

PENERAPAN ALGORITMA UNTUK MENDETEKSI KEMIRIPAN JUDUL SKRIPSI PADA PROGRAM STUDI INFORMATIKA

Sheva Ricko Apriansyah¹, M. Julkarnain^{*1}

¹Informatika, Rakayasa Sistem, Universitas Teknologi Sumbawa, Indonesia

m.julkarnain@uts.ac.id

ABSTRAK

Program Studi Informatika Universitas Teknologi Sumbawa mengalami kendala dalam pengecekan judul skripsi mahasiswa yang hendak mengerjakan tugas akhir skripsi, karena dilakukan dengan menggunakan sistem informasi pendaftaran skripsi berbasis *website*, namun algoritma yang dipakai masih sangat buruk (*linear*). Penelitian ini menerapkan algoritma *winnowing* dalam pengecekan judul skripsi. Algoritma ini akan mencari *fingerprint* (kesamaan di dalam dua teks) dengan cara mengubah *n-gram* dari sebuah teks menjadi bentuk numerik yang disebut dengan nilai *hash*. Teknik untuk menentukan nilai ini adalah *Hashing*. Hasil dari penelitian ini adalah sebuah aplikasi pendeteksi kemiripan judul skripsi berbasis web yang dibangun menggunakan metode pengembangan sistem *waterfall*, metode penelitian yang digunakan adalah metode penelitian kualitatif dan jenis penelitian kualitatif studi kasus, pemodelan *Unified Modelling Language* (UML) dan bahasa pemrograman PHP dengan *framework Laravel* serta *database MySQL* serta diuji menggunakan metode *Black Box Testing*.

Kata Kunci: skripsi, kemiripan judul, algoritma *winnowing*

ABSTRACT

The Informatics Study Program at the Sumbawa University of Technology experienced problems in checking the thesis titles of students who wanted to work on the final thesis assignment because it is done using a website-based thesis registration information system, but the algorithm used is still very bad (linear). This study applies the winnowing algorithm to checking the thesis title. This algorithm will look for a fingerprint (similarity in two texts) by converting n-grams of a text into a numeric form called a hash value. The technique for determining this value is Hashing. The result of this research is a web-based thesis title similarity detector application that was built using the waterfall system development method, the research method used is a qualitative research method and a type of qualitative case study research, Unified Modeling Language (UML) modeling, and the PHP programming language with the Laravel framework and MySQL database and tested using the Black Box testing method..

Keywords: *thesis, title similarity, winnowing algorithm*

PENDAHULUAN

Menurut Undang-Undang Dasar Negara Republik Indonesia No. 20 Tahun 2003 tentang Sistem Pendidikan Nasional, bahwasanya kaidah pendidikan tinggi di Indonesia ialah subsistem dari kaidah pendidikan tinggi. Setiap perguruan tinggi harus memiliki gelar

Strata 1 (S1), adalah dengan menyudahkan tugas akhir berupa skripsi sepikiran dengan Peraturan Pemerintah Republik Indonesia No. 60 tahun 1999 yaitu ujian akhir program sarjana, yaitu dengan menyelesaikan ujian skripsi. Untuk memulai, mahasiswa harus menemukan topik penelitian yang akan mereka pelajari. Untuk melakukan ini, mereka dapat melakukan banyak hal, seperti membaca jurnal penelitian lokal, nasional, dan internasional, mengikuti penelitian yang dilakukan oleh dosen mereka, dan membaca kumpulan tesis mahasiswa terdahulu, dan mengamati permasalahan yang timbul baik di kampus maupun diluar kampus (Astuti, 2017).

Pada Universitas Teknologi Sumbawa khususnya di Fakultas Rekayasa Sistem Program Studi Informatika, sebelum mahasiswa mengambil tugas akhir skripsi mahasiswa harus melewati alur pengajuan skripsi terlebih dahulu. Pengajuan skripsi ini dimulai dengan, pertama mengecek syarat dari pengambilan skripsi sesuai dengan Peraturan Rektor no. 10 tahun 2020, kedua mahasiswa menyiapkan berkas pendaftaran skripsinya, ketiga mengisi form pendaftaran skripsi dari Program Studi, keempat menunggu pembagian dosen pembimbing, kelima melakukan konsultasi judul skripsi dengan dosen pembimbing. Menurut Zakaria (dalam Mutadin, 2002) saat mahasiswa mengerjakan skripsi, banyak kendala yang dihadapi, seperti: (1) Mencari judul skripsi; (2) Proses pengumpulan data dan kajian literature yang lama; (3) Kesulitan mengungkapkan gagasan secara tertulis; (4) Kesulitan membagi waktu antara skripsi dan kegiatan lain, seperti bekerja; (5) Takut bertemu dengan dosen pembimbing. Pada kendala pertama diatas, mahasiswa harus mencari judul skripsi dan melakukan konsultasi judul skripsi kepada dosen pembimbing. Bagian ini menjadi bagian yang akan menentukan tema ataupun topik yang akan diteliti oleh mahasiswa. Setelah mengkonsultasikan judul skripsi yang akan diteliti, barulah mahasiswa akan memulai mengerjakan skripsi yang sesuai dengan pedoman mengerjakan skripsi. Akan tetapi beberapa mahasiswa terkendala pada seminar proposal karena tema penelitian yang diajukan sama dengan tema skripsi yang telah diteliti oleh mahasiswa terdahulu, yang mengakibatkan mahasiswa harus merubah judul skripsi yang akan diteliti.

Saat ini aplikasi pendeteksi kemiripan judul skripsi pada Program Studi Informatika sudah tersedia, namun belum mumpuni. Aplikasi tersebut dibuat oleh Olvira Nanda mahasiswi angkatan 2019, aplikasi yang dibuat belum diimplementasi ke web dan algoritma yang dipakai masih sangat jelek (linear). Sehingga membuat mahasiswa kesulitan dalam menentukan judul skripsi karena adanya persamaan judul skripsi dengan yang terdahulu.

Peneliti bermaksud menyediakan aplikasi untuk mempermudah Program Studi Informatika dalam memilah judul skripsi yang belum ada ataupun yang sudah ada, agar dosen pembimbing mudah untuk menyetujui judul skripsi tanpa khawatir sudah adanya judul skripsi yang telah diteliti oleh mahasiswa terdahulu. Mahasiswapun dipermudah karena tidak terhambat untuk mengerjakan skripsi hanya karena judul skripsi yang telah ada sebelumnya. Oleh karena itu diperlukan suatu aplikasi yang dapat dengan cepat dan akurat menentukan persentase kemiripan suatu judul. Dalam sistem ini, Sekretaris Program Studi nantinya hanya tinggal memasukkan judul yang akan dikirim sesuai formulir yang diberikan dan sistem akan mengecek dan menampilkan hasilnya secara otomatis. Hasil ini dapat diperhitungkan saat memutuskan apakah akan diterima atau ditolak judul skripsi mahasiswa tersebut.

Winnowing adalah algoritma yang digunakan untuk proses *document fingerprinting* (Schleimer et al., 2003). Tujuan proses ini adalah untuk menemukan penjiplakan, termasuk komponen kecil yang serupa yang ditemukan dalam sejumlah besar dokumen. *File teks* adalah input dari proses *fingerprinting* dokumen, dan sekumpulan nilai hash yang disebut *fingerprint* berfungsi sebagai dasar untuk membandingkan file teks yang telah dimasukkan. *Rolling hash* adalah algoritma yang digunakan untuk menghitung nilai *hash* dalam *winnowing*. Algoritma *Winnowing* menghapus karakter yang tidak relevan (seperti spasi, tanda baca, dan karakter lain) untuk memenuhi persyaratan *insensitivity whitespace* dari algoritma pendeteksi plagiarisme. Ini berarti hanya karakter yang berupa huruf atau angka yang akan diproses lebih lanjut.

Menurut (Fauzi & Wibawa, 2018) algoritma dokumen *fingerprint*, *winnowing*, melakukan proses pengecekan kesamaan kata dan menghasilkan kumpulan nilai hash yang dibangun dari perhitungan ASCII untuk setiap karakter. Algoritma ini disebut *fingerprint*, dan digunakan untuk memeriksa plagiarisme. Proses algoritma *Winnowing* meliputi penghapusan karakter yang tidak relevan, pembentukan rangkaian *n-gram*, penghitungan *hash (rolling hash)* untuk setiap *gram*, pembentukan *window*, pemilihan *hash fingerprint* terkecil dari setiap layar, dan penghitungan persamaan dengan *koefisien Jaccard*.

Metode *waterfall*, atau metode air terjun, sering disebut sebagai siklus hidup klasik. Pendekatan sistematis dan berurutan untuk pengembangan perangkat lunak digambarkan oleh model ini, yang sebenarnya disebut sebagai "Model Sequential Linear". Pendekatan ini dimulai dengan spesifikasi kebutuhan pengguna dan kemudian melewati fase perencanaan

(planning), permodelan (modelling), konstruksi (contruction), dan penyerahan sistem ke pengguna (deployment). Akhir dari fase ini adalah dukungan untuk perangkat lunak lengkap yang dihasilkan (Sudiarta & Akmalludin, 2018).

Penelitian kualitatif adalah deskriptif dan menggunakan analisis. Proses dan pemaknaan adalah hal yang paling penting dalam penelitian ini, dan dasar teori yang dibangun berdasarkan temuan lapangan akan menjadi dasar penelitian. Karena hasil penelitian kualitatif memerlukan analisis mendalam, peneliti harus terlibat langsung dengan kondisi atau peristiwa yang diteliti. Penelitian kualitatif biasanya mengumpulkan data melalui observasi dan wawancara. Setelah itu, peneliti menganalisis data tersebut untuk menghasilkan konsep atau teori baru jika temuan penelitian bertentangan dengan teori yang sudah ada (Aulia et al., 2023).

Desain sistem yang digunakan peneliti adalah *Unified Modelling Language* (UML). Menurut Suendri (2019) dalam (Windu dan Grace, (2013)), *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, mendefinisikan dan membuat perangkat lunak. UML adalah metodologi untuk mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.

PHP adalah bahasa *scripting* yang dieksekusi di server dan digunakan untuk membuat halaman web dinamis seperti *guestbook*, statistik pengunjung, *polling*, *email*, dan lainnya (Dwi Krisbiantoro & Prih Diantono Abda'u, 2021).

Laravel adalah framework aplikasi web berbasis PHP open source yang menggunakan konsep *Model-View-Controller* (MVC) (Jannah & Rahardi, 2021).

MySQL adalah perangkat lunak sistem manajemen basis data SQL atau DBMS yang multithread dan multiuser yang mendukung bahasa database SQL sebagai bahasa manajemen data interaktif. MySQL juga merupakan database engine atau server database yang mendukung bahasa database SQL sebagai bahasa manajemen data interaktif (Rahimi Fitri, 2020).

Black-box didasarkan sepenuhnya pada spesifikasi program, dan tidak menggunakan pengetahuan tentang cara kerja bagian dalam kode program. pengujian black-box bertujuan untuk memverifikasi bahwa program memenuhi persyaratan yang ditentukan, tanpa pengetahuan tentang implementasi program (Bierig et al., 2021).

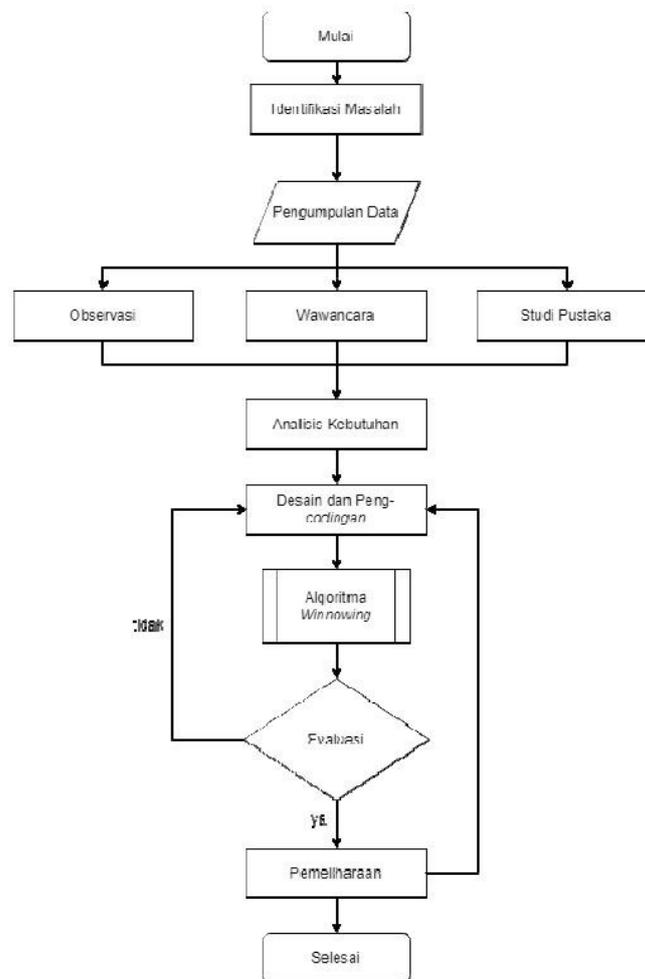
Penulis melakukan penelusuran literatur tentang berbagai penelitian dan menemukan beberapa penelitian terkait sebagai pendukung penelitian ini. Penelitian pertama dilakukan oleh (Rihi et al., 2022) yang terdapat dua *workflow* sistem yaitu sistem lama dan baru. *Workflow* sistem baru dimulai dari mahasiswa yang mengajukan judul skripsi ke sistem, kemudian sistem akan mendeteksi kesamaan judul skripsi menggunakan algoritma *winnowing*, dan menunjukkan persentase kemiripan judul skripsi yang diajukan sehingga kaprodi dan admin dapat membuat keputusan terkait menerima judul mahasiswa atau tidak. Penelitian kedua dilakukan oleh (Putra Bayu Pratama et al., 2021) Persentase kemiripan judul terhadap data judul skripsi yang ada pada perguruan tinggi adalah salah satu kriteria dalam penerimaan judul skripsi yang diajukan. Perhitungan persentase kemiripan ini relatif sulit dilakukan tanpa adanya piranti pendukung seperti sistem komputer mengingat jumlah data judul yang dibandingkan cukup banyak. Pencarian data judul skripsi yang ada pada perguruan tinggi dilakukan dengan mencari buku skripsi pada perpustakaanannya, ini membuat pencarian data memakan waktu relatif lama dan informasi yang didapat untuk suatu tema atau objek penelitian tertentu lebih sedikit dari jumlah data aktualnya karena tidak memungkinkan untuk menfilter data fisik seperti buku skripsi. Dengan hasil bahwa algoritma *winnowing* dapat digunakan untuk menghitung tingkat kemiripan kata dengan mencari kesamaan *fingerprint* dalam kalimat yang dibandingkan. Penelitian ketiga dilakukan oleh (Islamiyati & Fikri, 2022) yang berjudul Penerapan Algoritma *Knuth-Morris-Pratt* dalam Mendeteksi Tingkat Kemiripan Judul Skripsi Berbasis Web, salah satu upaya yang dapat membantu dalam hal pengecekan kemiripan dua judul skripsi adalah by system dengan pencocokan string (*string matching*) pada teks yang terdapat pada judul Skripsi/TA. *String matching* adalah mencari sebuah string yang terdiri dari beberapa karakter yang biasa disebut (*pattern*) dalam jumlah besar [1]. Terdapat beberapa algoritma pada *string matching* ini diantaranya Brute Force, *Winnowing*, *Knuth-Morris-Pratt*, *Booyer-Moore*, *Rabin-Karip* dan sebagainya. Dimana hasil menunjukkan bahwa aplikasi bekerja dengan lancar dan menampilkan hasil analisis berupa jumlah kata yang mirip dan persentasenya. Fitur-fitur yang disediakan yaitu mengelompokkan user menjadi 3 diantaranya admin, staff, dan mahasiswa. Penelitian keempat dilakukan oleh (Setyawan et al., 2022) Mahasiswa yang akan membuat skripsi terlebih dulu harus mengajukan judul skripsi dan selanjutnya diikuti proposal skripsi. Proposal skripsi tersebut diajukan ke pengelola program studi untuk dinilai, sejauh mana kelayakannya untuk menjadi skripsi nantinya. Salah satu yang menjadi unsur penilaian adalah sejauh mana proposal skripsi tersebut memiliki kemiripan dengan proposal-

proposal yang lain. Hal ini dilakukan agar skripsi mahasiswa yang dibuat nantinya tidak sekedar mengulang skripsi yang sudah ada, atau skripsi berbeda dengan skripsi lain, namun masih terlalu banyak kemiripannya. Algoritma *K-Means* dan *Winowing* menunjukkan bahwa dapat digunakan untuk menghitung kemiripan sebagian isi proposal skripsi dan dapat digunakan dalam sistem klusterisasi. Penelitian kelima dilakukan oleh (Fauzi & Wibawa, 2018) Tidak sedikit mahasiswa yang bingung dalam pengajuan tersebut dikarenakan mahasiswa tersebut tidak mengetahui bahwa proposal yang akan diajukan itu telah ada atau tidak sebelumnya, judul yang sama tidak menutup kemungkinan isi dari judul tersebut juga sama, akan tetapi isi juga dapat berbeda walaupun dengan judul yang sama. Dan jika isi itu sama, mahasiswa akan diduga melakukan plagiat dan akan memungkinkan bahwa tugas akhir yang diajukan tersebut harus diubah dikemudian hari. Dimana aplikasi pendeteksian plagiat atau kemiripan teks yang telah dirancang dapat digunakan tanpa harus menggunakan biaya, terbuka bagi mahasiswa sehingga dapat menggunakan untuk melihat hasil pengecekan langsung terkait kemiripan yang telah ditentukan.

Berdasarkan latar belakang masalah yang telah dijelaskan diatas, maka dapat diperoleh sebuah solusi yaitu menerapkan algoritma *winnowing* untuk mendeteksi kemiripan judul skripsi berbasis web yang efisien pada Program Studi Informatika Universitas Teknologi Sumbawa.

METODOLOGI PENELITIAN

Metode penelitian yang digunakan pada penelitian ini yaitu metode kualitatif. Teknik pengumpulan data pada penelitian ini dilakukan melalui observasi langsung yang dilakukan di Program Studi Informatika Universitas Teknologi Sumbawa, wawancara dengan Sekretaris Program Studi Informatika, dan melalui studi Pustaka dengan membaca buku, jurnal dan hasil penelitian dari peneliti lain yang mempunyai keterkaitan dengan penelitian yang dilakukan. Berikut beberapa tahapan- tahapan dalam penelitian ini:



Gambar 1. Alur Penelitian

Peneliti memulai dengan menentukan masalah yang ada terkait sistem yang sedang berjalan beserta dukungan studi pustaka, observasi dan wawancara untuk dapat membantu mengidentifikasi masalah utamanya, dimana dalam hal masalahnya utamanya adalah cara mendeteksi kemiripan judul skripsi pada Program Studi Informatika Universitas Teknologi Sumbawa. Selanjutnya analisa kebutuhan yaitu penulis mencoba melakukan analisa kebutuhan perangkat keras maupun perangkat lunak untuk dapat membantu menyelesaikan aplikasi yang akan dibuat. Pada tahap desain dan pengcodingan, peneliti melakukan desain terhadap tampilan utama dari aplikasi agar lebih menarik, setelah melakukan desain maka akan dilakukan pengcodingan dari desain yang telah dibuat. Pada tahap ini juga akan diterapkan algoritma yang digunakan yaitu algoritma *wInnowing*. Algoritma *WInnowing* mencakup beberapa langkah yang dilakukan untuk mendapatkan hasil berupa document *fingerprinting*. Tahap selanjutnya adalah evaluasi dengan melakukan pengujian terhadap aplikasi yang telah dibuat. Pengujian dilakukan oleh para ahli

dibidangnya yang akan menjadi diterima atau tidaknya aplikasi yang dibuat. Tahap terakhir ialah pemeliharaan, dimana peneliti akan melakukan pemeliharaan pada aplikasi yang telah dibuat berupa perbaikan kesalahan dan penyesuaian aplikasi sesuai kebutuhan.

HASIL DAN PEMBAHASAN

Tahapan-tahapan untuk mengimplementasikan algoritma *winnowing* untuk mendeteksi presentase kemiripan judul skripsi adalah sebagai berikut: Ambil dua judul skripsi yang akan dibandingkan

Teks judul 1:

Sistem Informasi Akademik SMAN 1 Moyo Utara Berbasis Web

Teks judul 2

Sistem Informasi Akademik SMK Negeri 3 Sumbawa Besar Berbasis Web

Pembuangan karakter yang tidak relevan

Hapus semua huruf yang bukan A-Z dan a-z. kemudian ubah menjadi huruf kecil semua.

Teks judul 1 akan berbentuk menjadi:

sisteminformasiakademiksmanmoyoutaraberbasisweb

Teks judul 2 akan berbentuk menjadi:

sisteminformasiakademiksmknegerisumbawabe sarberbasisweb

Pembentukan rangkaian n-gram

Pembentukan rangkaian gram dari teks yang telah dibersihkan dengan ukuran banyak karakter 7:

Pada teks judul 1 akan terbentuk 41 rangkaian n-gram yaitu:

sistemi istemin steminf teminfo eminfor minform informa nformas formasi ormasia
rmasiak masiaka asiakad siakade iakadem akademi kademik ademiks demiksm emiksm
miksmn iksmanm ksmnmo smanmoy manmoyo anmoyou nmoyout moyouta oyoutar
youtara outarab utarabe taraber araberb raberba aberbas berbasi erbasis rbasisw basiswe
asisweb

Sedangkan pada teks judul 2 terbentuk 49 rangkaian n-gram yaitu:

sistemi istemin steminf teminfo eminform minform informa nformas formasi ormasia
 rmasiak masiaka asiakad siakade iakadem akademi kademik ademiks demiksm emiksmk
 miksmkn iksmkne ksmkneg smknege mkneger knegeri negeris egerisu gerisum erisumb
 risumba isumbaw sumbawa umbawab mbawabe bawabes awabesa wabesar abesarb
 besarbe esarber sarberb arberba rberbas berbasi erbasis rbasisw basiswe asisweb

Perhitungan fungsi *hash* untuk setiap *n-gram*

Penghitungan nilai-nilai *hash* dari setiap *gram* menggunakan rolling hash dengan basis 2.

Hasil semua perhitungan nilai hash pada teks judul 1 yaitu:

14215, 13820, 14302, 13995, 13256, 13693, 13531, 13737,
 13499, 14039, 13977, 13459, 13066, 13817, 13023, 12711,
 13113, 12645, 12983, 13263, 13708, 13573, 13817, 14059,
 13509, 13183, 14066, 14149, 14460, 14809, 14228, 14349,
 13836, 12922, 13525, 12573, 12835, 13241, 13673, 12855,
 13264

Hasil semua perhitungan nilai hash pada judul 2 yaitu:

14215, 13820, 14302, 13995, 13256, 13693, 13531, 13737,
 13499, 14039, 13977, 13459, 13066, 13817, 13023, 12711,
 13113, 12645, 12983, 13273, 13728, 13605, 13873, 14151,
 13696, 13545, 13509, 13055, 13591, 13496, 14161, 13849,
 14355, 14088, 13301, 12765, 13083, 13864, 12594, 12873,
 13316, 13802, 12981, 13661, 12835, 13241, 13673, 12855,
 13264

Pembentukan Window dari Nilai Hash

Bentuk window dari nilai-nilai hash dengan ukuran 4:

Window teks 1

Window Teks 2

W-1 = {14215, 13820, 14302, 13995}
 W-2 = {13820, 14302, 13995, 13256}
 W-3 = {14302, 13995, 13256, 13693}
 W-4 = {13995, 13256, 13693, 13531}
 W-5 = {13256, 13693, 13531, 13737}
 W-6 = {13693, 13531, 13737, 13499}
 W-7 = {13531, 13737, 13499, 14039}
 W-8 = {13737, 13499, 14039, 13977}
 W-9 = {13499, 14039, 13977, 13459}
 W-10 = {14039, 13977, 13459, 13066}
 W-11 = {13977, 13459, 13066, 13817}
 W-12 = {13459, 13066, 13817, 13023}
 W-13 = {13066, 13817, 13023, 12711}
 W-14 = {13817, 13023, 12711, 13113}
 W-15 = {13023, 12711, 13113, 12645}
 W-16 = {12711, 13113, 12645, 12983}
 W-17 = {13113, 12645, 12983, 13273}
 W-18 = {12645, 12983, 13263, 13708}
 W-19 = {12983, 13263, 13708, 13573}

W-20 = {13263, 13708, 13573, 13817}
 W-21 = {13708, 13573, 13817, 14059}
 W-22 = {13573, 13817, 14059, 13509}
 W-23 = {13817, 14059, 13509, 13183}
 W-24 = {14059, 13509, 13183, 14066}
 W-25 = {13509, 13183, 14066, 14149}
 W-26 = {13183, 14066, 14149, 14460}
 W-27 = {14066, 14149, 14460, 14809}
 W-28 = {14149, 14460, 14809, 14228}
 W-29 = {14460, 14809, 14228, 14349}
 W-30 = {14809, 14228, 14349, 13836}
 W-31 = {14228, 14349, 13836, 12922}
 W-32 = {14349, 13836, 12922, 13525}
 W-33 = {13836, 12922, 13525, 12573}
 W-34 = {12922, 13525, 12573, 12835}
 W-35 = {13525, 12573, 12835, 13241}
 W-36 = {12573, 12835, 13241, 13673}
 W-37 = {12835, 13241, 13673, 12855}
 W-38 = {13241, 13673, 12855, 13264}

W-24 = {14151, 13696, 13545, 13509}
 W-25 = {13696, 13545, 13509, 13055}
 W-26 = {13545, 13509, 13055, 13591}
 W-27 = {13509, 13055, 13591, 13496}
 W-28 = {13055, 13591, 13496, 14161}
 W-29 = {13591, 13496, 14161, 13849}
 W-30 = {13496, 14161, 13849, 14355}
 W-31 = {14161, 13849, 14355, 14088}
 W-32 = {13849, 14355, 14088, 13301}
 W-33 = {14355, 14088, 13301, 12765}
 W-34 = {14088, 13301, 12765, 13083}
 W-35 = {13301, 12765, 13083, 13864}
 W-36 = {12765, 13083, 13864, 12594}
 W-37 = {13083, 13864, 12594, 12873}
 W-38 = {13864, 12594, 12873, 13316}
 W-39 = {12594, 12873, 13316, 13802}
 W-40 = {12873, 13316, 13802, 12981}
 W-41 = {13316, 13802, 12981, 13661}
 W-42 = {13802, 12981, 13661, 12835}
 W-43 = {12981, 13661, 12835, 13241}
 W-44 = {13661, 12835, 13241, 13673}
 W-45 = {12835, 13241, 13673, 12855}

W-1 = {14215, 13820, 14302, 13995}
 W-2 = {13820, 14302, 13995, 13256}
 W-3 = {14302, 13995, 13256, 13693}
 W-4 = {13995, 13256, 13693, 13531}
 W-5 = {13256, 13693, 13531, 13737}
 W-6 = {13693, 13531, 13737, 13499}
 W-7 = {13531, 13737, 13499, 14039}
 W-8 = {13737, 13499, 14039, 13977}
 W-9 = {13499, 14039, 13977, 13459}
 W-10 = {14039, 13977, 13459, 13066}
 W-11 = {13977, 13459, 13066, 13817}
 W-12 = {13459, 13066, 13817, 13023}
 W-13 = {13066, 13817, 13023, 12711}
 W-14 = {13817, 13023, 12711, 13113}
 W-15 = {13023, 12711, 13113, 12645}
 W-16 = {12711, 13113, 12645, 12983}
 W-17 = {13113, 12645, 12983, 13273}
 W-18 = {12645, 12983, 13263, 13708}
 W-19 = {12983, 13263, 13708, 13573}

Pemilihan *Fingerprint* dari Setiap Window lalu memilih nilai *hash* terkecil dari setiap *window* untuk dijadikan sebagai *fingerprint*, apabila ada nilai *hash* yang sama dengan yang sebelumnya maka nilai *hash* tersebut tidak akan ditulis:

Fingerprint teks 1

[13820, 1]	[13256, 2]	[13499, 6]	[13459, 9]
[13066, 10]	[13023, 12]	[12711, 13]	[12645, 16]
[12983, 19]	[13263, 20]	[13573, 21]	[13509, 22]
[13183, 23]	[14066, 27]	[14149, 28]	[14228, 29]
[13836, 30]	[12922, 31]	[12573, 33]	[12835, 37]
[12855, 38]			

Fingerprint teks 2

[13820, 1]	[13256, 2]	[13499, 6]	[13459, 9]
[13066, 10]	[13023, 12]	[12711, 13]	[12645, 16]
[12983, 19]	[13273, 20]	[13605, 21]	[13545, 23]
[13509, 24]	[13055, 25]	[13496, 29]	[13849, 31]
[13301, 32]	[12765, 33]	[12594, 36]	[12873, 40]
[12981, 41]	[12835, 42]	[12855, 46]	

Jaccard's Similarity Coefficient

Tahap yang akhir adalah menghitung presentasi kemiripan (*similarity*) dari sebuah teks dilakukan menggunakan *Jaccard's Similarity Coefficient*, dimana nilai *hash* dari teks 1 adalah: {13820, 13256, 13499, 13459, 13066, 13023, 12711, 12645, 12983, 13263, 13573, 13509, 13183, 14066, 14149, 14228, 13836, 12922, 12573, 12835, 12855} dan teks 2 : {13820, 13256, 13499, 13459, 13066, 13023, 12711, 12645, 12983, 13273, 13605, 13545, 13509, 13055, 13496, 13849, 13301, 127655, 12594, 12873, 12981, 12835, 12855}.

Nilai $|H d1 \cap H d2| = \{13820, 13256, 13499, 13459, 13066, 13023, 12711, 12645, 12983, 12835, 12855\}$ dan nilai $|H d1 \cup H d2| = \{13820, 13256, 13499, 13459, 13066, 13023, 12711, 12645, 12983, 13263, 13573, 13509, 13183, 14066, 14149, 14228, 13836, 12922, 12573, 13273, 13605, 13545, 13509, 13055, 13496, 13849, 13301, 127655, 12594, 12873, 12981, 12835, 12855\}$. Sehingga

$$\text{Similarity}(d_1, d_2) = \frac{11}{33} \times 100\% = 33\%$$

atau

$$\text{Similarity} = \frac{11}{44 - 11} \times 100\% = 33\%$$

Dari proses tersebut dapat dilihat bahwa teks 1 dan teks 2 memiliki persamaan sebesar 33%. Algoritma *winnowing* dapat menghitung tingkat kemiripan kata dengan mencari kesamaan dari *fingerprint*. Sebelum menggunakan algoritma *winnowing*, peneliti terlebih dahulu menentukan banyak karakter, basis, dan *window*. Penentuan berupa angka dari banyak karakter, basis, dan *window*, akan mempengaruhi nilai presentase kemiripan judul skripsi. Setelah itu barulah menggunakan algoritma *winnowing* sesuai dengan tatapan-tahapan yang berlaku.

KESIMPULAN DAN SARAN

Dari hasil analisis dan perancangan yang dilakukan sebelumnya, penerapan algoritma *winnowing* untuk mendeteksi kemiripan judul skripsi pada Program Studi Informatika ini

telah berhasil dan selesai dibangun menggunakan pemrograman PHP dengan *framework Laravel* dan *database MySQL*, serta telah diuji secara fungsional. Aplikasi pendeteksi kemiripan judul skripsi ini dibangun dengan tujuan untuk mempermudah mahasiswa untuk mengambil judul skripsi yang akan diteliti sekaligus mempermudah program studi dalam menentukan diterima atau tidaknya judul skripsi guna menghindari judul skripsi yang sudah diambil oleh mahasiswa terdahulu. Pengguna aplikasi dapat melakukan pencarian informasi mengenai judul skripsi sekaligus menguji kemiripan judul skripsi, dengan begitu pengguna dapat mengetahui berapa presentase dari kemiripan judul skripsi yang tengah diuji. Algoritma *winnowing* dapat menghitung tingkat kemiripan kata dengan mencari kesamaan dari *fingerprint*. Dengan hasil perhitungan algoritma *winnowing*, pengguna terutama mahasiswa dapat terbantu dalam menentukan judul skripsi yang akan diajukan ke program studi.

Adapun saran untuk penelitian selanjutnya antara lain penambahan data judul skripsi mahasiswa 2019 sampai yang terbaru, penambahan fitur-fitur yang digunakan pada fitur *menu* berupa *edit* dan *hapus*, aplikasi dapat dikembangkan berbasis *android*, dan dapat dikembangkan untuk document teks yang lebih banyak atau kompleks, tidak hanya judul skripsi tapi dapat diterapkan untuk dokumen utuh skripsi atau jurnal penelitian.

DAFTAR PUSTAKA

- Astuti, W. (2017). Analisis String Matching Pada Judul Skripsi Dengan Algoritma Knuth-Morris Pratt (Kmp). *ILKOM Jurnal Ilmiah*, 9(2), 167–172. <https://doi.org/10.33096/ilkom.v9i2.136.167-172>
- Aulia, T., Salman, M., & Lubis, N. K. (2023). Kecendrungan Penelitian Minat Zakat Di Jurnal Google Scholar Tahun 2022. *MUQADDIMAH: Jurnal Ekonomi, Manajemen, Akuntansi Dan Bisnis*, 1(1), 178–189. <https://doi.org/10.21111/IEJ.V2I1.970>
- Bierig, R., Brown, S., Galván, E., & Timoney, J. (2021). *Essentials of Software Testing*. Cambridge University Press. <https://books.google.co.id/books?id=zQg3EAAAQBAJ>
- Dwi Krisbiantoro, M. K., & Prih Diantono Abda' u, M. K. (2021). *Dasar Pemrograman Web dengan bahasa HTML, PHP, dan Database MySQL*. Zahira Media Publisher. <https://books.google.co.id/books?id=hGEzEAAAQBAJ>
- Fauzi, R. M., & Wibawa, J. C. (2018). *Implementasi Algoritma Winnowing untuk Mendeteksi Kemiripan Teks pada Artikel*. Universitas Komputer Indonesia.
- Islamiyati, D. S., & Fikri, A. (2022). Penerapan Algoritma Knuth-Morris-Pratt dalam Mendeteksi Tingkat Kemiripan Judul Skripsi Berbasis Web. *Journal of Information System Research (JOSH)*, 3(2), 58–63. <https://doi.org/10.47065/josh.v3i2.1168>

- Jannah, I. N., & Rahardi, R. G. (2021). *Play with Laravel - UDACODING: Seri 1*. UDACODING. <https://books.google.co.id/books?id=NFMfEAAAQBAJ>
- Putra Bayu Pratama, N., Mustaqiem, & Minarni. (2021). Pengembangan Sistem Informasi Pengelolaan Judul Skripsi dan Tugas Akhir dengan Fitur Deteksi Kemiripan Menggunakan Algoritma Winnowing. *Terapan Informatika Nusantara*, 2(5), 271–278.
- Rahimi Fitri, S. K. M. K. (2020). *Pemrograman Basis Data Menggunakan MySQL*. Deepublish. <https://books.google.co.id/books?id=y9kZEAAAQBAJ>
- Rihi, M. P., Pekuwali, A. A., & Sitaniapessy, D. A. (2022). *Winnowing Algorithm for Detecting the Similarity of Informatic Engineering Undergraduate Thesis Title Algoritma Winnowing untuk Mendeteksi Kesamaan Judul Skripsi Teknik Informatika*. 2(October), 42–52.
- Schleimer, S., Wilkerson, D. S., & Aiken, A. (2003). Winnowing: Local Algorithms for Document Fingerprinting. *Proceedings of the ACM SIGMOD International Conference on Management of Data*.
- Setyawan, D., Coefficient, J., Komputer, I., Widya, U., & Klaten, D. (2022). *Implementasi Algoritma Winnowing dan K-Means*. 2, 17–26.
- Sudiarta, I. W., & Akmalludin, R. F. (2018). *Pemrograman C untuk penyelesaian kasus*. Penerbit Periuk. <https://books.google.co.id/books?id=eeSpDwAAQBAJ>
- Suendri, S. (2019). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). *Algoritma: Jurnal Ilmu Komputer Dan Informatika*, 2(2), 1.
- Zakaria, D. (2017). Tingkat Stres Mahasiswa Ketika Menempuh Skripsi. *Fakultas Psikologi Universitas Muhammadiyah Malang*, 1–45.
https://eprints.umm.ac.id/43367/1/jiptummp_p-gdl-dhickyza-47398-1-naskahp-i.pdf