

Perancangan Fitur Deteksi Kemiripan Dokumen Jawaban Tugas Mahasiswa pada Sistem Manajemen Pembelajaran dengan Metode *K-Shingling* dan *Cosine Similarity*

Komang Nova Artawan¹, Rukmi Sari Hartati², Yoga Divayana³, Made Sudarma⁴

[Submission: 15-12-2022, Accepted: 09-02-2023]

Abstract— Technology development can be used to support the teaching and learning process so it can runs more effectively and efficiently, and one of the applications that can be used to support this process is the Learning Management System (LMS). LMS provides an integrated platform starting from delivering learning materials to students, and evaluating learning process. The evaluation method that can be done at the LMS is giving assignments to students, and students can do submission by uploading their documents assignment answer. The common problem is many students do plagiarism by duplicate the answers of other students. To overcome these problems, this study will implement feature in Course Assignment Menu in LMS to detect the similarity of student assignment document using *K-Shingling* and *Cosine Similarity* methods. The *K-Shingling* algorithm is used to form *shingles* or word fragments based from all text in the task answer document that has gone through the *pre processing* stage, then each document have *shingle* value vector which will be used to compare the similarity values between documents using *Cosine Similarity*. The results obtained in this study is the developed feature is capable to detecting the similarity percentage of the assignment answer documents from each student, and from that percentage value it can indicate whether the students assignment answers is plagiarism to other students assignment answers.

Keyword : LMS; plagiarism; *K-Shingling*; *Cosine Similarity*

Intisari— Perkembangan teknologi dapat digunakan untuk mendukung proses belajar mengajar agar berjalan dengan lebih efektif dan efisien, salah satu aplikasi yang dapat digunakan untuk menunjang hal tersebut adalah Sistem Manajemen Pembelajaran atau yang sering dikenal dengan LMS (*Learning Management System*). LMS menyediakan wadah yang terintegrasi dimulai dari penyampaian materi hingga evaluasi pembelajaran. Metode evaluasi pembelajaran yang dapat dilakukan pada LMS adalah dengan pemberian tugas kepada mahasiswa, dan mahasiswa dapat melakukan pengumpulan dengan mengunggah dokumen jawaban mereka. Hal yang sering menjadi permasalahan adalah adanya indikasi plagiarisme yang dilakukan mahasiswa dengan menduplikasi jawaban mahasiswa lainnya. Untuk mengatasi permasalahan tersebut, akan dikembangkan fitur pada menu Tugas Mata Kuliah LMS yang dapat mendeteksi kemiripan dokumen jawaban tugas mahasiswa dengan menggunakan metode *K-Shingling* dan *Cosine Similarity*. Algoritma *K-Shingling* digunakan untuk membentuk *shingle* yang

merupakan *substring* atau potongan kata yang dibentuk dari keseluruhan teks dalam dokumen jawaban tugas yang telah melalui tahap *preprocessing*, kemudian untuk tiap dokumen dibangkitkan suatu vektor nilai *shingle* yang akan digunakan untuk membandingkan nilai kemiripan antar dokumen menggunakan *Cosine Similarity*. Hasil yang didapat dalam penelitian ini adalah sistem yang dikembangkan mampu mendeteksi persentase kemiripan dokumen jawaban tugas dari tiap mahasiswa, dan dari nilai persentase tersebut dapat mengindikasikan apakah jawaban tugas mahasiswa merupakan plagiat terhadap jawaban mahasiswa lainnya atau tidak.

Kata Kunci— LMS; Kemiripan Dokumen; *K-Shingling*; *Cosine Similarity*

I. PENDAHULUAN

Saat ini perkembangan teknologi informasi sudah sangat pesat. Pemanfaatan teknologi telah mempengaruhi berbagai macam aspek kehidupan. Salah satu bidang pemanfaatan teknologi adalah pada bidang Pendidikan, mulai dari pendidikan tingkat dasar hingga tingkat atas pada universitas. Pihak - pihak yang terlibat pada pendidikan tingkat universitas seperti dosen sebagai pengajar, dan mahasiswa sebagai pelajar saat ini sudah diwajibkan memahami betapa pentingnya penggunaan teknologi untuk membantu proses belajar mengajar, sebab teknologi informasi dapat dimanfaatkan untuk mendukung proses belajar mengajar agar berjalan dengan lebih efektif dan efisien. Komunikasi yang terjalin antara dosen dan mahasiswa tidak hanya terbatas pada ruang kelas namun sudah bergeser dengan pemanfaatan teknologi di luar kelas, sehingga dosen dapat berkomunikasi dalam proses belajar mengajar dengan mahasiswa secara online melalui internet dengan memanfaatkan media dan aplikasi yang saat ini tersedia. Salah satu aplikasi yang dapat menunjang proses pembelajaran secara online adalah Sistem Manajemen Pembelajaran atau yang lebih dikenal dengan nama *Learning Management System (LMS)*.

LMS adalah sebuah aplikasi perangkat lunak yang digunakan untuk mengatur dan memberikan konten belajar yang di dalamnya termasuk materi, silabus, tugas, kuis dan ujian [1]. LMS menyediakan wadah yang terintegrasi untuk penyampaian materi, manajemen pembelajaran yang dapat diakses oleh peserta didik, pembuat materi, dan administrator, dan membantu dalam perencanaan, distribusi, dan evaluasi sebuah proses pembelajaran [2]. Melalui LMS, perkuliahan dapat dilakukan secara online, dimana dosen dapat menyampaikan materi pembelajaran kepada mahasiswa, membuka ruang diskusi, serta memberikan tugas terkait materi yang diberikan, sedangkan dari segi mahasiswa mereka dapat

^{1, 2, 3, 4} Pascasarjana Jurusan Teknik Elektro Fakultas Teknik Universitas Udayana, Jln. P.B Sudirman Denpasar-Bali 80234 INDONESIA (e-mail: komang.nopha@gmail.com, rukmisari@unud.ac.id, yoga@unud.ac.id, msudarma@unud.ac.id)



mengunduh materi perkuliahan yang diberikan oleh dosen, melakukan diskusi terkait materi, dan dapat mengumpulkan tugas yang diberikan oleh dosen. Penggunaan LMS mulai sering ditemukan di dunia pendidikan terlebih sejak pandemi COVID menyerang, dimana seluruh kegiatan pada semua bidang terpaksa harus dilakukan secara online atau daring [3], termasuk aktivitas pembelajaran, mulai dari penyampaian materi, hingga pengumpulan tugas dilakukan secara online. Tugas diberikan kepada mahasiswa sebagai upaya untuk evaluasi pemahaman mereka terhadap materi perkuliahan yang disampaikan secara online, sehingga ini akan berpengaruh terhadap penilaian mahasiswa itu sendiri [4], disamping itu juga akan memberikan umpan balik kepada dosen mengenai bagaimana cara mereka memberikan materi berikutnya agar lebih mudah dipahami oleh mahasiswa.

Namun dalam metode pengumpulan tugas mahasiswa secara online, hal yang sering terjadi adalah adanya indikasi pada jawaban tugas mahasiswa yang hampir sama persis dengan tugas mahasiswa yang lainnya, karena mereka meniru jawaban tugas dari teman lainnya dengan melakukan *copy-paste* jawaban, yang mana hal ini sudah termasuk kedalam kategori plagiarisme [4]. Beberapa penyebab seseorang melakukan tindakan plagiarisme adalah karena kurangnya ide atau pemahaman dalam membuat suatu karya, kurang yakin dengan hasil dari karyanya sendiri, dan kurangnya pemahaman tentang plagiarisme dan undang-undang hak cipta [5]. Hal ini tentu tidak baik apabila menjadi kebiasaan bagi mahasiswa, karena mereka yang sebenarnya kurang paham mengenai materi pembelajaran, dengan melakukan plagiat dari jawaban mahasiswa yang lain mereka terlihat memahami materi dengan baik. Padahal yang diharapkan melalui pengumpulan tugas ini adalah jawaban yang benar - benar berasal dari pemahaman mahasiswa itu sendiri, sehingga mahasiswa menjadi terbiasa untuk jujur dan mampu mengukur kemampuan diri sendiri dalam memahami materi pembelajaran, begitupun dosen juga dapat mengetahui tingkat pemahaman mahasiswa didiknya.

Berdasarkan latar belakang yang telah dijelaskan diatas, untuk mengatasi permasalahan yang telah dijelaskan diatas, peneliti akan merancang fitur yang akan diimplementasikan pada LMS, yaitu pada menu Tugas Mata Kuliah untuk dapat melakukan deteksi tingkat kemiripan dokumen tugas dari masing – masing mahasiswa dengan menggunakan algoritma *K-Shingling* dan *Cosine Similarity*. Metode *Agile Development* akan digunakan pada penelitian ini sebagai metode pengembangan fitur pada sistem LMS. Ruang lingkup dari pengembangan fitur dalam penelitian ini adalah hanya membandingkan kemiripan antar dokumen tugas tiap mahasiswa, tidak seperti *tools* yang saat ini tersedia yaitu *Turnitin* dan lain lain yang membandingkan kemiripan dokumen dengan sumber – sumber dari internet. Sehingga pengembangan fitur ini berguna untuk memberi peringatan awal kepada mahasiswa apabila tugas mereka terindikasi memiliki nilai kemiripan yang tinggi terhadap jawaban mahasiswa yang lain, sehingga membuat mahasiswa menjadi lebih jujur dalam membuat tugas mereka dengan jawaban sendiri.

II. STUDI PUSTAKA

A. Kajian Pustaka Teori

1) *Learning Management System (LMS)* : LMS dapat membantu proses pembelajaran yang harus dilakukan secara online, oleh karena itu agar pengguna LMS tidak kesulitan dalam penggunaannya, LMS sebaiknya memiliki beberapa kriteria dan fitur berikut yang berfungsi untuk memudahkan pengguna dan membantu proses transfer ilmu, fitur tersebut di antaranya adalah [1] :

- Tampilan Antarmuka yang Menarik

LMS harus memiliki tampilan antarmuka yang menarik, sehingga pengguna menjadi mudah memahami fitur-fitur didalamnya dan tidak akan kebingungan saat menggunakan aplikasi ini. Tampilan antarmuka yang menarik juga membuat proses pembelajaran menjadi lebih interaktif, dan mahasiswa akan lebih nyaman dalam mengoperasikan LMS.

- Pendaftaran Daring

Pengguna LMS dapat menggunakan fitur-fitur didalamnya apabila telah terdaftar didalam kelas matakuliah. Umumnya, harus ada laman khusus sebagai tempat pendaftaran kelas agar pengguna dapat mengakses semua terkait materi dan silabus pembelajaran, namun dalam penelitian ini, mahasiswa pengguna LMS sudah otomatis terdaftar dalam kelas apabila telah mengambil matakuliah tersebut dalam studi mereka.

- Kelas Daring

Proses pembelajaran secara online yang melibatkan dosen dan mahasiswa terjadi didalam kelas daring ini, dosen dapat melakukan upload materi pembelajaran yang berupa silabus dan materi dalam bentuk video, animasi, artikel, e-book, dan lain-lain, dan mahasiswa dapat mengakses materi dan berinteraksi dengan dosen dan mahasiswa lainnya dalam kelas daring.

- Tugas dan Kuis

Untuk melakukan evaluasi terhadap pemahaman mahasiswa terhadap materi pembelajaran yang sudah diberikan, dosen juga perlu memberikan Tugas dan Kuis kepada mahasiswa. Fitur ini dapat digunakan oleh dosen untuk membuat pertanyaan dan petunjuk tugas, sedangkan mahasiswa dapat mengunggah jawaban dari tugas mereka.

- Video Conference

Sebagai fitur tambahan, fitur ini dapat memungkinkan dosen dan mahasiswa bertatap muka secara online untuk membahas materi pembelajaran dengan dosen dan mahasiswa lainnya..

Pada penelitian ini akan dilakukan pengembangan pada fitur Tugas dan Kuis pada LMS, peneliti akan mengembangkan fitur untuk dapat mendeteksi kemiripan dokumen dari tugas yang dikumpulkan oleh mahasiswa.

2) *K-Shingling*: Algoritma ini bekerja dengan cara membuat sebuah *shingle* yang berisi beberapa kata dengan jumlah yang ditetapkan [6]. Angka yang menentukan jumlah kata dalam satu *shingle* ini disebut dengan *gram*, dan dalam penelitian ini disebutkan sebagai variabel *k*. Proses *shingling* dilakukan pada tiap dokumen yang akan dibandingkan, sehingga menghasilkan gabungan *shingle* dari beberapa dokumen. Pada tiap *shingle* akan dibangkitkan suatu nilai *shingle fingerprint* yang akan dihitung dari tiap dokumen [7], kemudian nilai ini dibandingkan berdasarkan dokumen yang akan dicari kemiripannya. Nilai kemiripan antara dua set *shingle* umumnya dapat diperoleh menggunakan persamaan *Jaccard Coefficient*, yang merupakan nilai perbandingan antara

jumlah *shingle fingerprint* yang sama (*intersection*) dari dua dokumen dengan jumlah total gabungan *shingle fingerprint* (*union*) [19]. Proses perhitungan tersebut dirumuskan dalam persamaan berikut :

$$r(A, B) = \frac{|S_a \cap S_b|}{|S_a \cup S_b|} \quad (1)$$

Berdasarkan rumus diatas, S_a adalah nilai *shingle fingerprint* dari dokumen A, dan S_b adalah *shingle fingerprint* dari dokumen B. *Intersection* merupakan jumlah dari *shingle fingerprint* yang bernilai sama pada dokumen A dan dokumen B, sedangkan *union* adalah nilai total gabungan dari *shingle fingerprint* dokumen A dan dokumen B.

Tahapan yang dilakukan dalam algoritma *K-Shingling* sebagai berikut [6]:

- Lakukan tahapan *preprocessing* pada teks dokumen yang telah diekstraksi.
- Pembentukan *shingle* dimulai dari kata pertama sampai dengan k kata berikutnya, kemudian *shingle* berikutnya dari kata kedua sampai k kata berikutnya, lakukan pembentukan *shingle* sampai dengan kata terakhir dalam dokumen.
- Untuk tiap *shingle* yang telah terbentuk, hitung nilai *fingerprint* sebagai ciri dari tiap dokumen yang akan dihitung kemiripannya.
- Gunakan rumus untuk menghitung nilai kemiripan dokumen berdasarkan nilai *shingle fingerprint* yang telah dihitung dari tiap dokumen.

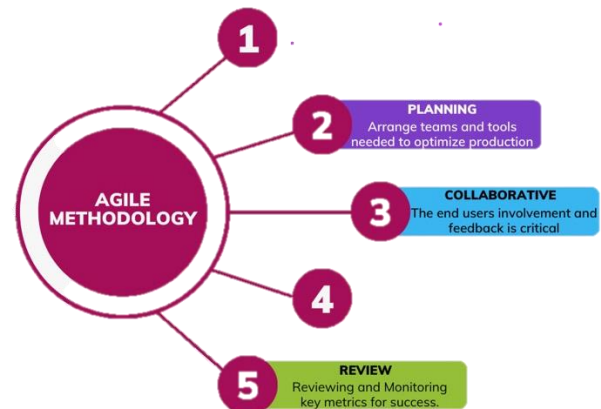
3) *Cosine Similarity*: *Cosine Similarity* merupakan metode yang digunakan untuk menghitung tingkat kesamaan (*similarity*) antar dua buah objek [11]. Metode yang digunakan adalah melakukan perhitungan ukuran kesamaan antara dua buah vektor dalam sebuah ruang dimensi yang didapat dari nilai cosinus sudut dari perkalian dua buah vektor yang dibandingkan [10]. Untuk notasi himpunan *Cosine Similarity* dapat menggunakan persamaan sebagai berikut.

$$sim(A, B) = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n A_i^2 \times \sum_{i=1}^n B_i^2}} \quad (2)$$

Dalam kasus perhitungan kemiripan antar dokumen, A dan B adalah dua buah objek dokumen berbeda yang akan diuji kemiripannya, sedangkan n adalah jumlah *shingle* yang terbentuk dari kedua dokumen, dan A_i adalah nilai *shingle* ke- i pada dokumen A sedangkan B_i adalah nilai *shingle* ke- i pada dokumen B.

4) *Agile Development* : merupakan salah satu metode untuk mengembangkan sistem perangkat lunak (*software*). Kata *Agile* yang berarti gesit, cepat, ringan, bergerak bebas, lincah, digunakan untuk menjelaskan bahwa konsep proses pengembangan perangkat lunak ini berbeda dengan konsep model proses yang sudah ada. Metode ini dicetuskan oleh Kent Beck dan 16 rekannya, yang menjelaskan bahwa metode pengembangan *Agile* adalah suatu cara atau metode dalam pengembangan perangkat lunak yang dilakukan secara Komang Nova Artawan: Perancangan Fitur Deteksi Kemiripan...

sekaligus melalui kolaborasi dengan melibatkan pengguna akhir sistem dalam satu tim, sehingga metode ini dapat membantu pengembangan perangkat lunak menjadi lebih cepat [12].



Gambar 1: Skema *Agile Development*

Gambar 1 : menunjukkan siklus dari metode *Agile Development* ini terdiri dari 5 tahap, dimulai dari definisi konsep dan kebutuhan sistem yang akan dikembangkan, menentukan rencana pengembangan sistem yang meliputi penentuan alat dan anggota yang akan terlibat, eksekusi proses pengembangan sistem yang melibatkan pengguna akhir, melakukan perilisan sistem secara bertahap dan rutin untuk mendapatkan masukan dari pengguna, dan tetap mengacu pada kebutuhan sistem di awal untuk keberhasilan pengembangan sistem [1].

B. Kajian Pustaka Penelitian

Beberapa penelitian sebelumnya yang dijadikan referensi pada penelitian ini diantaranya adalah yang dilakukan oleh Jahnae Miller pada tahun 2020 dengan judul *An Empirical Comparison of Code Similarity Algorithms* [8], dalam penelitian tersebut dinyatakan bahwa metode *Shingling* lebih baik daripada metode *Winnowing* dan *AST* dari segi penggunaan CPU, penggunaan memori, dan nilai *absolute error* margin nya. Kemudian penelitian yang dilakukan oleh Siti Hardiyanti pada tahun 2018 yang berjudul *Perbandingan Distance Based Similarity Measure Pada Algoritma Rabin Karp Untuk Menghitung Kemiripan Teks* [13] didapatkan hasil bahwa metode *Cosine Similarity* memiliki akurasi paling tinggi yaitu sebesar 83,2% dengan $k\text{-gram} = 2$ dibandingkan dengan metode *Dice Coefficient* dan *Jaccard Coefficient*. Berdasarkan penelitian sebelumnya tersebut, diketahui bahwa metode *Shingling* dan *Cosine Similarity* merupakan yang terbaik ketika dibandingkan dengan metode yang serupa, oleh karena itu, penelitian kali ini akan menerapkan metode *K-Shingling* dan *Cosine Similarity* untuk melakukan deteksi kemiripan dokumen tugas mahasiswa pada LMS yang telah dirancang.



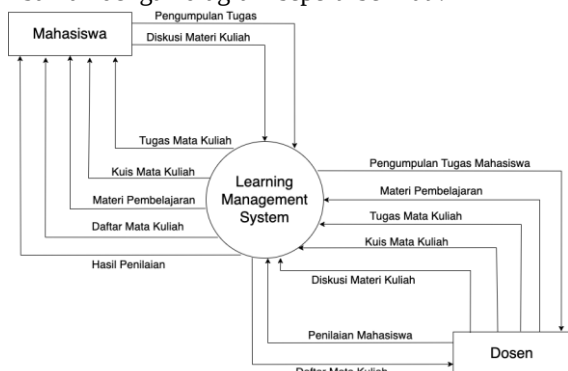
Penelitian yang serupa telah dilakukan oleh Irwan Saputra Simanullang pada tahun 2020 dengan judul Perancangan Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma *Shingling* [5], namun pada penelitian tersebut bertujuan untuk membuat aplikasi berbasis *desktop* yang menerima inputan berupa dua file dokumen dengan ekstensi *doc* dan *txt* untuk mendeteksi kemiripan antara 2 dokumen tersebut. Perbedaan penelitian tersebut dengan penelitian yang akan dilakukan kali ini adalah pada metode similarity yang digunakan, dimana penelitian tersebut menggunakan metode *Jaccard Similarity*. Selain itu, metode *Cosine Similarity* juga telah digunakan pada penelitian yang dilakukan oleh Eric Siswanto, dan Yo Ceng Giap pada tahun 2020 dengan judul Implementasi Algoritma *Rabin-Karp* dan *Cosine Similarity* Untuk Pendeteksi Plagiarisme Pada Dokumen [9]. Hasil yang didapatkan pada penelitian tersebut adalah metode *Rabin-Karp* dan *Cosine Similarity* dapat memproses perbandingan plagiarisme dengan jumlah karakter sebanyak 42432 karakter memakan waktu sebanyak 12,49 hingga 16,17 detik. Perbedaan penelitian tersebut dengan penelitian kali ini adalah pada penelitian tersebut menggunakan algoritma *Rabin-Karp*.

III. METODOLOGI

Dalam penelitian ini, akan dilakukan pengembangan fitur untuk mendeteksi kemiripan dokumen tugas mahasiswa yang akan diimplementasikan pada LMS. Metode yang akan digunakan untuk proses pengembangan fitur ini adalah Agile Development yang terbagi menjadi beberapa tahapan sebagai berikut :

A. Melakukan analisis konsep dan definisi kebutuhan sistem.

Konsep dan kebutuhan dari pengembangan fitur yang akan diimplementasikan pada sistem LMS ini secara umum dapat digambarkan dengan diagram seperti berikut :



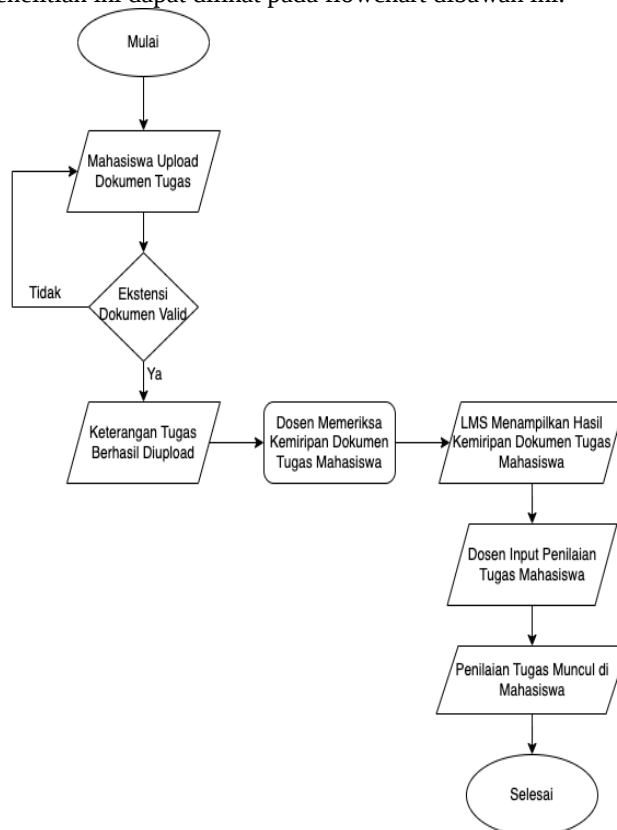
Gambar 2: Data Flow Diagram (DFD) Penggunaan LMS

Pada Gambar 2, terlihat bagaimana interaksi antara dosen dan mahasiswa pada sistem LMS ini. Salah satu fitur yang akan dikembangkan dalam penelitian ini akan diterapkan pada menu Tugas Mata Kuliah, yaitu fitur untuk mendeteksi kemiripan dokumen jawaban tugas mahasiswa. Ketika dosen telah memberikan tugas pada matakuliah yang diampunya, dan mahasiswa telah melakukan unggah dokumen jawaban mereka sebagai pengumpulan tugas, fitur yang dikembangkan ini nantinya dapat digunakan oleh dosen untuk melakukan perbandingan kemiripan dokumen jawaban tugas dari masing masing mahasiswa, apabila terdapat nilai kemiripan yang tinggi dengan salah satu dokumen tugas mahasiswa lainnya, maka

jawaban tugas mahasiswa tersebut terindikasi plagiat. Yang menjadi batasan dalam penelitian ini adalah deteksi kemiripan dokumen dilakukan dengan memeriksa kemiripan dokumen jawaban tugas mahasiswa, dengan tiap dokumen lain dari seluruh mahasiswa dalam satu matakuliah, tidak memeriksa terhadap kemiripan dokumen terhadap sumber-sumber dari internet seperti yang dilakukan oleh *software plagiarism checker* seperti Turnitin dan sebagainya.

B. Menentukan rencana untuk pengembangan sistem.

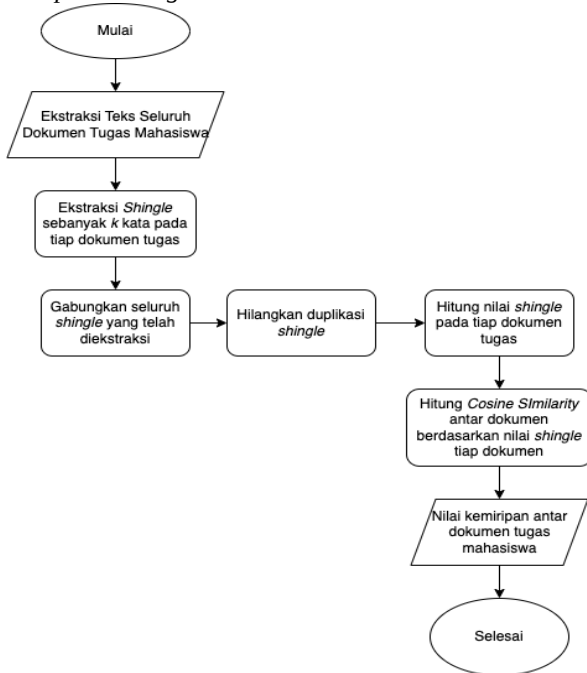
Secara umum, bagaimana fitur untuk deteksi kemiripan dokumen tugas mahasiswa yang akan dikembangkan dalam penelitian ini dapat dilihat pada flowchart dibawah ini.



Gambar 3: Flowchart Pengembangan Fitur Deteksi Kemiripan Dokumen Tugas Mahasiswa pada LMS

Gambar 3 menjelaskan bagaimana cara kerja dari fitur yang akan dikembangkan dalam penelitian ini. Pada saat interaksi dengan menu Tugas Mata Kuliah, mahasiswa akan melakukan pengumpulan tugas dengan mengupload dokumen mereka, kemudian sistem akan memvalidasi ekstensi dari dokumen mereka. Apabila dokumen tugas dari tiap mahasiswa telah diupload, kemudian dosen dapat melakukan pemeriksaan tugas masing-masing mahasiswa dengan melakukan pengecekan kemiripan dokumen jawaban mereka, yang diimplementasikan menggunakan algoritma *K-Shingling* dan *Cosine Similarity*. Berdasarkan nilai kemiripan yang didapat, dosen akan menentukan penilaian terhadap tugas mahasiswa apakah tugas tersebut terindikasi plagiat dengan tugas mahasiswa lain atau tidak.

Cara kerja dari metode pengecekan kemiripan dokumen yang akan diterapkan dalam penelitian ini dapat dilihat pada *flowchart* dibawah ini.



Gambar 4: Flowchart Algoritma K-Shingling dan Cosine Similarity untuk Deteksi Kemiripan Dokumen Tugas Mahasiswa pada LMS

Algoritma yang digunakan untuk melakukan deteksi kemiripan dokumen tugas mahasiswa pada penelitian ini adalah K-Shingling dan Cosine Similarity. Metode K-Shingling akan membagi teks pada suatu dokumen yang sebelumnya telah dibersihkan dengan tahapan *pre-processing* menjadi beberapa *shingle* yang terdiri dari sejumlah *k*-kata (*substring*) [14]. Kemudian dilakukan pengecekan terhadap kumpulan *shingle* yang telah terbentuk untuk menghilangkan duplikasi *shingle*. Setelah mendapatkan seluruh *shingle* dari dokumen yang akan dicari kemiripannya yang telah dibersihkan dari duplikasi, kemudian dilakukan proses perhitungan nilai *shingle* yang berupa vector nilai yang terdiri dari nilai 0 atau 1, nilai tersebut menandakan apakah *shingle* tersebut ditemukan pada dokumen yang diuji atau tidak. Setelah didapatkan nilai *shingle* dokumen, kemudian dengan Cosine Similarity akan dihitung nilai kemiripan dari dua buah dokumen berdasarkan vektor nilai *shingle* dokumen tersebut. Nilai Cosine Similarity ini menandakan persentase kemiripan masing-masing dokumen, dimana terdapat nilai ambang batas (*threshold*) yang digunakan untuk menentukan apakah dua buah dokumen terindikasi melakukan plagiat, rentang nilai tersebut adalah < 60%, 60 – 70%, 70 – 80%, dan > 80% [17].

C. Melibatkan pengguna sistem (end-users)

Dosen dan mahasiswa sebagai pengguna sistem dilibatkan dari awal proses pengembangan sistem untuk memberikan masukan terkait fitur yang akan diterapkan pada LMS. Dalam penelitian ini, dosen dilibatkan untuk membuat tugas pada salah satu mata kuliah yang diampunya, dan kemudian dapat

menggunakan fitur untuk mendeteksi kemiripan dokumen tugas mahasiswa. Sedangkan mahasiswa dilibatkan untuk melakukan pengumpulan tugas dengan mengunggah dokumen jawaban tugas mereka.

D. Memulai pengembangan sistem

Setelah konsep dan kebutuhan sistem telah terdefinisi, tahapan yang dilakukan berikutnya adalah memulai proses pengembangan fitur pada sistem secara bertahap dan menulis ke pengguna agar mendapatkan masukan atau penyesuaian terhadap sistem yang akan langsung diterapkan dan diimplementasikan nantinya.

E. Melakukan pemeriksaan dan pemantauan sistem

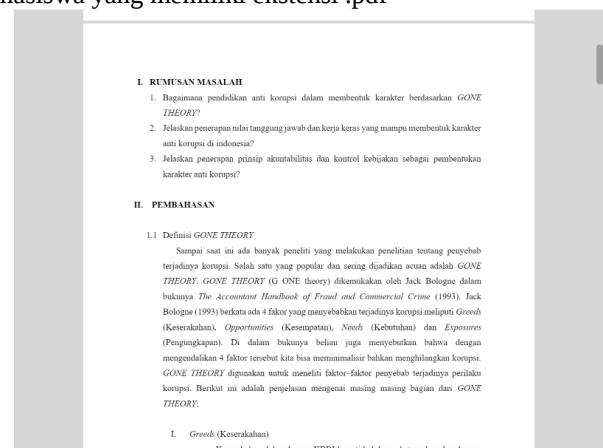
Tahap akhir adalah memeriksa kembali pada konsep dan kebutuhan sistem yang telah didefinisikan di awal untuk menentukan keberhasilan pengembangan sistem. Apabila sistem sudah mencakup semua kebutuhan tersebut, maka pengembangan fitur dalam penelitian ini dinyatakan telah berhasil dan akan dikembangkan lagi untuk menyesuaikan dengan kebutuhan mendatang.

IV. HASIL DAN PEMBAHASAN

Hasil yang didapat dari pengembangan fitur pada LMS untuk mendeteksi kemiripan dokumen tugas mahasiswa dapat dijelaskan menjadi beberapa poin, yaitu bagaimana cara kerja metode K-Shingling dan Cosine Similarity yang digunakan, dan bagaimana hasil perancangan sistem dalam bentuk user interface nya.

A. Hasil Implementasi Metode

Tahapan awal yang dilakukan dalam penerapan metode K-Shingling adalah melakukan ekstraksi teks dari dokumen tugas mahasiswa yang memiliki ekstensi .pdf



Gambar 5: Contoh Teks pada Dokumen Tugas Mahasiswa

Gambar 5 menunjukkan contoh dokumen teks jawaban tugas mahasiswa, teks tersebut akan melalui tahapan *preprocessing* yang dapat dilakukan dengan berbagai cara, antara lain menghilangkan tanda baca, mengubah kapitalisasi (*case*



folding), koreksi ejaan, dan menambah, menghapus, atau menata ulang kata [20]. Tahapan *preprocessing* ini penting untuk dilakukan, karena seperti tanda baca atau karakter lain dalam kalimat, misalnya tanda baca koma, titik, atau spasi berlebihan tersebut memiliki frekuensi kemunculan tinggi pada dokumen, tetapi memiliki bobot informasi yang sangat rendah [18]. Teks yang telah melalui *preprocessing* ini selanjutnya akan diproses untuk pembuatan *shingle*.

Shingle diimplementasikan dengan metode jendela geser dengan ukuran jendela yang dilambangkan dengan k yang ditempatkan pada teks dokumen sehingga didapatkan sebagian kecil kata (*substring*) [15]. *Shingle* terbentuk dari sejumlah k kata sesuai ukuran jendela yang digunakan, yang dimulai dari kalimat awal hingga akhir dokumen, misalnya digunakan nilai $k = 5$, maka contoh dari *shingle* yang terbentuk berdasarkan dokumen tugas mahasiswa diatas adalah sebagai berikut :

TABEL I
CONTOH *K-SHINGLE* DARI SUATU DOKUMEN TUGAS MAHASISWA

rumusan masalah bagaimana pendidikan anti
masalah bagaimana pendidikan anti korupsi
bagaimana pendidikan anti korupsi dalam
pendidikan anti korupsi dalam membentuk
anti korupsi dalam membentuk karakter
korupsi dalam membentuk karakter berdasarkan
dalam membentuk karakter berdasarkan gone
membentuk karakter berdasarkan gone theory
karakter berdasarkan gone theory jelaskan
berdasarkan gone theory jelaskan penerapan
gone theory jelaskan penerapan nilai
theory jelaskan penerapan nilai tanggung
jelaskan penerapan nilai tanggung jawab

Dalam metode *Shingling*, substring yang diperoleh dari dokumen ditempatkan dalam satu set, dan tidak memiliki pengulangan, oleh karena itu pengulangan *shingle* tidak diperhitungkan dalam menghitung dokumen serupa [16], sehingga perlu dilakukan pengecekan agar tidak ada pengulangan *shingle*.

Shingle yang telah dibersihkan dari duplikasi ini kemudian digabung menjadi satu set untuk dibandingkan pada kedua dokumen sehingga didapat nilai *shingle* seperti yang terlihat pada tabel II.

TABEL III
NILAI *K-SHINGLING* DARI 2 DOKUMEN YANG DIBANDINGKAN

<i>Shingle</i>	Dokumen 1	Dokumen 2
rumusan masalah bagaimana pendidikan anti	1	0

masalah bagaimana pendidikan anti korupsi	1	0
bagaimana pendidikan anti korupsi dalam	1	1
pendidikan anti korupsi dalam membentuk	1	0
anti korupsi dalam membentuk karakter	1	1
korupsi dalam membentuk karakter berdasarkan	1	0
dalam membentuk karakter berdasarkan gone	1	1
membentuk karakter berdasarkan gone theory	1	0
karakter berdasarkan gone theory jelaskan	1	1
berdasarkan gone theory jelaskan penerapan	1	1
gone theory jelaskan penerapan nilai	1	1
theory jelaskan penerapan nilai tanggung	1	0
jelaskan penerapan nilai tanggung jawab	1	1

Nilai *K-Shingling* yang dihasilkan dari masing-masing dokumen menandakan apakah *shingle* tersebut ditemukan pada dokumen bersangkutan, apabila *shingle* tersebut ditemukan, maka nilai nya 1, sedangkan jika *shingle* tidak ditemukan pada dokumen, maka nilai nya 0. Setelah didapatkan nilai *K-Shingling* dari kedua dokumen, kemiripan dokumen dapat diukur berdasarkan proporsi nilai *shingle* yang ditemukan dalam pasangan dokumen teks dari vektor kata yang didefinisikan untuk mewakili dokumen [17], dan kemudian vektor tersebut akan dibandingkan menggunakan *Cosine Similarity*

Dengan menggunakan rumus pada “(2)”, maka didapat nilai sebagai berikut :

$$sim(A, B) = \frac{(1 \times 0) + (1 \times 0) + (1 \times 1) + (1 \times 0) + (1 \times 1) + (1 \times 0) + \dots + (1 \times 1)}{\sqrt{1^2 + 1^2 + 1^2 + 1^2 + \dots + 1^2} \times \sqrt{0^2 + 0^2 + 1^2 + 0^2 + \dots + 1^2}}$$

$$sim(A, B) = \frac{7}{\sqrt{13} \times \sqrt{7}}$$

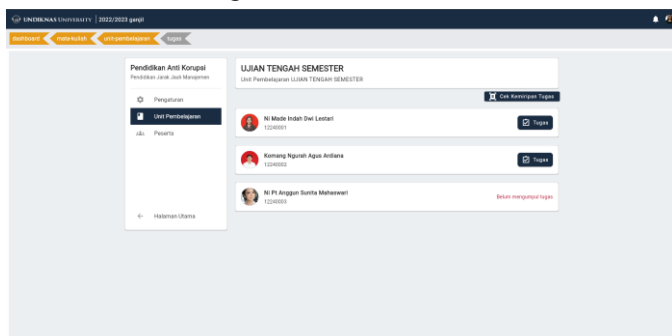
$$sim(A, B) = \frac{7}{\sqrt{91}}$$

$$sim(A, B) = \frac{7}{9.53939201417} = 0.7337993857$$

Nilai *Cosine Similarity* dihitung berdasarkan nilai penjumlahan dari perkalian nilai *shingle* antar dokumen, kemudian dibagi dengan nilai akar dari penjumlahan nilai *shingle* dokumen 1 dikali dengan nilai akar dari penjumlahan nilai *shingle* dokumen 2. Berdasarkan perhitungan tersebut,

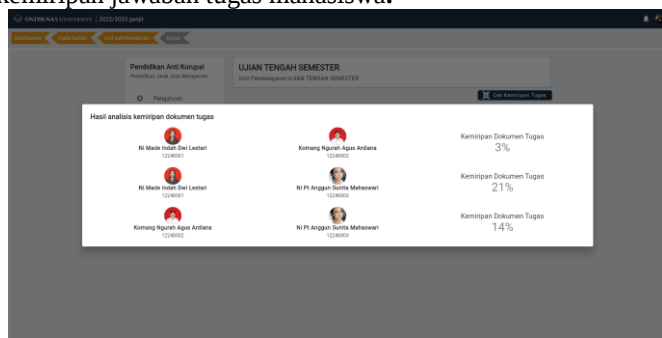
nilai *Cosine Similarity* yang didapat adalah 0.7337, maka kemiripan dari dokumen 1 dan dokumen 2 adalah 73.37%.

B. Hasil Perancangan Sistem



Gambar 6: Menu Tugas Mahasiswa pada Menu LMS

Gambar 6 menunjukkan menu yang dapat diakses oleh dosen ketika masuk ke menu tugas matakuliah, pada menu tersebut ditampilkan daftar mahasiswa beserta keterangan tugas yang telah mereka kumpulkan. Selain itu, fitur yang akan dikembangkan dalam penelitian ini juga berada pada menu tugas matakuliah ini, dimana dosen dapat melakukan deteksi kemiripan jawaban tugas mahasiswa.



Gambar 7: Tampilan Fitur Hasil Deteksi Kemiripan Tugas Mahasiswa

Pada gambar 7 terlihat contoh hasil deteksi kemiripan dokumen tugas mahasiswa. Nilai yang ditampilkan adalah berupa persentase kemiripan dokumen yang telah dihitung menggunakan algoritma *K-Shingling* dan *Cosine Similarity*. Berdasarkan nilai persentase yang ditampilkan tersebut akan menunjukkan indikasi apakah dokumen jawaban tugas dari satu mahasiswa merupakan plagiat terhadap jawaban tugas mahasiswa yang lainnya.

V. KESIMPULAN DAN SARAN

Pada penelitian ini, pengembangan fitur untuk mendeteksi kemiripan dokumen tugas mahasiswa akan diterapkan LMS yaitu pada menu Tugas Mahasiswa yang dapat diakses oleh dosen, dan metode *K-Shingling* dan *Cosine Similarity* dapat digunakan untuk mengembangkan fitur tersebut. Berdasarkan nilai kemiripan dokumen yang didapatkan oleh sistem, itu akan menunjukkan indikasi apakah jawaban tugas mahasiswa melakukan plagiat terhadap jawaban tugas mahasiswa yang Komang Nova Artawan: Perancangan Fitur Deteksi Kemiripan...

lain. Metode yang digunakan untuk mengembangkan fitur ini akan dioptimalkan dengan menggunakan metode *preprocessing* teks yang baik, karena teks yang diekstraksi dari suatu dokumen mewakili isi dokumen tersebut dan akan mempengaruhi hasil deteksi kemiripan antar dokumen jawaban tugas mahasiswa. Selain itu algoritma *K-Shingling* ini juga dapat dilakukan pengujian untuk menentukan berapa nilai *k* yang terbaik untuk pembuatan *shingle* dalam dokumen. Pengembangan fitur ini dapat membantu dosen dalam memeriksa kemiripan antara jawaban tugas mahasiswa sebelum memberikan penilaian terhadap jawaban mahasiswa.

REFERENSI

- [1] Yauma, Alfath., Fitri, Iskandar., Ningsih, Sari. "Learning Management System (LMS) pada E-Learning Menggunakan Metode Agile dan Waterfall berbasis Website". Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi), 323-328. 2021.
- [2] (2021) Universitas Medan Area. [Online]. Available: <https://uma.ac.id/berita/apa-itu-learning-management-system-40lms41>
- [3] Fitriani, Yuni. "Analisa Pemanfaatan Learning Management System (LMS) Sebagai Media Pembelajaran Online Selama Pandemi COVID-19". Journal of Information System, Informatics and Computing. Vol.4 No.2, Desember 2020.
- [4] Putra, Miftakhul Ilmi S. Mutaqin, Imam. "Analisis Deteksi Plagiarisme Pada LMS (Learning Management Systems) Untuk Pembelajaran Online Calon Guru Madrasah Ibtidaiyah". Jpdi: Jurnal Pendidikan Dasar Islam, Vol. 3, No. 1 : 01-15. April 2021.
- [5] Simanullang, Irwan Saputra. "Perancangan Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma Shingling". Jurnal Sistem Komputer dan Informatika (JSON) Volume 2, Nomor 1. Hal: 36-41, Sep 2020.
- [6] Sariwating, Andry Vegard. "Perancangan dan Implementasi Aplikasi Deteksi Kemiripan Citra Digital Menggunakan Algoritma Shingling dan Redundant Pixel Removal". S. Kom. Artikel Ilmiah. Universitas Kristen Satya Wacana. 2016.
- [7] Adiansyah, Muhamad Yusuf. "Perancangan dan Implementasi Aplikasi Deteksi Kemiripan Dokumen Menggunakan Algoritma Shingling dan MD5 Fingerprint". S. Kom. Artikel Ilmiah. Universitas Kristen Satya Wacana, Des. 2014.
- [8] Miller, Jahnae. "An Empirical Comparison of Code Similarity Algorithms". NYUAD's Capstone Project 2, Abu Dhabi, UAE. Spring 2020.
- [9] Siswanto, Eric., Giap, Yo Ceng. "Implementasi Algoritma Rabin-Karp dan Cosine Similarity Untuk Pendeteksi Plagiarisme Pada Dokumen". Jurnal Algor-Vol.1 No.2. 2020.
- [10] Meilina, Lely., Kumara, I Nyoman Satya., Setiawan, Nyoman. "Literature Review Klasifikasi Data Menggunakan Metode Cosine Similarity dan Artificial Neural Network". Majalah Ilmiah Teknologi Elektro, Vol. 20, No. 2, Juli – Desember 2021.
- [11] Setyadi, I Wayan Adi., Khrisne, Duman Care., Suyadnya, I Made Arsa. "Automatic Text Summarization Menggunakan Metode Graph dan Ant Colony Optimization". Majalah Ilmiah Teknik Elektro, Vol. 17, No. 1, Januari – April 2018.
- [12] (2020) Universitas Raharja. [Online]. Available: <https://raharja.ac.id/2020/04/05/metode-agile/>
- [13] Hardiyanti, Siti. "Perbandingan Distance Based Similarity Measure Pada Algoritma Rabin Karp Untuk Menghitung Kemiripan Teks". S. Kom. Skripsi. Universitas Gadjah Mada. 2018.
- [14] Li, Peng., Qiao, Tianling., Guang, Yongxing., Zhang, Lan. "A New Shingling Similar Text Detection Algorithm" in Proceedings of the Second International Symposium on Simulation and Process Modelling, 2020, paper, page 83.



- [15] Varol, Cihan., Hari, Sairam. "Detecting Near-Duplicate Text Documents With A Hybrid Approach". *Journal of Information Science*, April, 2015.
- [16] Azgomi, Hossen., Mahsayeh, Masumeh Ghasemi., Mohammadic, Masoud., Rad, Milad Moradi. "A Method For Finding Similar Documents Relying on Adding Repetition of Symbols in Length Based Filtering". *Indian J.Sci.Res.* 2(1) : 81-84, 2014.
- [17] Mishra, Asha Rani., Panchal, V.K. "A Novel Approach to Capture the Similarity in Summarized Text Using Embedded Model". *International Journal on Smart Sensing and Intelligent Systems*, Issue 1 Vol. 15, 2022
- [18] Firdaus, Pasnur och Wabdillah. "Implementasi Cosine Similarity Untuk Peningkatan Akurasi Pengukuran Kesamaan Dokumen Pada Klasifikasi Dokumen Berita Dengan K Nearest Neighbour" *Jurnal Teknologi Informasi dan Komunikasi* , Vol. 1, Nomor 1, 2019.
- [19] Manaa, Mehdi Ebady., Abdulameer, Ghufra. "Web Documents Similarity Using K-Shingle Tokens and MinHash Technique". *Journal of Engineering and Applied Sciences* 13 (6): 1449-1505, 2018.
- [20] Finansyah, Achmad Yohni Wahyu., Afiahayati, Sutanto, Vincent Michael. "Performance Comparison of Similarity Measure Algorithm as Data Preprocessing Stage: Text Normalization in Bahasa Indonesia". *Scientific Journal of Informatics*, Vol. 9, No. 1, May 2022.