

**PENERAPAN ALGORITMA HORSPPOOL SEBAGAI
PENDETEKSI KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus Teknik Informatika Universitas
Muhammadiyah Ponorogo)**

SKRIPSI

Diajukan Sebagai Salah satu Syarat
Untuk Memperoleh Gelar Sarjana Jenjang Strata Satu (S1)
Pada Program Studi Teknik Informatika Fakultas Teknik
Universitas Muhammadiyah Ponorogo



FUAD AINUR RAFIQ

19533156

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PONOROGO**

2024

HALAMAN PENGESAHAN SKRIPSI

Nama : Fuad Ainur Rafiq
Nim : 19533156
Program Studi : Teknik Informatika
Fakultas : Teknik
Judul Skripsi : Penerapan Algoritma Horspool Sebagai Pendeteksi
Kemiripan Judul Skripsi (Studi Kasus Teknik
Informatika Universitas Muhammadiyah Ponorogo)

Isi dan formatnya telah disetujui dan dinyatakan memenuhi syarat
Untuk melengkapi persyaratan guna memperoleh Gelar Sarjana
pada Program Studi Teknik Informatika Fakultas Teknik
Universitas Muhammadiyah Ponorogo

Ponorogo, 2 Agustus 2024

Menyetujui

Dosen Pembimbing Utama



(Arin Yuli Astuti, S.Kom., M.Kom.)
NIK. 19890717/201309 13

Dosen Pembimbing Pendamping



(Dyah Mustikasari, S.T., M.Eng.)
NIK. 19871007 201609 13

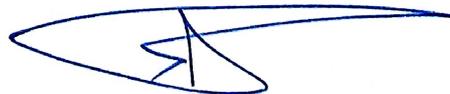
Mengetahui

Dekan Fakultas Teknik,



(Edy Kurniawan, ST., M.T.)
NIK. 19771026 200810 12

Ketua Program Studi
Teknik Informatika,



(Adi Fajaryanto C., S. Kom, M.Kom.)
NIK. 19840924 201309 13

PERNYATAAN ORISINALITAS SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Fuad Ainur Rafiq

Nim : 19533156

Program Studi : Teknik Informatika

Dengan ini menyatakan bahwa Skripsi saya dengan judul : “Penerapan Algoritma Horspool Sebagai Pendeteksi Kemiripan Judul Skripsi (Studi Kasus Teknik Informatika Universitas Muhammadiyah Ponorogo)” bahwa berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang saya rancang/ teliti di dalam Naskah Skripsi ini adalah hasil dari pemikiran saya. Tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam Naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur plagiatisme, saya bersedia Ijazah saya dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Demikian pernyataan ini saya dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Ponorogo, 12 Juli 2024

Mahasiswa,



Fuad Ainur Rafiq

NIM.19533156

HALAMAN BERITA ACARA UJIAN

Nama : Fuad Ainur Rafiq
Nim : 19533156
Program Studi : Teknik Informatika
Fakultas : Teknik
Judul Skripsi : Penerapan Algoritma Horspool Sebagai Pendeteksi
Kemiripan Judul Skripsi (Studi Kasus Teknik
Informatika Universitas Muhammadiyah Ponorogo)

Telah diuji dan dipertahankan dihadapan

Dosen penguji tugas akhir jenjang Strata Satu (S1) pada

Hari : Kamis

Tanggal : 25 Juli 2024

Dosen Penguji

Ketua Penguji

(Arin Yuli A., M.Kom.)
NIK. 19890717 201309 13

Anggota Penguji I

(Ir. Moh. Bhanu S., S.T., M.Kom.)
NIK. 19800225 201309 13

Anggota Penguji II

(Ghulam Asrofi B., S.T., M.Eng.)
NIK. 19870723 202109 12

Mengetahui

Dekan Fakultas Teknik

(Eddy Kurniawan, S.T., M.T.)
NIK. 19771026 200810 12

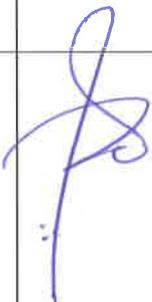
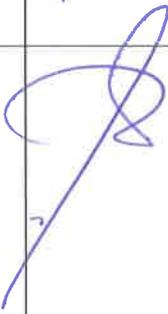
Ketua Program Studi
Teknik Informatika

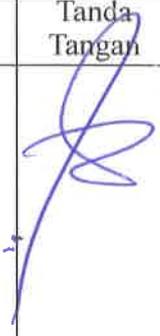
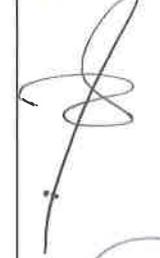
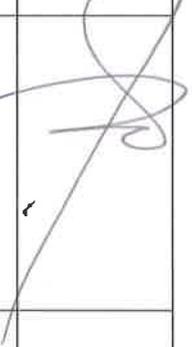
(Adi Fajaryanto C., S.Kom., M.Kom.)
NIK. 19840924 201309 13

BERITA ACARA BIMBINGAN SKRIPSI

Nama : Fuad Ainur Rafiq
NIM : 19533156
Judul Skripsi : Penerapan Algoritma Horspool Sebagai
Pendeteksi Kemiripan Judul Skripsi
Dosen Pembimbing I : Arin Yuli Astuti, S.Kom., M.Kom

PROSES PEMBIMBINGAN

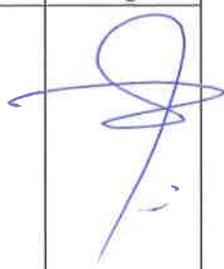
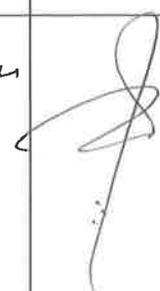
No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	15/23 103	Bab I	Revisi bab I	
2	27/23 103	bab I	revisi masalah Glossary	
3	12/23 104	Revisi	Revisi bab I	
4	05/05 ²³	bab I	Revisi bab I	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	19/23 06	bab 8	Variabel belum ada	
6	08/23 06	Bab 11	Variabel ciri dasarnya - Teori lebih di perbanyak.	
7	16/23 06	Bab 11	Algoritma persentase • belum ada	
8	22/23 06	ABC Supro	ABC Supro	
9				
10				

**BERITA ACARA
BIMBINGAN SKRIPSI**

Nama : Fuad Ainur Rafiq
NIM : 19533156
Judul Skripsi : Penerapan Algoritma Horspool Sebagai
Pendeteksi Kemiripan Judul Skripsi
Dosen Pembimbing I : Atin Yuli Astuti, S.Kom., M.Kom.

PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	15/23/21	Bab III	Revisi bab III	
2	18/05/24	Bab IV	Aplikasi	
3	24/06/24	Bab V	ERD sama men disamakan	
4	26/06/24	Bab IV	ERD	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
11				
12				
13				
14				
15				
16	10/24 /07	All Sidemg	All sidemg	

BERITA ACARA BIMBINGAN SKRIPSI

Nama : Fuad Ainur Rafiq
 NIM : 19533156
 Judul Skripsi : Penerapan Algoritma Horspool Sebagai
 Pendeteksi Kemiripan Judul Skripsi
 Dosen Pembimbing II : Dyah Mustikasari, S.T., M. Eng

PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	31/5 2023	Bab 1-2	Penulisan tabel. Daftar pustaka kutipan.	
2	22/6 2023	Bab 1-3	Perbaiki kesalahan caption tabel, gambar, persamaan/ rumus. Sebut di dlm paragraf.	
3	24/6 2023		Gamba 3.2. 3.7. 3.8. 3.3. 3.4 3.5 3.6.	
4			Gamba 3.14. 3.16. Belum disebut di paragraf.	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	27/6 ²³		Perbaiki di caption gambar. (tergeser).	
6	3/7 ²³		AEC sempro	
7				
8				
9				
10				

BERITA ACARA BIMBINGAN SKRIPSI

Nama : Fuad Ainur Rafiq
NIM : 19533156
Judul Skripsi : Penerapan Algoritma Horspool sebagai
Perdeteksi Kemiripan Judul Skripsi
Dosen Pembimbing II : Dyah Mustikasari, S.T., M.Eng

PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	5/12/2023	Bab 3	perbaikan Bab 3	
2	10/7/2024		Ace sedang	
3				
4				

SURAT KETERANGAN HASIL PLAGIASI SKRIPSI



UNIVERSITAS MUHAMMADIYAH PONOROGO LEMBAGA LAYANAN PERPUSTAKAAN (L2P)

Jl. Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia

Telp (0352) 481124, Fax. (0352) 461796,

e-mail : akademik@umpo.ac.id Website : www.umpo.ac.id

Akreditasi Institusi B oleh BAN-PT

(SK Nomor : 77/SK/BAN-PT/Ak-PPJ/PT/IV/2020)

NPP.3502102D2014337

SURAT KETERANGAN HASIL *SIMILIARITY CHECK* KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah dengan rincian sebagai berikut:

Nama : Fuad Ainur Rafiq

NIM : 19533156

Prodi : Teknik Informatika

Judul : Penerapan Algoritma Horspool sebagai Pendeteksi Kemiripan Judul Skripsi (Studi Kasus=Teknik Informatika Universitas Muhammadiyah Ponorogo)

Dosen pembimbing :

1. Arin Yuli Astuti, S.Kom., M.Kom

2. Dyah Mustikasari, ST, M.Eng

Telah dilakukan check plagiasi berupa Skripsi di L2P Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar 17 %

Demikian keterangan ini dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 18 Juli 2024

Kepala L2P



Ayu Wulansari, S.Kom, M.A

NIK. 197608 11 200111 21

Nb: Dosen pembimbing dimohon untuk mengecek kembali keaslian soft file karya ilmiah yang telah diperiksa melalui Turnitin perpustakaan

SURAT KETERANGAN HASIL PLAGIASI ARTIKEL



UNIVERSITAS MUHAMMADIYAH PONOROGO
LEMBAGA LAYANAN PERPUSTAKAAN
Jalan Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia
Telp. (0352) 481124, Fax (0352) 461796, e-mail : lib@umpo.ac.id
website : www.library.umpo.ac.id
TERAKREDITASI A
(SK Nomor 00012/ LAP.PT/ I.2017)
NPP. 3502102D2014337

SURAT KETERANGAN HASIL *SIMILARITY CHECK* KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah ilmiah dengan rincian sebagai berikut :

Nama : Fuad Ainur Rafiq
NIM : 19533156
Judul : Penerapan Algoritma Horspool Sebagai Pendeteksi Kemiripan Judul Skripsi
(Studi Kasus Teknik Informatika Universitas Muhammadiyah Ponorogo)
Fakultas / Prodi : Teknik Informatika

Dosen pembimbing :

1. Arin Yuli Astuti, S.Kom., M.Kom.
2. Dyah Mustikasari, S.T., M.Eng.

Telah dilakukan check plagiasi berupa **Artikel** di Lembaga Layanan Perpustakaan Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar **14 %**

Demikian surat keterangan dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 6/Agustus/2024
Kepala Lembaga Layanan Perpustakaan



Ayu Wulansari, S.Kom, M.A
NIK. 19760811 201111 21

NB: Dosen pembimbing dimohon untuk melakukan verifikasi ulang terhadap kelengkapan dan keaslian karya beserta hasil cek Turnitin yang telah dilakukan

MOTTO

“Barang siapa yang menempuh jalan untuk mencari ilmu, maka Allah akan memudahkan baginya jalan menuju surga”.

(HR. Muslim)

“Allah tidak mengatakan hidup ini mudah, tetapi Allah berjanji bahwa sesungguhnya bersama kesulitan ada kemudahan”.

(QS. Al-Insyirah : 5-6)

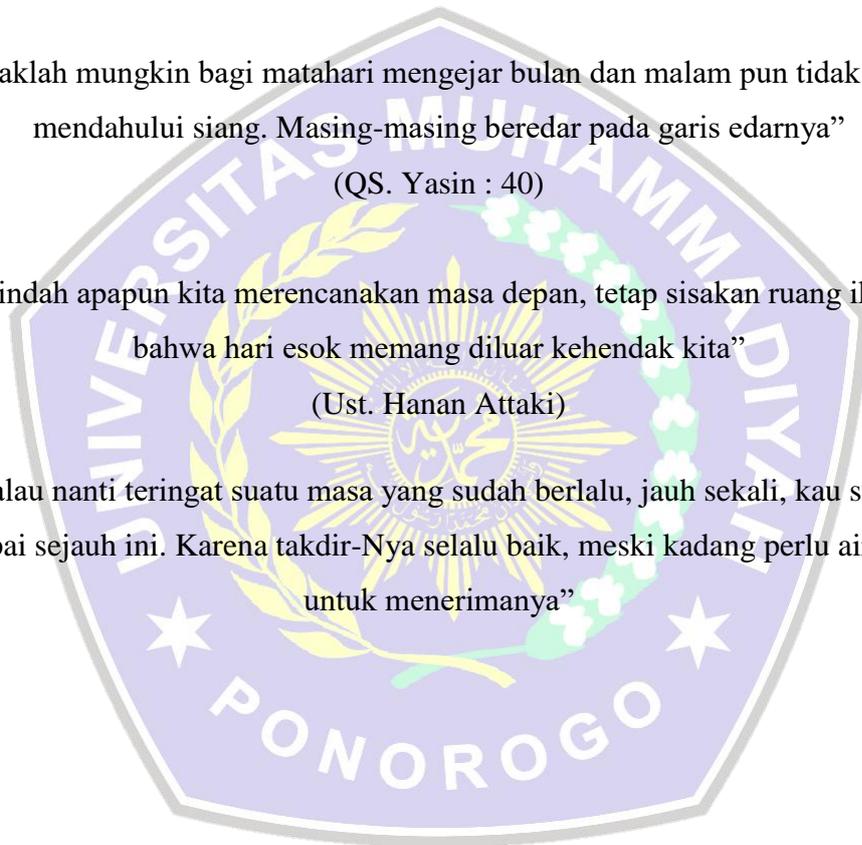
“Tidaklah mungkin bagi matahari mengejar bulan dan malam pun tidak dapat mendahului siang. Masing-masing beredar pada garis edarnya”

(QS. Yasin : 40)

“Seindah apapun kita merencanakan masa depan, tetap sisakan ruang ikhlas bahwa hari esok memang diluar kehendak kita”

(Ust. Hanan Attaki)

“Kalau nanti teringat suatu masa yang sudah berlalu, jauh sekali, kau sudah sampai sejauh ini. Karena takdir-Nya selalu baik, meski kadang perlu air mata untuk menerimanya”



HALAMAN PERSEMBAHAN

Alhamdulillah puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufiq serta hidayahnya sehingga saya dapat menyelesaikan pendidikan jenjang Strata Satu (S1) ini dengan segala solusi atas permasalahan dan hambatan yang saya temui. Dikelilingi kedua orang tua dan orang - orang baik lainnya membuat saya tetap semangat dalam perjalanan ini. Untuk itu, saya persembahkan skripsi ini untuk :

1. Skripsi ini saya persembahkan sepenuhnya kepada dua orang hebat dalam hidup saya, Bapak dan Ibu. Keduanya lah yang membuat segalanya menjadi mungkin sehingga saya bisa sampai pada tahap di mana skripsi ini akhirnya selesai. Terima kasih atas segala pengorbanan, nasihat dan doa baik yang tidak pernah berhenti Bapak dan Ibu berikan.
2. Dosen pembimbing skripsi, Ibu Arin Yuli Astuti dan Ibu Dyah Mustikasari, terima kasih banyak atas ilmunya, sudah dinasehati, sudah membantu, sudah diajari, mengarahkan dan memotivasi saya sampai skripsi ini selesai.
3. Saya ucapkan terimakasih kepada teman-teman terdekat saya, Tri Sakti Budiono, Nandika Bima, Erik Wahyudi dan Salma Safira, mereka yang terus dan tiada henti selalu memberikan semangat dan motivasi dalam proses penyusunan skripsi.
4. Terimakasih sebanyak-banyaknya untuk diri saya sendiri, Fuad Ainur Rafiq yang telah bertahan dalam menikmati proses panjang pengerjaan skripsi ini. Telah melalui berbagai macam hambatan baik dalam pengerjaan maupun dalam revisi. Semoga skripsi ini menjadi karya terbaik saya, dan memotivasi saya untuk lebih belajar lagi.
5. Serta seluruh sahabat dan teman-teman semua, terima kasih telah memberikan semangat, motivasi dan inspirasi dalam menyelesaikan skripsi ini. Semoga doa dan semua hal baik yang kalian berikan menjadikan diri saya untuk bisa lebih baik kedepannya.

**PENERAPAN ALGORITMA HORSPPOOL SEBAGAI PENDETEKSI
KEMIRIPAN JUDUL SKRIPSI
(Studi Kasus Teknik Informatika Universitas Muhammadiyah Ponorogo)**

Fuad Ainur Rafiq, Arin Yuli Astuti, Dyah Mustikasari

Program Studi Teknik Informatika, Fakultas Teknik
Universitas Muhammadiyah Ponorogo
email : fuadainur385@gmail.com

ABSTRAK

Seorang mahasiswa harus mampu menyelesaikan tugas-tugas sesuai dengan bidang yang dipilih agar mendapat gelar di akhir semester, termasuk dalam mengerjakan tugas akhir atau skripsi. Tahapan penyusunan skripsi, yang pertama mahasiswa lakukan adalah menentukan tema. Tema bisa didapat dengan banyak membaca jurnal atau literature yang sesuai dengan minat mahasiswa. Tema tersebut akan menjadi patokan terbentuknya judul skripsi. Dalam menentukan judul skripsi, mahasiswa sering mengalami kesulitan sehingga beberapa kali harus merubah judul skripsi dan dalam menentukan judul skripsi tersebut akan sangat mungkin terjadinya kemiripan judul dengan judul yang sudah ada sebelumnya. Sebagai langkah awal pencegahan agar tidak terjadi kemiripan judul skripsi, diperlukan cara mendeteksi kemungkinan kemiripan dengan mencocokkan serta membandingkan skripsi tersebut dengan skripsi sebelumnya yang telah dipublikasikan. Dalam penelitian yang dilakukan ini menghasilkan sebuah sistem yang dapat digunakan untuk melakukan pencocokan string pada dokumen dan dilengkapi dengan jumlah presentase kemiripan judul skripsi. Pada program ini akan dilengkapi jumlah presentase kemiripan judul skripsi.

Kata Kunci: mahasiswa, judul skripsi, algoritma horspool, string matching, presentase

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, bahwa penulis telah menyelesaikan skripsi dengan judul “Penerapan Algoritma Horspool Sebagai Pendeteksi Kemiripan Judul Skripsi (Studi Kasus Teknik Informatika Universitas Muhammadiyah Ponorogo)”. Dalam penyusunan skripsi ini, tidak sedikit hambatan yang penulis hadapi, namun penulis menyadari bahwa kelancaran dalam penyusunan materi ini tidak lain berkat bantuan, dorongan dan bimbingan pihak lain, oleh karena itu penulis mengucapkan terima kasih kepada :

1. Edy Kurniawan, S.T., M.T, selaku Dekan Fakultas Teknik Universitas Muhammadiyah Ponorogo.
2. Adi Fajaryanto Cobantoro, S.Kom., M.Kom, selaku Ketua Program Studi Teknik Informatika Universitas Muhammadiyah Ponorogo
3. Arin Yuli Astuti, S.Kom., M.Kom, selaku Dosen Pembimbing Utama
4. Dyah Mustikasari, S.T., M.Eng, selaku Dosen Pembimbing Pendamping
5. Bapak dan Ibu Dosen Program Studi Teknik Informatika Universitas Muhammadiyah Ponorogo yang dengan kesabaran serta ketulusan hati memberikan bekal ilmu selama perkuliahan di Universitas Muhammadiyah Ponorogo.
6. Teman-teman Teknik Informatika Angkatan 2019 Universitas Muhammadiyah Ponorogo yang telah memberikan semangat, dukungan, serta bantuannya dalam penyusunan skripsi ini.

Penulis telah berusaha semaksimal mungkin dalam menyelesaikan Skripsi ini, akan tetapi penulis sadar bahwa ini masih jauh dari sempurna. Oleh karenanya, kritik dan saran yang bersifat membangun senantiasa penulis harapkan. Semoga karya sederhana ini bermanfaat bagi kita semua.

Ponorogo, 5 Agustus 2024

Fuad Ainur Rafiq

DAFTAR ISI

HALAMAN PENGESAHAN SKRIPSI.....	ii
PERNYATAAN ORISINALITAS SKRIPSI	iii
HALAMAN BERITA ACARA UJIAN	iv
BERITA ACARA BIMBINGAN SKRIPSI	v
SURAT KETERANGAN HASIL PLAGIASI SKRIPSI	xii
SURAT KETERANGAN HASIL PLAGIASI ARTIKEL	xiii
MOTTO	xiv
HALAMAN PERSEMBAHAN	xv
ABSTRAK	xvi
KATA PENGANTAR	xvii
DAFTAR ISI.....	xviii
DAFTAR GAMBAR	xx
DAFTAR TABEL.....	xxii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	3
1.4 Batasan Masalah.....	3
1.5 Manfaat.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Penelitian Terdahulu.....	5
2.2 Landasan Teori	8
2.2.1 Plagiarisme.....	8
2.2.2 String Matching.....	9
2.2.3 Algoritma Horspool	9
2.2.4 Jaccard Similarity	14
2.2.5 MySQL.....	16
2.2.6 PHP MyAdmin.....	17
2.2.7 Metode Waterfall	17
2.2.8 White Box Testing	19

BAB III METODE PENELITIAN.....	20
3.1 Tahapan Penelitian	20
3.1.1. Studi Literatur	20
3.1.2. Pengumpulan Data	20
3.1.3. Analisis Kebutuhan	21
3.1.4. Implementasi Algoritma.....	21
3.1.5. Jaccard Similarity	29
3.1.6. Perancangan Sistem	30
BAB IV HASIL DAN PEMBAHASAN	40
4.1 Implementasi Sistem	40
4.2 Implementasi <i>Interface</i>	40
4.3 Pengujian White Box testing	51
BAB V PENUTUP.....	60
5.1 Kesimpulan.....	60
5.2 Saran	60
DAFTAR PUSTAKA	61



DAFTAR GAMBAR

Gambar 2. 1 Menentukan indeks karakter[19].....	11
Gambar 2. 2 menghitung value[19]	11
Gambar 2. 3 tabel Bad Match praproses[19]	12
Gambar 2. 4 tabel BadMatch di pencarian sistematis[19]	12
Gambar 2. 5 pencarian sistematis 1[19].....	13
Gambar 2. 6 pencarian sistematis 2[19].....	13
Gambar 2. 7 pencarian sistematis 3[19].....	14
Gambar 2. 8 pencarian sistenatis 4[19].....	14
Gambar 2. 9 Contoh dua buah dokumen [21].....	15
Gambar 2. 10 Metode Waterfall.....	17
Gambar 3. 1 Alur Tahapan Penelitian.....	20
Gambar 3. 2 flowchart system	30
Gambar 3. 3 flowchart algoritma horspool	31
Gambar 3. 4 DFD level 0	32
Gambar 3. 5 DFD level 1	32
Gambar 3. 6 ERD.....	33
Gambar 3. 7 Desain Database	33
Gambar 3. 8 Login admin	34
Gambar 3. 9 tampilan dashboard admin	35
Gambar 3. 10 halaman input judul.....	35
Gambar 3. 11 tampilan upload judul.....	36
Gambar 3. 12 tampilan judul berhasil di upload.....	36
Gambar 3. 13 daftar judul skripsi terdahulu.....	37
Gambar 3. 14 tampilan logout admin.....	37
Gambar 3. 15 login user	38
Gambar 3. 16 halaman deteksi judul user	38
Gambar 3. 17 Halaman hasil pencarian kemiripan	39
Gambar 3. 18 logout user	39

Gambar 4. 1 Halaman Login.....	41
Gambar 4. 2 Dashboard.....	41
Gambar 4. 3 input judul skripsi terdahulu.....	42
Gambar 4. 4 data judul skripsi terdahulu	42
Gambar 4. 5 menambahkan data dosen.....	43
Gambar 4. 6 data dosen tersimpan	43
Gambar 4. 7 menambahkan admin.....	44
Gambar 4. 8 data admin tersimpan	44
Gambar 4. 9 profil admin.....	45
Gambar 4. 10 halaman login.....	45
Gambar 4. 11 halaman dashboard.....	46
Gambar 4. 12 input judul skripsi baru.....	46
Gambar 4. 13 presentase kemiripan.....	47
Gambar 4. 14 hasil presentase judul	47
Gambar 4. 15 profil user	51
Gambar 4. 16 Case Login.....	51



DAFTAR TABEL

Tabel 2. 1 Tabel penelitian terdahulu.....	5
Tabel 3. 1 Panjang pattern.....	22
Tabel 3. 2 Panjang pattern setelah dikurangi huruf yang sama	22
Tabel 3. 3 tabel bad match	23
Tabel 3. 4 pencocokan string 1	24
Tabel 3. 5 pencocokan string 2	24
Tabel 3. 6 pencocokan string 3	25
Tabel 3. 7 Panjang pattern.....	25
Tabel 3. 8 Panjang pattern setelah dikurangi huruf yang sama.....	26
Tabel 3. 9 Tabel Bad Match.....	26
Tabel 3. 10 Pencocokan string 1	27
Tabel 3. 11 Pencocokan string 2	27
Tabel 3. 12 Pencocokan string 3	28
Tabel 3. 13 Pencocokan string 4	28
Tabel 4. 1 Test Case Login	52
Tabel 4. 2 Test Case Plagiasi	53
Tabel 4. 3 identifikasi.....	53
Tabel 4. 4 Tabel Uji	54

BAB I

PENDAHULUAN

1.1 Latar Belakang

Batas jangkauan universitas, seorang mahasiswa harus mampu menyelesaikan tugas-tugas sesuai dengan bidang yang dipilih agar mendapat gelar di akhir semester, termasuk dalam mengerjakan tugas akhir. Dalam proses menyelesaikan jenjang pendidikan strata 1 (S1), mahasiswa harus mempersiapkan dari awal semester karena tugas akhir ini ada di akhir semester. Mahasiswa diwajibkan untuk menyelesaikan sebuah tugas akhir atau skripsi. Skripsi atau tugas terakhir mahasiswa merupakan tugas yang wajib disusun dan dikerjakan hingga selesai oleh seorang mahasiswa untuk dapat memperoleh gelar sarjana[1]. Dalam hal ini, tugas akhir yang dimaksud adalah skripsi. Untuk mengerjakan skripsi harus secara individu dan tidak boleh ada kesamaan dengan skripsi mahasiswa lain[2]. Dalam hal ini adalah plagiasi.

Pada tahun 2010, Menteri Pendidikan pada saat itu telah menyusun Pencegahan dan Penanggulangan Plagiat di Institusi Pendidikan Tinggi yang dirancang dalam Permendiknas No. 17 Tahun 2010. Berdasarkan Pasal 1 (1) plagiat yaitu tindakan yang dilakukan dengan cara sadar atau terduga maupun tanpa diduga untuk mendapatkan, menjiplak dan memperoleh gagasan atau pernyataan guna menyelesaikan penelitian ilmiah, mengambil keseluruhan atau sedikit dari penelitian ilmiah lain yang dikenal sebagai penelitian ilmiah sendiri, dengan tidak mencantumkan sumber secara lengkap serta benar[3].

Tahapan penyusunan skripsi, yang pertama mahasiswa lakukan adalah menentukan tema. Tema bisa didapat dengan banyak membaca jurnal atau literature yang sesuai dengan minat mahasiswa[4]. Tema tersebut akan menjadi patokan terbentuknya judul skripsi[5]. Dalam menentukan judul skripsi, mahasiswa sering mengalami kesulitan sehingga beberapa kali harus merubah judul skripsi dan dalam menentukan judul skripsi tersebut akan sangat mungkin terjadinya kemiripan judul dengan judul yang sudah ada sebelumnya[6].

Dari data perpustakaan Universitas Muhammadiyah Ponorogo tahun 2023, di tahun 2020 sampai 2022 terdapat 332 skripsi prodi Teknik Informatika, dimana 34 diantaranya memiliki kemiripan judul skripsi. Kemiripan judul skripsi bisa terjadi karena beberapa alasan, seperti topik yang sedang populer dalam bidang tersebut atau karena topik tersebut memang merupakan topik yang penting dan relevan[7]. Jika beberapa mahasiswa mengalami kemiripan judul skripsi dengan angkatan sebelumnya perlu dilakukan tindakan untuk memastikan bahwa topik yang diambil mahasiswa tidak terlalu mirip dengan angkatan sebelumnya.

Sebagai langkah awal pencegahan agar tidak terjadi kemiripan judul skripsi, diperlukan cara mendeteksi kemungkinan kemiripan dengan mencocokkan serta membandingkan skripsi tersebut dengan skripsi sebelumnya yang telah dipublikasikan. Untuk mengetahui adanya kemiripan dapat menggunakan algoritma string matching. String matching atau diartikan sebagai pencocokan string merupakan suatu algoritma yang digunakan untuk melakukan pencarian semua kemunculan string pendek pattern $[0 \dots n-1]$ yang disebut pattern di string yang lebih panjang teks $[0 \dots m-1]$ yang disebut teks[8]. String Matching memiliki sifat yaitu mencari sebuah string yang terdiri dari beberapa karakter (yang biasa disebut dengan pattern) dan sejumlah besar teks[9].

Beberapa algoritma dapat diterapkan dalam proses string matching ini, salah satunya yaitu algoritma Horspool. Algoritma Horspool adalah sebuah algoritma pencarian string yang digunakan untuk mencari sebuah pola string atau pattern dalam sebuah teks[10]. Algoritma Horspool melakukan pencocokan string dimulai dari karakter paling kanan dari pattern yang akan dicocokkan dengan text[11]. Untuk melewati proses ketika terjadi ketidakcocokan karakter pattern dengan karakter text, Algoritma Horspool menggunakan bad-match secara berulang hingga seluruh karakter pattern ditemukan memiliki kecocokan terhadap text[12]. Dalam kasus ketidakcocokan (karakter tidak terdapat pada pattern) terjadi, window digeser oleh panjang teks atau panjang pattern[9].

Berdasarkan penjelasan diatas, pada penelitian ini penulis akan mengimplementasikan algoritma Hoorspol pada sistem pendeteksi kemiripan judul skripsi. Dalam penelitian yang dilakukan ini menghasilkan sebuah sistem yang dapat digunakan untuk melakukan pencocokan string pada dokumen dan dilengkapi dengan jumlah presentase kemiripan judul skripsi. Prosedur yang dijalankan dalam penelitian ini adalah program akan melakukan pencarian string kemudian membandingkan antara string yang terdapat pada database dengan string pada dokumen yang diuji. Pada program ini akan dilengkapi jumlah presentase kemiripan judul skripsi.

Dengan dilakukannya penelitian ini diharapkan nantinya dapat mengetahui hasil implementasi algoritma Hoorspol pada sistem pendeteksi kemiripan judul skripsi, sehingga dapat mengetahui apakah judul skripsi tersebut memiliki kemiripan atau tidak.

1.2 Rumusan Masalah

Bagaimana mengimplementasikan Algoritma Horspool untuk mendeteksi kemiripan judul skripsi mahasiswa?

1.3 Tujuan

Untuk mengetahui hasil implementasi algoritma Horspool pada sistem pendeteksi kemiripan judul skripsi.

1.4 Batasan Masalah

Agar tidak melebar dari latar belakang yang sudah penulis sampaikan, maka batasan masalah sebagai berikut :

1. Cek kemiripan judul skripsi di prodi Teknik Informatika Universitas Muhammadiyah Ponorogo.
2. Cek kesamaan judul skripsi mahasiswa menggunakan Algoritma Horspool
3. Menggunakan bahasa pemrograman PHP, MySQL
4. Mahasiswa yang mengajukan judul langsung ke dosen dan dosen menentukan judul setelah mengecek judul melalui aplikasi

5. Diterima atau tidak diterima judul tersebut oleh dosen tergantung jumlah presentase dari kesamaan judul dalam aplikasi tersebut
6. Pencarian data menggunakan variabel text, panjang text, pola, nilai similaritas.
7. Judul skripsi yang diterima memiliki presentase maksimal 25%.

1.5 Manfaat

1. Membantu dosen dalam menentukan judul skripsi mahasiswa
2. Mempermudah dosen dalam membandingkan judul skripsi mahasiswa.



BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Hasil penelitian terdahulu seperti pada tabel 2.1 memberikan gambaran tentang penelitian yang akan dilakukan sebagai pedoman maupun acuan penulisan dalam melakukan penelitian guna menambah wawasan pengetahuan dan menambah bahan bacaan objek penelitian yang sedang dibuat.

Tabel 2. 1 Tabel penelitian terdahulu

No	Judul	Penulis (Tahun)	Isi	Perbedaan
1	Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi dengan Algoritma WInnowing	Nur Alamsyah (2017)[13]	Pada penelitian ini membahas pengajuan judul skripsi mahasiswa yang akan diajukan agar tidak sama dengan yang telah ada.	Perbedaan penelitian ini dengan penelitian yang akan dibuat adalah di penggunaan algoritma. Algoritma pada penelitian ini menggunakan WInnowing sedangkan algoritma di penelitian yang akan dibuat menggunakan Horspool. Untuk

				objek di kedua penelitian ini sama yaitu kemiripan judul skripsi
2	Penerapan Algoritma Horspool untuk Apikasi Pencarian Abstrak Skripsi Berbasis Web	Elisa Malat, Eka Arriyanti dan hanifah Eka Wati (2015)[10]	Penelitian ini membahas tentang pencarian abstrak skripsi untuk kajian empiris dari penelitian yang dikerjakan dengan menggunakan algoritma Horspool	Perbedaan terletak pada objek penelitian, dimana pada objek yang akan diteliti ada pada judul skripsi sedangkan pada penelitian ini ada pada abstrak skripsi. Untuk metode yang digunakan sama, yaitu menggunakan algoritma Horspool.
3	Perancangan Aplikasi untuk Menghitung Presentase Kemiripan Proposal dan Isi Skripsi dengan Algoritma Rabin-Karp	Ledisma Juliana Purba dan Lambot Sitorus (2018)[14]	Pada penelitian ini menjelaskan perancangan aplikasi pendeteksi plagiasi guna menghindari kesamaan judul, naskah pada proposal dan isi dari skripsi masing-	Perbedaan terletak pada objek dan metode yang digunakan. Untuk penelitian ini menggunakan objek judul, naskah, dan isi skripsi. Lalu untuk metode

		masing mahasiswa.	menggunakan algoritma Rabin-Karp. Untuk penelitian yang akan dilakukan menggunakan objek judul skripsi dan untuk metode menggunakan algoritma Horspool.
4	Penerapan Algoritma Horspool pada Aplikasi Katalog Perpustakaan	Fince Tinus Waruwu dan Rivalri Kristianto Hondro (2019)[12]	<p>Pada penelitian ini menjelaskan banyaknya buku dalam perpustakaan tidak mungkin jika dicari manual. Dari masalah tersebut dibangun sebuah aplikasi pencarian jenis buku yang dibutuhkan.</p> <p>Pada penelitian ini, perbedaan dengan penelitian yang akan dibuat adalah di objek. Penelitian ini menggunakan jenis buku sedangkan penelitian yang akan dibuat menggunakan judul skripsi. Untuk metode yang digunakan sama yaitu algoritma Horspool</p>

5	Pembuatan Sistem Pendeteksi Kemiripan Pengajuan Topik Skripsi Menggunakan Metode Rabin-Karp	Ahmad Husein Siti Nur Salam (2018)[2]	Penelitian ini menjelaskan bahwa dalam menulis tugas kuliah, tugas akhir berpotensi melakukan penjiplakan. Termasuk juga dalam pengajuan topik skripsi.	Perbedaan terletak pada objek dan penggunaan metode. Di penelitian ini menggunakan metode Rabin-Karp dan objek skripsi, sedangkan pada penelitian yang akan dibuat menggunakan metode Horspool dan objek judul skripsi.
---	---	---------------------------------------	---	---

Pada penelitian yang akan dilakukan untuk mendeteksi kemiripan judul skripsi menggunakan Algoritma Horspool. Kemiripan judul bisa dilakukan dengan pergeseran dan pencocokan antar karakter. Pencocokan inilah yang nantinya bisa untuk mendeteksi bahwa judul skripsi tersebut mirip atau tidak dengan yang sudah ada. Nantinya output yang dihasilkan berupa sistem pendeteksi kemiripan judul skripsi dengan jumlah presentase kemiripan.

2.2 Landasan Teori

2.2.1. Plagiarisme

Plagiarisme merupakan tindakan yang dilakukan baik sengaja ataupun tidak sengaja untuk memperoleh nilai-nilai dari karya ilmiah dengan mengutip sebagian atau keseluruhan tanpa

memberikan sumber dengan tujuan menjadikan karya tersebut hak milik sendiri[15]. Literasi informasi termasuk salah satunya mengenai penghindaran kegiatan plagiarisme tidak hanya dilakukan oleh mahasiswa dan dosen saja namun dimulai sejak dini yaitu pada siswa dan guru[16].

2.2.2. String Matching

String matching merupakan suatu pendekatan yang berfungsi untuk menemukan kecocokan antar dua string berbeda, yaitu string pattern dan string text. String pattern dan string text terdiri dari kumpulan karakter yang disebut dengan kata alfabet yang disimbolkan dengan Σ (sigma) dan memiliki ukuran σ (tao) [17]. Pencocokan string juga dapat digunakan untuk mencari bitdalam sejumlah besar file binary, dalam algoritma string matching, teks diasumsikan berada didalam memory, sehingga bila mencari string di dalam sebuah teks, maka semua isi teks perlu dibaca terlebih dahulu kemudian disimpan didalam memorystring matching fokus pada pencarian satu, atau lebih umum, semua kehadiran sebuah kata (lebih umum disebut pattern) dalam sebuah teks. Semua kehadiran yang dibahas menegeluaran semua kehadiran pola dalam teks, pola dinotasikan sebagai $x = x[0..m-1]$; m adalah panjangnya[18].

2.2.3. Algoritma Horspool

Algoritma Horspool adalah sebuah metode pencarian teks (string matching) yang mencocokkan pola string dari kanan ke kiri dan bergeser sesuai tabel nilai pergeseran yang telah ditentukan[10]. Algoritma horspool hampir sama kerjanya dengan algoritma boyer moore dimana melakukan kesamaan antara pattern yang dikasih indeks dengan teks. Algoritma Horspool tidak efisien untuk kata yang pendek. Pada algoritma horspool diberikan nilai pergeseran karakter dari yang sebelah kanan jendela. Pada pengamatan dari yang pertama diberikan nilai pergeseran yang

akan diukur pada setiap karakter. Selanjutnya dilakukan persamaan pattern mulai ke dari kanan baru kekiri sampai terdapat hasil kesimpulan sama atau tidak sama[12].

Menurut [17] terdapat dua tahap pada pencocokan string menggunakan algoritma Horspool, yaitu :

1. Tahap praproses, pada tahap ini dilakukan observasi pattern terhadap teks untuk membangun sebuah tabel bad-match yang berisi nilai shift ketika ketidakcocokan antara pattern dan teks terjadi. Secara sistematis, langkah-langkah yang dilakukan algoritma Horspool pada tahap praproses adalah:

- a. Algoritma Horspool melakukan pencocokan karakter terkanan pada pattern.
- b. Setiap karakter pada pattern ditambah ke dalam tabel badmatch dan dihitung nilai shiftnya.
- c. Karakter yang berada pada ujung pattern tidak dihitung dan tidak dijadikan karakter terkanan dari karakter yang sama dengannya.
- d. Apabila terdapat dua karakter yang sama dan salah satunya bukan karakter terkanan, maka karakter dengan indeks terbesar yang dihitung nilai shift-nya.
- e. Algoritma Horspool menyimpan panjang dari pattern sebagai panjang nilai shift secara default apabila karakter pada teks tidak ditemukan dalam pattern.
- f. Nilai (value) shift yang akan digunakan dapat dicari dengan perhitungan panjang dari pattern dikurang indeks terakhir karakter dikurang 1, untuk masing-masing karakter, $value = m - I - 1 \dots \dots \dots (2.1)$

2. Tahap pencarian Secara sistematis, langkah-langkah yang dilakukan algoritma *Horspool* pada tahap praproses adalah:

- a. Dilakukan perbandingan karakter paling kanan *pattern* terhadap *window*.

- b. Tabel *bad-match* digunakan untuk melewati karakter ketika ketidakcocokan terjadi.
 - c. Ketika ada ketidakcocokan, maka karakter paling kanan pada *window* berfungsi sebagai landasan untuk menentukan jarak *shift* yang akan dilakukan.
3. Proses perhitungan Horspool

Proses perhitungan Algoritma Horspool dimulai dengan menentukan setiap karakter dengan nilai Index. Nilai index diawali mulai angka 0 dan berakhir sampai dengan jumlah karakter pada kalimat tersebut.

Pattern : MUSA				
	M	U	S	A
	0	1	2	3

Gambar 2. 1 Menentukan indeks karakter[19]

Pada Gambar 2.1 terdapat angka 0, 1, 2, dan 3. Angka tersebut adalah nilai indeks pada setiap karakter dalam kalimat MUSA. Setelah menentukan nilai indeks, selanjutnya adalah menghitung nilai Value. Nilai value didapat dengan menggunakan rumus (2.1)

$value = 4 - 0 - 1 = 3$ $value = 4 - 1 - 1 = 2$ $value = 4 - 2 - 1 = 1$ * : karakter yang tidak dikenali

Gambar 2. 2 menghitung value[19]

Setelah nilai value terhitung seperti pada gambar 2.2, langkah selanjutnya adalah membuat tabel Bad Match yang berisi nilai indeks dan nilai value

Karakter	Index	Value
M	0	3
U	1	2
S	2	1
*	-	4

Gambar 2. 3 tabel Bad Match praproses[19]

Gambar 2.3 menunjukkan tabel Bad Match yang berisi nilai indeks dan nilai value. Proses selanjutnya adalah tahap pencarian secara sistematis. Pada awal tahap ini adalah pembuatan tabel Bad Match, dapat dilihat pada gambar 2.4.

P	M	U	S	A
i	0	1	2	3
V	3	2	1	4

Gambar 2. 4 tabel BadMatch di pencarian sistematis[19]

Setiap teks dan pattern masing-masing diberi nilai m dan i, dimana m sebagai panjang pattern dan i sebagai indeks. Gambar 2.4 menunjukkan nilai pergeseran bad-match dengan menghitung nilai v seperti yang telah dilakukan pada gambar 2.2. Tahap selanjutnya adalah tahap awal pencarian, dilakukan pencocokan karakter paling kanan pattern terhadap window. Apabila terjadi ketidakcocokan, akan dilakukan pergeseran ke kanan untuk melewati karakter yang tidak cocok dimana nilai pergeserannya terdapat pada tabel bad-match. Karakter paling kanan teks pada window berfungsi sebagai landasan untuk menentukan jarak geser yang akan dilakukan. Hal ini terlihat pada Gambar 2.5 berikut:

<i>m</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P	M	U	S	A												
<i>i</i>	0	1	2	3												

Gambar 2. 5 pencarian sistematis 1[19]

Pada Gambar 2.5 ketidakcocokan karakter “Spasi” dan “A”. karakter “Spasi” tidak terdapat pada Tabel Bad-Match sehingga diganti oleh tanda (*). Tanda (*) Bernilai sebesar 4 sehingga dilakukan pergeseran sebanyak 4 kali. Hal ini terlihat pada Gambar 2.6.

<i>m</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P					M	U	S	A								
<i>i</i>					0	1	2	3								

Gambar 2. 6 pencarian sistematis 2[19]

Pada Gambar 2.6, terdapat ketidakcocokan kembali antara karakter “I” dan “A”. Karakter “I” tidak terdapat pada Tabel Bad-Match sehingga diganti oleh tanda (*). Tanda (*) bernilai sebesar 4 sehingga dilakukan pergeseran sebanyak 4 kali. Terlihat pada Gambar 2.7

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P									M	U	S	A				
i									0	1	2	3				

Gambar 2. 7 pencarian sistematis 3[19]

Pada Gambar 2.7 menunjukkan ketidakcocokan pattern dan teks terjadi pada karakter “S” dan “A”. Karakter “S” terdapat pada Tabel bad-match yang bernilai sebesar 2 sehingga dilakukan pergeseran sebanyak 2 kali. Hal ini terlihat terlihat pada Gambar 2.8.

m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T	D	O	A		N	A	B	I		M	U	S	A		A	S
P										M	U	S	A			
i										0	1	2	3			

Gambar 2. 8 pencarian sistenatis 4[19]

Pada Gambar 2.8, window telah berada pada teks dan semua pattern cocok dengan teks. Seluruh pencocokan karakter menggunakan algoritma horspool telah selesai dan berhenti pada pencarian sistematis keempat.

2.2.4. Jaccard Similarity

Jaccard Similarity atau Jaccard Coefficient merupakan algoritma yang fungsinya untuk membandingkan dua sample yaitu dokumen yang satu dengan yang lainnya berdasarkan kata yang dimilikinya. Jaccard similarity biasanya digunakan untuk membandingkan dokumen dan menghitung nilai kemiripan (similarity) dari dua buah objek atau dokumen[20]. Rumus Jaccard Similarity ditunjukkan pada (2.2)

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \dots\dots\dots(2.2)$$

$A \cap B$ = Jumlah kata yang sama (\cup = Union)

$A \cup B$ = Jumlah seluruh kata (\cap = Intersection)

Pada perhitungan Jaccard, nilai $|A \cap B|$ merupakan jumlah fingerprint yang sama antara dokumen A dengan dokumen B. Untuk dapat mengetahui nilai dokumen A sama dengan dokumen B dilakukan penyimpanan setiap nilai pada dokumen A kemudian dibandingkan dengan setiap nilai pada dokumen B, apabila sesuai maka nilai irisan ditambahkan dan disimpan. Bila semua nilai sudah dibandingkan maka proses berhenti dan nilai $|A \cap B|$ sudah diketahui. Pada proses $|A \cup B|$ bisa dihitung dengan mencari nilai jumlah fingerprint pada dokumen A kemudian ditambah dengan jumlah fingerprint pada dokumen B[21].

Misalkan terdapat dua buah dokumen yang dilakukan pembobotan dan pengindeksan. Isi dari kedua dokumen yang digunakan tersebut dapat dirincikan, dapat dilihat pada gambar 2.9 sebagai berikut:

Dokumen 1 = sarjana teknik informatika unissula
 Dokumen 2 = sarjana teknik sipil unissula

Gambar 2. 9 Contoh dua buah dokumen [21]

Kemudian memisahkan dua dokumen di atas menjadi array; (1)sarjana, (2)teknik, (3)informatika, (4)sipil , (5)unissula. Berarti memiliki dua set yang berbeda yaitu Dokumen 1 dan Dokumen 2.

Dokumen 1 = A

Dokumen 2 = B

$$A = \{ 1,2,3,5\} \text{ dan } B = \{1,2,4,5\} \dots\dots\dots(2.3)$$

Kemudian mencari Union dari kedua dokumen tersebut. Union adalah jumlah kata secara keseluruhan dari dua dokumen yang sedang dihitung. Dari array diatas bisa dilihat bahwa jumlah kata secara keseluruhan adalah 5 kata. Union dari Dokumen 1 dan 2 adalah sebagai berikut :

$$A \cup B = \{1,2,3,4,5\} \dots\dots\dots(2.4)$$

Setelah berhasil mendapatkan hasil Union pada persamaan (2.4), selanjutnya adalah mencari Intersection diantara dua dokumen tersebut. Intersection adalah jumlah kata yang sama dari dua dokumen yang sedang dihitung. Jika dilihat dari Dokumen A dan Dokumen B, ada beberapa kata yang sama dari kedua dokumen tersebut, antara lain : [1]sarjana, [2]teknik , [5]unissula. Intersection dari Dokumen A dan B adalah :

$$A \cap B = \{1,2,5\} \dots\dots\dots(2.5)$$

Langkah selanjutnya adalah menghitung kemiripan dari kedua dokumen tersebut dengan rumus (2.2)

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{3}{5} = 0,6$$

Berdasarkan nilai persamaan yang diperoleh maka dapat ditetapkan bahwa nilai kemiripan dari dokumen 1 dan dokumen 2 adalah 0,6. Hasil tersebut dikalikan dengan 100 karena untuk mencari presentase kemiripan, jadi presentase dari kedua dokumen tersebut sebesar 60%.

2.2.5. MySQL

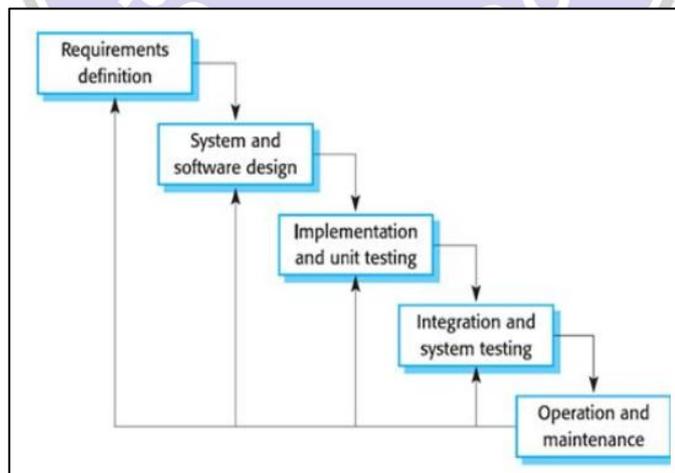
MySQL adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang databse sebagai sumber dan pengelolaan datanya[22]. MySQL bisa dikatakan perangkat lunak sistem manajemen basis data SQL atau DBMS yang multithread, multi-user dengan sekitar

6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License [23]. MySQL merupakan database yang awalnya hanya berjalan pada sistem Unix dan Linux. Seiring berjalannya waktu dan banyaknya peminat yang menggunakan database ini, MySQL merilis versi yang dapat diinstal pada hampir semua platform, termasuk Windows. Lisensi dari MySQL adalah freeware[24]..

2.2.6. PHP MyAdmin

PHP merupakan kependekan dari Hypertext Preprocessor. PHP tergolong sebagai perangkat lunak open source yang diatur dalam aturan general purpose licences (GPL). Bahasa pemrograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa diletakkan pada script HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis[25]. Karena PHP bersifat open source, siapa pun dapat menggunakannya tanpa dipungut biaya. Pemrograman PHP memungkinkan program dijalankan pada sistem operasi apa pun dengan browser web[26].

2.2.7. Metode Waterfall



Gambar 2. 10 Metode Waterfall

Metode waterfall seperti pada gambar 2.10 merupakan model pengembangan sistem informasi yang sistematis dan sekuensial[27]. Metode Waterfall memiliki tahapan-tahapan sebagai berikut :

1. Requirements analysis and definition

Layanan sistem, kendala, dan tujuan ditetapkan oleh konsultasi dengan pengguna dan menjelaskan secara rinci spesifikasi sistem

2. Sistem and software design

Perancangan tampilan tatap muka secara keseluruhan baik perangkat keras maupun perangkat lunak. Perancangan perangkat melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat.

3. Implementation and unit testing

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

4. Integration and sistem testing

Unit-unit pada perangkat digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan atau tidak. Setelah pengujian, perangkat lunak dapat dikirimkan ke pengguna..

5. Operation and maintenance

Tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. Maintenance melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

2.2.8. White Box Testing

Metode ini untuk pengujian aplikasi atau sistem. Pengujian ini banyak dilakukan karena metode ini sangat baik untuk pengujian. Pengujian White Box telah berkembang sebuah analisis sistem yang dapat menggambarkan secara tepat kode sumber pemrograman, prosedur sistem, dan jalur. Metode pengujian White box diimplementasikan guna menentukan kapasitas sistem yang diciptakan bisa bekerja secara efisien dan sistematis. Pengujian white box merupakan cara pengujian yang menerapkan tatanan control yang sesuai prosedur. Pengujian ini dapat digunakan untuk menganalisa aplikasi setelah implementasi sistem[28]



BAB III METODE PENELITIAN

3.1 Tahapan Penelitian



Gambar 3. 1 Alur Tahapan Penelitian

3.1.1. Studi Literatur

Studi literatur bertujuan untuk mempelajari dan memperjelas teori yang berkaitan dengan pengerjaan skripsi Penerapan Algoritma Horspool sebagai Pendeteksi Kemiripan Judul Skripsi ini. Teori yang dapat menunjang dalam mengerjakan skripsi ini dapat diperoleh dari penelitian sebelumnya berupa jurnal, buku, dan dokumen yang memiliki topik berhubungan dengan skripsi ini. Penelitian sebelumnya menjadikan sebuah perbandingan dan menjadi sebuah masukan untuk penelitian yang sedang dilakukan.

3.1.2. Pengumpulan Data

Pada proses pengumpulan data dengan mencatat judul skripsi 3 tahun terakhir Teknik Informatika dan wawancara di Perpustakaan Universitas Muhammadiyah Ponorogo. Dalam proses wawancara,

diajukan pertanyaan dan didapatkan data yang dapat mendukung pengerjaan penelitian ini.

3.1.3. Analisis Kebutuhan

Permasalahan saat ini adalah masih adanya judul skripsi mahasiswa yang mirip dengan judul yang sudah ada sebelumnya. Kemiripan judul skripsi bisa terjadi karena beberapa alasan, seperti topik yang sedang populer dalam bidang tersebut atau karena topik tersebut memang merupakan topik yang penting dan relevan[7]. Jika beberapa mahasiswa mengalami kemiripan judul skripsi dengan angkatan sebelumnya perlu dilakukan tindakan untuk memastikan bahwa topik yang diambil mahasiswa tidak terlalu mirip dengan angkatan sebelumnya. Untuk menyelesaikan permasalahan tersebut dapat dibantu dengan sistem pendeteksi kemiripan dengan menerapkan algoritma Horspool. Sistem dilengkapi dengan jumlah presentase kemiripannya.

Pada tahap ini dilakukan analisa terhadap kebutuhan untuk membangun sebuah sistem pendeteksi kemiripan judul skripsi. Sistem ini memiliki kebutuhan fungsional antara lain :

- 1) Sistem mampu menampilkan, menambah, dan merubah data judul skripsi
- 2) Sistem mampu menampilkan judul-judul skripsi
- 3) Sistem mampu menampilkan hasil presentase kemiripan judul skripsi

3.1.4. Implementasi Algoritma

Dalam pembahasan ini diperlukan data berupa karakter teks dan petern yang sudah ditentukan, yang nantinya akan dilakukan pencocokan sehingga didapatlah hasilnya berupa kemiripan judul sesuai dengan permintaan. Adapun Penerapan Algoritma Horspool pada sistem pendeteksi kemiripan judul skripsi terdiri dari beberapa langkah-langkah yaitu sebagai berikut :

1. Tahap Praproses

Pada tahap ini dilakukan observasi pattern terhadap teks untuk membangun sebuah tabel badmatch yang berisi nilai shift. Ditunjukkan pada tabel 3.1

Pattern :

Tabel 3. 1 Panjang pattern

K	E	P	U	T	U	S	A	N
0	1	2	3	4	5	6	7	8

Di tabel 3.1, terdapat dua karakter yang sama, yaitu huruf "U" maka yang dihitung adalah karakter huruf "U" dengan indeks terbesar.

Tabel 3. 2 Panjang pattern setelah dikurangi huruf yang sama

K	E	P	T	U	S	A	N
0	1	2	3	4	5	6	7

Pada gambar 3.2, huruf "U" yang semula ada dua, dikurangi satu dan yang dihitung adalah huruf "U" dengan indeks terbesar. Selanjutnya, menentukan nilai (value) dengan rumus (2.1)

$$v = m - i - 1$$

$$K v = 8 - 0 - 1 = 7$$

$$E v = 8 - 1 - 1 = 6$$

$$P v = 8 - 2 - 1 = 5$$

$$T v = 8 - 3 - 1 = 4$$

$$U v = 8 - 4 - 1 = 3$$

$$S v = 8 - 5 - 1 = 2$$

$$A v = 8 - 6 - 1 = 1$$

* $v = 8$

Huruf terakhir pada pattern yaitu huruf “N” tidak dihitung dan menjadi karakter tidak dikenali dengan nilai sama dengan jumlah karakternya yaitu 8, terlihat pada tabel 3.3.

Tabel 3. 3 tabel bad match

Karakter	Index	Value
K	0	7
E	1	6
P	2	5
T	3	4
U	4	3
S	5	2
A	6	1
*	-	8

2. Tahap pencarian secara sistematis

Setelah memperoleh nilai Bad Match seperti pada tabel 3.3, tahap selanjutnya adalah melakukan tahap pencarian atau pencocokan string sebagai berikut, seperti pada gambar 3.4

a) Pencocokan String 1

Tabel 3. 4 pencocokan string 1

Indeks	0	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	
											0	1	2	3	4	5	6	7	8
Teks	P	E	N	D	U	K	U	N	G	K	E	P	U	T	U	S	A	N	
pattren	K	E	P	U	T	U	S	A	N										

Pada tabel 3.4 diatas, proses pencocokan karakter pada tabel 3.2 menunjukkan bahwa karakter “N” pattern tidak cocok dengan karakter “G” teks, sehingga dilakukan pergeseran. Karakter “G” tidak terdapat pada tabel Bad Match, sehingga dapat diartikan menjadi karakter tidak dikenal. Pergeseran karakter sebanyak 8 langkah, ditunjukkan pada gambar 3.5.

b) Pencocokan String 2

Tabel 3. 5 pencocokan string 2

Indeks	0	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	
											0	1	2	3	4	5	6	7	8
Teks	P	E	N	D	U	K	U	N	G	K	E	P	U	T	U	S	A	N	
pattren										K	E	P	U	T	U	S	A	N	

Pada gambar 3.5, setelah bergeser 8 langkah, tidak terdapat kecocokan pada ujung pattern dengan teks. Karakter “N” pattern tidak cocok dengan karakter “S” teks. maka selanjutnya dilakukan pergeseran sebanyak nilai Bad Match karakter “S” yaitu 2 langkah, seperti pada gambar 3.6.

c) Pencocokan String 3

Tabel 3. 6 pencocokan string 3

Indeks	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Teks	P	E	N	D	U	K	U	N	G		K	E	P	U	T	U	S	A	N
pattren											K	E	P	U	T	U	S	A	N

Di gambar 3.6, proses pencocokan karakter “N” pattern dengan karakter “N” pada text terjadi kecocokan, karakter “A” pattern dengan karakter “A” pada text terjadi kecocokan, karakter “S” pattern dengan karakter “S” pada teks terjadi kecocokan, karakter “U” pattern dengan karakter “U” pada text terjadi kecocokan, karakter “T” pattern dengan karakter “T” pada text terjadi kecocokan, karakter “U” pattern dengan karakter “U” pada text terjadi kecocokan, karakter “P” pattern dengan karakter “P” pada text terjadi kecocokan, karakter “E” pattern dengan karakter “E” pada text terjadi kecocokan, karakter “K” pattern dengan karakter “K” pada text terjadi kecocokan, maka proses pencocokan string berhenti pada tahap pencocokan string yang ketiga.

3. Tahap Praproses.

Pattern

Tabel 3. 7 Panjang pattern

P	A	K	A	R
0	1	2	3	4

Pada tabel 3.7, pattern diatas terdapat dua karakter yang sama, yaitu huruf “A” maka yang dihitung adalah karakter huruf “A” dengan indeks terbesar, pattern yang sudah dikurangi huruf “A” ada pada tabel 3.8.

Tabel 3. 8 Panjang pattern setelah dikurangi huruf yang sama

P	K	A	R
0	1	2	3

Pada tabel 3.8, pattern setelah dikurangi huruf “A”. Selanjutnya menentukan nilai (value) dengan rumus (2.1)

$$v = m - i - 1$$

$$P \ v = 4 - 0 - 1 = 3$$

$$K \ v = 4 - 1 - 1 = 2$$

$$A \ v = 4 - 2 - 1 = 1$$

$$* \ v = 4$$

Huruf terakhir pada pattern yaitu huruf “R” tidak dihitung dan menjadi karakter tidak dikenali dengan nilai sama dengan jumlah karakternya yaitu 8, terlihat pada tabel 3.9.

Tabel 3. 9 Tabel Bad Match

karakter	indeks	value
P	0	3
K	1	2
A	2	1
*	-	4

4. Tahap pencarian secara sistematis

a) Pencocokan String 1

Tabel 3. 10 Pencocokan string 1

Indeks	0	1	2	3	4	5	6	7	8	9	10	11
Teks	S	I	S	T	E	M		P	A	K	A	R
Pattern	P	A	K	A	R							

Proses pencocokan karakter pada tabel 3.10 menunjukkan bahwa karakter “R” pattern tidak cocok dengan karakter “E” teks, sehingga dilakukan pergeseran. Karakter “E” tidak terdapat pada tabel Bad Match, sehingga dapat diartikan menjadi karakter tidak dikenal. Pergeseran karakter sebanyak 4 langkah.

b) Pencocokan String 2

Tabel 3. 11 Pencocokan string 2

Indeks	0	1	2	3	4	5	6	7	8	9	10	11
Teks	S	I	S	T	E	M		P	A	K	A	R
Pattern					P	A	K	A	R			

Proses pergeseran sebanyak 4 langkah pada tabel 3.11, tidak terdapat kecocokan pada ujung pattern dengan teks. Karakter “R” pattern tidak cocok dengan karakter “A” teks maka selanjutnya dilakukan pergeseran sebanyak nilai Bad Match karakter “A” yaitu 1 langkah, ditunjukkan dengan gambar 3.12.

c) Pencocokan String 3

Tabel 3. 12 Pencocokan string 3

Indeks	0	1	2	3	4	5	6	7	8	9	10	11
Teks	S	I	S	T	E	M		P	A	K	A	R
Pattern						P	A	K	A	R		

Pada tabel 3.12 ini, ujung pattern masih tidak cocok dengan teks. Karakter “R” pattern tidak cocok dengan karakter “K” teks. Maka selanjutnya dilakukan pergeseran sesuai dengan nilai bad match karakter “K” teks yaitu sebanyak 2 langkah. Proses pergeseran pada tabel 3.13.

d) Pencocokan String 4

Tabel 3. 13 Pencocokan string 4

Indeks	0	1	2	3	4	5	6	7	8	9	10	11
Teks	S	I	S	T	E	M		P	A	K	A	R
Pattern								P	A	K	A	R

Pada tabel 3.13, proses pencocokan karakter “R” pattern dengan karakter “R” pada text terjadi kecocokan, karakter “A” pattern dengan karakter “A” pada text terjadi kecocokan, karakter “K” pattern dengan karakter “K” pada teks terjadi kecocokan, karakter “A” pattern dengan karakter “A” pada text terjadi kecocokan, karakter “A” pattern dengan karakter “A” pada text terjadi kecocokan, karakter “P” pattern dengan karakter “P” pada

text terjadi kecocokan, maka proses pencocokan string berhenti pada tahap pencocokan string yang keempat.

3.1.5. Jaccard Similarity

Terdapat dua dokumen yang akan dilakukan penghitungan presentase yaitu :

Dokumen 1 (A) = penerapan metode ahp untuk mencari pemain voli terbaik

Dokumen 2 (B) = penerapan metode saw untuk rekrutmen karyawan

Kemudian memisahkan ke dua dokumen tersebut menjadi array :

(1)penerapan, (2)metode, (3)ahp, (4)saw, (5)untuk, (6)mencari, (7)rekrutmen, (8)pemain, (9)voli, (10)karyawan, (11)terbaik

$A = \{1, 2, 3, 5, 6, 8, 9, 11\}$

$B = \{1, 2, 4, 5, 7, 10\}$

Kemudian mencari jumlah kata keseluruhan dari dua dokumen tersebut, dari perhitungan di atas bisa dilihat jumlah kata keseluruhan yaitu 11 kata.

Jadi $A \cup B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$(3.1)

Setelah mencari jumlah seluruh kata, selanjutnya mencari kata yang sama diantara dua dokumen tersebut. Kata yang sama yaitu (1)penerapan, (2)metode, (5)untuk.

Jadi untuk jumlah $A \cap B = \{1, 2, 5\}$(3.2)

Langkah selanjutnya adalah menghitung kemiripan dari dua dokumen tersebut berdasarkan persamaan (2.2). Hasil perhitungan dikali 100 karena untuk mencari persentasenya.

$$(A, B) = \frac{3}{11} = 0,27 \times 100 = 27\%$$

Jadi nilai persamaan yang diperoleh didapat nilai kemiripan antara dua dokumen tersebut (A dan B) adalah 27%

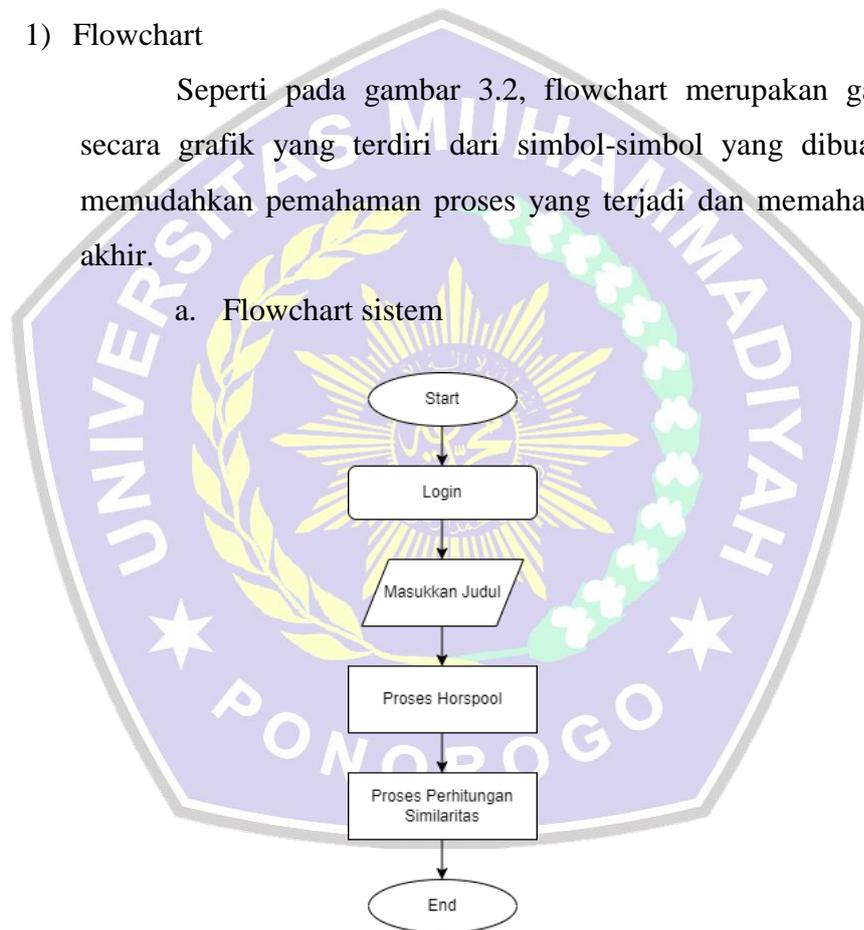
3.1.6. Perancangan Sistem

Perancangan sistem merupakan sebuah tahapan yang dilakukan untuk mempermudah proses pengimplementasian dari sistem yang akan dibangun. Pada penelitian ini perancangan sistem menggunakan flowchart, yang merupakan alat yang digunakan untuk menggambarkan langkah dan urutan prosedur dari suatu program[29].

1) Flowchart

Seperti pada gambar 3.2, flowchart merupakan gambaran secara grafik yang terdiri dari simbol-simbol yang dibuat untuk memudahkan pemahaman proses yang terjadi dan memahami hasil akhir.

a. Flowchart sistem

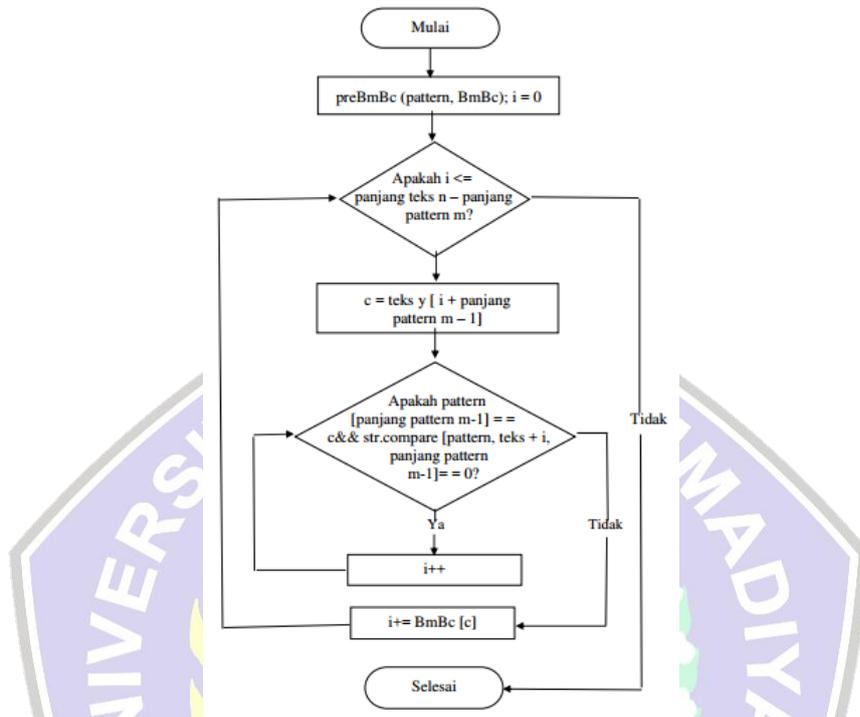


Gambar 3. 2 flowchart system

b. Flowchart Algoritma Horspool

Aturan pergeseran bila terjadi ketidakcocokan adalah berdasarkan nilai shift table yang telah dicari sebelumnya, sehingga setiap huruf mempunyai shift

tersendiri berdasarkan perhitungan. Untuk flowchart algoritma Horspool dapat dilihat pada Gambar 3.4 berikut.

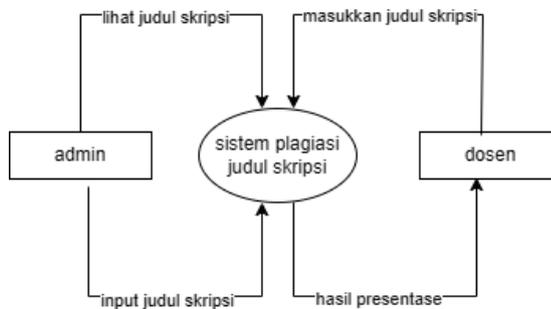


Gambar 3. 3 flowchart algoritma horspool

2) Data Flow Diagram

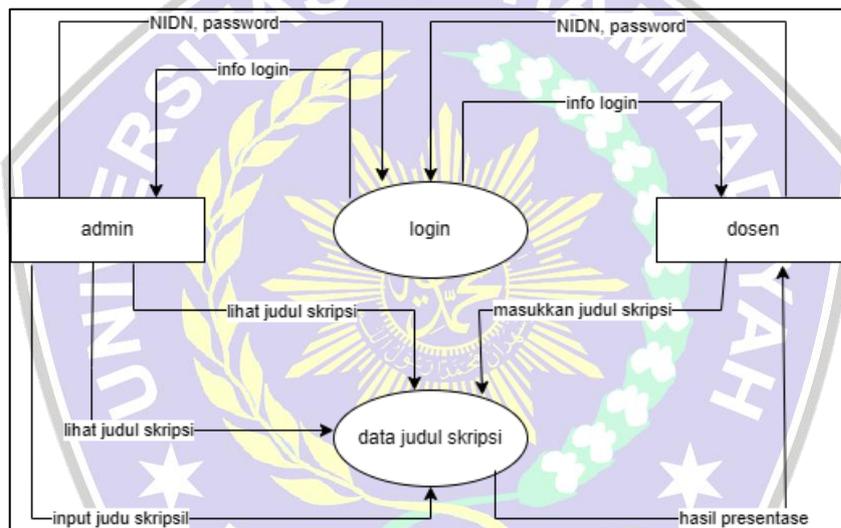
Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan ke mana tujuan data yang keluar dari sistem, di mana data tersimpan, proses apa yang menghasilkan data, dan interaksi antara data tersimpan dan proses yang dikenakan pada data tersebut. Untuk DFD level 0 dan DFD level 1 dapat dilihat pada gambar 3.5 dan 3.6 berikut.

a. DFD Level 0



Gambar 3. 4 DFD level 0

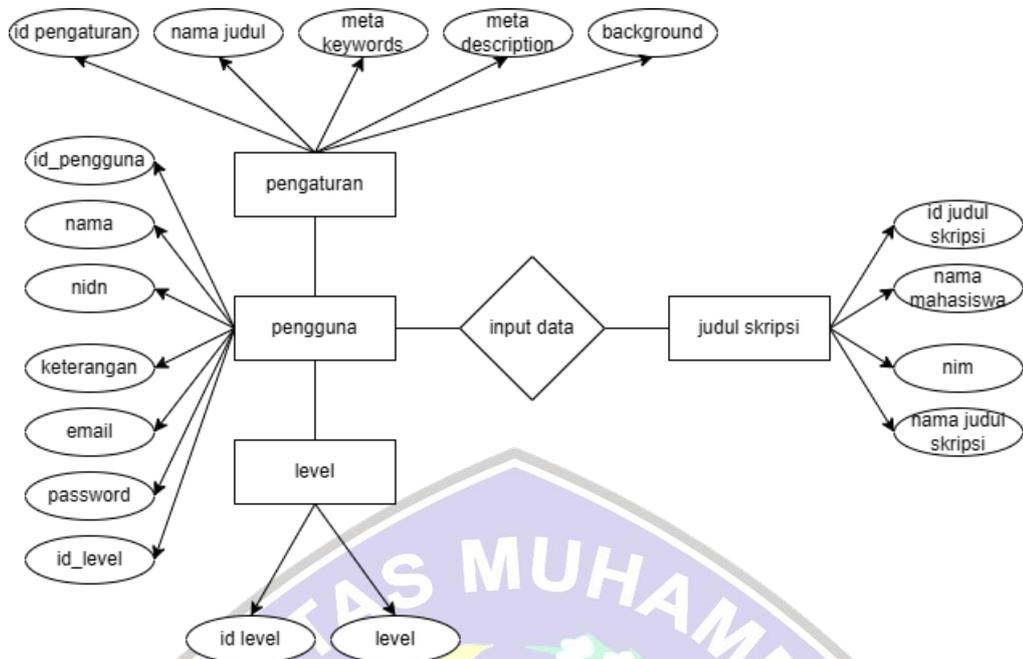
b. DFD Level 1



Gambar 3. 5 DFD level 1

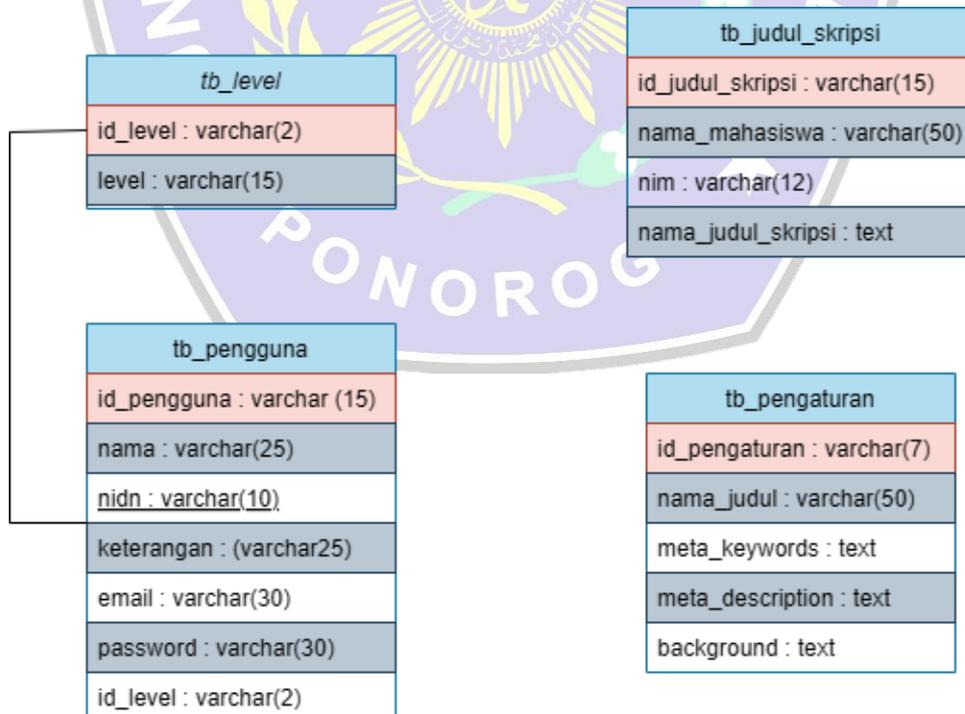
3) ERD

ERD berfungsi untuk menjelaskan objek data utama yang akan diproses oleh sistem, bagaimana komposisi dari masing-masing objek data termasuk atributnya, hubungan antara masing-masing objek data dan objek yang lainnya dan bagaimana hubungan antara objek dengan proses yang mentransformasikannya[27]. Untuk ERD dapat dilihat pada gambar 3.7.



Gambar 3. 6 ERD

4) Desain Database



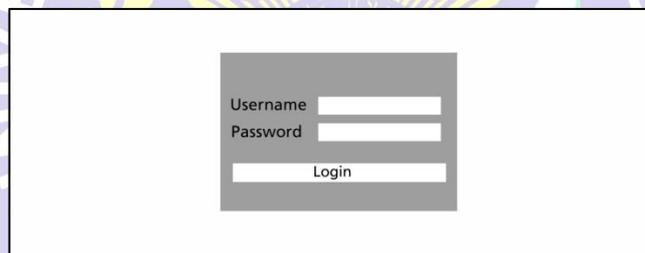
Gambar 3. 7 Desain Database

Pada gambar 3.7 terdapat tahap proses perencanaan dan perancangan struktur atau format tabel dalam sebuah basis data yang mencakup pengorganisasian, penentuan atribut, dan hubungan antar tabel disebut sebagai rancangan tabel database.

5) Desain Interface

Perancangan antarmuka bertujuan untuk mempermudah pengguna dalam berinteraksi dengan sistem yang pada umumnya tampilan antarmuka bersifat sederhana. Antarmuka yang dirancang diharapkan dapat memperhatikan faktor pengguna sehingga menghasilkan aplikasi yang mudah dipahami. Perancangan antarmuka ini disesuaikan dengan kerangka menu seperti yang telah dituliskan sebelumnya.

1. Login Admin
 - a. Halaman Login admin

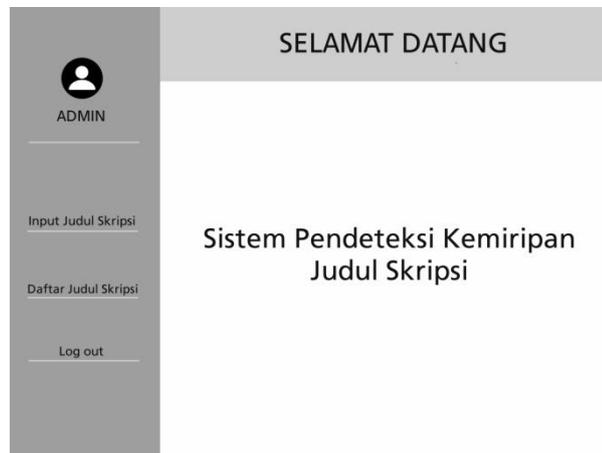


The image shows a login form with three input fields: 'Username', 'Password', and 'Login'. The 'Login' field is a button. The form is centered on a white background.

Gambar 3. 8 Login admin

Pada gambar 3.8, halaman login admin terdapat username, password dan login. Admin diminta memasukkan username dan password setelah itu login.

b. Dashboard admin

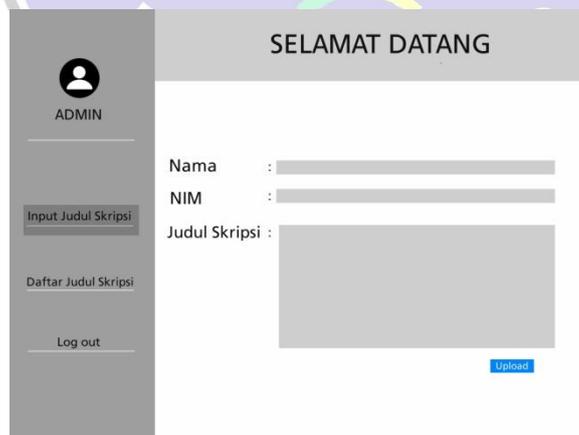


Gambar 3. 9 tampilan dashboard admin

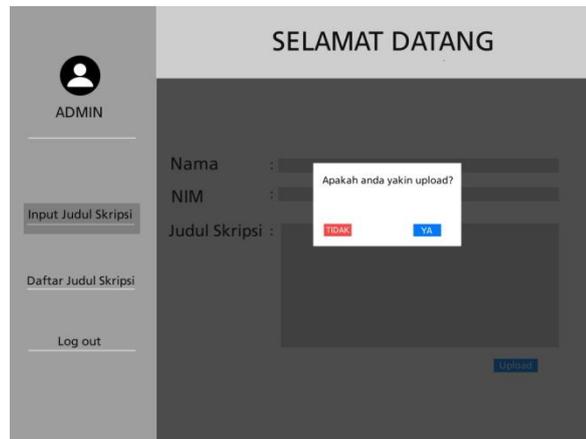
Gambar 3.9 menampilkan dashboard admin, terdapat menu input judul skripsi, daftar judul skripsi dan logout.

c. Halaman input judul skripsi admin

Halaman input judul skripsi terdapat nama mahasiswa, NIM mahasiswa dan judul skripsi. Admin dapat menambahkan judul skripsi terdahulu melalui halaman ini. Setelah itu judul dapat diupload, tampilan halaman upload dan setelah apload judul ada pada gambar 3.10 dan gambar 3.11.

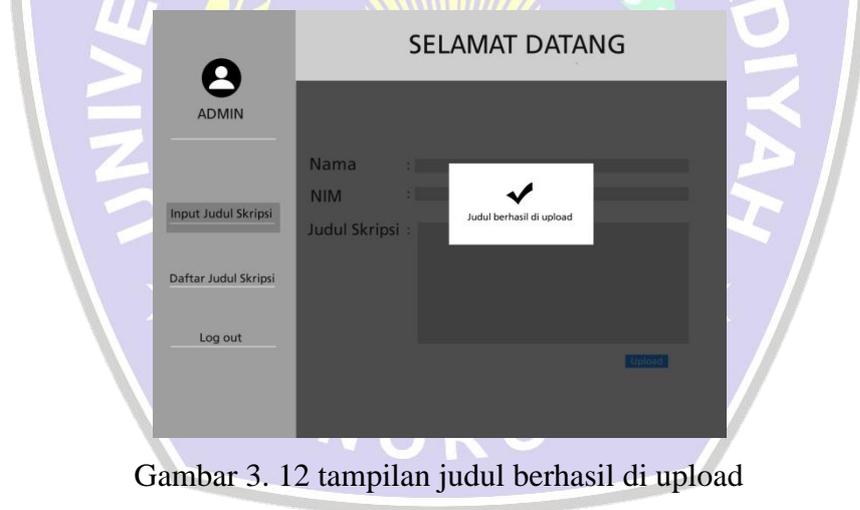


Gambar 3. 10 halaman input judul



Gambar 3. 11 tampilan upload judul

Pada gambar 3.12, tampilan menu ketika judul berhasil di upload oleh admin. Nantinya judul-judul tersebut akan jadi pembandingan dengan judul baru yang di masukkan oleh user.



Gambar 3. 12 tampilan judul berhasil di upload

d. Halaman daftar judul skripsi



Gambar 3. 13 daftar judul skripsi terdahulu

Pada gambar 3.13, halaman daftar judul skripsi, terdapat judul skripsi terdahulu yang sudah berhasil diupload oleh admin dan judul terbaru berada di urutan atas. Di halaman ini, admin dapat menghapus judul skripsi.

e. Halaman logout admin

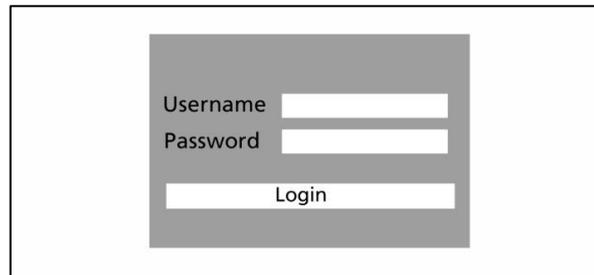


Gambar 3. 14 tampilan logout admin

Tampilan menu logout admin seperti pada gambar 3.14. terdapat pilihan Ya dan Tidak. Apabila dipilih Ya, akan keluar ke halaman login. Sedangkan apabila dipilih tidak, akan tetap pada halaman input judul skripsi.

2. Login User

a. Halaman login user



A screenshot of a login form. It features a dark gray background with three white input fields. The first field is labeled 'Username', the second 'Password', and the third is a button labeled 'Login'.

Gambar 3. 15 login user

Pada gambar 3.15, halaman login user terdapat username, password dan login. User diminta memasukkan username dan password setelah itu pilih login.

b. Halaman deteksi kemiripan judul



A screenshot of a user dashboard. The top right corner says 'SELAMAT DATANG'. On the left, there is a sidebar with a user profile icon labeled 'USER', an 'Input Judul Skripsi' button, and a 'Log out' button. The main area contains a search form with the text 'Masukkan judul skripsi :', an input field, and a 'Cari' button.

Gambar 3. 16 halaman deteksi judul user

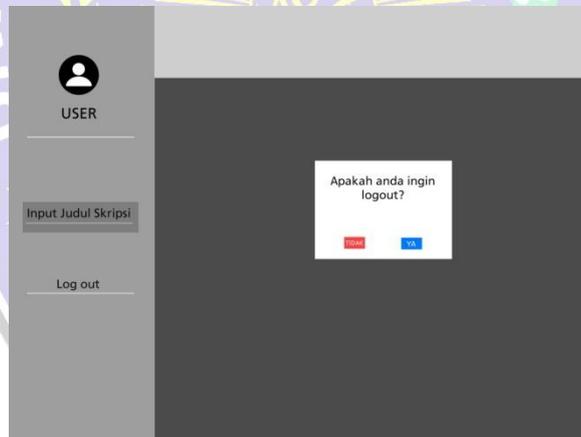
Pada gambar 3.16, terdapat halaman input judul skripsi. User dapat memasukkan judul skripsi baru untuk di cari tingkat kemiripannya. Tampilan setelah judul dicari tingkat kemiripannya terdapat pada gambar 3.17.



Gambar 3. 17 Halaman hasil pencarian kemiripan

c. Halaman logout user

Tampilan menu logout user seperti pada gambar 3.18. terdapat pilihan Ya dan Tidak. Apabila dipilih Ya, akan keluar ke halaman login. Sedangkan apabila dipilih tidak, akan tetap pada halaman input judul skripsi.



Gambar 3. 18 logout user

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Pada bab ini menjelaskan implementasi aplikasi pendeteksi kemiripan judul skripsi yang berdasarkan analisa kebutuhan dan proses dari perancangan aplikasi tersebut. Dalam implementasi ini akan ada beberapa hal terkait dengan sistem yang akan dibangun dan juga memperhatikan ketentuan-ketentuan agar aplikasi ini berjalan dengan baik. Hasil dari penelitian ini akan berguna untuk mencari kesamaan kata yaitu kemiripan judul skripsi dengan menampilkan tingkat presentase kemiripannya. Pembahasan tentang aplikasi yang akan dimuat terdiri dari spesifikasi sistem, dan implementasi Interface.

4.2 Implementasi Interface

Pada bab ini menjelaskan tentang bagaimana implementasi antarmuka aplikasi yang mengacu pada bab sebelumnya yaitu desain antarmuka. Antarmuka ini akan menjadi salah satu patokan keberhasilan dari membangun aplikasi kemiripan judul ini. Bab ini akan menjelaskan bagaimana user interface dari sistem ini yang dibangun, menjelaskan apa saja fitur-fitur yang ada didalamnya. Sistem ini akan menampilkan antar muka admin dan user saja, karena memang focus aplikasi ini hanya untuk mencari persamaan yang hasilnya berupa presentase.

A. Login Admin

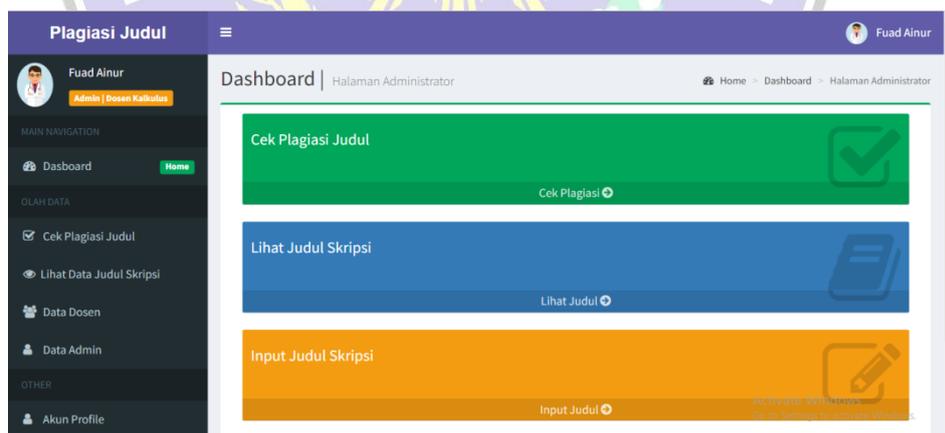
1. Halaman Login



Gambar 4. 1 Halaman Login

Pada gambar 4.1, halaman ini adalah tampilan awal dari sistem yang di bangun dengan melakukan sebuah login untuk admin system pendeteksi kemiripan judul skripsi dengan memasukkan data berupa username dan password.

2. Halaman Dashboard

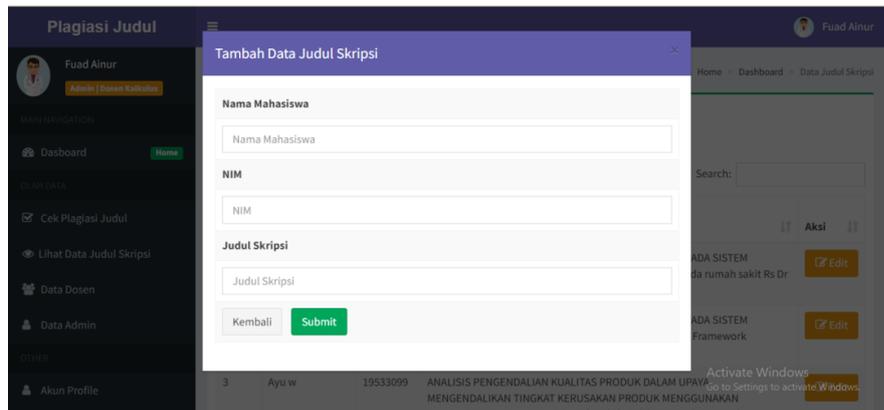


Gambar 4. 2 Dashboard

Pada halaman ini adalah halaman Dashboard dari sistem dengan memberikan beberapa menu yang dapat digunakan oleh admin, terlihat

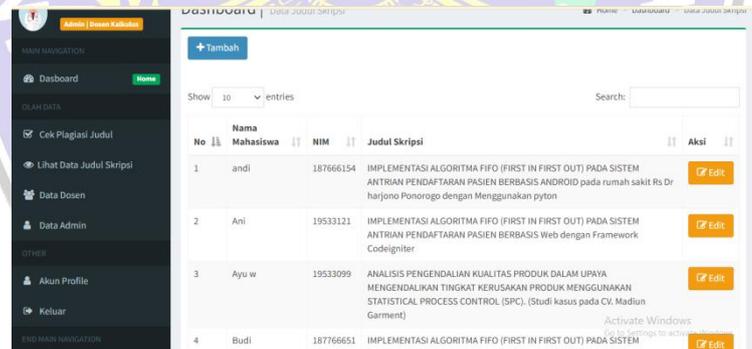
pada gambar 4.2.. Halaman dashboard berisi menu lihat data judul skripsi beserta input judul terdahulu, data dosen, data admin, dan akun profile.

3. Menu Lihat Data Judul Skripsi



Gambar 4. 3 input judul skripsi terdahulu

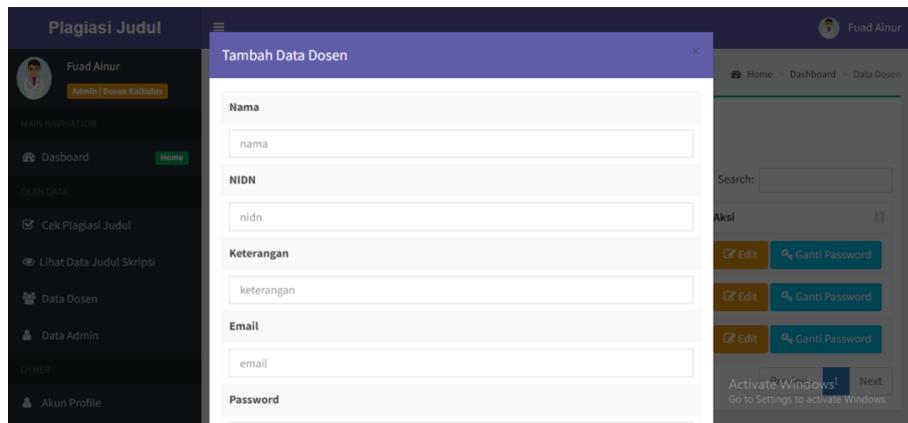
Pada gambar 4.3, admin dapat menambahkan judul skripsi angkatan sebelumnya dengan memasukkan nama mahasiswa, NIM, dan judul skripsi.



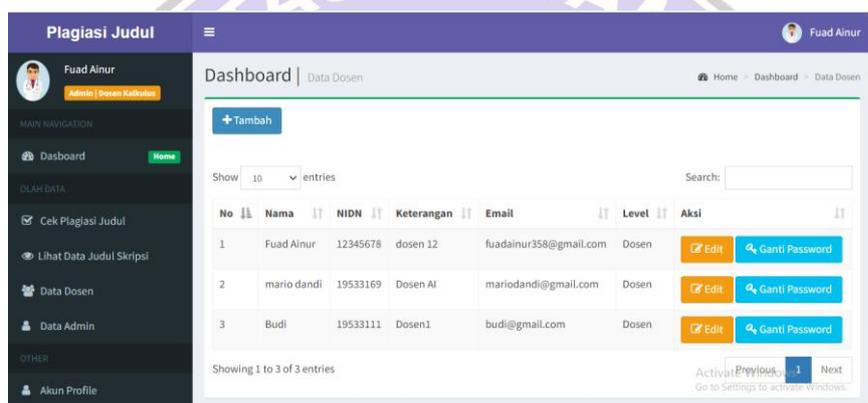
Gambar 4. 4 data judul skripsi terdahulu

Pada halaman ini berfungsi untuk mengolah judul skripsi angkatan sebelumnya, terlihat pada gambar 4.4. Setelah judul dimasukkan, judul skripsi otomatis tersimpan. Judul skripsi yang telah disimpan juga dapat di edit apabila ada kesalahan sebelumnya.

4. Data dosen



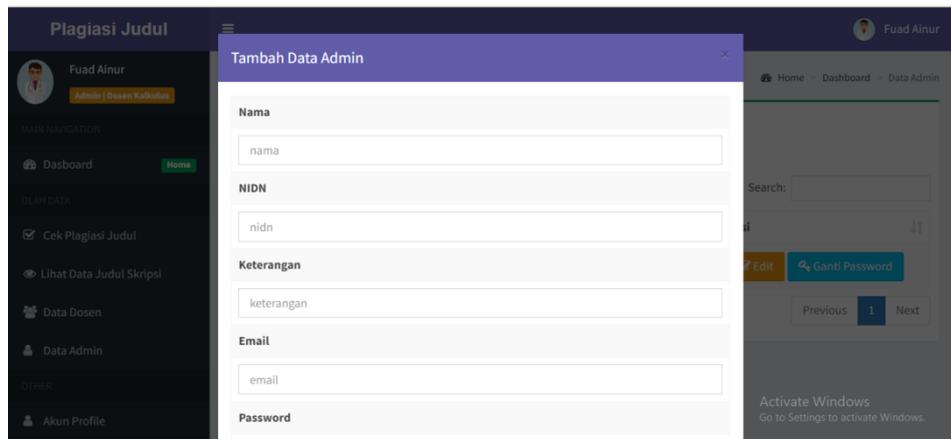
Gambar 4. 5 menambahkan data dosen



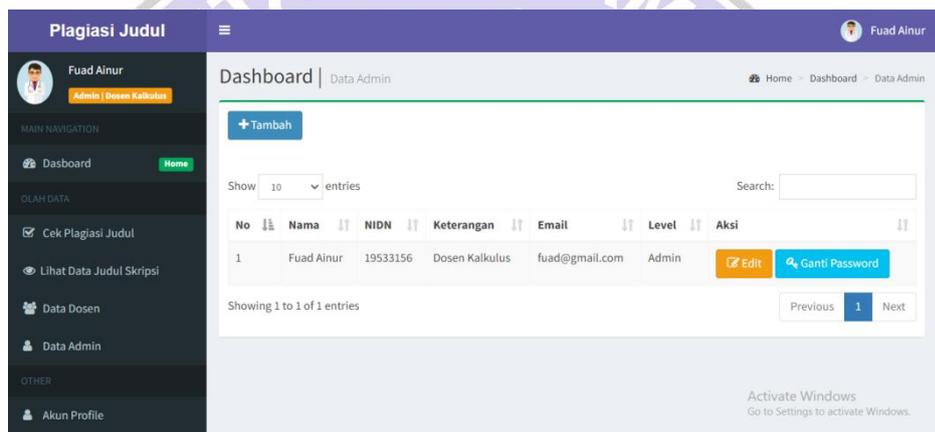
Gambar 4. 6 data dosen tersimpan

Pada halaman ini, admin dapat menambahkan user agar dapat melakukan login ke system dengan memasukkan nama dosen, nidn, keterangan, email, dan password, terlihat pada gambar 4.5. Setelah data dosen tersimpan, admin dapat mengedit data dosen tersebut, ada pada gambar 4.6.

5. Data admin



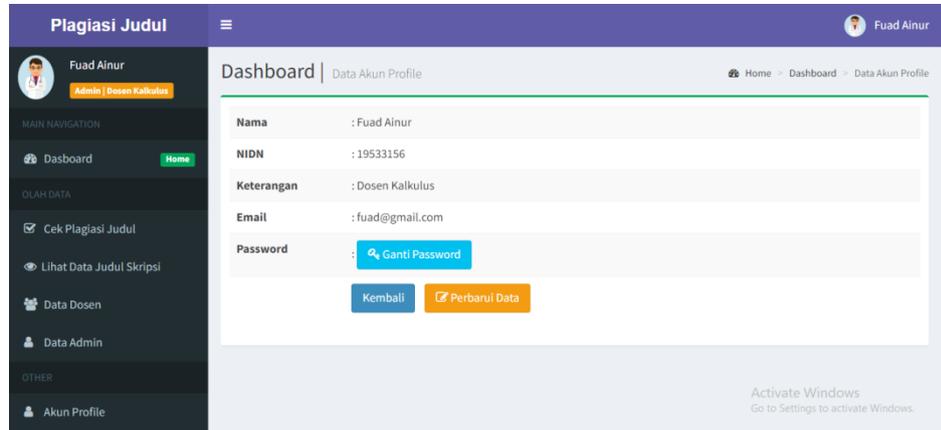
Gambar 4. 7 menambahkan admin



Gambar 4. 8 data admin tersimpan

Halaman data admin adalah untuk menambahkan admin baru dan mengolah data admin tersebut. Pada gambar 4.7, untuk menambahkan admin perlu memasukkan nama, nidn, keterangan, email, dan password. Untuk tampilan mengolah dan mengedit data admin ada pada gambar 4.8.

6. Akun Profile

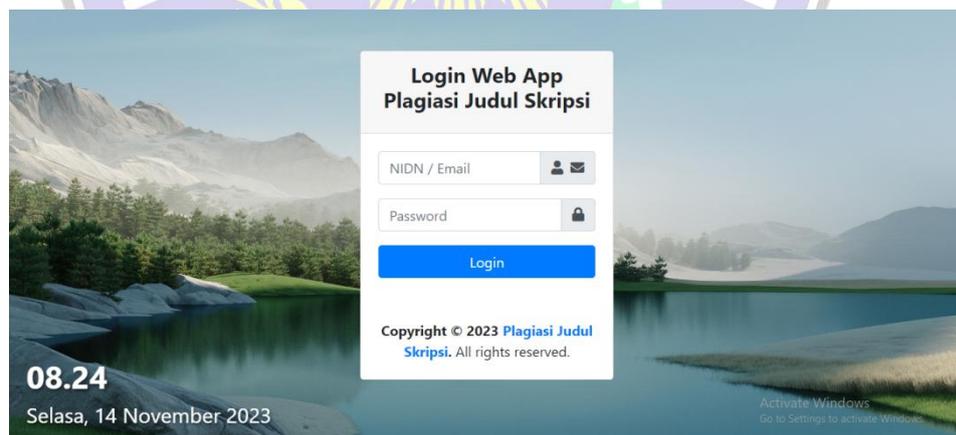


Gambar 4. 9 profil admin

Pada halaman akun profile, berisi informasi admin yang sedang login pada system. Dalam halaman ini juga dapat mengolah dan mengedit data admin, terlihat pada gambar 4.9.

B. Login User

1. Halaman login.

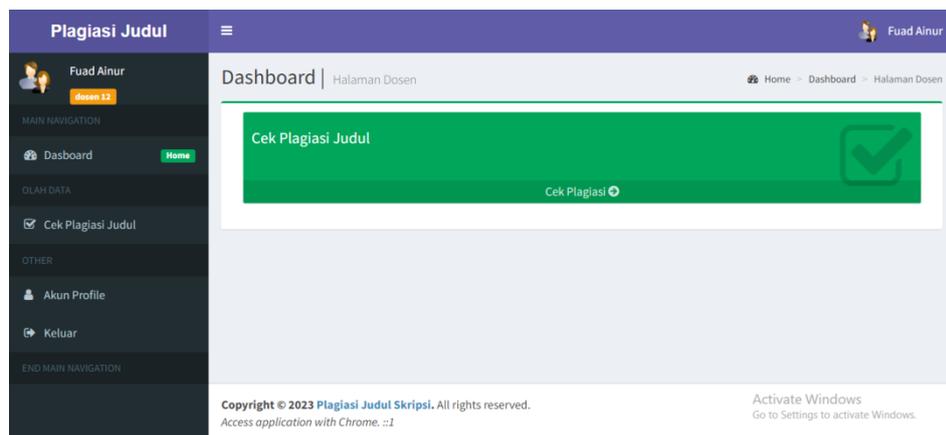


Gambar 4. 10 halaman login

Pada gambar 4.10, halaman ini adalah tampilan awal dari sistem yang di bangun dengan melakukan sebuah login untuk user dalam sistem

pendeteksi kemiripan judul skripsi dengan memasukkan data berupa username dan password.

2. Halaman dashboard



Gambar 4. 11 halaman dashboard

Gambar 4.11 menunjukkan halaman dashboard user. Pada dashboard user terdapat menu cek plagiasi judul skripsi dan akun profile.

3. Halaman cek plagiasi judul



Gambar 4. 12 input judul skripsi baru

Plagiasi Judul

Fuad Ainur

Dashboard | Hasil Cek Plagiasi

Judul skripsi anda: **implementasi algoritma fisher-yates shuffle pada permainan matching card dalam media pembelajaran bahasa indonesia**

Show 10 entries Search:

No	Nama Mahasiswa	NIM	Judul Skripsi	Persentase
21	Yulia	19533441	implementasi algoritma fisher-yates shuffle pada permainan matching card dalam media pembelajaran bahasa korea berbasis web.	75.00%
8	Gigi	19533211	implementasi algoritma decision tree pada internet of things inkubator tempe kedelai	14.29%
16	Risma	18111321	implementasi algoritma k-means clustering pada sistem informasi persebaran penyakit demam berdarah berbasis webgis	13.04%
4	Budi	18776651	implementasi algoritma fifo (first in first out) pada sistem antrian pendaftaran pasien berbasis android	12.50%

Gambar 4. 13 presentase kemiripan

Pada gambar 4.12, menunjukkan halaman cek plagiasi. user dapat melakukan cek plagiasi dengan memasukkan judul skripsi baru atau judul yang mau diajukan untuk dicek tingkat kemiripannya dengan judul skripsi angkatan sebelumnya. Lalu untuk tampilan setelah judul dicari kemiripannya terdapat pada gambar 4.13.

Judul skripsi anda: **Algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur**

Show 10 entries Search:

No	Nama Mahasiswa	NIM	Judul Skripsi	Persentase
24	Robot	19663374	algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur	100.00%
27	Udin	19663300	algoritma decision tree pada iot untuk perangkat hama ngengat tanaman bawang	97.50%
16	Gigi	19533211	implementasi algoritma decision tree pada internet of things inkubator tempe kedelai	22.22%

Gambar 4. 14 hasil presentase judul

Pada gambar 4.14, terdapat satu judul baru dan tiga judul pembanding yang ketiganya memiliki presentase kemiripan berbeda.

Dokumen 1 (A) = Algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur

Dokumen 2 (B) = Algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur

- a. Memisahkan ke dua dokumen tersebut menjadi array : (1)algoritma, (2)decision, (3)tree, (4)pada, (5)iot, (6)untuk, (7)pengelolaan, (8)suhu, (9)kandang, (10)ayam, (11)petelur

$$A = \{1,2,3,4,5,6,7,8,9,10,11\}$$

$$B = \{1,2,3,4,5,6,7,8,9,10,11\}$$

Kemudian mencari jumlah kata keseluruhan dari dua dokumen tersebut dan dari perhitungan di atas bisa dilihat jumlah kata keseluruhan yaitu 16 kata.

$$\text{Jadi } A \cup B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$$

Setelah mencari jumlah seluruh kata, selanjutnya mencari kata yang sama diantara dua dokumen tersebut. Kata yang sama yaitu (1)algoritma, (2)decision, (3)tree, (4)pada, (5)iot, (6)untuk, (7)pengelolaan, (8)suhu, (9)kandang, (10)ayam, (11)petelur

$$\text{Jadi untuk jumlah } A \cap B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$$

Langkah selanjutnya adalah menghitung kemiripan dari dua dokumen tersebut berdasarkan persamaan. Hasil perhitungan dikali 100 karena untuk mencari persentasenya.

$$(A, B) = \frac{11}{11} = 1 \times 100 = \mathbf{100\%}$$

Jadi nilai persamaan yang diperoleh didapat nilai kemiripan antara dua dokumen tersebut (A dan B) adalah 100%.

Dokumen 1 (A) = Algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur

Dokumen 2 (B) = Algoritma decision tree pada iot untuk perangkat hama ngengat tanaman bawang

- b. Memisahkan ke dua dokumen tersebut menjadi array : (1)algoritma, (2)decision, (3)tree, (4)pada, (5)iot, (6)untuk, (7)pengelolaan, (8)perangkat, (9)suhu, (10)hama, (11)kandang, (12)ngengat, (13)ayam, (14)tanaman, (15)petelur, (16)bawang.

$$A = \{1, 2, 3, 4, 5, 6, 7, 9, 11, 13, 15\}$$

$$B = \{1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16\}$$

Kemudian mencari jumlah kata keseluruhan dari dua dokumen tersebut dan dari perhitungan di atas bisa dilihat jumlah kata keseluruhan yaitu 16 kata.

$$\text{Jadi } A \cup B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$$

Setelah mencari jumlah seluruh kata, selanjutnya mencari kata yang sama diantara dua dokumen tersebut. Kata yang sama yaitu (1)algoritma, (2)decision, (3)tree, (4)pada, (5)iot, (6)untuk

$$\text{Jadi untuk jumlah } A \cap B = \{1, 2, 3, 4, 5, 6\}$$

Langkah selanjutnya adalah menghitung kemiripan dari dua dokumen tersebut berdasarkan persamaan (2.2). Hasil perhitungan dikali 100 karena untuk mencari persentasenya.

$$(A, B) = \frac{6}{16} = 0,375 \times 100 = \mathbf{37,5\%}$$

Jadi nilai persamaan yang diperoleh didapat nilai kemiripan antara dua dokumen tersebut (A dan B) adalah 37,5%

Dokumen 1 (A) = Algoritma decision tree pada iot untuk pengelolaan suhu kandang ayam petelur

Dokumen 2 (B) = Implementasi algoritma decision tree pada internet of things incubator tempe kedelai

- c. Memisahkan ke dua dokumen tersebut menjadi array : (1)implementasi, (2)algoritma, (3)decision, (4)tree, (5)pada, (6)iot, (7)internet, (8)untuk, (9)pengelolaan, (10)suhu, (11)of, (12)kandang, (13)things, (14)ayam, (15)incubator, (16)petelur, (17)tempe, (18)kedelai

$$A = \{2, 3, 4, 5, 6, 8, 9, 10, 12, 14, 16\}$$

$$B = \{1, 2, 3, 4, 5, 7, 11, 13, 15, 17, 18\}$$

Kemudian mencari jumlah kata keseluruhan dari dua dokumen tersebut dan dari perhitungan di atas bisa dilihat jumlah kata keseluruhan yaitu 16 kata.

$$\text{Jadi } A \cup B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18\}$$

Setelah mencari jumlah seluruh kata, selanjutnya mencari kata yang sama diantara dua dokumen tersebut. Kata yang sama yaitu (1)algoritma, (2)decision, (3)tree, (4)pada.

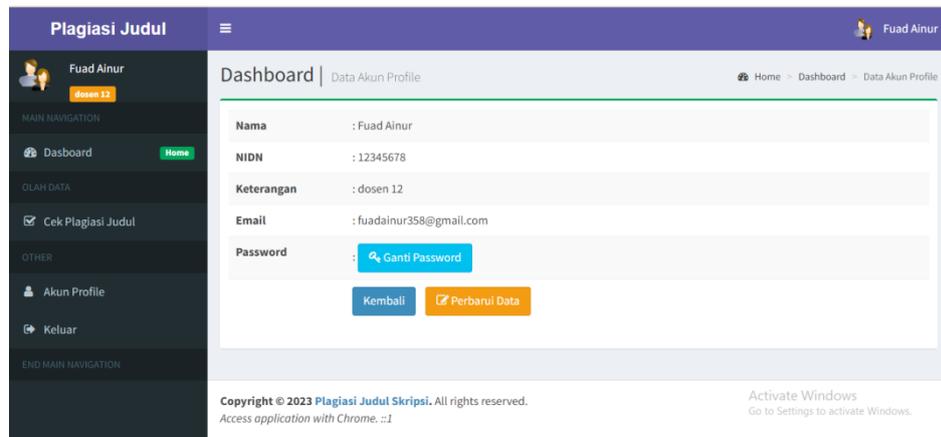
$$\text{Jadi untuk jumlah } A \cap B = \{1, 2, 3, 4\}$$

Langkah selanjutnya adalah menghitung kemiripan dari dua dokumen tersebut berdasarkan persamaan (2.2). Hasil perhitungan dikali 100 karena untuk mencari persentasenya.

$$(A, B) = \frac{4}{18} = 0,22 \times 100 = \mathbf{22\%}$$

Jadi nilai persamaan yang diperoleh didapat nilai kemiripan antara dua dokumen tersebut (A dan B) adalah 22%

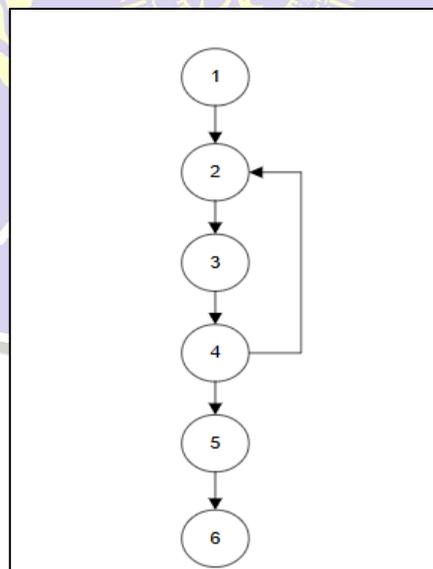
4. Akun profil



Gambar 4. 15 profil user

Pada halaman akun profile, berisi informasi user yang sedang login pada sistem. Dalam halaman ini juga dapat mengolah dan mengedit data user, terlihat pada gambar 4.15

4.3 Pengujian White Box testing



Gambar 4. 16 Case Login

Jalur 1 = 1-2-3-4-5-6

Jalur 2 = 1-2-3-4-2-3-5-6

Setelah diketahui jumlah jalur maka dilanjutkan dengan pengujian path yang berisikan jalur, skenario, dan hasil pengujian

Tabel 4. 1 Test Case Login

Path	1
Jalur	1-2-3-4-5-6
Skenario	<p>Start</p> <p>Masukkan username dan password</p> <p>Klik login</p> <p>Validasi data benar</p> <p>System menampilkan halaman utama</p> <p>End.</p>
Path	2
Jalur	1-2-3-4-2-3-5-6
Scenario	<p>Start</p> <p>Masukkan username dan password</p> <p>Klik login</p> <p>Validasi data salah</p> <p>Muncul pesan error. Masukkan kembali username dan password</p> <p>Klik login</p> <p>System menampilkan halaman utama</p> <p>end</p>

Halaman Pengujian	Berhasil
-------------------	----------

Tabel 4. 2 Test Case Plagiasi

Path	1
Jalur	1-2-3-4-5-6
Skenario	Start Cek plagiasi Masukkan judul skripsi Klik search System menampilkan halaman hasil presentase plagiasi End.
Halaman Pengujian	Berhasil

Setelah hasil pengujian pada tabel 4.2 berstatus berhasil, selanjutnya dilakukan Identifikasi dan kasus uji web plagiasi judul skripsi untuk proses pengujian white box.

Tabel 4. 3 identifikasi

No	Nama Class	Nama Method	Tujuan
1	Periksa	cek_plagiasi	Input judul baru untuk

cek
plagiasi

Berikut merupakan hasil dari proses identifikasi pengujian yang telah dilakukan dan dapat dilihat pada tabel 4.3 terkait hasil atau tujuan pengujiannya

Tabel 4. 4 Tabel Uji

Nomor uji	1
Input	<pre>//Lihat Data public function input(\$value='') { \$view = array('judul' =>'Cek Plagiasi Judul Skripsi', 'aksi' =>'input',); \$this->load- >view('dosen/cek_plagiasi/input',\$view); } public function calculate_similarity() { \$userTitle = \$this->input- >post('nama_judul_skripsi'); \$titles = \$this->M_judul_skripsi->view()- >result(); // Threshold kemiripan yang diperbolehkan \$threshold = 1; // Ganti dengan nilai threshold yang Anda inginkan // Variabel untuk menyimpan data judul skripsi yang sesuai \$matchingData = array(); // Default kosong // Hitung kemiripan dengan judul dari database \$similarTitles = array(); foreach (\$titles as \$dbTitle) {</pre>

```

        $jaccardSimilarity = $this-
>calculateJaccardSimilarity($userTitle, $dbTitle-
>nama_judul_skripsi);
        if ($jaccardSimilarity >= $threshold) {
            // Simpan data judul skripsi, nama
mahasiswa, dan nim ke dalam array
            $matchingData[] = array(
                'nama_mahasiswa' => $dbTitle-
>nama_mahasiswa,
                'nim' => $dbTitle->nim,
                'judul_skripsi' => $dbTitle-
>nama_judul_skripsi,
                'kemiripan' =>
min($jaccardSimilarity, 100)
            );
        }
    }
}

```

Output

```

<?php $this->load->view('template/header'); ?>
<style>
    .highlight {
        background-color: yellow; /* Ganti warna latar belakang
sesuai preferensi Anda */
    }
</style>
<h4>Judul skripsi anda: <strong><?= $userTitle ?></strong></h4>
<div class="table-responsive">
    <table id="example1" class="table table-bordered table-
striped">
    <thead>
    <tr>
    <th>No</th>

```

```
<th>Nama Mahasiswa</th>
```

```
<th>NIM</th>
```

```
<th>Judul Skripsi</th>
```

```
<th>Persentase</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<?php
```

```
    // Membuat array asosiatif yang berisi judul, nama
    mahasiswa, nim, dan kemiripan
```

```
    $dataForSorting = array();
```

```
    foreach ($matchingData as $data) {
```

```
        $dataForSorting[] = array(
```

```
            'nama_mahasiswa' => $data['nama_mahasiswa'],
```

```
            'nim' => $data['nim'],
```

```
            'judul' => $data['judul_skripsi'],
```

```
            'kemiripan' => min($data['kemiripan'], 100)
```

```
        );
```

```
    }
```

```
    // Fungsi untuk mengurutkan array berdasarkan kemiripan
```

```
    // usort($dataForSorting, function ($a, $b) {
```

```
        // return $a['kemiripan'] < $b['kemiripan'];
```

```
// });
```

```
$no = 1;
```

```
foreach ($dataForSorting as $data) {
```

```
    $nama_mahasiswa = $data['nama_mahasiswa'];
```

```
    $nim = $data['nim'];
```

```
    $nama_judul_skripsi = $data['judul'];
```

```
    $similarity = $data['kemiripan'];
```

```
    $colorClass = "";
```

```
    if ($similarity >= 60) {
```

```
        $colorClass = 'bg-red'; // Merah untuk kemiripan di  
        atas 60% dan 100%
```

```
    } elseif ($similarity >= 45) {
```

```
        $colorClass = 'bg-yellow'; // Kuning untuk kemiripan  
        di antara 45% dan 59.9%
```

```
    } elseif ($similarity >= 26) {
```

```
        $colorClass = 'bg-blue'; // Biru untuk kemiripan di  
        antara 26% dan 44.9%
```

```
    } elseif ($similarity >= 0) {
```

```
        $colorClass = 'bg-green'; // Hijau untuk kemiripan di  
        antara 0% dan 25.9%
```

```
    }
```

```
// Membagi judul menjadi kata-kata
```

```
$userTitleWords = explode(' ', strtolower($userTitle));
```

```
$judulSkripsiWords = explode(' ',  
strtolower($nama_judul_skripsi));
```

```
// Menyimpan kata-kata yang sama
```

```
$matchingWords = array_intersect($userTitleWords,  
$judulSkripsiWords);
```

```
// Menyusun ulang judul dengan kata-kata yang sama  
diberi latar belakang warna
```

```
$highlightedTitle = "";
```

```
foreach ($judulSkripsiWords as $word) {
```

```
    if (in_array($word, $userTitleWords)) {
```

```
        $highlightedTitle .= "<span  
class='highlight'>$word</span>";
```

```
    } else {
```

```
        $highlightedTitle .= "$word ";
```

```
    }
```

```
}
```

```
?>
```

```
<tr>
```

```
    <td><?= $no++ ?></td>
```

```
    <td><?= $nama_mahasiswa ?></td>
```



```
<td><?= $nim ?></td>
```

```
<td><?= rtrim($highlightedTitle) ?></td>
```

```
<td>
```

```
<span class="<?= $colorClass ?> font-weight-  
bold"><?= number_format($similarity, 2) ?> %</span>
```

```
</td>
```

```
</tr>
```

```
<?php } ?>
```

```
</tbody>
```

```
</table>
```

```
</div>
```

```
<?php $this->load->view('template/footer'); ?>
```

Pada tabel 4.4 menjelaskan proses algoritma berfungsi pada setiap menginput data berupa judul baru. Pengujian tersebut berlaku pada setiap data yang juga akan di input pada web aplikasi. Hasil dari pengujian white box memenuhi hasil yang diharapkan.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan dari hasil penelitian di atas adalah jumlah kesamaan kata yang sangat berpengaruh pada hasil cek kemiripan judul ini, diperlukan penggunaan rumus perhitungan yang tepat agar memperoleh hasil yang benar. Presentase kemiripan pada dua dokumen yang dibandingkan atau di uji akan menghasilkan nilai presentase kemiripan yang berbeda apabila jumlah kata sama dari kedua dokumen uji tersebut yang dimasukan berbeda juga nilainya. Hasil presentase tergantung dari banyaknya jumlah kata yang sama pada dokumen 1 dan 2, dokumen 1 sebagai dokumen uji dan dokumen 2 sebagai dokumen pembanding. Jaccard similarity yang digunakan untuk menghitung besarnya kemiripan antar dokumen juga cukup baik karena mudah untuk menemukan kesamaan dari dua dokumen dan membaginya dengan jumlah item kedua yang diperoleh dengan menggabungkan kedua kertas. Deteksi plagiarisme menggunakan Algoritma Horspool dengan Jaccard Similarity efektif untuk menganalisis kesamaan dua dokumen dan menentukan presentase kemiripan.

5.2 Saran

Pada aplikasi deteksi kemiripan judul skripsi ini masih banyak kekurangannya, penelitian selanjutnya diharapkan bisa menambah beberapa fitur baru untuk mendukung berjalannya aplikasi ini. Penelitian selanjutnya direncanakan untuk bisa mendeteksi kemiripan apabila pengguna ingin mengambil salah satu bagian dari artikel ilmiah yang menggunakan berbahasa Inggris lalu bisa menerjemahkannya ke dalam bahasa Indonesia dengan baik dan benar.

DAFTAR PUSTAKA

- [1] E. Alimudin and T. Yuliati, "Deteksi Kemiripan Judul dengan Metode Term Weighting pada Sistem Informasi Pengajuan Judul Skripsi," vol. 1, pp. 20–25, 2016.
- [2] A. Husein and S. N. Alam, "Pembuatan Sistem Pendeteksi Kemiripan Pengajuan Topic Skripsi Menggunakan Metode Rabin - Karp," *Semin. Nas. Sist. Inf. dan Teknol. Inf. 2018*, pp. 270–274, 2018.
- [3] J. P. Utama, "TINDAK PIDANA PLAGIARISME JASA PEMBUATAN SKRIPSI SEBAGAI PELANGGARAN HAK CIPTA," vol. 2, no. 3, pp. 201–212, 2013.
- [4] M. H. Irfani, "Pengaruh Kemampuan Numerik dan Algoritma terhadap Kemampuan Pemrograman dalam Pilihan Tema Skripsi," *Sistemasi*, vol. 10, no. 1, p. 64, 2021, doi: 10.32520/stmsi.v10i1.1088.
- [5] F. Syah, N. Saputra, and S. Maharani, "Sistem Deteksi Kemiripan Judul Skripsi Dengan Metode Term Weight Dan Word Embedding Di Program Studi Informatika Universitas PGRI Yogyakarta," vol. 9, no. 2, pp. 1–15, 2022.
- [6] B. H. Irawan, M. S. H. Simarankir, and E. Erlinna, "Deteksi Kemiripan Judul Skripsi Menggunakan Algoritma Levenshtein Distance Pada Kampus Stmik Mic Cikarang," *Eduatic - Sci. J. Informatics Educ.*, vol. 7, no. 2, pp. 143–149, 2021, doi: 10.21107/edutic.v7i2.10051.
- [7] N. L. P. Merawati and S. Hartati, "Sistem Rekomendasi Topik Skripsi Menggunakan Metode Case Based Reasoning," *J. Ilm. Teknol. Infomasi Terap.*, vol. 4, no. 3, pp. 174–183, 2018, doi: 10.33197/jitter.vol4.iss3.2018.164.
- [8] S. WULAN, "ANALISIS PENERAPAN STRING MATCHING DALAM

KOMPARASI DATA KEPESERTAAN JAMINAN KESEHATAN MASYARAKAT (JAMKESMAS),” *Phys. Rev. E*, 2011, [Online]. Available:
<http://www.ainfo.inia.uy/digital/bitstream/item/7130/1/LUZARDO-BUIATRIA-2017.pdf>

- [9] J. Adirachman *et al.*, “Pengembangan Game Edukasi Tebak Asmaul Husna,” 2018.
- [10] E. Malat, E. Arriyanti, and H. E. Wati, “PENCARIAN ABSTRAK SKRIPSI BERBASIS WEB,” pp. 1–6.
- [11] N. Marbun, S. Aripin, E. Buulolo, P. T. Informatika, U. S. Utara, and P. T. Informatika, “Implementasi algoritma horspool pada aplikasi istilah fashion,” vol. 3, pp. 549–551, 2019, doi: 10.30865/komik.v3i1.1641.
- [12] F. T. Waruwu and R. K. Hondro, “Penerapan Algoritma Horspool pada Aplikasi Katalog Buku Perpustakaan,” no. September, pp. 881–885, 2019.
- [13] N. Alamsyah, “Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi Dengan Algoritma Winnowing,” *Technol. J. Ilm.*, vol. 8, no. 4, p. 205, 2017, doi: 10.31602/tji.v8i4.1119.
- [14] L. J. Purba and L. Sitorus, “Perancangan Aplikasi Untuk Menghitung Persentase Kemiripan Proposal Dan Isi Skripsi Dengan Algoritma Rabin-Karp,” *J. Tek. Inform. Unika St. Thomas*, vol. 3, no. 1, pp. 17–25, 2018.
- [15] A. Y. Diny, “Self-Efficacy Dan Perilaku Plagiarisme Di Media Sosial Pada Mahasiswa Pengguna Media Sosial,” pp. 1–98, 2018, [Online]. Available: <https://eprints.umm.ac.id/39234/>
- [16] N. C. Wahyuni, “When Plagiarism Is A Matter,” *Rec. Libr. J.*, vol. 4, no. 1, p. 8, 2019, doi: 10.20473/rlj.v4-i1.2018.8-14.
- [17] E. Mesi, D. Oktarina, and S. Matching, “Penerapan Algoritma Horspool Pada Sistem Pendataan Obat Pada Apotek Fajar Mas,” 2021.

- [18] D. B. Cibro, "Perancangan Aplikasi Biografi Tokoh Penerima Penghargaan Nobel Dengan Menerapkan Algoritma Horspool Boyer Moore," vol. 1, no. 6, pp. 362–367, 2021.
- [19] W. Fuadi, R. Risawandi, and R. Yanti, "Aplikasi Doa Para Nabi Dan Rasul Dalam Al-Qur'an Menggunakan Algoritma Horspool Berbasis Android," *TECHSI - J. Tek. Inform.*, vol. 12, no. 1, p. 1, 2020, doi: 10.29103/techsi.v12i1.1702.
- [20] S. Sunardi, A. Yudhana, and I. A. Mukaromah, "Implementasi Deteksi Plagiarisme Menggunakan Metode N-Gram Dan Jaccard Similarity Terhadap Algoritma Winnowing," *Transmisi*, vol. 20, no. 3, p. 105, 2018, doi: 10.14710/transmisi.20.3.105-110.
- [21] S. Utomo, I. Much, I. Subroto, and A. Riansyah, "Deteksi Plagiat Tugas Akhir dengan Metode Jaccard Similarity," vol. 4, no. 2, pp. 132–141, 2022.
- [22] M. Suhartanto, "Pembuatan Website Sekolah Menengah Pertama Negeri 3 Delanggu Dengan Menggunakan Php Dan MySQL," *J. Speed-Sentra Penelit. Enginerring dan Edukasi*, vol. 4, no. 1, pp. 1–8, 2012.
- [23] A. Solichin, "MySQL Dari Pemula Higga Akhir," pp. 1–117, 2010.
- [24] J. Sovia, Rini dan Febio, "MEMBANGUN APLIKASI E-LIBRARY MENGGUNAKAN HTML, PHP SCRIPT, DAN MYSQL DATABASE Rini Sovia dan Jimmy Febio," *Processor*, vol. 6, no. 2, pp. 38–54, 2011.
- [25] A. B. Hikmah, "Cara Cepat Membangun Website dari Nol," *cara cepat membangun website dari nol Stud.*, 2015.
- [26] M. R. Arief, "Pemrograman Web Dinamis menggunakan PHP dan MySql," 2012.
- [27] R. S. Pressman, *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku Dua)*. 2002.
- [28] A. Fahma Rosyada, I. Sukirman, M. Afrizal Nur, and A. Saifudin,

“Pengujian Sistem Informasi Aplikasi Perpustakaan basis Website Menggunakan White Box Testing,” *J. Multidisiplin ilmu*, vol. 1, no. 6, pp. 1034–1039, 2022, [Online]. Available: <https://journal.mediapublikasi.id/index.php/oktal>

- [29] I. Ahmad, R. I. Borman, G. G. Caksana, and J. Fakhrurozi, “IMPLEMENTASI STRING MATCHING DENGAN ALGORITMA BOYER- MOORE UNTUK MENENTUKAN TINGKAT KEMIRIPAN PADA PENGAJUAN JUDUL SKRIPSI / TA MAHASISWA (STUDI KASUS : UNIVERSITAS XYZ),” vol. 4, no. 1, pp. 53–58, 2021.

