

PENERAPAN ALGORITMA *WINNOWER* PADA SISTEM PENGELOLAAN KERJA PRAKTIK DENGAN PENDEKATAN *HUMAN-CENTERED DESIGN*

(STUDI KASUS: PROGRAM STUDI S-1 INFORMATIKA
UNIVERSITAS BENGKULU)

Rizky Amanah Putra¹, Ferzha Putra Utama^{2*}, Aan Erlansari³

¹ Program Studi Informatika, Fakultas Teknik, Universitas Bengkulu
^{2,3} Program Studi Sistem Informasi, Fakultas Teknik, Universitas Bengkulu
^{1,2,3} Jl. WR. Supratman Kandang Limun Bengkulu 38371A Indonesia
(tel: 0736-341022; fax: 0736-341022)

¹rizkyamanahp@gmail.com
²fputama@unib.ac.id
³aan_erlanshari@unib.ac.id

Abstrak: Kerja praktik adalah mata kuliah wajib berbasis praktik yang ditempuh oleh setiap mahasiswa Program Studi Informatika Fakultas Teknik Universitas Bengkulu. Program kerja praktik secara mandiri dilakukan setiap mahasiswa sebagai salah satu syarat skripsi. Pengelolaan kerja praktik sejauh ini masih dilakukan secara manual, sehingga terdapat keterbatasan dalam pemeriksaan kesamaan dari program kerja praktik terdahulu. Kami mengusulkan sebuah sistem untuk dapat mengelola program kerja praktik dengan lebih baik. Sistem ini menerapkan algoritma *Winnower* untuk mengetahui tingkat kesamaan program kerja praktik yang ditinjau dari aspek judul, lokasi, dan uraian singkat. Lebih lanjut, pengembangan sistem ini berdasarkan metode *Human-Centered Design* untuk memenuhi kebutuhan *UI/UX* calon penggunanya. Pengukuran sistem dilakukan dengan dua metode, secara fungsional dan persepsi pengguna ahli. Pengujian fungsional sistem dengan *blackbox* menghasilkan keberhasilan sebesar 100% dari 93 skenario, sedangkan *heuristic evaluation* oleh lima responden ahli menghasilkan total rata-rata skor keseluruhan 80,8% (sangat baik). Algoritma *winnower* cukup efektif dalam melakukan *similarity check* yang dibuktikan dari pengujian sampel dihasilkan kemiripan 3,92% hingga 16,67%.

Kata Kunci: Sistem informasi, Kerja Praktik, *Winnower*, *Human-Centered Design*, *Heuristic Evaluation*

Abstract: *The internship is a practice-based compulsory course taken by every student of the Informatics Study Program, Faculty of Engineering, University of Bengkulu. Each student performs the internship program independently as one of the thesis requirements. Management of the internship so far is still done manually, so there are limitations in checking the similarity of the previous apprenticeship program. We propose a system to manage the internship program better. This system applies the Winnower algorithm to determine the level of similarity of the internship program in terms of the title, location, and brief description. Furthermore, the development of this system is based on the Human-Centered Design method to*

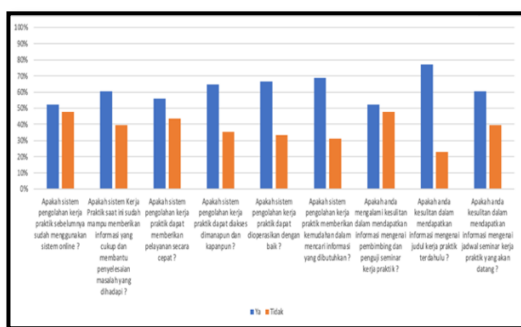
meet prospective users' UI/UX needs. System measurement is done by two methods, functionally and expert user perception. Functional testing of the system with a black box resulted in a success of 100% of 93 scenarios. In comparison, the heuristic evaluation by five expert respondents resulted in a total average score of 80.8% (excellent). The winnower algorithm is quite effective in carrying out a similarity check, as evidenced by the sample testing resulting in a similarity of 3.92% to 16.67%.

Keywords: *Information System, Practical Work, Winnower, Human-Centered Design, Heuristic Evaluation*

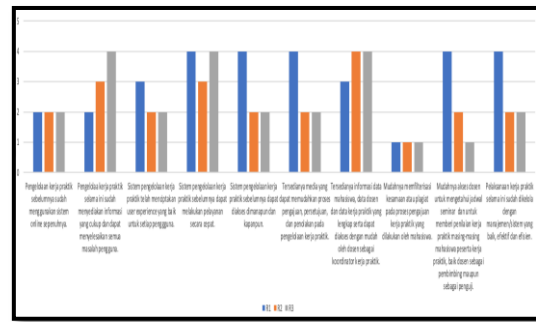
I. PENDAHULUAN

Kerja Praktik (KP) merupakan mata kuliah wajib yang berbasis praktik lapangan dengan bobot 3 SKS dilakukan secara mandiri dan harus ditempuh oleh setiap mahasiswa Program Studi Informatika Fakultas Teknik Universitas Bengkulu sebagai salah satu syarat untuk mengambil mata kuliah Skripsi. Mata kuliah merupakan kuliah lapangan yang pelaksanaannya dilakukan pada suatu lembaga/instansi/perusahaan selama satu semester. Tujuan utama KP ialah untuk meningkatkan pengetahuan yang dipelajari mahasiswa terhadap masalah yang terjadi di dunia nyata sehingga mahasiswa dapat menemukan solusi yang tepat untuk menghadapinya. Dalam pengelolaannya selama ini, prodi Informatika belum menggunakan sistem informasi untuk mengelola kebutuhan dalam pelaksanaan KP. Platform yang digunakan hanya sebatas aplikasi gratis dari Google seperti *Google Forms* dan *Google Classroom*, sehingga proses yang berjalan dirasa kurang efektif dan efisien.

Berdasarkan hasil survei yang ditunjukkan pada Gambar 1 dengan responden 48 mahasiswa aktif dan Gambar 2 menunjukkan hasil survei pada 3 dosen Program Studi Informatika, ditemukan beberapa masalah yang terjadi dalam pengelolaan KP.



Gambar 1. Hasil survei mahasiswa



Gambar 2. Hasil survei dosen

Permasalahan yang diungkapkan responden seperti terhambatnya administrasi yang belum dilakukan secara *online*, belum tersedianya *user experience* yang baik, kesulitan dalam mengetahui topik yang terindikasi plagiat, penjadwalan, dan lain sebagainya menyebabkan pelaksanaan KP selama ini masih dianggap belum efektif dan efisien. Berdasarkan aturan KP yang telah disepakati oleh Program Studi, beberapa aturan KP yang perlu diperhatikan antara lain:

- Setiap judul dan tempat KP yang diajukan mahasiswa tidak boleh sama dengan yang telah diajukan sebelumnya, atau sedang diajukan oleh mahasiswa lain.
- Judul yang diajukan dapat dilanjutkan pada semester kedua jika tidak diselesaikan pada semester pertama.

Kemudian untuk sistematis pelaksanaan kegiatan KP, mahasiswa yang ingin mengajukan KP diwajibkan untuk membuat sebuah kerangka acuan. Kerangka acuan merupakan suatu dokumen yang berisi informasi-informasi KP yang akan diajukan yang berisi data diri mahasiswa dan rencana kegiatan KP. Banyaknya topik KP yang diusulkan tiap mahasiswa memberikan kesulitan pengelola KP di prodi Informatika untuk mengetahui ada/tidaknya plagiasi. Penerapan metode *winning* bertujuan untuk membantu

menemukan tingkat kesamaan judul, lokasi, dan uraian singkat KP yang diajukan oleh mahasiswa.

Untuk membantu mengatasi masalah tersebut dibutuhkan suatu sistem yang dapat mengelola KP yang ramah pengguna. Salah satu pendekatan yang dapat dilakukan dalam pengembangan sistem yang ramah pengguna tersebut adalah dengan menerapkan metode *Human-Centered Design* (HCD). Dalam prosesnya pendekatan ini berfungsi untuk menciptakan sebuah aplikasi yang berfokus terhadap kebutuhan pengguna. HCD dilakukan dengan memahami pengguna terlebih dahulu dan mengetahui apa saja kebutuhan pengguna. HCD merupakan metode yang tepat karena melibatkan calon pengguna terlibat langsung dalam pengembangan suatu aplikasi yang berpusat pada kebutuhan pengguna. Metode HCD dapat mengidentifikasi permasalahan dari awal perancangan suatu produk agar solusi dapat ditemukan secara cepat. Metode ini memberikan tuntunan kolaborasi antara pengembang dan pengguna sistem berdasarkan pengetahuan pengguna atau *User Experience* (UX) dan menampung apa yang diinginkan pengguna dalam sistem secara langsung [1]. Proses desain *user experience* dalam pembangunan sebuah aplikasi bertujuan untuk menghasilkan tingkat kepuasan pengguna yang baik. Pendekatan HCD merupakan pendekatan iteratif guna *men-generate* desain solusi yang tepat untuk memenuhi kebutuhan calon pengguna dengan meningkatkan tingkat efektivitas serta efisiensi pengguna.

Menurut penelitian yang dilakukan oleh Wijaya menunjukkan hasil berupa evaluasi desain dengan *usability testing* mendapatkan hasil sebesar 96,39%. Dengan demikian, dapat disimpulkan bahwa desain solusi memberikan persepsi yang positif dari segi *user experience* dan dari segi

usability. Pengembangan sistem berbasis HCD ini dilakukan berdasarkan analisis masalah yang ditemukan dari hasil kuisioner yang telah dilakukan sebelumnya, penerapan HCD ini ditujukan untuk memperoleh/mengembangkan sistem sehingga dapat lebih baik diterima oleh pengguna nantinya [2].

Untuk membantu memudahkan pemeriksaan dan kecocokan poin-poin tersebut maka sistem yang dikembangkan akan menggunakan algoritma *winning*. Sehingga sistem KP ke depannya lebih mampu dalam melakukan proses filter judul dan lokasi KP yang serupa. Pada dasarnya algoritma *winning* merupakan sebuah algoritma yang melakukan perbandingan dan pencocokan string untuk mendeteksi tingkat kesamaan sebuah dokumen atau teks [3]. Algoritma *winning* berjalan dengan proses *input* data teks, kemudian dilakukan *preprocessing* dokumen untuk menentukan *n-gram* dan nilai *hash* dari masing-masing kata pada teks. Algoritma *winning* dimulai dari proses pembentukan *window* untuk melakukan pemilihan atau penyaringan kata-kata pada teks yang akan diproses nantinya. Selanjutnya akan menentukan *fingerprint* untuk menjadi pembanding sehingga dapat dihitung nilai kemiripannya (tingkat kesamaan). Pemilihan *n-gram* yang semakin kecil akan memperoleh nilai persentase kemiripan yang besar. Hal ini terjadi karena pada *n-gram* yang lebih sedikit, pemotongan *string* yang lebih kecil akan menyebabkan kemungkinan ditemukannya karakter yang sama akan lebih sering terjadi. Hal tersebut akan mempengaruhi nilai tingkat kemiripan menjadi lebih besar. Semakin besarnya *n-gram* berbanding lurus dengan jumlah karakter di dalamnya yang memungkinkan nilai tingkat kemiripan menjadi lebih kecil [4]. Pemilihan nilai

n-gram akan memengaruhi proses algoritma *winnowing* untuk menemukan *gram* pada dokumen, semakin kecil nilai *n-gram* maka semakin besar pula kemungkinan ditemukannya *fingerprints* yang sama, sehingga nilai kemiripan antar dokumen akan lebih besar. Pemilihan nilai basis yang besar akan menghasilkan nilai *hashing* yang lebih besar pula sehingga kemungkinan tabrakan nilai *hash* akan lebih sedikit [5].

Untuk mengukur kualitas sistem, dilakukan evaluasi atau pengujian perangkat lunak dengan menerapkan *heuristic evaluation*. *Heuristic Evaluation* adalah suatu cara pemeriksaan *usability* untuk perangkat lunak komputer yang membantu mengenali masalah-masalah *usability* pada rancangan antarmuka yang mana nantinya pengujian ini akan melibatkan beberapa evaluator yang merupakan pakar untuk mengevaluasi sistem. Secara umum pengujian ini ialah pengujian yang melibatkan ahli secara langsung [6]. *Heuristic Evaluation* merupakan salah satu metode yang sangat populer untuk mengukur tingkat pengalaman pengguna untuk mengidentifikasi masalah yang pada rancangan antarmuka. Metode *Heuristic evaluation* mempunyai 10 prinsip dasar sebagai acuan pendukung penentuan keputusan pengguna itu sendiri [7]. Untuk menggunakan evaluasi ini dibutuhkan *software* yang akan diteliti atau *storyboard* untuk sistem yang akan dibuat [8].

Untuk mengetahui sejauh mana kelayakan sistem maka diperlukan pengujian *blackbox* pada sistem untuk mendapatkan hasil yang lebih maksimal [9]. *Tester* akan mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program [10]. *Equivalence Partitioning* (EP) merupakan salah satu jenis teknik dalam *blackbox testing*, teknik EP

digunakan untuk menguji masukan dan membagi masukan berdasarkan fungsinya ke dalam beberapa kelompok yang mana akan memudahkan proses *test case* dalam pengujian [11].

II. LANDASAN TEORI

A. Human Centered Design

Dalam mencapai hasil yang baik untuk memenuhi kebutuhan *user* ada banyak metode yang dapat digunakan seperti contohnya *design thinking*, *user-experience design*, *human-centered design*, dan masih banyak metode lainnya. Proses desain memiliki pendekatan yang diberi istilah “*design thinking*”. Namun penggunaan istilah *design thinking* dalam desain, biasanya hanya berfokus pada aspek teknis dan praktis, dan tidak dapat sepenuhnya memahami perilaku serta kebutuhan konsumen yang sesungguhnya. Oleh karena itu, *human-centered design* muncul dan menjadi perubahan dalam pemikiran desain yang lebih humanistik. *Human-centered design* menggunakan pendekatan empati untuk memahami perilaku laten konsumen yang potensial, serta sering diabaikan dan sejauh ini belum dipertimbangkan. Melalui *human-centered design*, orang memiliki hak untuk tumbuh dengan produk desain, bukan hanya menjadi komoditas di industri desain. Dalam *design thinking*, pendekatan yang lumrah adalah pendekatan berbasis masalah. Metode ini berfokus pada penyebab kesalahan dan faktor keahlian dalam konteks praktis. Akan tetapi kemudian, banyak hal berasal dari masalah manusia yang spesifik. Misalnya, apa kesulitan dan tantangan dalam menggunakan produk ini? Bagaimana pengguna menggunakan produk desain untuk memenuhi kebutuhan mereka? Apa kemungkinan situasi yang tidak terduga dengan menggunakan produk desain? [12]. Berikut

merupakan perbandingan beberapa metode yang pada Tabel 1. hampir serupa dengan HCD yang dapat dilihat

Tabel 1. Perbandingan Metode

	<i>Design Thinking</i>	<i>User Experience Design</i>	<i>Human Centered Design</i>
Fokus	Teknis, praktis, dan hasil akhir (hanya mengutamakan identifikasi dan penyelesaian masalah)	Hanya berfokus untuk memenuhi pemahaman dan kebutuhan pengguna saja	Sangat kompleks dan mendetail terhadap karakteristik, psikologis, persepsi manusia sebagai pengguna
Perkembangan dan ketahanan	Hanya menyelesaikan masalah di satu waktu tertentu dan solusi yang dihasilkan tidak bertahan lama	Sesuai kebutuhan dan permintaan pengguna saja	Inovatif (dinamis, mengikuti tren dan kebutuhan pengguna)
Lingkup Pengguna	Umum	Spesifik	Spesifik

Berdasarkan hasil perbandingan beberapa metode pada Tabel 1, HCD menjadi salah satu yang paling baik dan cocok untuk digunakan dalam pengembangan ini. Pendekatan HCD digunakan untuk merancang sebuah desain aplikasi yang berorientasi kepada manusia sebagai pengguna sehingga UX *designer* mengetahui apa yang dibutuhkan pengguna secara nyata dan langsung [13]. Dengan kata lain, *user experience* sistem yang bagus tidak akan menyulitkan pengguna untuk mencapai tujuan mereka. Entah itu dari desain UI yang *friendly*, sistem yang ringan untuk diakses, menu yang tidak berbelit-belit, dan lain sebagainya. Sebaliknya, *user experience* yang buruk membuat pengguna frustrasi karena kesulitan untuk mendapatkan apa yang mereka inginkan [14], [15].

B. Algoritma *Winnowing*

Algoritma *winnowing* adalah salah satu algoritma pencocokan string. Jadi pada dasarnya algoritma *winnowing* merupakan sebuah algoritma yang melakukan perbandingan dan pencocokan string untuk mendeteksi tingkat kesamaan sebuah dokumen atau teks [16], [17]. Pada pendeteksiannya, algoritma *winnowing* harus memenuhi kebutuhan mendasar yaitu [3]:

- a) *Whitespace insensitivity* yaitu pencarian kalimat mirip seharusnya tidak terpengaruh oleh spasi, jenis huruf (kapital atau normal),

tanda baca dan sebagainya.

- b) *Noise surpression* yaitu menghindari penemuan kecocokan dengan panjang kata yang terlalu kecil atau kurang relevan seperti “the” dan bukan merupakan kata yang umum digunakan.
- c) *Position independence* yaitu penemuan kesamaan harus tidak bergantung pada posisi kata-kata sehingga kata dengan urutan posisi berbeda masih dapat dikenali jika terjadi kesamaan.

Berikut Langkah-langkah yang dapat dilakukan dalam penerapan Algoritma *Winnowing*:

1. Pembuangan karakter yang tidak relevan.
2. Pembentukan rangkaian *n-gram*.
3. Perhitungan fungsi *hash* untuk setiap *n-gram*.
4. Pembentukan *window* dari nilai *hash*.
5. Pemilihan *fingerprint* dari setiap *window*.
6. Persamaan *Jaccard Coeficient*.

Dalam beberapa aplikasi akan sangat berguna Jika tidak hanya menyimpan *fingerprints* dari sebuah dokumen, tetapi juga posisi dan *fingerprints* dalam dokumen. Sebagai contoh, kita ingin menampilkan informasi posisi dari sub-string yang cocok. Implementasi algoritma *Winnowing* yang efisien juga sebaiknya menyimpan posisi dari *fingerprint* yang didapatkan dimana posisi pertama

dimulai dari 0 dari nilai *hash* pada gram yang terbentuk sebelumnya [18].

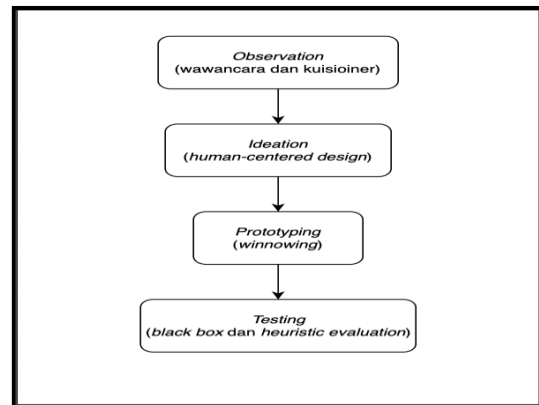
C. Heuristic Evaluation

Heuristic evaluation adalah suatu cara pemeriksaan *usability* untuk perangkat lunak komputer yang membantu mengenali masalah-masalah *usability* pada rancangan antarmuka [19], [20]. *Heuristic evaluation* juga merupakan salah satu metode yang paling banyak digunakan untuk mengukur tingkat kenyamanan pengguna dalam hal interaksi manusia dan komputer (IMK). Tujuan utama *Heuristic evaluation* adalah untuk mengidentifikasi masalah yang berkaitan dengan rancangan antarmuka. Metode *Heuristic evaluation* menggunakan 10 prinsip dalam mengevaluasi pada tingkat *usability* yang dijadikan acuan untuk menentukan keputusan yang sudah diambil oleh pengguna [7].

III. METODOLOGI PENELITIAN

A. Metode Pengembangan Sistem

HCD merupakan pendekatan iteratif yang proses penerapan setiap tahapnya dapat diulang, sehingga menghasilkan *output* yang diinginkan dapat dicapai [21]. Pendekatan *Human-Centered Design* sangat bertumpu dengan aspek *User Experience*. Penerapan HCD berfungsi untuk meningkatkan efektivitas, efisiensi serta kepuasan pengguna. Dalam siklus iteratif HCD terdapat 4 aktifitas yaitu *observation*, *ideation*, *prototyping*, dan *testing* [22].



Gambar 3. Ilustrasi HCD

Pemodelan HCD pada Gambar 3 dijelaskan sebagai berikut:

a) Observation

Tahapan pertama yang dilakukan observasi, analisis masalah, dan pengumpulan data terhadap kebutuhan pengguna sehingga dapat ditemukan solusi tepat untuk mengatasi masalah yang terjadi pada pengguna. Pada tahap ini dilakukan proses wawancara dan pengumpulan data dengan menggunakan metode kuisioiner langsung terhadap calon pengguna sistem nantinya.

b) Ideation

Tahapan kedua yaitu melakukan proses *generate* solusi atas permasalahan yang sudah ditemukan pada tahapan *observation* sebelumnya. Pada tahapan inilah unsur *human-centered design* berperan penting sebagai acuan untuk merancang dan membangun sistem yang akan berfokus kepada *user experience* calon pengguna nantinya.

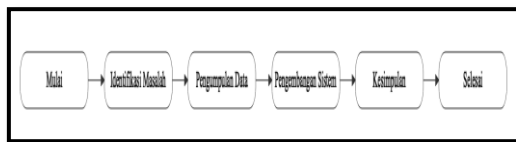
c) Prototyping

Tahapan ketiga yaitu merancang dan membangun sebuah model *prototype* yang dapat diuji cobakan langsung oleh calon pengguna. Pada tahapan ini juga algoritma *winnowing* diterapkan.

d) *Testing*

Tahapan terakhir yaitu melakukan pengujian sistem. Pada tahapan ini merupakan proses uji coba yang dilakukan untuk mengetahui sejauh mana fungsional sistem dan kualitas sistem yang telah dihasilkan. Pengujian ini dilakukan dengan menggunakan metode *black box* dan *heuristic evaluation*. *Heuristic evaluation* nantinya akan dilakukan oleh beberapa evaluator yang merupakan pakar dari sistem tersebut.

B. *Diagram Alir Penelitian*



Gambar 4. Diagram alir penelitian

Secara umum proses tahapan yang akan dilakukan berdasarkan Gambar 4 yaitu:

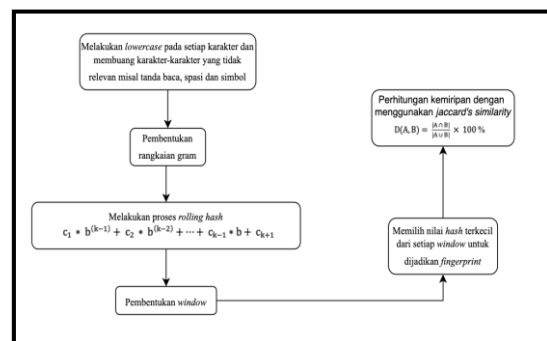
- a) Identifikasi masalah diperlukan untuk mengetahui apa saja yang menjadi masalah pada sistem sebelumnya. Proses identifikasi masalah meliputi latar belakang, tujuan, manfaat dan ruang lingkup penelitian.
- b) Tahapan selanjutnya merupakan pengumpulan data. Pada tahapan ini data-data akan dikumpulkan dalam penelitian ini berperan sebagai bahan pendukung. Data yang dikumpulkan merupakan data dosen, mahasiswa dan KP.
- c) Tahapan selanjutnya ialah pengembangan sistem. Pada tahapan ini pendekatan *human-centered design* dan algoritma *winning* diimplementasikan.
- d) Selanjutnya ialah kesimpulan. Pada tahap ini dilakukan untuk mengetahui sejauh mana hasil dan tercapai atau tidaknya tujuan dari penelitian. Tahap ini akan menghasilkan

kesimpulan dari hasil implementasi pendekatan *Human Centered Design* sebagai aspek utama dalam mencapai kebutuhan penyelesaian masalah calon pengguna dan algoritma *winning* pada sistem informasi pengelolaan KP dalam menentukan kemungkinan plagiasi terhadap KP yang diajukan oleh mahasiswa yang nantinya akan diukur dengan uji coba atau pengujian oleh pakar yang merupakan beberapa dosen yang menjadi ahli dalam bidang pengelolaan KP itu sendiri di Program Studi Informatika.

C. *Diagram Alir Algoritma Winning*

Algoritma *winning* adalah salah satu algoritma pencocokan *string*. Jadi pada dasarnya algoritma *winning* merupakan sebuah algoritma yang melakukan perbandingan dan pencocokan string untuk mendeteksi tingkat kesamaan sebuah dokumen atau teks [7]. Berikut langkah-langkah yang dapat dilakukan dalam penerapan Algoritma *Winning* [3]:

- a) Pembuangan karakter yang tidak relevan.
- b) Pembentukan rangkaian *n-gram*.
- c) Perhitungan fungsi *hash* untuk setiap *n-gram*.
- d) Pembentukan *window* dari nilai *hash*.
- e) Pemilihan *fingerprint* dari setiap *window*.
- f) Persamaan *Jaccard Coefficient*.



Gambar 5. Alir algoritma *winning*

Gambar 5 merupakan proses kerja algoritma *winnowing*, pada tahap pertama dilakukan proses *lowercase* dan pembuangan karakter yang tidak relevan pada teks yang akan digunakan, setelah proses tersebut dilakukan tahapan selanjutnya adalah membentuk rangkaian *n-gram* pada teks di mana nilai $n = 5$, setelah rangkaian *n-gram* terbentuk maka proses selanjutnya yakni *rolling hash* yang bertujuan untuk menghitung *hash* dari setiap *n-gram* yang terbentuk sebelumnya dengan Rumus (1).

$$c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \dots + c_{k-1} * b + c_{k+1} \dots \dots (1)$$

keterangan:

c = nilai ASCII karakter

b = basis bilangan prima

k = nilai *k-gram* / banyaknya karakter pada *gram*

Kemudian setelah nilai masing-masing *hash* ditemukan maka dilakukan proses pembentukan *window* di mana nilai $w = 4$ dan diambil nilai terkecil setiap *window* untuk dijadikan *fingerprint*. Tahapan terakhir ialah menentukan nilai kemiripan dengan membandingkan *fingerprint* A dan *fingerprint* B menggunakan Rumus (2), yaitu persamaan *jaccard's similarity*.

$$D(A, B) = \frac{|A \cap B|}{|A \cup B|} \times 100 \% \dots \dots (2)$$

Dalam prosesnya, algoritma *winnowing* yang baik akan menyimpan posisi *fingerprint* pertama dimulai dari 0 dari nilai *hash gram* yang terbentuk sebelumnya. Beberapa kasus menyimpulkan penyimpanan posisi *fingerprints* pada sebuah dokumen akan sangat mendukung proses berjalannya algoritma *winnowing* [18].

D. Metode Pengujian

Metode *heuristic evaluation* menggunakan 10 prinsip dasar dalam mengevaluasi tingkat *usability* untuk menentukan pendukung keputusan yang akan diambil pengguna nantinya [6]. Desain

interaksi dalam *heuristic evaluation* menurut Jakob Nielsen mempunyai 10 prinsip dasar, di antaranya [23]: *Visibility of system status*, *Match between system and the real world*, *User control and freedom*, *Consistency and standards*, *Error prevention*, *Recognition rather than recall*, *Flexibility and efficiency of use*, *Aesthetic and minimalist design*, *Help users recognize, diagnose, and recover from errors*, dan *Help and documentation*

IV. ANALISIS DAN PERANCANGAN

A. Analisis Sistem

Sistem yang akan dihasilkan nantinya yaitu berupa sebuah sistem informasi pengelolaan KP dengan meneegggunakan pendekatan *human-centered design* untuk menciptakan pengalaman calon pengguna dan algoritma *winnowing* untuk membantu mengetahui tingkat kesamaan KP yang diajukan.

B. Analisis Fungsional

Analisis fungsional merupakan analisis yang berisikan proses ataupun fungsi yang dapat dilakukan oleh sistem nantinya. Terdapat 3 *user* dalam sistem yaitu mahasiswa, dosen, dan koordinator KP.

C. Perancangan Sistem

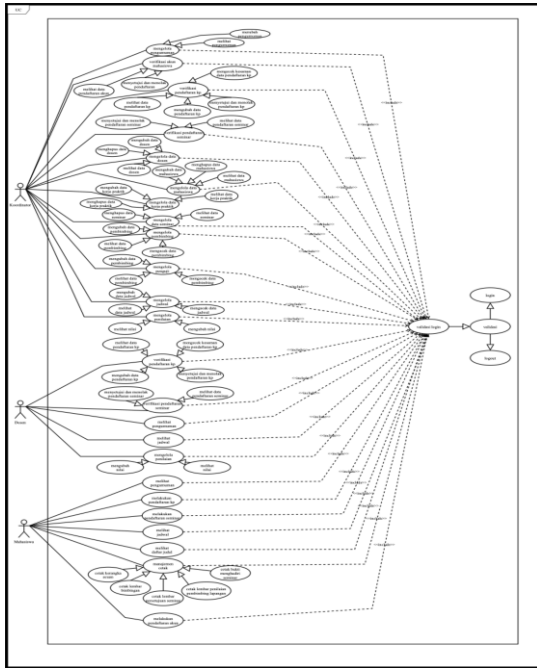
Perancangan sistem yang digunakan yakni perancangan UML (*Unified Modelling Language*) serta ERD (*Entity Relationship Diagram*). Untuk menggambarkan hubungan dan interaksi yang ada pada sistem dapat dilihat pada Gambar 6 yang merupakan *usecase diagram* dan juga Gambar 7 merupakan penggambaran hubungan antara entitas dan basis data pada sistem yang merupakan ERD.

Pemodelan ERD (*Entity Relationship Diagram*) dimaksudkan untuk memodelkan

rancangan basis data. Berdasarkan Gambar 7, terdapat 7 entitas yakni mahasiswa, dosen, KP, seminar, lokasi, nilai dan kegiatan. Setiap entitas memiliki atributnya masing-masing dan setiap entitas juga mempunyai relasi (hubungan) antar entitas lainnya.

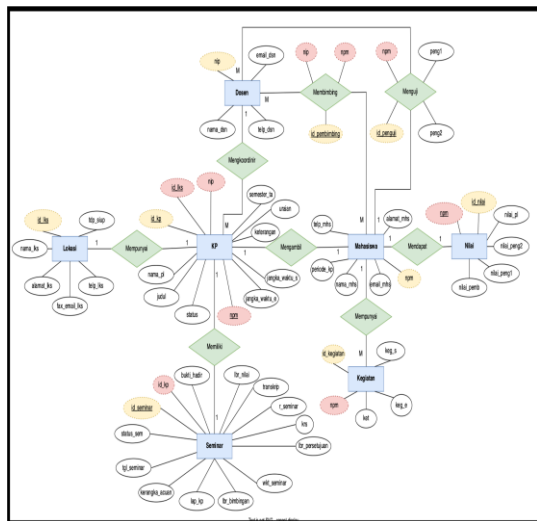
D. Perancangan Antarmuka (interface)

a) Usecase Diagram

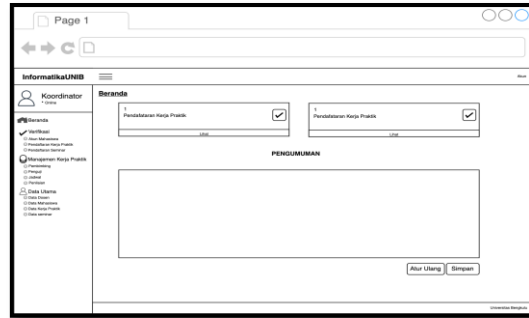


Gambar 6. Usecase diagram

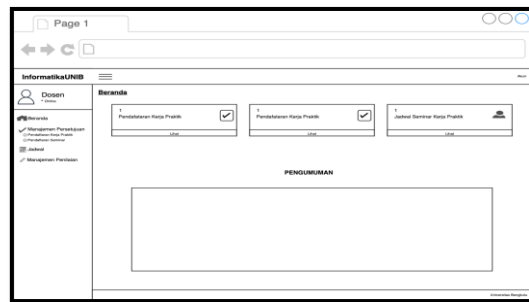
b) ERD (Entity Relationship Diagram)



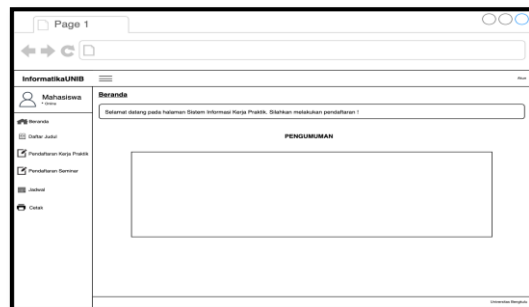
Gambar 7. Entity Relationship Diagram (ERD)



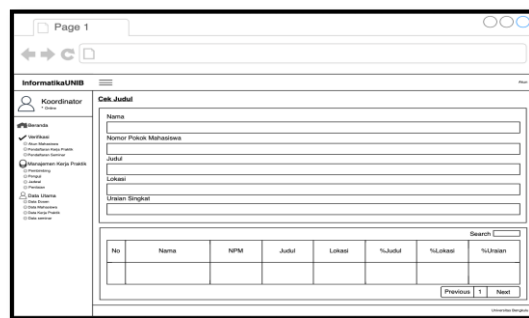
Gambar 8. Rancangan UI halaman user koordinator



Gambar 9. Rancangan UI halaman user dosen



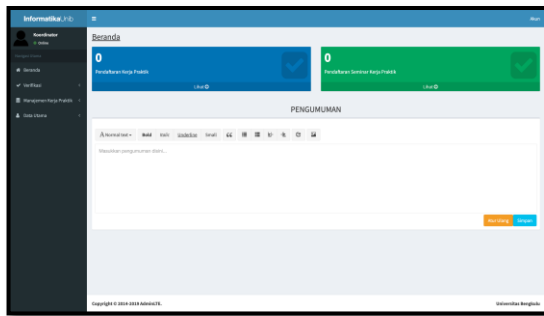
Gambar 10. Rancangan UI halaman user mahasiswa



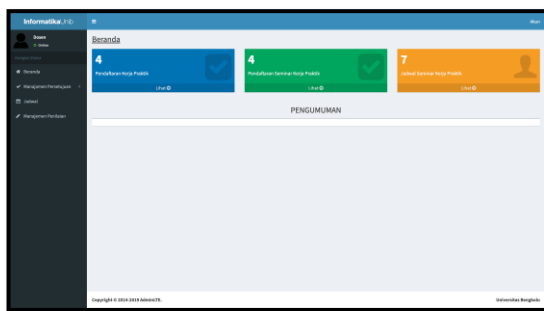
Gambar 11. Rancangan UI halaman cek kesamaan judul

V. HASIL DAN PEMBAHASAN

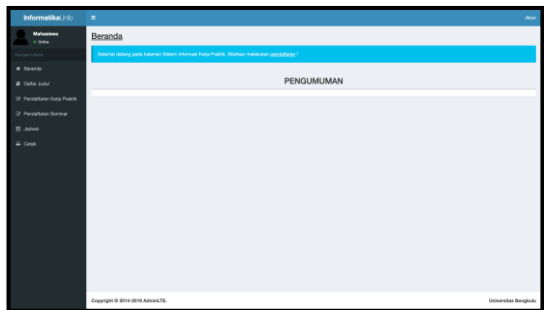
A. Implementasi Antarmuka (interface)



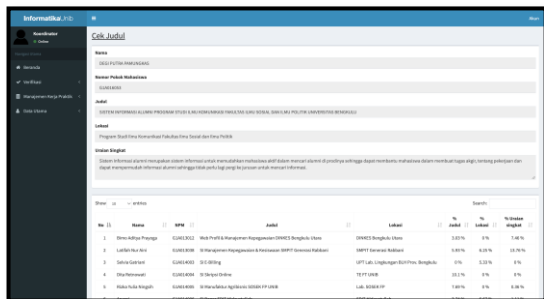
Gambar 12. UI Halaman *user* koordinator



Gambar 13. UI halaman *user* dosen



Gambar 14. UI halaman *user* mahasiswa



Gambar 15. UI halaman cek kesamaan judul

B. Pengujian Sistem

a) Pengujian *Black Box*

Pada pengujian ini terdapat 93 skenario dari *black box testing*. Dari 93 skenario yang diuji, menghasilkan nilai pengujian fungsional sebesar:

$$\text{Keberhasilan Fungsional} = \frac{93}{93} \times 100\% = 100\%$$

b) Pengujian *Heuristics Evaluation*

Pada tahap ini terdapat 10 poin yang dijadikan acuan untuk dilakukan pengujian dengan skor penilaian yang memiliki rentang nilai 1-5 yaitu sangat tidak setuju (1), tidak setuju (2), ragu-ragu (3), setuju (4), dan sangat setuju (5). Pengujian ini dilakukan oleh 5 responden yang untuk menilai UI/UX sistem. Pengujian ini dilakukan dengan melibatkan 3 sivitas akademika internal yang terdiri dari 3 dosen serta 2 penguji eksternal. Berikut merupakan data hasil pengujian yang dapat dilihat pada Tabel 2.

Tabel 2. Hasil pengujian *heuristic evaluation*

Heuristics	Skor				
	R1	R2	R3	R4	R5
<i>Visibility of system status</i>	4	4	5	4	4
<i>Match between system and the real world</i>	4	4	5	5	3
<i>User control and freedom</i>	5	3	4	4	4
<i>Consistency and standards</i>	4	4	5	3	3
<i>Error prevention</i>	4	4	5	3	5
<i>Recognition rather than recall</i>	4	4	5	4	5
<i>Flexibility and efficiency of use</i>	5	4	5	4	5
<i>Aesthetic and minimalist design</i>	5	4	5	4	2
<i>Help users recognize, diagnose, and recover from errors</i>	4	4	5	4	5
<i>Help and documentation</i>	3	2	3	3	1

Pada Tabel 2 merupakan tabel hasil pengujian dari 5 responden yang memberikan skor pada setiap masing-masing poin *heuristic* yang dijadikan acuan dalam pengujian. Kemudian hasil pengujian ini akan dihitung dengan menggunakan perhitungan skala *likert* dengan Rumus (3).

Total Skor = T × Pn (3)

Keterangan:

T = Total jumlah responden yang memilih

Pn = Pilihan angka skor likert

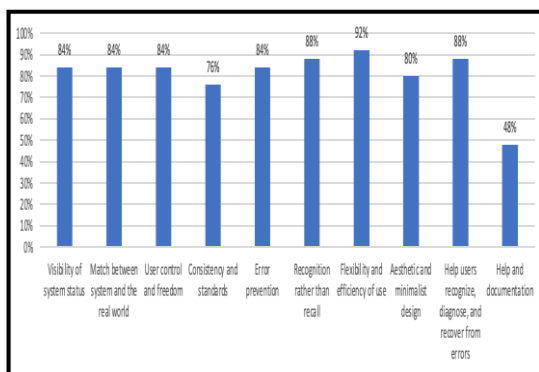
$$\text{Index \%} = \frac{\text{Total Skor}}{Y} \times 100 \%$$

Keterangan:

Y= Skor tertinggi likert x jumlah responden

Hasil persentase:

1. *Visibility of system status*
 $\frac{21}{25} \times 100\% = 84\%$
2. *Match between system and the real world*
 $\frac{21}{25} \times 100\% = 84\%$
3. *User control and freedom*
 $\frac{21}{25} \times 100\% = 84\%$
4. *Consistency and standards*
 $\frac{19}{25} \times 100\% = 76\%$
5. *Error prevention*
 $\frac{21}{25} \times 100\% = 84\%$
6. *Recognition rather than recall*
 $\frac{22}{25} \times 100\% = 88\%$
7. *Flexibility and efficiency of use*
 $\frac{23}{25} \times 100\% = 92\%$
8. *Aesthetic and minimalist design*
 $\frac{20}{25} \times 100\% = 80\%$
9. *Help users recognize, diagnose, and recover from errors*
 $\frac{22}{25} \times 100\% = 88\%$
10. *Help and documentation*
 $\frac{12}{25} \times 100\% = 48\%$



Gambar 17. Grafik persentase pengujian *heuristic evaluation*

Pada Gambar 17 merupakan grafik hasil persentase dari pengujian *heuristic evaluation*.

Kemudian total keseluruhan rata-rata persentase pengujian *heuristic evaluation* menggunakan skala *likert* yakni 80,8% yang mana ini mengindikasikan sistem masuk dalam kriteria interpretasi sangat baik berdasarkan perhitungan skala *likert*.

c) Pengujian Algoritma *Winnowing*

Pengujian ini dilakukan untuk mengetahui kesesuaian penerapan algoritma *winnowing* yang digunakan. Pengujian ini dilakukan dengan membandingkan hasil perhitungan manual dengan hasil dari persentase yang dihasilkan oleh sistem. Sampel data yang diuji coba dalam pembahasan ini terdapat pada Tabel 3 yaitu tabel yang berisi data nama mahasiswa dan judul KP.

Tabel 3. Data judul kerja praktik

No.	Nama Mahasiswa	Judul
1.	Rizky Amanah Putra	Sistem Informasi Pengajuan Cuti Pegawai
2.	Kurnia Nutriana Sitepu	SI Cuti Pegawai
3.	Faisen Bakti	SI Tambahan Penghasilan Pegawai
4.	Mahdarina Nasution	SIM Cuti Pegawai
5.	M. Yanuar Afif Furqon	SI Sasaran Kerja Pegawai Dosen
6.	Syifa Mawaddah	SI Penilaian Kinerja Pegawai Honoror

Perhitungan Algoritma *Winnowing*:

Langkah 1: *lowercase* pada setiap karakter dan pembuangan karakter yang tidak penting.

Langkah 2: pembentukan rangkaian *gram*, dimana $k = 5$.

Langkah 3: melakukan proses *rolling hash*, di mana $b = 2$.

$$c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \dots + c_{k-1} * b + c_{k+1}$$

Keterangan :

c =Nilai ASCII karakter

b =basis bilangan prima

k =nilai *k-gram* / banyaknya karakter pada *gram*

Langkah 4: pembentukan *window*, di mana $w = 4$.

Langkah 5: membentuk *fingerprints* dari *hash* terkecil masing-masing window.

Langkah 6: perhitungan kemiripan dengan menggunakan *jaccard's similarity*

$$D(A, B) = \frac{|A \cap B|}{|A \cup B|} \times 100$$

Perhitungan manual data ke-1 (data yang dibandingkan):

1. *Lowercase* pada setiap karakter dan pembuangan karakter yang tidak penting. sisteminformasipengajuancutipegawai
2. Pembentukan rangkaian *gram*, di mana $k = 5$.
siste istem stemi temin eminf minfo infor nform forma ormas rmasi masip asipe sipen ipeng penga engaj ngaju gajua ajuan juanc uancu ancun cuti cutip utipe tipeg ipega pegaw egawa gawai

3. Melakukan proses *rolling hash*, dimana $b = 2$.

$$c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \dots + c_{k-1} * b + c_{k+1}$$

Keterangan:

c : Nilai ASCII karakter

b : basis bilangan prima

k : nilai *k-gram* / banyaknya karakter pada *gram*

$$\begin{aligned} h_{(siste)} &= c_{(s)} * 2^{(5-1)} + c_{(i)} * 2^{(5-2)} + c_{(s)} \\ &\quad * 2^{(5-3)} + c_{(t)} * 2^{(5-4)} \\ &\quad + c_{(e)} * 2^{(5-5)} \\ &= 115 * 2^4 + 105 * 2^3 \\ &\quad + 115 * 2^2 + 116 * 2^1 \\ &\quad + 101 * 2^0 \\ &= 115 * 16 + 105 * 8 \\ &\quad + 115 * 4 + 116 * 2 + 101 \\ &\quad * 1 \\ &= 1840 + 840 + 460 + 232 \\ &\quad + 101 = \mathbf{3473} \end{aligned}$$

Kemudian untuk menghitung nilai *hash gram* berikutnya digunakan persamaan:

$$h = c_{k-1} * b + c_{k+1}$$

$$\begin{aligned} h_{(istem)} &= (h_{(siste)} - c_{(s)} * 2^{5-1}) * 2 + c_{(m)} \\ &= (3473 - 115 * 16) * 2 \\ &\quad + 109 \\ &= (3473 - 1840) * 2 + 109 \\ &= 1633 * 2 + 109 = \mathbf{3375} \end{aligned}$$

3473 3375 3495 3420 3230 3339 3304 3357
3291 3433 3419 3302 3217 3440 3303 3343 3208
3301 3179 3172 3339 3403 3178 3357 3306 3545
3449 3283 3325 3163 3199

4. Pembentukan *window*, di mana $w = 4$.

[3473|3375|3495|3420], [3375|3495|3420|3230],
[3495|3420|3230|3339], [3420|3230|3339|3304],
[3230|3339|3304|3357], [3339|3304|3357|3291],
[3304|3357|3291|3433], [3357|3291|3433|3419],
[3291|3433|3419|3302], [3433|3419|3302|3217],
[3419|3302|3217|3440], [3302|3217|3440|3303],
[3217|3440|3303|3343], [3440|3303|3343|3208],
[3303|3343|3208|3301], [3343|3208|3301|3179],
[3208|3301|3179|3172], [3301|3179|3172|3339],
[3179|3172|3339|3403], [3172|3339|3403|3178],
[3339|3403|3178|3357], [3403|3178|3357|3306],
[3178|3357|3306|3545], [3357|3306|3545|3449],
[3306|3545|3449|3283], [3545|3449|3283|3325],
[3449|3283|3325|3163], [3283|3325|3163|3199]

5. Memilih nilai *hash* terkecil dari setiap *window* untuk dijadikan *fingerprint*.

[3473|**3375**|3495|3420], [3375|3495|3420|**3230**],
[3495|3420|**3230**|3339], [3420|**3230**|3339|3304],
[**3230**|3339|3304|3357], [3339|3304|3357|**3291**],
[3304|3357|**3291**|3433], [3357|**3291**|3433|3419],
[**3291**|3433|3419|3302], [3433|3419|3302|**3217**],
[3419|3302|**3217**|3440], [3302|**3217**|3440|3303],
[**3217**|3440|3303|3343], [3440|3303|3343|**3208**],
[3303|3343|**3208**|3301], [3343|3208|3301|**3179**],

[3208|3301|3179|**3172**], [3301|3179|**3172**|3339],
[3179|**3172**|3339|3403], [**3172**|3339|3403|3178],
[3339|3403|**3178**|3357], [3403|**3178**|3357|3306],
[**3178**|3357|3306|3545], [3357|**3306**|3545|3449],
[3306|3545|3449|**3283**], [3545|3449|**3283**|3325],
[3449|3283|3325|**3163**], [3283|3325|**3163**|3199]

Finger print: 3375, 3230, 3230, 3230, 3230,
3291, 3291, 3291, 3291, 3217,3217, 3217, 3217,
3208, 3208, 3179, 3172, 3172, 3172, 3172, 3178,
3178, 3178, 3306, 3283, 3283, 3163, 3163

Perhitungan manual data ke-2 (data pembandingan):

1. *Lowercase* pada setiap karakter dan pembuangan karakter yang tidak penting. *sicutipegawai*
2. Pembentukan rangkaian *gram*, dimana $k = 5$. *sicut icuti cutip utipe tipeg ipega pegaw egawa gawai*
3. Melakukan proses *rolling hash*, dimana $b = 2$.
$$c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \dots + c_{k-1} * b + c_{k+1}$$

Keterangan :

c = Nilai ASCII karakter

b =basis bilangan prima

k =nilai k -gram / banyaknya karakter pada *gram*

$$h_{(sicut)} = c_{(s)} * 2^{(5-1)} + c_{(i)} * 2^{(5-2)} + c_{(c)} * 2^{(5-3)} + c_{(u)} * 2^{(5-4)} + c_{(t)} * 2^{(5-5)}$$

$$= 115 * 2^4 + 105 * 2^3 + 99 * 2^2 + 117 * 2^1 + 116 * 2^0$$

$$= 115 * 16 + 105 * 8 + 99 * 4 + 117 * 2 + 116 * 1$$

$$= 1840 + 840 + 396 + 234 + 116 = \mathbf{3426}$$

Kemudian untuk menghitung nilai *hash gram* berikutnya digunakan persamaan :

$$h = c_{k-1} * b + c_{k+1}$$

$$h_{(icuti)} = (h_{(sicut)} - c_{(s)} * 2^{5-1}) * 2 + c_{(i)}$$

$$= (3426 - 115 * 16) * 2 + 105$$

$$= (3426 - 1840) * 2 + 105 = 1586 * 2 + 105 = \mathbf{3277}$$

3426 3277 3306 3545 3449 3283 3325
3163 3199

4. Pembentukan *window*, dimana $w = 4$.

[3426|3277|3306|3545], [3277|3306|3545|3449],
[3306|3545|3449|3283], [3545|3449|3283|3325],
[3449|3283|3325|3163], [3283|3325|3163|3199]

5. Membentuk *fingerprints* dari *hash* terkecil masing-masing *window*.

[3426|**3277**|3306|3545], [**3277**|3306|3545|3449],
[3306|3545|3449|**3283**], [3545|3449|**3283**|3325],
[3449|3283|3325|**3163**], [3283|3325|**3163**|3199]

Finger print: 3277, 3277, 3283, 3283, 3163, 3163

6. Perhitungan kemiripan dengan menggunakan *jaccard's similarity*

Finger print A = 3375, 3230, 3230, 3230, 3230,
3291, 3291, 3291, 3291, 3217,3217, 3217, 3217,
3208, 3208, 3179, 3172, 3172, 3172, 3172,
3178, 3178, 3178, 3306, **3283, 3283, 3163, 3163**

Finger print B = 3277, 3277, **3283, 3283, 3163, 3163**

$$D(A, B) = \frac{|A \cap B|}{|A \cup B|} \times 100 \%$$

$$= \frac{4}{30} \times 100\% = 13,33 \%$$

Tabel 4. Perbandingan Persentase Kesamaan Pada Sistem

	D-2	D-3	D-4	D-5	D-6
D-1	13,33%	4,26	16,67%	6,82%	3,92%

Pada Tabel 4 merupakan daftar data judul KP yang dijadikan bahan sampel untuk dilakukan pengujian algoritma *winnowing*, data yang dijadikan bahan pengujian merupakan data dari 6

mahasiswa yang terdiri dari 1 data pembandingan dan 5 data yang dibandingkan. Kemudian pada Tabel 4 merupakan perbandingan hasil persentase kesamaan antara tiap masing-masing data seperti persentase kesamaan antara data ke-1 dan data ke-2 senilai 13,33%, persentase kesamaan antara data ke-1 dan data ke-3 senilai 4,26%, persentase kesamaan data ke-1 dan data ke-4 senilai 16,67%, persentase kesamaan data ke-1 dan data ke-5 senilai 6,82%, persentase kesamaan data ke-1 dan data ke-6 senilai 3,92%.

VI. KESIMPULAN

Berdasarkan hasil dan pembahasan yang telah dijabarkan sebelumnya, maka kesimpulan yang dapat diambil adalah penelitian ini telah menghasilkan sebuah sistem informasi pengelolaan KP yang berbasis UX dengan menerapkan pendekatan HCD. Algoritma *winnowing* yang diterapkan juga terbukti dapat membantu mengetahui tingkat plagiat atau kesamaan pengajuan KP. Hal itu dilihat berdasarkan hasil pengujian algoritma *winnowing* yang telah dijabarkan pada bab sebelumnya, untuk mencari masing-masing *fingerprint* data, setiap *n-gram* yang ada maka dilakukan proses *rolling hash* data, kemudian didapatkanlah beberapa nilai persentase kemiripan antar sampel data yang diuji, nilai persentase kemiripan judul KP sangat kecil yakni berkisar 3,92-16,67%. Pengujian *black box* telah menghasilkan keberhasilan fungsional sebesar 100%. Berdasarkan hasil pengujian *heuristic evaluation* yang diuji dengan menggunakan perhitungan menggunakan skala likert telah menghasilkan nilai persentase skor keseluruhan sebesar 80,8% dengan interpretasi kualitas sistem sangat baik.

REFERENSI

- [1] E. G. Ramadhan, "Human Centered Design," *UNIKOM Codelabs*, 2017.
- [2] H. Wijaya, H. Tolle, and H. M. Az-Zahra, "Perancangan User Experience Aplikasi Pemesanan Katering Sekolah Dengan Menggunakan Metode Human-Centered Design," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, p. 964X, 2019.
- [3] S. Astutik, A. D. Cahyani, and M. K. Sophan, "Sistem Penilaian Esai Otomatis Pada E-Learning Dengan Algoritma Winnowing," *J. Inform.*, vol. 12, no. 2, pp. 47–52, 2014.
- [4] A. H. Pratomo and A. P. Suryotomo, "Implementasi Pengecekan Plagiarisme Proposal Tugas Akhir Mahasiswa Teknik Informatika UPN Veteran Yogyakarta," in *Seminar Nasional Informatika (SEMNASIF)*, 2020, vol. 1, no. 1, pp. 221–229.
- [5] O. Kharisman, B. Susanto, and S. Suwarno, "Implementasi Algoritma Winnowing Untuk Mendeteksi Kemiripan Pada Dokumen Teks," *J. Inform.*, vol. 9, no. 1, 2013.
- [6] U. Ependi, F. Panjaitan, and H. Hutrianto, "System Usability Scale Antarmuka Palembang Guide Sebagai Media Pendukung Asian Games XVIII," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 3, no. 2, pp. 80–86, 2017.
- [7] N. Dalimunthe, F. Nazari, and K. Purba, "Evaluasi Website Pemko Pekanbaru Menggunakan Metode Heuristic Evaluation," *J. Ilm. Rekayasa dan Manaj. Sist. Inf.*, vol. 5, no. 2, pp. 245–250, 2019.
- [8] A. Ridwan, "Pengukuran usability aplikasi menggunakan evaluasi heuristik," *J. Inform. Komput.*, vol. 12, no. 3, pp. 220–222, 2007.
- [9] B. A. Priyaungga, D. B. Aji, M. Syahroni, N. T. S. Aji, and A. Saifudin, "Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions," *J. Teknol. Sist. Inf. Dan Apl.*, vol. 3, no. 3, pp. 150–157, 2020.
- [10] T. Hidayat and M. Muttaqin, "Pengujian sistem informasi pendaftaran dan pembayaran wisuda online menggunakan black box testing dengan metode equivalence partitioning dan boundary value analysis," 2020.
- [11] A. Aristoteles, W. Wardiyanto, and A. A. Pratama, "Sistem Pakar Diagnosa Penyakit Pada Ikan Budidaya Air Tawar dengan Metode Forward Chaining," *J. Komputasi*, vol. 3, no. 2, 2016.
- [12] D. Winograd, T., & Woods, "The Challenge of

- Human-Centered Design. HumanCentered Systems: Information, Interactivity, and Intelligence,” 1997.
- [13] I. N. Arifin, H. Tolle, and R. I. Rokhmawati, “Evaluasi dan Perancangan User Interface untuk Meningkatkan User Experience menggunakan Metode Human-Centered Design dan Heuristic Evaluation pada Aplikasi Ezyschool,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, p. 964X, 2018.
- [14] M. M. Haekal, “User Experience (UX): Pengertian dan Tips Penerapannya untuk Pemula,” 2020. .
- [15] A. S. Wijaya, “Human Centered Design dan Perbedaan dengan User Centered Design,” 2019. .
- [16] E. Y. Puspaningrum, B. Nugroho, A. Setiawan, and N. Hariyanti, “Detection of text similarity for indication plagiarism using winnowing algorithm based K-gram and jaccard coefficient,” in *Journal of Physics: Conference Series*, 2020, vol. 1569, no. 2, p. 22044.
- [17] D. Leman, M. Rahman, F. Ikorasaki, B. S. Riza, and M. B. Akbbar, “Rabin karp and Winnowing algorithm for Statistics of text document plagiarism detection,” in *2019 7th International Conference on Cyber and IT Service Management (CITSM)*, 2019, vol. 7, pp. 1–5.
- [18] N. Elbegbayan, “Winnowing, a Document Fingerprinting Algorithm,” *TDDC03 Proj. Spring*, 2005.
- [19] M. I. S. B. Khairat, Y. Priyadi, and M. Adrian, “Usability Measurement in User Interface Design Using Heuristic Evaluation & Severity Rating (Case Study: Mobile TA Application based on MVVM),” in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 2022, pp. 974–979.
- [20] I. N. Arifin, H. Tolle, and R. I. Rokhmawati, “Evaluasi dan Perancangan User Interface untuk Meningkatkan User Experience menggunakan Metode Human-Centered Design dan Heuristic Evaluation pada Aplikasi Ezyschool,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, p. 964X, 2019.
- [21] International Organization for Standardization, “ISO 9241-210: Ergonomics of human–system interaction - Human-centred design for interactive systems,” 2010.
- [22] E. G. Ramadhan, “Human Centered Design,” 2013. .
- [23] Jakob Nielsen, “10 Usability Heuristics for User Interface Design.” .