengan hash lainnya untuk mencari kesamaan. Hasil dari proses pencarian kesamaan ini akan dikembalikan kepada

pengguna, memberikan informasi mengenai persentase plagiarisme yang terdeteksi.

5 Proses Algoritma Rabin-Karp

Pada proses algoritma rabin-karp dalam sistem pendeteksi plagiarisme ini terdapat beberapa tahapan, yaitu sebagai berikut:

5.1 Upload

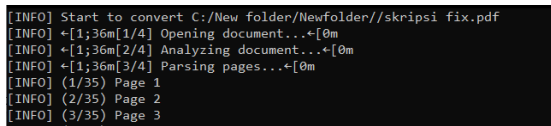


Gambar 7 Tahapan Upload

Pada gambar 7, merupakan tampilan awal dalam tahapan ini yang dimana pengguna mengupload dua file pdf dokumen skripsi yang ingin dilihat apakah terdapat plagiarisme dari kedua dokumen tersebut.

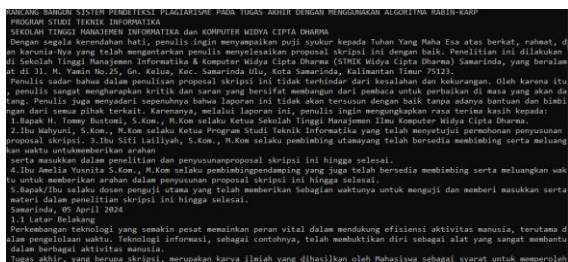
5.2 Text Pre-Processing

Pada tahap ini dilakukan text preprocessing yang dimana dilakukan berbagai proses yaitu :



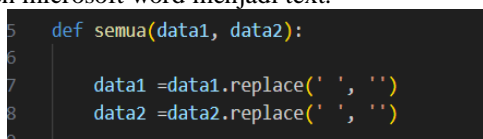
Gambar 8 Pengubahan dokumen pdf menjadi microsoft word

Pada gambar 8, menunjukkan proses pengubahan dokumen PDF menjadi Microsoft Word.



Gambar 9 Pengubahan file microsoft word menjadi text

Pada gambar 9, menunjukkan proses pengubahan dokumen microsoft word menjadi text.

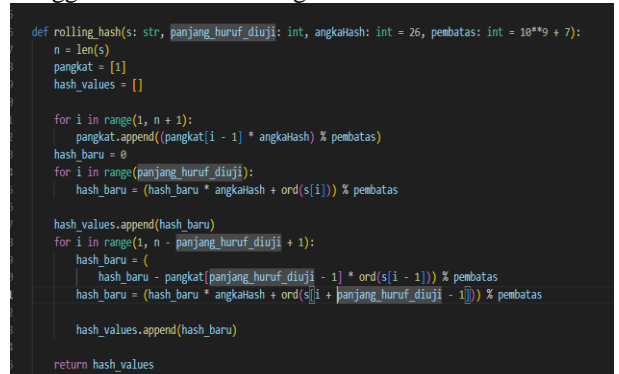


Gambar 10 Menghapus Semua Spasi

Pada gambar 10, merupakan codingan untuk mengubah semua spasi yang ada dalam file.

5.3 Hashing

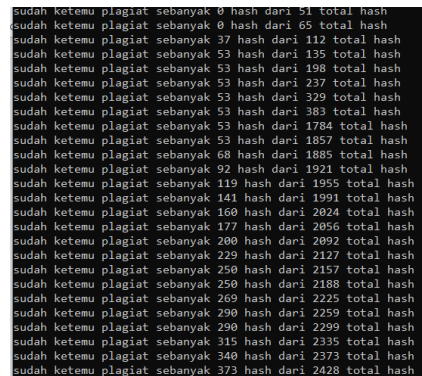
Pada tahap ini setiap kalimat akan diubah menjadi nilai numerik unik yang disebut hash dengan menggunakan rumus rolling hash.



Gambar 11 Rumus Rolling Hash

Pada gambar 11, merupakan rumus rolling hash digunakan untuk menghitung nilai hash dari setiap kalimat efisien. Ini berguna untuk pencocokan pola dan deteksi duplikasi, memungkinkan perbandingan substring yang cepat tanpa harus menghitung ulang hash dari awal setiap kali.

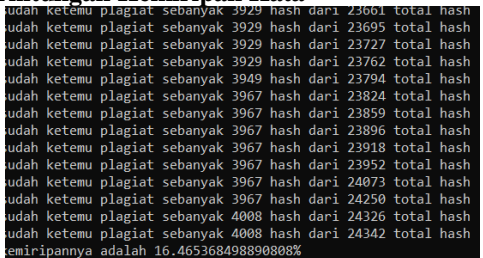
5.4 Pencocokan kata dengan algoritma rabin-karp



Gambar 12 Pencocokan Hash

Pada gambar 12, dilakukan proses pencocokan hash dari kedua dokumen yang bertujuan untuk mengukur kesamaan teks antara dua dokumen tersebut.

5.5 Perhitungan Kemiripan Kata

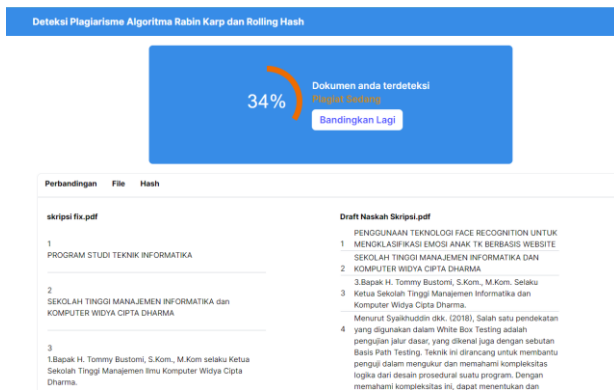


Gambar 13 Perhitungan Kemiripan Kata

Pada gambar 13, dilakukan perhitungan kemiripan kata dari semua hash yang didapatkan dari kedua dokumen tersebut. Semakin banyak nilai hash yang sama, semakin tinggi tingkat kemiripan kata.

5.6 Hasil

Algoritma ini akan menampilkan persentase kemiripan antara dua dokumen. Persentase ini menunjukkan seberapa mirip isi kedua dokumen. Semakin tinggi persentasenya, semakin mirip isi kedua dokumen.



Gambar 14 Hasil Akhir Proses Sistem

Pada gambar 14, menampilkan text dalam bentuk paragraf-paragraf dari dokumen yang di proses sistem yang nilai hashnya terdeteksi plagiat dengan dokumen pembandingnya. Hasil akhir dari proses sistem yang dimana dapat disimpulkan bahwa plagiarisme yang terdeteksi dari kedua dokumen yang telah diupload dan diproses oleh sistem adalah 34%.

6. KESIMPULAN

Berdasarkan pembahasan pada bab – bab sebelumnya dan pengujian yang dibuat, maka dapat ditarik kesimpulan sebagai berikut:

1. Sistem Pendeteksi Plagiarisme pada Tugas Akhir Dengan Algoritma Rabin-Karp telah berhasil dibuat dan dapat dipergunakan untuk mendeteksi plagiat pada skripsi mahasiswa.
2. Sistem yang dibuat mampu untuk memproses keseluruhan dokumen pada skripsi, dan mendeteksi plagiat pada skripsi mahasiswa.

7. SARAN

Saran yang dapat diambil dari proses analisis sampai pada pembuatan skripsi ini adalah sistem diharapkan dapat dikembangkan lagi dengan, seperti berikut:

1. Menambahkan halaman lain untuk bisa membaca history data-data skripsi yang pernah diupload ke dalam sistem pendeteksi tersebut.
2. Melakukan optimalisasi di kecepatan dalam mencocokkan hash dokumen skripsi yang diproses oleh sistem pendeteksi tersebut.

8. DAFTAR PUSTAKA

Candra, B., Arfyanti, I., & Harianto, K. (2020). Sistem Informasi Manajemen Turnamen Futsal Berbasis

Web Pada Rumah Futsal Melak. *Jurnal Informatika Wicida*, 9(2), 47–53. <https://doi.org/10.46984/inf-wcd.1237>

Cholifah, W. N., Sagita, S. M., & Knowledge, S. (2018). PENGUJIAN BLACK BOX TESTING PADA APLIKASI ACTION & STRATEGY BERBASIS ANDROID. 3(2), 206–210.

Hidayah, A., Indirawati, D., Agustina, D. S., & Amalia, S. (2020). Sistem Informasi Pelatihan Lisensi Penerbangan Pada Citilink Berbasis Web. *JTIM: Jurnal Teknik Informatika Mahakarya*, 03(2), 37–44.

Lubis, F., Silaban, A. F., Siregar, A. S. A. J., Girsang, A. A., Situmorang, D. N. B., Purba, G. S. A., Siregar, N. A., & Devi, T. A. (2023). Analisis Pentingnya Parafrase pada Penulisan Artikel Ilmiah Sebagai Upaya Menghindari Plagiarisme. *Jurnal Pendidikan Non Formal*, 1(2), 9. <https://doi.org/10.47134/jpn.v1i2.150>

Nelly Sofi, & Riza Dharmawan. (2022). Perancangan Aplikasi Bengkel Csm Berbasis Android Menggunakan Framework Flutter (Bahasa Dart). *Jurnal Teknik Dan Science*, 1(2), 53–64. <https://doi.org/10.56127/jts.v1i2.125>

Nurjani, Y., & Inda Aditya, M. (2023). Pendataan Jumlah Nasabah Bri Yang Ikut Serta Dalam Asuransi Anak Perusahaan Bri Dan Jumlah Nasabah Bri Yang Tidak Ikut Serta Dalam Asuransi Bri Di Kanca Muara Bulian. *FORTECH (Journal of Information Technology)*, 7(1), 32–36. <https://doi.org/10.53564/fortech.v7i1.951>

Pemrograman, A. D. A. N. (2020). Pseudocode. *Definitions*. <https://doi.org/10.32388/ft77dy>

Priambodo, J. (2018). Pendeteksian Plagiarisme Menggunakan Algoritma Rabin-Karp. *Jurnal Informatika Universitas Pamulang*, 3(1), 39–45.

Rudianto, B., & Achyani, Y. E. (2022). Rancang Bangun Sistem Informasi Simpan Pinjam Pada Koperasi Berbasis Web. *Journal of Information System, Applied, Management, Accounting and Research*, 6(1), 77. <https://doi.org/10.52362/jisamar.v6i1.669>

Satyaningrat, L. M. W., Hamijaya, P. D. N., & Rahmah, K. (2023). Analisis Pemodelan Data Flow Diagram pada Sistem Basis Data Wisata Kuliner di Kota Balikpapan. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 3(2), 236–246. <https://doi.org/10.57152/malcom.v3i2.920>

Surahman, A., Prastowo, A. T., & Aziz, L. A. (2022). Rancang Alat Keamanan Sepeda Motor Honda Beat Berbasis Sim Gsm Menggunakan Metode Rancang Bangun. *Jurnal Teknologi Dan Sistem Tertanam*, 3(1). <https://doi.org/10.33365/jtst.v3i1.1918>

Syam, M. A., Wijaya, M. A. Z., Khalisah, L. N., Nst, M. A. B. W., & Yahfizham, Y. (2024). Macam Dan Fungsi Perangkat Lunak Yang Perlu Dipahami Anak Muda Masa Kini. *Nusantara Journal of Multidisciplinary Science*, 2(1), 85–98.

Syukron, A. (2019). Perancangan Sistem Informasi Administrasi Kependudukan Desa Berbasis Website Pada Desa Winong. *Bianglala Informatika*, 7(1), 16–21. <https://doi.org/10.31294/bi.v7i1.5790>