

**PERANCANGAN SISTEM PENILAIAN UJIAN OTOMATIS
DENGAN ALGORITMA *SIMILARITY TEXT* DAN
OPTIMASI DENGAN ALGORITMA *STEMMER*
SASTRAWI PADA SEKOLAH DHARMA
BAKTI LUBUK PAKAM**

SKRIPSI

Oleh:

**TINIS ROY
NIM.2044068**



**JENJANG PENDIDIKAN STRATA-1
PROGRAM STUDI TEKNIK INFORMATIKA
STMIK TIME
MEDAN
2024**

LEMBAR PENGESAHAN

**PERANCANGAN SISTEM PENILAIAN UJIAN OTOMATIS
DENGAN ALGORITMA *SIMILARITY TEXT* DAN
OPTIMASI DENGAN ALGORITMA *STEMMER*
SASTRAWI PADA SEKOLAH DHARMA
BAKTI LUBUK PAKAM**

SKRIPSI

Diajukan Untuk Melengkapi Persyaratan Guna
Mendapat Gelar Sarjana Strata Satu
Program Studi Teknik Informatika

Oleh:

TINIS ROY

NIM.2044068

Disetujui Oleh:

Dosen Pembimbing I



David, S.Kom., M.TI

Dosen Pembimbing II



Wilson, S.Kom.,

Medan, 29 Juli 2024

Diketahui dan Disahkan Oleh:

Ketua STMIK TIME

Dr. Edi Wijaya, S.Kom., M.Kom., M.M

Ketua Program Studi



Robet, M.Kom

Kepada

Yth Ketua STMIK TIME

Dr. Edi Wijaya, S.Kom., M.Kom., M.M

di Medan

Hal : **Surat Pernyataan Bebas Plagiat**

Saya yang bertanda tangan di bawah ini :

Nama	: Tinis Roy
NIM	: 2044068
Program Studi	: Teknik Informatika
Angkatan	: 2020
Judul Skripsi	: Perancangan Sistem Penilaian Ujian Otomatis Dengan Algoritma <i>Similarity Text</i> dan Optimasi Dengan Algoritma <i>Stemmer Sastrawi</i> Pada Sekolah Dharma Bakti Lubuk Pakam

Dengan ini menyatakan bahwa skripsi yang Saya tulis adalah karya ilmiah asli dari pikiran saya sendiri.

Apabila kemudian hari terbukti bahwa skripsi tersebut merupakan Plagiat, maka saya bersedia gelar dan ijazah yang saya peroleh ditinjau kembali.

Demikian surat pernyataan ini Saya perbuat dengan pikiran yang sehat, tanpa tekanan dari siapapun.

Medan, 30 Juli 2024



Yang membuat pernyataan
Tinis Roy

Diketahui oleh



Pembantu Ketua I
Hendri, M.Kom

ABSTRAK

Penelitian ini bertujuan untuk merancang sistem penilaian ujian otomatis berbasis *text mining* di Sekolah Dharma Bakti Lubuk Pakam. Latar belakang dari penelitian ini adalah kebutuhan akan solusi yang dapat mempermudah dan mempercepat proses koreksi jawaban *essay* yang sering kali memerlukan waktu dan usaha yang signifikan dari pengajar. Sistem yang diusulkan mengintegrasikan algoritma *Similarity Text* dan *Stemmer Sastrawi* untuk meningkatkan akurasi penilaian. Teknik *text mining* digunakan untuk melakukan *preprocessing* pada teks jawaban peserta ujian sebelum dibandingkan dengan jawaban sebenarnya menggunakan algoritma *Similarity Text*. Penelitian sebelumnya telah menunjukkan keberhasilan dalam penerapan algoritma untuk penilaian otomatis, namun belum optimal dalam tahap *preprocessing*. Dengan implementasi *Stemmer Sastrawi*, diharapkan akurasi penilaian dapat ditingkatkan. Hasil penelitian ini menunjukkan bahwa sistem penilaian ujian otomatis berbasis *text mining* dapat mempermudah proses penilaian jawaban *essay* di Sekolah Dharma Bakti Lubuk Pakam, membuatnya lebih cepat dan akurat. Algoritma yang diterapkan terbukti mampu memberikan penilaian yang tepat dan akurat, sehingga dapat mengurangi beban kerja pengajar dan meningkatkan efisiensi dalam proses penilaian.

Kata Kunci: Penilaian Ujian Otomatis, *Text Mining*, *Similarity Text*, *Stemmer Sastrawi*, Sekolah Dharma Bakti, Koreksi Jawaban *Essay*

ABSTRACT

This research aims to design an automatic exam grading system based on text mining at Dharma Bakti School in Lubuk Pakam. The background of this research is the need for a solution that can facilitate and speed up the process of grading essay answers, which often requires significant time and effort from teachers. The proposed system integrates the Similarity Text algorithm and Sastrawi Stemmer to improve grading accuracy. Text mining techniques are used to preprocess the exam answers before comparing them with the correct answers using the Similarity Text algorithm. Previous research has shown the success of algorithm application for automatic grading, but it has not been optimal in the preprocessing stage. With the implementation of Sastrawi Stemmer, it is expected that grading accuracy can be enhanced. The results of this research indicate that the automatic exam grading system based on text mining can facilitate the process of grading essay answers at Dharma Bakti School in Lubuk Pakam, making it faster and more accurate. The implemented algorithm has proven capable of providing precise and accurate grading, thereby reducing the workload of teachers and increasing efficiency in the grading process.

Keywords: Automatic Exam Grading, Text Mining, Similarity Text, Sastrawi Stemmer, Dharma Bakti School, Essay Grading

KATA PENGANTAR

Ucapan puji dan syukur kepada hadirat Tuhan Yang Maha Esa atas segala berkat, hikmat, dan pertolongan-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi ini tepat pada waktunya.

Tugas Akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan Program Strata I pada STMIK TIME. Penulis berharap agar hasil dari penelitian yang dilakukan ini dapat membawa banyak manfaat bagi berbagai pihak.

Dalam menyelesaikan tugas akhir ini, penulis banyak mendapat bantuan dari berbagai pihak. Oleh karena itu, melalui kesempatan ini penulis ingin menyampaikan terima kasih kepada:

1. Bapak David, S.Kom., M.TI, selaku Pembimbing I yang telah membimbing penulis selama mengerjakan skripsi ini.
2. Bapak Wilson, S.Kom., M.M, selaku Pembimbing II yang telah membimbing penulis selama mengerjakan skripsi ini.
3. Bapak Simon Kangga Lee, selaku Ketua Yayasan STMIK TIME Medan.
4. Bapak Dr. Edi Wijaya, S.Kom., M.Kom., M.M, selaku Ketua STMIK TIME Medan.
5. Bapak Hendri, M.Kom., selaku Pembantu Ketua I STMIK TIME Medan.
6. Bapak Robet, M.Kom., selaku Ketua Program Studi Teknik Informatika STMIK TIME Medan.
7. Seluruh Dosen STMIK TIME Medan yang telah mendidik dan membimbing penulis selama masa perkuliahan.
8. Teristimewa kepada orang tua dan keluarga tercinta, yang telah memberikan dukungan baik moral maupun materi kepada penulis selama ini.
9. Seluruh teman yang telah banyak membantu penulis dan memberikan dorongan serta semangat dalam menyelesaikan skripsi ini
10. Semua yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak dapat disebutkan satu persatu

Penulis menyadari dalam proses penyelesaian skripsi yang berjudul “Perancangan Sistem Penilaian Ujian Otomatis Dengan Algoritma *Similarity Text* dan Optimasi Dengan Algoritma *Stemmer Sastrawi* Pada Sekolah Dharma Bakti Lubuk Pakam” masih jauh dari sempurna, mengingat keterbatasan kemampuan yang penulis miliki dan terbatasnya waktu yang digunakan. Untuk itu, penulis mengharapkan saran, kritikan, dan masukan yang membangun demi perkembangan dan kemajuan penelitian selanjutnya. Semoga penulisan skripsi ini dapat bermanfaat bagi pihak yang berkepentingan.

Akhir kata penulis berharap agar laporan penulisan Tugas Akhir ini dapat berguna dan bermanfaat bagi pembaca.

Medan, 30 Juli 2024

Penulis,

A handwritten signature in black ink, appearing to be 'Tinis Roy', with a stylized 'T' and 'R'.

Tinis Roy

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	v
LAMPIRAN.....	vi
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	ix
DAFTAR LAMPIRAN	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	4
1.4 Tujuan dan Manfaat Penelitian	4
1.5 Metodologi Penelitian.....	5
1.6 Analisis Sistem.....	7
1.7 Perancangan Sistem	8
1.8 Sistematika Penulisan	8
BAB II TINJAUAN PUSTAKA.....	10
2.1 Kecerdasan Buatan.....	10
2.2 <i>Natural Language Processing (NLP)</i>	14
2.3 <i>Text Mining</i>	16
2.4 Algoritma Nazief dan Adriani	19
2.5 <i>Stemmer</i> Sastrawi.....	20
2.6 Algoritma <i>Similarity Text</i>	20
2.7 <i>Use Case Diagram</i>	22
2.8 <i>Entity Relationship Diagram</i>	23
2.9 Penelitian Terdahulu	24
BAB III METODE PENELITIAN	25

3.1	Analisis Masalah.....	25
3.1.1	Analisis Proses Yang Berjalan Sesuai di Lapangan	25
3.1.2	Analisis Algoritma Yang Diusulkan	26
3.1.3	Analisis Sistem Usulan.....	29
3.2	Perancangan	34
3.2.1	Perancangan Tampilan	35
3.2.2	Perancangan Basis Data	53
BAB IV HASIL DAN PEMBAHASAN		56
4.1	Hasil	56
4.2	Pembahasan.....	68
BAB V KESIMPULAN DAN SARAN		70
5.1	Kesimpulan	70
5.2	Saran	70
DAFTAR PUSTAKA		72
LAMPIRAN		

DAFTAR GAMBAR

Gambar 3.1 <i>Use Case Diagram</i> Sistem Penilaian Ujian Otomatis.....	30
Gambar 3.2 Rancangan Tampilan Awal	35
Gambar 3.3 Rancangan Tampilan Daftar Akun Guru	36
Gambar 3.4 Rancangan Tampilan Masuk Guru.....	37
Gambar 3.5 Rancangan Tampilan Beranda Guru	38
Gambar 3.6 Rancangan Tampilan Kelola Data Siswa	39
Gambar 3.7 Rancangan Tampilan Tambah Data Siswa.....	40
Gambar 3.8 Rancangan Tampilan Ubah Data Siswa	41
Gambar 3.9 Rancangan Tampilan Hapus Data Siswa	42
Gambar 3.10 Rancangan Tampilan Tambah Data Soal	43
Gambar 3.11 Rancangan Tampilan Ubah Data Soal	44
Gambar 3.12 Rancangan Tampilan Hapus Data Soal.....	45
Gambar 3.13 Rancangan Tampilan Lihat Hasil Ujian.....	46
Gambar 3.14 Rancangan Tampilan Detail Hasil Ujian.....	47
Gambar 3.15 Rancangan Tampilan Hasil Akhir	48
Gambar 3.16 Rancangan Tampilan Beranda Siswa	49
Gambar 3.17 Rancangan Tampilan Jawab Soal.....	50
Gambar 3.18 Rancangan Tampilan Tambah Jawaban	51
Gambar 3.19 Rancangan Tampilan Ubah Jawaban	52
Gambar 3.20 ERD Sistem Penilaian Ujian Otomatis di Sekolah Dharma Bakti Lubuk Pakam	53
Gambar 4.1 Tampilan Awal.....	56
Gambar 4.2 Tampilan Daftar Akun Guru	57
Gambar 4.3 Tampilan Masuk Guru	58
Gambar 4.4 Tampilan Beranda Guru	58
Gambar 4.5 Tampilan Kelola Data Siswa.....	59
Gambar 4.6 Tampilan Tambah Data Siswa	60
Gambar 4.7 Tampilan Ubah Data Siswa.....	60

Gambar 4.8 Tampilan Hapus Data Siswa	61
Gambar 4.9 Tampilan Tambah Data Soal.....	61
Gambar 4.10 Tampilan Ubah Data Soal	62
Gambar 4.11 Tampilan Hapus Data Soal.....	62
Gambar 4.12 Tampilan Lihat Hasil Ujian.....	63
Gambar 4.13 Tampilan Detail Hasil Ujian	64
Gambar 4.14 Tampilan Hasil Akhir.....	65
Gambar 4.15 Tampilan Beranda Siswa.....	65
Gambar 4.16 Tampilan Jawab Soal	66
Gambar 4.17 Tampilan Tambah Jawaban.....	67
Gambar 4.18 Tampilan Ubah Jawaban	67

DAFTAR TABEL

Tabel 1.1 Jadwal Penelitian.....	6
Tabel 2.1 Simbol <i>Use Case Diagram</i> [14].....	22
Tabel 2.2 Simbol <i>Entity Relationship Diagram</i> [14]	23
Tabel 3.1 Contoh Soal, Kunci Jawaban, dan Jawaban Siswa	27
Tabel 3.2 Skenario <i>Use Case</i> : Daftar Akun Guru.....	30
Tabel 3.3 Skenario <i>Use Case</i> : <i>Login</i>	31
Tabel 3.4 Skenario <i>Use Case</i> : Kelola Data Siswa	31
Tabel 3.5 Skenario <i>Use Case</i> : Kelola Data Soal.....	32
Tabel 3.6 Skenario <i>Use Case</i> : Lihat Hasil Ujian	33
Tabel 3.7 Skenario <i>Use Case</i> : Menjawab Soal Ujian	34
Tabel 3.8 Struktur Tabel Guru	53
Tabel 3.9 Struktur Tabel Hasil	54
Tabel 3.10 Struktur Tabel Jawaban.....	54
Tabel 3.11 Struktur Tabel Siswa.....	54
Tabel 3.12 Struktur Tabel Penilaian.....	55
Tabel 3.13 Struktur Tabel Soal	55

DAFTAR LAMPIRAN

Lampiran 1 Daftar Riwayat Hidup

Lampiran 2 Surat Keputusan Dosen Pembimbing Skripsi

Lampiran 3 Kartu Bimbingan Dosen I

Lampiran 4 Kartu Bimbingan Dosen II

Lampiran 5 *Listing* Program

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ujian merupakan salah satu alat penilaian dalam proses belajar mengajar yang dilaksanakan secara menyeluruh untuk menentukan kualitas pembelajaran dan merupakan suatu alat ukur untuk mengukur performa peserta didik. Namun di sisi lain ujian merupakan suatu pekerjaan rumah dan kesibukan tambahan yang harus dilakukan oleh pengajar terutama dalam proses koreksi [1]. Soal ujian biasanya memiliki beragam jenis jawaban ada yang menggunakan tipe pilihan berganda dan ada yang berupa jawaban *essay*. Jawaban dengan tipe *essay* memberikan beragam kemungkinan jawaban, tergantung pemahaman masing-masing siswa. Hasil jawaban tipe *essay* bukan hanya benar atau salah tetapi juga bisa menghasilkan nilai hampir benar. Sehingga skor nilai untuk jawaban *essay* bisa beragam [2].

Sekolah Dharma Bakti merupakan sebuah sekolah yang beralamat di jalan Bidan Bakaran Batu No. 8, Lubuk Pakam. Seperti sekolah pada umumnya, sekolah Dharma Bakti berfokus dalam kegiatan belajar mengajar anak-anak dari tingkat SD hingga SMA. Selain kegiatan belajar mengajar juga terdapat adanya proses ujian pada waktu tertentu. Terdapat banyak sekali bentuk soal ujian seperti soal pilihan berganda dan *essay*. Soal dengan tipe pilihan berganda lebih mudah dikoreksi namun berbeda dari soal dengan tipe *essay* dikarenakan soal *essay* dikoreksi dengan cara membaca satu persatu jawaban peserta didik. Salah satu kekurangan pada jawaban dengan jenis *essay* yaitu sulitnya menilai jawaban dan membutuhkan

banyak waktu untuk mengoreksi. Apalagi jika jumlah peserta didik yang diajar oleh pengajar sangat banyak maka memerlukan waktu yang lama juga dalam proses koreksi. Di sisi lain, pengajar juga memiliki kendala sendiri-sendiri dalam proses koreksi jawaban, terkadang koreksi harus dibawa pulang oleh pengajar, sedangkan dirumah para pengajar juga memiliki tanggung jawab yang tidak bisa ditinggalkan. Seorang pengajar bukan hanya berkewajiban mengajar dan mengoreksi jawaban siswa, namun juga memiliki kewajiban administrasi lainnya seperti menyusun Rencana Pelaksanaan Pembelajaran, menyusun silabus, program tahunan dan lain sebagainya, dimana kewajiban tersebut membutuhkan waktu yang tidak sedikit. Sejumlah pengajar menyatakan bahwa kelelahan akan mengakibatkan inkonsistensi dalam pemberian skor (*human error*) [3].

Dari uraian permasalahan tersebut, maka solusi untuk mengatasinya adalah dengan membangun sebuah aplikasi penilaian jawaban *essay* secara otomatis. Aplikasi penilaian jawaban *essay* akan mengimplementasikan algoritma tertentu sehingga jawaban dari peserta ujian akan dibandingkan dengan jawaban yang benar kemudian disimpulkan persentase kebenarannya. Dengan adanya aplikasi tersebut maka dapat memudahkan pihak pengajar melakukan koreksi secara tepat dan akurat. Solusi membangun aplikasi penilaian jawaban *essay* otomatis sudah pernah dibahas pada penelitian sebelumnya seperti penelitian pada tahun 2021 membahas mengenai penilaian *essay* pendek otomatis dengan pencocokan kata dengan menggunakan teknik *Natural Language Processing* (NLP) dan frasa nomina. Hasil penelitian menunjukkan bahwa implementasi kedua teknik ini mendapatkan akurasi dan nilai uji sangat baik dengan nilai *Pearson Correlation* sebesar 0,83 [4]. Penelitian terbaru pada tahun 2022 membahas perancangan aplikasi koreksi ujian

essay otomatis dengan menggunakan metode *Cosine Similarity*. Hasil penelitian menunjukkan bahwa aplikasi yang dibangun dapat mempersingkat waktu pengajar dalam mengoreksi jawaban peserta ujian [5].

Penelitian yang dilakukan sebelumnya sudah cukup baik, namun fokus penelitian sebelumnya lebih ke algoritma bukan ke proses *preprocessing* teks jawaban. Padahal akurasi dari *text mining* sangat berpengaruh dari tahapan *preprocessing* yang dilakukan. Oleh sebab itu pada penelitian ini akan diimplementasikan teknik berbasis *text mining* yaitu algoritma *Similarity Text* dan optimasi dengan *Stemmer Sastrawi* sehingga diharapkan dapat meningkatkan akurasi dari proses penilaian jawaban essay. Algoritma *Stemmer Sastrawi* akan melakukan proses *preprocessing* terhadap teks jawaban peserta ujian dan kemudian selanjutnya algoritma *Similarity Text* akan melakukan perbandingan terhadap jawaban peserta ujian dengan jawaban sebenarnya sehingga dengan adanya proses *preprocessing* yang ketat maka diharapkan akurasi penilaian essay juga akan semakin akurat.

Berdasarkan uraian yang telah dipaparkan, maka akan dilakukan penelitian dengan mengambil judul “**PERANCANGAN SISTEM PENILAIAN UJIAN OTOMATIS BERBASIS TEXT MINING PADA SEKOLAH DHARMA BAKTI LUBUK PAKAM**”.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang diuraikan, maka rumusan masalah pada penelitian ini yaitu:

1. Bagaimana membangun sebuah sistem penilaian ujian otomatis yang dapat menyelesaikan permasalahan para pengajar ketika melakukan koreksi nilai ujian dikarenakan menghabiskan waktu yang lama serta terkadang salah dikarenakan kecapean (*human error*)?
2. Bagaimana keakuratan dari *algoritma Similarity Text* dan optimasi dengan *algoritma Stemmer Sastrawi* dalam melakukan penilaian ujian *essay*?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini yaitu:

1. Simulasi dibatasi pada tingkat pendidikan SMA di sekolah Dharma Bakti.
2. Teknik *text mining* dibatasi dengan penerapan algoritma *Similarity Text* dan optimasi dengan *Stemmer Sastrawi*.
3. *User* yang terlibat dalam aplikasi terdiri dari pengajar dan peserta ujian.
4. Belum memiliki fitur kompleks yang membahas mengenai ujian secara *online*.

1.4 Tujuan dan Manfaat Penelitian

Tujuan yang diharapkan dalam penelitian ini adalah:

1. Untuk membangun sebuah sistem penilaian ujian otomatis yang dapat menyelesaikan permasalahan para pengajar ketika melakukan koreksi nilai ujian dikarenakan menghabiskan waktu yang lama serta terkadang salah dikarenakan kecapean (*human error*)

2. Untuk menganalisis keakuratan dari *algoritma Similarity Text dan optimasi dengan algoritma Stemmer Sastrawi* dalam melakukan penilaian ujian *essay*.

Manfaat yang diharapkan dalam penelitian ini adalah:

1. Meringankan pekerjaan pengajar dalam melakukan koreksi jawaban ujian peserta ujian dengan adanya aplikasi yang dibangun.
2. Proses penilaian menjadi lebih akurat dengan adanya aplikasi yang dibangun karena dapat mencegah terjadinya *human error* dari pihak pengajar ketika melakukan penilaian.
3. Hasil analisis algoritma *Similarity Text* dan optimasi dengan *Stemmer Sastrawi* dapat dijadikan acuan untuk diterapkan pada kasus-kasus lainnya seperti untuk mendeteksi plagiat.

1.5 Metodologi Penelitian

Metodologi penelitian yang digunakan pada dimana perincian pelaksanaannya adalah sebagai berikut:

1. Jadwal Penelitian

Berikut ini akan diuraikan tabel jadwal penelitian dari awal penelitian hingga penelitian selesai yang dapat dilihat pada Tabel 1.1.

Tabel 1.1 Jadwal Penelitian

NO	Waktu Ket.	2023								2024															
		November				Desember				Januari				Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Rumusan Masalah																								
2	Pengumpulan Data																								
3	Analisis Sistem																								
4	Perancangan Sistem																								
5	Uji Coba Sistem																								
6	Penulisan Laporan																								

2. Bahan dan Alat

Berikut ini akan dijelaskan bahan dan alat yang digunakan yaitu berupa *hardware* dan *software* yang digunakan dalam proses penelitian.

Hardware yang digunakan yaitu sebuah laptop dengan spesifikasi:

- Processor Intel® Core™ i3-7020U CPU @ 2,30GHz
- RAM 8GB.
- Perangkat *keyboard*.
- Perangkat *touchpad*.
- Display* dengan resolusi 1920 x 1080 pixel.

Software yang digunakan yaitu:

- Sistem Operasi Windows 10 64 Bit.
- Sublime Text 3.
- Balsamiq Mockup 3 untuk rancangan prototipe.
- XAMPP Control Panel sebagai *software server* lokal.

3. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan pada penelitian ini antara lain sebagai berikut:

a. Metode Observasi

Melakukan observasi langsung ke sekolah Dharma Bakti untuk mengumpulkan data-data soal ujian dan jawaban ujian peserta ujian yang akan digunakan proses *training* dan *testing*.

b. Metode Eksperimen

Melakukan eksperimen pada aplikasi penilaian ujian *essay* secara otomatis untuk mengetahui kebutuhan-kebutuhan yang diperlukan pada aplikasi yang akan dibangun.

c. Metode Studi Pustaka

Mengumpulkan data-data teori melalui jurnal, media cetak, ataupun sumber-sumber referensi dari internet.

1.6 Analisis Sistem

Analisis sistem pada penelitian ini terbagi menjadi 3 tahapan proses yaitu:

1. Analisis sistem berjalan yaitu melakukan analisis sistem berjalan yang digunakan pada saat ini, khususnya sistem berjalan yang digunakan pengajar dalam melakukan koreksi nilai ujian.
2. Analisis algoritma yang digunakan yaitu algoritma *Similarity Text* dan Stemmer Sastwari dengan menguraikan cara kerja algoritma tersebut dalam melakukan penilaian ujian *essay*.

3. Analisis sistem usulan yaitu menggambarkan sistem usulan yang akan dibangun beserta fitur-fitur yang tersedia yang dimodelkan dengan menggunakan *tools* pemodelan sistem *Use Case Diagram*.

1.7 Perancangan Sistem

Perancangan sistem pada penelitian ini terbagi menjadi 2 tahapan yaitu:

1. Perancangan *prototype* tampilan dengan menggunakan *software* Balsamiq Mockup 3.
2. Perancangan basis data yang menunjukkan relasi antar tabel dari setiap basis data yang dimodelkan dengan menggunakan *tools* *Entity Relationship Diagram* (ERD).

1.8 Sistematika Penulisan

Untuk mempermudah memahami permasalahan yang dibahas, maka skripsi ini akan disusun dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Bab I menjelaskan tentang latar belakang, identifikasi masalah, batasan masalah, tujuan dan manfaat penelitian, metode penelitian serta sistematika penulisan.

BAB II LANDASAN TEORI

Bab II membahas mengenai teori-teori yang berhubungan dengan judul penulisan skripsi.

BAB III METODE PENELITIAN

Bab III membahas tentang analisis dan perancangan. Analisis ini akan menguraikan tentang analisis sistem berjalan, analisis metode yang diusulkan, dan analisis sistem usulan.

BAB IV HASIL DAN PEMBAHASAN

Bab IV merupakan tahapan yang penulis lakukan dalam mengembangkan sistem, desain, hasil *testing* dan implementasinya. Dari hasil yang didapat akan dilakukan pembahasan yang berisikan tentang hasil penelitian yang didapatkan.

BAB V KESIMPULAN DAN SARAN

Bab V berisikan mengenai kesimpulan yang didapatkan setelah melakukan penelitian dengan proses analisa terhadap teori dan hasil pengujian. Dan juga terdapat saran yang berisikan hal-hal yang belum berhasil dan masih mengalami masalah berdasarkan hasil pengujian yang dilakukan.

BAB II

TINJAUAN PUSTAKA

2.1 Kecerdasan Buatan

Kecerdasan buatan atau yang sering dikenal dengan *Artificial Intelligence* (AI) merupakan kemampuan mesin atau sistem komputer untuk meniru atau menunjukkan kecerdasan manusia. Definisi ini melibatkan kemampuan sistem untuk mengumpulkan informasi, memahami konteks, melakukan analisis, membuat keputusan, dan belajar dari pengalaman untuk menghadapi tugas-tugas yang kompleks. Tujuan utama AI adalah untuk mengembangkan sistem yang dapat melakukan tugas-tugas yang biasanya memerlukan kecerdasan manusia. Beberapa tujuan utama AI meliputi [6]:

1. Pengolahan Bahasa Alami (*Natural Language Processing*).

AI bertujuan untuk mengembangkan sistem yang mampu memahami, menghasilkan, dan berinteraksi dengan bahasa manusia secara alami. Hal ini melibatkan kemampuan sistem untuk memahami konteks, sintaksis, semantik, dan pragmatik dalam bahasa manusia.

2. Pembelajaran Mesin (*Machine Learning*).

AI bertujuan untuk mengembangkan algoritma dan teknik yang memungkinkan mesin untuk belajar dari data dan pengalaman, tanpa diprogram secara eksplisit. Pembelajaran mesin melibatkan identifikasi pola, membuat prediksi, dan mengoptimalkan kinerja sistem berdasarkan data yang diberikan.

3. Pengenalan Pola (*Pattern Recognition*).

AI bertujuan untuk mengembangkan sistem yang dapat mengenali pola dan fitur dalam data, seperti pengenalan wajah, pengenalan suara, dan pengenalan tulisan tangan. Ini melibatkan penggunaan algoritma dan teknik untuk mengidentifikasi dan mengklasifikasikan pola-pola yang ada dalam data.

4. Penalaran dan Pengambilan Keputusan (*Reasoning and Decision Making*).

AI bertujuan untuk mengembangkan sistem yang dapat menganalisis informasi, melakukan penalaran, dan membuat keputusan berdasarkan pemahaman konteks dan aturan yang ditentukan. Ini melibatkan penggunaan logika formal, inferensi, dan pemodelan pengetahuan untuk memungkinkan sistem AI untuk mengambil keputusan yang masuk akal.

Komponen utama dalam sistem AI meliputi [6]:

1. Basis Pengetahuan (*Knowledge Base*).

Tempat penyimpanan informasi dan pengetahuan yang digunakan oleh sistem AI. Basis pengetahuan dapat berisi fakta, aturan, konsep, dan model yang relevan dengan domain atau tugas yang sedang dihadapi.

2. Mesin Inferensi (*Inference Engine*).

Mesin inferensi bertanggung jawab untuk melakukan proses logika dan inferensi berdasarkan informasi yang ada dalam basis pengetahuan. Ini melibatkan penerapan aturan, penalaran, dan manipulasi pengetahuan untuk mencapai kesimpulan atau membuat keputusan.

3. Antarmuka Pengguna (*User Interface*).

Antarmuka pengguna memungkinkan interaksi antara pengguna manusia dan sistem AI. Ini dapat berupa antarmuka grafis, perintah suara, atau interaksi berbasis teks yang memungkinkan pengguna untuk memberikan input, menerima output, dan berkomunikasi dengan sistem AI.

4. Algoritma dan Teknik Kecerdasan Buatan (*AI Algorithms and Techniques*).

Ada berbagai algoritma dan teknik yang digunakan dalam AI, termasuk pembelajaran mesin, jaringan saraf tiruan, logika proposisional, pemrosesan bahasa alami, dan banyak lagi. Algoritma ini memungkinkan sistem AI untuk memproses data, mengenali pola, belajar dari pengalaman, dan membuat keputusan.

Ada beberapa jenis AI yang digunakan dalam berbagai konteks dan aplikasi.

Dalam hal ini, kita akan menjelajahi beberapa jenis utama AI, termasuk sistem berbasis aturan, sistem pakar, pembelajaran mesin, jaringan saraf tiruan, dan AI yang terkait dengan robotika. Berikut adalah beberapa jenis AI yang umum [6]:

1. Sistem Berbasis Aturan.

Jenis AI ini menggunakan aturan-aturan yang ditentukan sebelumnya untuk menghasilkan output berdasarkan *input* yang diberikan. Sistem berbasis aturan terdiri dari himpunan aturan dan basis pengetahuan yang menggambarkan hubungan antara *input* dan *output*. Sistem ini bekerja dengan mencocokkan *input* dengan aturan yang sesuai untuk menghasilkan *output* yang diinginkan.

2. Sistem Pakar.

Sistem pakar adalah jenis AI yang menggabungkan pengetahuan ahli manusia dengan kemampuan komputasi untuk memecahkan masalah yang kompleks dalam domain tertentu. Sistem ini beroperasi dengan mengumpulkan pengetahuan dari para ahli manusia dan memanfaatkannya dalam membuat keputusan atau memberikan nasihat yang sesuai. Sistem pakar digunakan dalam bidang-bidang seperti kedokteran, keuangan, dan teknik.

3. Pembelajaran Mesin.

Pembelajaran mesin adalah cabang AI yang melibatkan pengembangan algoritma dan model statistik yang memungkinkan sistem untuk belajar dari data dan pengalaman. Dalam pembelajaran mesin, sistem menganalisis data untuk mengidentifikasi pola dan membuat prediksi atau keputusan berdasarkan informasi yang diberikan. Algoritma pembelajaran mesin dapat diklasifikasikan menjadi beberapa kategori, termasuk pembelajaran terawasi, pembelajaran tak terawasi, dan pembelajaran penguatan.

4. Jaringan Saraf Tiruan.

Jaringan saraf tiruan (*artificial neural networks*) adalah jenis AI yang terinspirasi oleh struktur dan fungsi jaringan saraf dalam otak manusia. Jaringan saraf tiruan terdiri dari banyak unit pemrosesan sederhana yang disebut neuron buatan, yang saling terhubung dan berinteraksi untuk memproses informasi. Jaringan saraf tiruan dapat digunakan untuk tugas-tugas seperti pengenalan pola, pengenalan wajah, dan pemrosesan bahasa alami.

5. AI dalam Robotika.

AI juga digunakan dalam pengembangan robotika, di mana sistem AI dikombinasikan dengan fisik robot untuk menghasilkan perilaku yang cerdas dan adaptif. Robotika AI melibatkan penggunaan algoritma pengolahan sensorik, perencanaan gerakan, dan pembelajaran untuk memungkinkan robot berinteraksi dengan lingkungan sekitarnya dan melakukan tugas-tugas tertentu.

2.2 *Natural Language Processing (NLP)*

Pengolahan bahasa alami atau dikenal dengan *Natural Language Processing (NLP)* adalah bidang AI yang berkaitan dengan pemahaman dan penghasilan bahasa manusia. Dengan bantuan teknik NLP, komputer dapat memahami dan menghasilkan teks secara alami, memungkinkan aplikasi seperti penerjemahan otomatis, *chatbot*, dan analisis sentimen [6]. NLP merupakan cabang penting dari kecerdasan artifisial yang biasanya berhubungan dengan interaksi antara mesin/ komputer dengan manusia menggunakan bahasa alami. Salah satu tujuan akhir NLP adalah untuk mendapatkan kecerdasan dari data atau konten tidak terstruktur yang diekspresikan dalam bahasa alami, seperti Bahasa Inggris, Bahasa Bengali, Bahasa Indonesia. Setiap bahasa memiliki struktur yang unik pada tata bahasa dan sintaksis, konvensi, serta teknik NLP bisa memungkinkan komputer untuk membaca teks, mendengar ucapan, menafsirkan, mengukur sentimen atau menambang pendapat (*opinion mining*), dan akhirnya menentukan bagian yang penting dalam sistem intelijen tersebut [7].

NLP digunakan dalam berbagai situasi untuk memecahkan berbagai jenis masalah, di antaranya [7]:

1. Mencari serangkaian karakter.

Pemrosesan otomatis bahasa alami memungkinkan untuk mengidentifikasi elemen tertentu dalam teks. Misalnya hal ini digunakan untuk mencari kemunculan kata atau lebih umumnya pada serangkaian karakter dalam dokumen.

2. Pengenalan entitas.

Misalnya untuk mendapatkan nama tempat, orang, organisasi dan produk dari teks.

3. Analisis sentimen.

Teknik ini digunakan untuk menentukan perasaan dan sikap orang terhadap suatu produk atau layanan. Hal ini berguna untuk memberikan umpan balik tentang caranya produk atau jasa yang telah dirasakan oleh pengguna/konsumen. Perusahaan di semua sektor industri dapat menganalisis data yang bersumber dari jejaring sosial untuk lebih memahami pendapat pelanggan. Tantangan utamanya adalah mengekstrak ulasan dalam bentuk teks dari konsumen dan menggunakannya secara otomatis untuk mengevaluasi produk atau merek terbaik.

4. Mesin pencari.

Pemanfaatan NLP secara ekstensif untuk berbagai tugas, seperti *query understanding*, *query expansion*, *question answering*, pencarian informasi (*information retrieval*), pemeringkatan, dan pengelompokan hasil (*clustering of results*).

5. Pesan elektronik.

Platform *email* menggunakan NLP untuk menyediakan serangkaian fitur, seperti melakukan klasifikasi *spam*, prioritas kotak masuk, dan pelengkap otomatis.

Sejak awal tahun 2000, analisis sentimen telah tumbuh menjadi salah satu bidang penelitian paling aktif dalam NLP. Hal ini juga dipelajari secara luas dalam penambangan data (*data mining*), penambangan web (*web mining*), penambangan teks (*text mining*), dan pengambilan informasi (*information retrieval*). Pada penelitian ini fokus analisis sentimen yang dilakukan adalah melalui *text mining* [7].

2.3 *Text Mining*

Text mining dapat didefinisikan secara luas sebagai proses intensif pengetahuan di mana pengguna berinteraksi dengan kumpulan dokumen dari waktu ke waktu dengan menggunakan seperangkat alat analisis. Penambangan teks berusaha untuk mengekstrak informasi yang berguna dari sumber data melalui identifikasi dan eksplorasi pola yang menarik. Dalam kasus penambangan teks, bagaimanapun, sumber data adalah kumpulan dokumen, dan pola yang menarik ditemukan bukan di antara catatan database yang diformalkan tetapi dalam data tekstual yang tidak terstruktur dalam koleksi dokumen. Oleh karena itu penambangan teks dan sistem penambangan data menunjukkan banyak kesamaan arsitektur tingkat tinggi. Contohnya, kedua jenis sistem ini bergantung pada rutinitas *preprocessing*, algoritma penemuan pola, dan elemen lapisan presentasi seperti alat visualisasi untuk meningkatkan penelusuran set jawaban [8].

Text mining sendiri memiliki beberapa tipe antara lain [8]:

1. *Search and Information Retrieval.*

Menyimpan dan menemukan kembali dokumen teks, termasuk mesin pencari dan kata kunci pencarian.

2. *Document Clustering.*

Pengelompokan dan pengkategorian istilah, potongan, paragraf, atau dokumen menggunakan metode mining.

3. *Document Classification.*

Pengelompokan dan pengkategorian istilah, potongan, paragraf, atau dokumen menggunakan metode *document classification*.

4. *Web Mining Data.*

Teknik ini diimplementasikan pada internet yang fokus pada skala dan antar hubungan pada *website*.

5. *Information Extraction.*

Identifikasi dan ekstraksi fakta yang relevan.

6. *Natural Language Processing.*

Pemrosesan bahasa tingkat rendah yang biasanya digunakan untuk bahasa komputasi.

7. *Concept Extraction.*

Pengelompokan kata dan frase dalam grup yang sama tahap *preprocessing text*.

Berikut adalah implementasi *text mining* pada kasus nyata [9]:

1. Klasifikasi *email* untuk menentukan email spam atau bukan sehingga digunakan oleh pemilik layanan email untuk memasukkan email yang

bukan spam ke folder inbox dan email spam dikirim ke folder spam atau junk.

2. Mengetahui sentimen terhadap tokoh publik dengan mengklasifikasikan komentar masyarakat di media sosial. Dengan pengetahuan ini dapat diimplementasikan dalam sebuah aplikasi yang memungkinkan penggunaanya mengetik nama tokoh publik, kemudian aplikasi akan menarik data dari media sosial dengan kata kunci nama tokoh tersebut. Selanjutnya aplikasi akan otomatis untuk menampilkan berapa persen sentimen positif dan negatif.
3. Aplikasi pemantauan keadaan fasilitas publik dengan mengklasifikasikan komentar penduduk suatu kota dari media sosial. Aplikasi secara otomatis menampilkan fasilitas publik yang rusak atau yang perlu perhatian.
4. Mengetahui kinerja pelayanan publik suatu lembaga pemerintahan dengan melakukan klasifikasi data komentar masyarakat baik dari survey atau media sosial. Dengan pengetahuan ini dapat dibuat aplikasi infografik yang menampilkan kinerja lembaga-lembaga pemerintahan berdasarkan komentar masyarakat tersebut.
5. Mengetahui sentimen terhadap produk atau jasa dengan mengklasifikasikan komentar netizen di media sosial. Sebagai implementasinya dapat dibuat infografik seperti aplikasi yang telah disebutkan pada kasus tokoh publik dan pelayanan publik di atas.
6. Aplikasi pemantauan keadaan saat terjadi bencana alam dengan mengklasifikasikan komentar masyarakat untuk membedakan pesan bencana yang ditulis oleh masyarakat yang merasakan bencana alam, pesan

bencana yang ditulis masyarakat namun bukan saksi mata langsung, dan pesan yang tidak berhubungan dengan bencana. Dengan membedakan ketiga kategori ini maka aplikasi dapat menentukan lokasi bencana dengan menggunakan geo lokasi dari komentar yang ditulis oleh masyarakat yang merasakan bencana alam tersebut.

2.4 Algoritma Nazief dan Adriani

Algoritma Nazief dan Adriani merupakan algoritma *stemming* yang populer untuk memproses teks berbahasa Indonesia. Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani memiliki tahapan sebagai berikut [10]:

1. Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tersebut adalah kata dasar. Maka algoritma berhenti.
2. Infleksi akhiran (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa partikel (“- lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus kata ganti posesif (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus penurunan akhiran (“-i”, “-an” atau “- kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a.
 - a. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
 - b. Akhiran yang dihapus (“-i”, “-an” atau “- kan”) dikembalikan, lanjut ke langkah 4.

4. Hapus penurunan awalan. Jika pada langkah 3 ada akhiran yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
 - a. Periksa tabel kombinasi awalan akhiran yang tidak diizinkan. Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
 - b. Pada langkah ini dilakukan perulangan sebanyak tiga kali. Tentukan tipe awalan kemudian hapus awalan. Jika kata dasar belum juga ditemukan lakukan langkah 5, jika sudah maka algoritma berhenti.
Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.
5. Melakukan *Recoding*.
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai kata dasar lalu proses diakhiri.

2.5 *Stemmer Sastrawi*

Stemmer Sastrawi merupakan sebuah *library* dari bahasa pemrograman PHP yang dibangun dengan berdasarkan pada algoritma Nazief dan Adriani (NA). Algoritma NA adalah aturan yang mengikuti pada aturan Bahasa Indonesia. Aturan tersebut adalah penentuan imbuhan yang di perbolehkan atau tidak. Imbuhan dikelompokkan menjadi imbuhan di awal kata, imbuhan di tengah kata, imbuhan di akhir kata, dan kombinasi imbuhan pada akhir dan awal kata [11].

2.6 *Algoritma Similarity Text*

Algoritma *Similarity Text* merupakan salah satu fungsi PHP yang merupakan algoritma sederhana untuk menghitung kesamaan antara dua *string*. *String* dimaksud dapat berupa angka, karakter, huruf atau kata yang dikelompokkan

secara Bersama maupun terpisah sebagai *string* dengan variabel panjang. Dalam prakteknya, *similar text* banyak digunakan di berbagai situs *web* untuk mencari keunikan teks, menghitung jumlah duplikasi hingga mengidentifikasi konten duplikat [12].

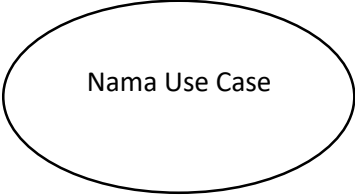
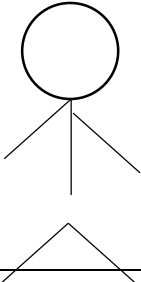
Berikut ini, akan diuraikan tahapan dari algoritma *Similarity Text* antara lain [13]:

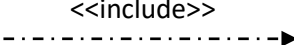
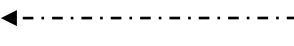
1. Menentukan 2 *string* yang akan diuji tingkat *similar text*-nya (misal Str_1 dan Str_2).
2. Menghitung panjang *string* pertama dan kedua, termasuk spasi (misal L_1 dan L_2).
3. Menghitung panjang *string* total dari *string* pertama dan kedua (misal $T_{str} = L_1 + L_2$).
4. Melakukan:
 - a. Perhitungan kesamaan karakter yang ada di *string* kedua dengan karakter di *string* pertama.
 - b. Simpan jumlah karakter yang sama tersebut (misal T_{kar}).
5. Menghitung nilai rata-rata panjang *string*, yaitu panjang *string* total 2 *string* dibagi dengan 2 (misal $P_{str} = T_{str}/2$).
6. Menghitung persentase *similar text* dengan cara membagi antara jumlah karakter yang sama (T_{kar}) dengan rata-rata panjang *string* (P_{str}) dikali dengan 100% (misal $S_{text} = (T_{kar}/P_{str}) * 100\%$).

2.7 Use Case Diagram

Diagram *Use Case* adalah salah satu alat pemodelan sistem yang merupakan komponen dari *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara pengguna dan sistem. Penggunaan Diagram *Use Case* dapat memberikan gambaran yang efektif dalam menjelaskan konteks suatu sistem dengan jelas, sehingga memperlihatkan batasan sistem tersebut dengan jelas. Terdapat beberapa simbol-simbol dari *Use Case Diagram* seperti yang ditunjukkan pada Tabel 2.1 berikut [14].

Tabel 2.1 Simbol Use Case Diagram [14]

Nama	Simbol	Deskripsi
<i>Use Case</i>		Simbol yang menggambarkan fungsi dari sistem umumnya diekspresikan dalam bentuk kata kerja di awal setiap frasa.
Aktor		Simbol yang mewakili pengguna yang terlibat dalam <i>Use Case</i> biasanya menyatakan orang-orang yang berinteraksi dengan sistem.
Asosiasi		Simbol yang menghubungkan antara <i>Use Case</i> dan aktor menggambarkan komunikasi antara keduanya. Asosiasi digunakan untuk menggambarkan hubungan statis antara elemen-elemen tersebut.

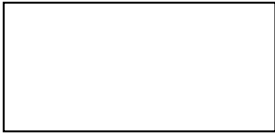
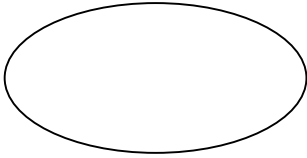
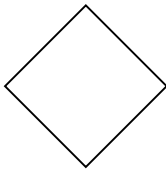
<i>Include</i>		Simbol ini menggambarkan bahwa <i>Use Case</i> secara keseluruhan merupakan bagian dari fungsionalitas <i>Use Case</i> lainnya.
<i>Extend</i>		Simbol yang serupa dengan " <i>include</i> ", namun menunjukkan bahwa sebuah <i>Use Case</i> tambahan dapat berdiri sendiri tanpa membutuhkan <i>Use Case</i> yang ditambahkan.
Sistem		Simbol yang membatasi bahwa setiap interaksi dengan simbol lain harus berada di dalamnya.

2.8 Entity Relationship Diagram

Entity Relationship Diagram merupakan teknik pendekatan *top-bottom* untuk desain *database*, yang dilakukan dengan proses berupa identifikasi informasi penting kemudian menggambarkannya dalam sebuah model. *Entity Relationship Diagram* adalah model yang berguna membantu dalam memahami data dan bagaimana data tersebut digunakan dalam sebuah perusahaan. Terdapat beberapa simbol-simbol dari *Use Case Diagram* seperti yang ditunjukkan pada Tabel 2.2 berikut [14].

Tabel 2.2 Simbol *Entity Relationship Diagram* [14]

Nama	Simbol	Deskripsi
------	--------	-----------

Entitas		Kumpulan dari objek yang dapat diidentifikasi secara unik biasanya merupakan nama dari tabel di dalam rancangan basis data.
Atribut		Gambaran karakteristik dari sebuah entitas yang biasanya dianggap sebagai nama kolom di dalam rancangan basis data.
Hubungan		Menyatakan jumlah himpunan relasi antar entitas.
Garis		Penghubung entitas, atribut dan hubungan.

2.9 Penelitian Terdahulu

Terdapat penelitian sebelumnya yang berkaitan dengan topik pada penelitian ini yaitu seperti penelitian pada tahun 2021 membahas mengenai penilaian *essay* pendek otomatis dengan pencocokan kata dengan menggunakan teknik *Natural Language Processing* (NLP) dan frasa nomina. Hasil penelitian menunjukkan bahwa implementasi kedua teknik ini mendapatkan akurasi dan nilai uji sangat baik dengan nilai *Pearson Correlation* sebesar 0,83 [4]. Penelitian terbaru pada tahun 2022 membahas perancangan aplikasi koreksi ujian *essay* otomatis dengan menggunakan metode *Cosine Similarity*. Hasil penelitian menunjukkan bahwa aplikasi yang dibangun dapat mempersingkat waktu pengajar dalam mengoreksi jawaban peserta ujian [5].

BAB III

METODE PENELITIAN

3.1 Analisis Masalah

Tahapan analisis masalah akan dianalisis secara mendalam masalah yang menjadi fokus pada penelitian ini. Analisis ini bertujuan untuk memahami akar permasalahan dengan lebih baik, serta untuk mengidentifikasi tantangan dan hambatan yang perlu diatasi dalam rangka mencapai solusi yang optimal. Dengan memahami dengan baik konteks dan kompleksitas masalah yang dihadapi, diharapkan dapat mengusulkan solusi yang tepat dan efektif dalam penyelesaiannya. Analisis masalah pada penelitian ini dibagi menjadi 3 yaitu analisis proses berjalan sesuai di lapangan, analisis algoritma yang diusulkan, dan analisis sistem usulan.

3.1.1 Analisis Proses Yang Berjalan Sesuai di Lapangan

Pada sub bab ini dilakukan analisis terhadap proses/sistem berjalan yang digunakan sekolah Dharma Bakti Lubuk Pakam khususnya sistem berjalan yang digunakan dalam melakukan proses koreksi ujian *essay*. Berikut ini uraikan sistem berjalannya antara lain:

1. Sekolah Dharma Bakti Lubuk Pakam, seperti sekolah pada umumnya, aktif dalam kegiatan belajar mengajar mulai dari tingkat SD hingga SMA. Salah satu aspek penting dari kegiatan ini adalah proses ujian, yang melibatkan berbagai jenis soal, termasuk soal pilihan berganda dan *essay*. Soal pilihan

berganda relatif lebih mudah untuk dikoreksi, tetapi soal *essay* seringkali menjadi tantangan karena memerlukan waktu dan tenaga ekstra dari para pengajar. Cara koreksi masih dilakukan secara konvensional oleh guru dengan membaca satu per satu jawaban dari siswa dan kemudian memberikan penilaian.

2. Pengajar di sekolah Dharma Bakti Lubuk Pakam sering kali mengalami kesulitan dalam menilai jawaban *essay* karena kompleksitasnya dikarenakan masih memanfaatkan cara manual. Selain itu, pengajar juga memiliki tanggung jawab administratif lainnya yang memerlukan waktu yang tidak sedikit. Hal ini menyebabkan kelelahan dan potensi *human error* dalam proses koreksi jawaban ujian.

Untuk mengatasi masalah ini, penelitian sebelumnya telah mengusulkan solusi berupa aplikasi penilaian jawaban *essay* secara otomatis. Namun, fokus penelitian sebelumnya lebih pada algoritma penilaian daripada proses *preprocessing* teks jawaban. Oleh karena itu, penelitian ini akan mengimplementasikan teknik berbasis *text mining*, seperti algoritma *Similarity Text* dan *Stemmer Sastrawi*, untuk meningkatkan akurasi dalam penilaian jawaban *essay*.

3.1.2 Analisis Algoritma Yang Diusulkan

Analisis algoritma yang diusulkan bertujuan untuk menguraikan cara kerja dari algoritma yang diusulkan dalam menyelesaikan permasalahan pada penelitian ini. Pada penelitian ini diusulkan 2 buah algoritma yaitu algoritma *Similarity Text* dan optimasi dengan *Stemmer Sastrawi* dalam melakukan penilaian ujian *essay*

secara otomatis. Berikut ini adalah analisis cara kerja dari algoritma yang diusulkan dalam bentuk contoh kasus sederhana.

Asumsikan terdapat 1 buah soal yang sudah memiliki kunci jawaban serta 1 buah jawaban dari siswa. Berikut ini, Tabel 3.1 menunjukkan contoh soal, kunci jawaban, dan jawaban dari siswa.

Tabel 3.1 Contoh Soal, Kunci Jawaban, dan Jawaban Siswa

Soal	Kunci Jawaban	Jawaban Siswa
Apa itu puisi?	Puisi adalah sastra yang ditulis dalam bentuk puisi dengan memperhatikan irama, rima, dan gaya bahasa tertentu untuk menyampaikan gagasan atau perasaan secara estetik.	Puisi adalah sastra dalam bentuk puisi yang menggunakan gaya bahasa khusus untuk menyampaikan ide dan perasaan secara estetik.

Tahap pertama adalah melakukan optimasi teks terlebih dahulu dengan algoritma *Stemmer Sastrawi* sehingga teks menjadi seperti berikut:

1. *String* Pertama (Kunci Jawaban):

"Puisi adalah sastra yang ditulis dalam bentuk puisi dengan memperhatikan irama, rima, dan gaya bahasa tertentu untuk menyampaikan gagasan atau perasaan secara estetik."

Setelah *stemming*: "puisi adalah sastra yang tulis dalam bentuk puisi dengan perhati irama, rima, dan gaya bahasa khusus untuk sampa gagas atau rasa secara estetik."

2. *String* Kedua (Jawaban Siswa):

"Puisi adalah sastra dalam bentuk puisi yang menggunakan gaya bahasa khusus untuk menyampaikan ide dan perasaan secara estetik."

Setelah *stemming*: "puisi adalah sastra dalam bentuk puisi yang guna gaya bahasa khusus untuk sampa ide dan rasa secara estetis."

Dengan langkah ini, telah berhasil dilakukan proses optimasi teks dengan *stemming* pada kedua *string* menggunakan algoritma *Stemmer Sastrawi*, sehingga didapatkan kata-kata dasar dari setiap kata dalam teks. Kemudian dapat dilanjutkan ke langkah-langkah berikutnya dalam perhitungan persentase kemiripan teks dengan algoritma *Similarity Text*. Berikut ini uraian cara kerjanya antara lain:

1. Menentukan 2 *string* yang akan diuji tingkat *similar text*-nya:
 - a. Str1 (jawaban siswa setelah *stemming*): "puisi adalah sastra dalam bentuk puisi yang guna gaya bahasa khusus untuk sampa ide dan rasa secara estetis."
 - b. Str2 (kunci jawaban setelah *stemming*): "puisi adalah sastra yang tulis dalam bentuk puisi dengan perhati irama, rima, dan gaya bahasa khusus untuk sampa gagasan atau rasa secara estetis."
2. Menghitung panjang *string* pertama dan kedua, termasuk spasi:
 - a. Panjang *string* pertama (L1): 73 karakter.
 - b. Panjang *string* kedua (L2): 100 karakter.
3. Menghitung panjang *string* total dari *string* pertama dan kedua:
 Total panjang string (Tstr) = $L1 + L2 = 73 + 100 = 173$.
4. Melakukan perhitungan kesamaan karakter:
 Perhitungan kesamaan karakter yang ada di *string* kedua dengan karakter di *string* pertama:

Dalam kedua string ini, terdapat 17 kata yang sama setelah stemming: "puisi", "adalah", "sastra", "dalam", "bentuk", "puisi", "yang", "guna", "gaya", "bahasa", "khusus", "untuk", "sampai", "ide", "dan", "rasa", "secara", "estetis".

Jumlah kata yang sama (T_{kar}) = 17.

5. Menghitung nilai rata-rata panjang *string*:

Panjang *string* total dibagi 2 (P_{str}) = $T_{str}/2 = 173/2 = 86,5$.

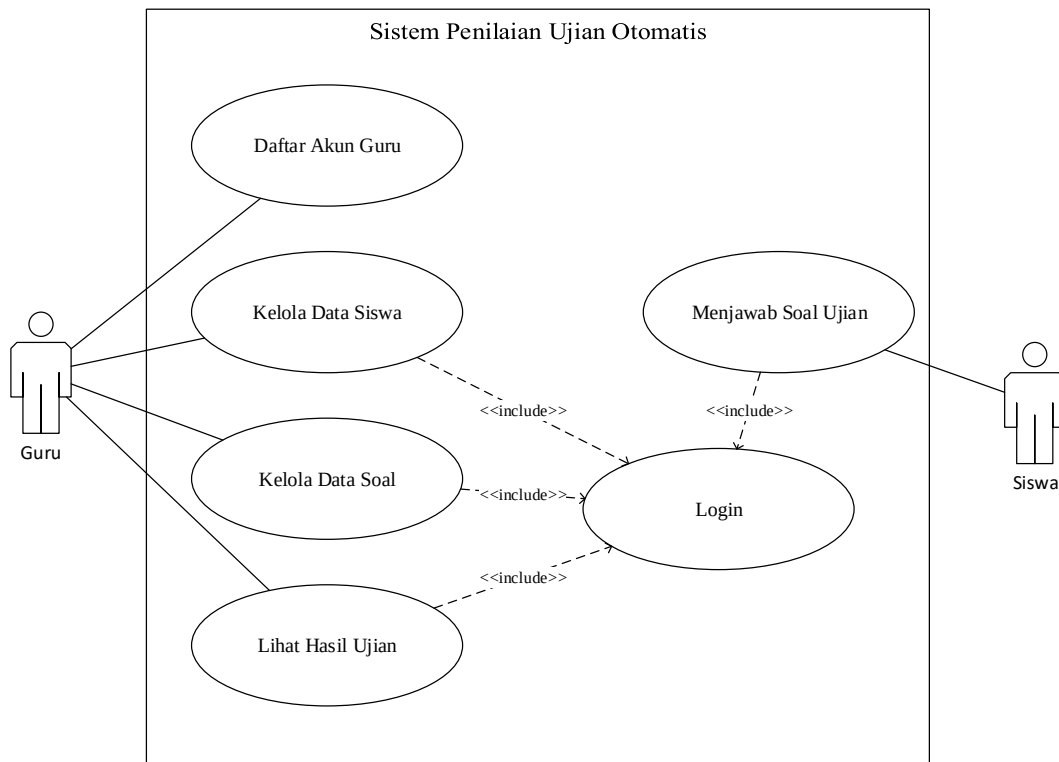
6. Menghitung persentase kemiripan:

Persentase kemiripan = $(T_{kar}/P_{str}) * 100\% = (17/86,5) * 100\% \approx 19,65\%$.

Dengan demikian, hasil persentase kemiripan antara jawaban siswa dan kunci jawaban setelah proses *stemming* dengan algoritma *Sastrawi* dan pengecekan kemiripan dengan algoritma *Similarity Text* adalah sekitar 19,65%. Dimana nantinya pengecekan nilai adalah seluruh total persentase dibagi dengan jumlah jawaban.

3.1.3 Analisis Sistem Usulan

Pada sub bab ini, dilakukan analisis mendalam terhadap sistem yang diusulkan sebagai solusi atas permasalahan yang telah diidentifikasi sebelumnya. Dalam proses analisis ini, kami akan menguraikan secara terperinci berbagai aspek kritis dari sistem usulan, khususnya pada fungsionalitas sistem serta fitur-fitur apa saja yang ada pada sistem usulan yang akan dikembangkan pada penelitian ini. Proses analisis sistem usulan pada penelitian ini menggunakan *Use Case Diagram*. Berikut ini, Gambar 3.1 menunjukkan *Use Case Diagram* dari sistem usulan yaitu sistem penilaian ujian otomatis berbasis *text mining*.



Gambar 3.1 Use Case Diagram Sistem Penilaian Ujian Otomatis

Tahapan berikutnya adalah menguraikan skenario dari setiap *Use Case* agar diketahui secara hasil analisis sistem usulan. Berikut ini, Tabel 3.2 hingga 3.7 menunjukkan skenario dari setiap *Use Case*.

Tabel 3.2 Skenario Use Case: Daftar Akun Guru

Daftar Akun Guru		Nama Use Case
Guru		Aktor
Melakukan pendaftaran akun guru agar dapat menggunakan fitur-fitur sistem		Deskripsi
Aksi Aktor	Respon Sistem	
1. Masuk ke halaman pendaftaran akun guru dengan menekan tombol daftar akun guru pada halaman akses masuk guru		Normal Flow Event
	2. Menampilkan halaman pendaftaran akun guru	
3. Mengisi data pada form pendaftaran		
4. Menekan tombol proses		

	5. Memanipulasi tabel “Guru”	
	6. Menampilkan pesan pendaftaran berhasil jika sesuai dan pesan pendaftaran gagal jika tidak sesuai dengan basis data	

Tabel 3.3 Skenario *Use Case*: Login

Login		Nama <i>Use Case</i>
Guru dan Siswa		Aktor
Melakukan <i>login</i> agar dapat menggunakan fitur-fitur yang terdapat pada aplikasi		Deskripsi
Aksi Aktor	Respon Sistem	
1. Masuk ke halaman masuk guru atau siswa		Normal Flow Event
	2. Menampilkan halaman masuk guru atau siswa	
3. Melakukan pengisian <i>form</i> masuk dan menekan tombol masuk		
	4. Sistem akan melakukan pengecekan basis data dan jika sesuai akan ditampilkan tampilan beranda aktor dan jika tidak sesuai akan ditampilkan pesan kesalahan kombinasi kata sandi tidak tepat	

Tabel 3.4 Skenario *Use Case*: Kelola Data Siswa

Kelola Data Siswa		Nama <i>Use Case</i>
Guru		Aktor
Melakukan pengelolaan data siswa yakni melihat, menambah, mengubah, dan menghapus data siswa		Deskripsi
Aksi Aktor	Respon Sistem	

1. Masuk ke halaman data siswa dengan menekan menu siswa yang terletak di <i>dropdown</i> proses data		<i>Normal Flow Event</i>
	2. Menampilkan halaman data siswa beserta <i>list</i> tabel data siswa yang pernah ditambahkan sebelumnya	
3. Memilih tombol yang diinginkan yaitu tombol tambah siswa untuk menampilkan <i>form</i> tambah data siswa, tombol ubah untuk menampilkan <i>form</i> ubah data siswa, dan tombol hapus untuk menghapus data siswa		
	4. Menampilkan halaman sesuai dengan tombol yang dipilih oleh aktor yang bersangkutan	

Tabel 3.5 Skenario *Use Case*: Kelola Data Soal

Kelola Data Soal		Nama <i>Use Case</i>
Guru		Aktor
Melakukan pengelolaan data soal ujian yakni melihat, menambah, mengubah, dan menghapus data soal ujian		Deskripsi
Aksi Aktor	Respon Sistem	
1. Masuk ke halaman data soal dengan menekan menu buat soal yang terletak di <i>dropdown</i> proses data		<i>Normal Flow Event</i>
	2. Menampilkan halaman data soal beserta <i>list</i> tabel data soal ujian yang pernah ditambahkan sebelumnya	

3. Memilih tombol yang diinginkan yaitu tombol tambah soal untuk menampilkan <i>form</i> tambah data soal, tombol ubah untuk menampilkan <i>form</i> ubah data soal, dan tombol hapus untuk menghapus data soal		
	4. Menampilkan halaman sesuai dengan tombol yang dipilih oleh aktor yang bersangkutan	

Tabel 3.6 Skenario *Use Case*: Lihat Hasil Ujian

Lihat Hasil Ujian		Nama <i>Use Case</i>
Guru		Aktor
Melihat hasil ujian yang telah dinilai secara otomatis dengan algoritma <i>Similarity Text</i> dan optimasi dengan <i>Stemmer Sastrawi</i>		Deskripsi
Aksi Aktor	Respon Sistem	
1. Masuk ke halaman cek hasil tes dengan menekan menu hasil tes yang terletak di <i>dropdown</i> proses data		Normal Flow Event
	2. Menampilkan halaman hasil ujian yang berisikan <i>list</i> data siswa yang melakukan ujian beserta nilai akhirnya	
3. Menekan tombol cek hasil		
	4. Menampilkan tabel hasil <i>text mining</i> dengan algoritma <i>Similarity Text</i> dan optimasi dengan <i>Stemmer Sastrawi</i> dalam melakukan penilaian ujian otomatis	
5. Menekan tombol hitung hasil akhir		

	6. Menampilkan nilai hasil akhir ujian	
--	--	--

Tabel 3.7 Skenario Use Case: Menjawab Soal Ujian

Menjawab Soal Ujian		Nama Use Case
Murid		Aktor
Menjawab soal ujian yang sedang diujikan		Deskripsi
Aksi Aktor	Respon Sistem	
1. Masuk ke halaman jawab soal ujian dengan menekan menu jawab soal yang terletak di <i>dropdown</i> proses data		Normal Flow Event
	2. Menampilkan halaman yang berisikan soal serta <i>form</i> untuk menjawab	
3. Menekan tombol jawab		
	4. Menampilkan <i>form</i> untuk menjawab soal ujian	
5. Memilih soal yang ingin dijawab serta mengisikan jawaban dan kemudian tekan tombol simpan		
	6. Menambahkan jawaban yang dijawab ke tabel hasil jawaban	

3.2 Perancangan

Dalam subbab ini, akan dibahas tentang perancangan sistem informasi penilaian ujian otomatis di sekolah Dharma Bakti Lubuk Pakam. Perancangan sistem ini merupakan langkah penting setelah analisis kebutuhan telah dilakukan. Pembahasan akan mencakup rancangan antarmuka pengguna serta basis data yang akan diimplementasikan dalam sistem baru. Dengan perancangan yang matang, diharapkan sistem informasi yang diusulkan dapat memberikan solusi yang efektif

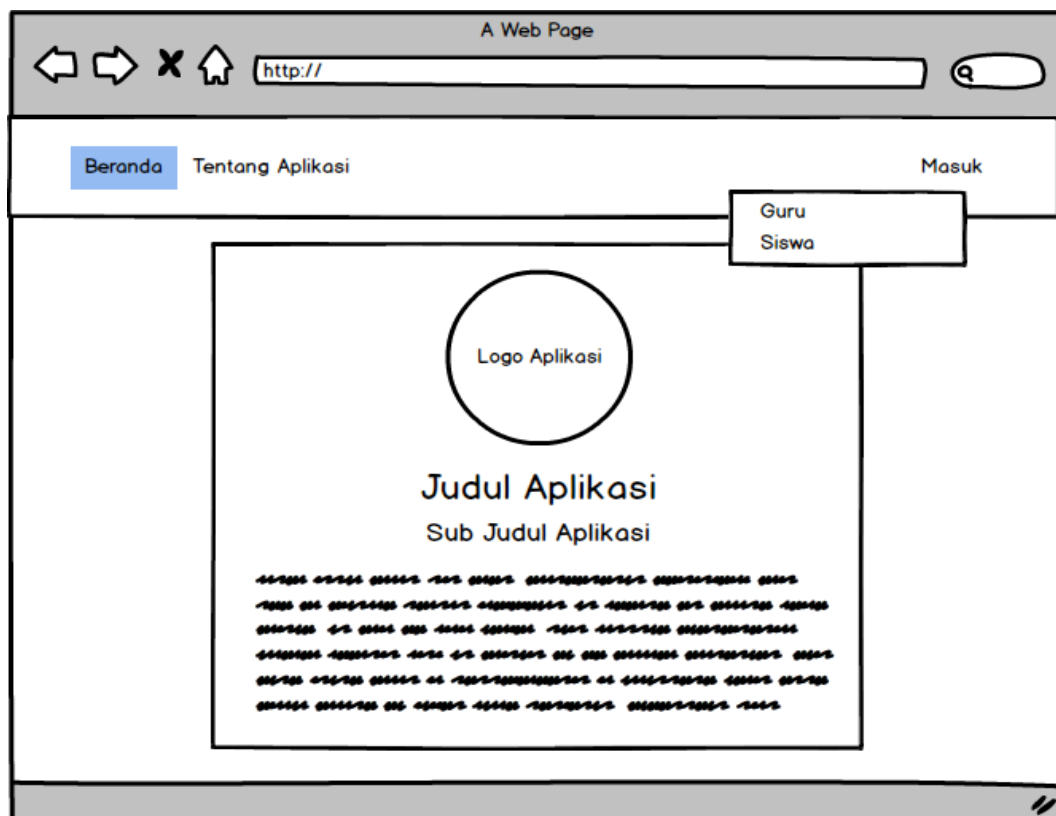
dan efisien untuk meningkatkan operasional sekolah khususnya terkait penilaian ujian otomatis.

3.2.1 Perancangan Tampilan

Perancangan tampilan digambarkan dengan menggunakan *software* perancangan sistem yaitu *Balsamiq Mockup 3*. User yang menggunakan sistem terdiri dari 2 yaitu guru dan siswa. Berikut ini akan diuraikan keseluruhan rancangan tampilan untuk *user* guru.

1. Rancangan Tampilan Awal

Pada bagian ini, akan dibahas mengenai rancangan tampilan awal dari sistem penilaian ujian otomatis. Tampilan awal memiliki peran yang penting karena menjadi titik awal interaksi pengguna dengan sistem.



Gambar 3.2 Rancangan Tampilan Awal

2. Rancangan Tampilan Daftar Akun Guru

Tampilan daftar akun guru akan mencakup formulir pendaftaran yang meminta informasi yang diperlukan, seperti NIP, nama, jenis kelamin, nomor telepon, alamat, *user name*, dan kata sandi. Tujuannya adalah menyediakan proses pendaftaran yang mudah dipahami dan diakses oleh para pengguna guru.

A Web Page

http://

Beranda Tentang Aplikasi

Masuk

Daftar Akun Guru

NIP

Nama Guru

Laki-Laki

Nomor Telepon

Alamat

Username

Password

Daftar

[Ingin Masuk Sebagai Guru?](#)

[Apakah Anda Ingin Kembali Ke Beranda?](#)

Gambar 3.3 Rancangan Tampilan Daftar Akun Guru

3. Rancangan Tampilan Masuk Guru

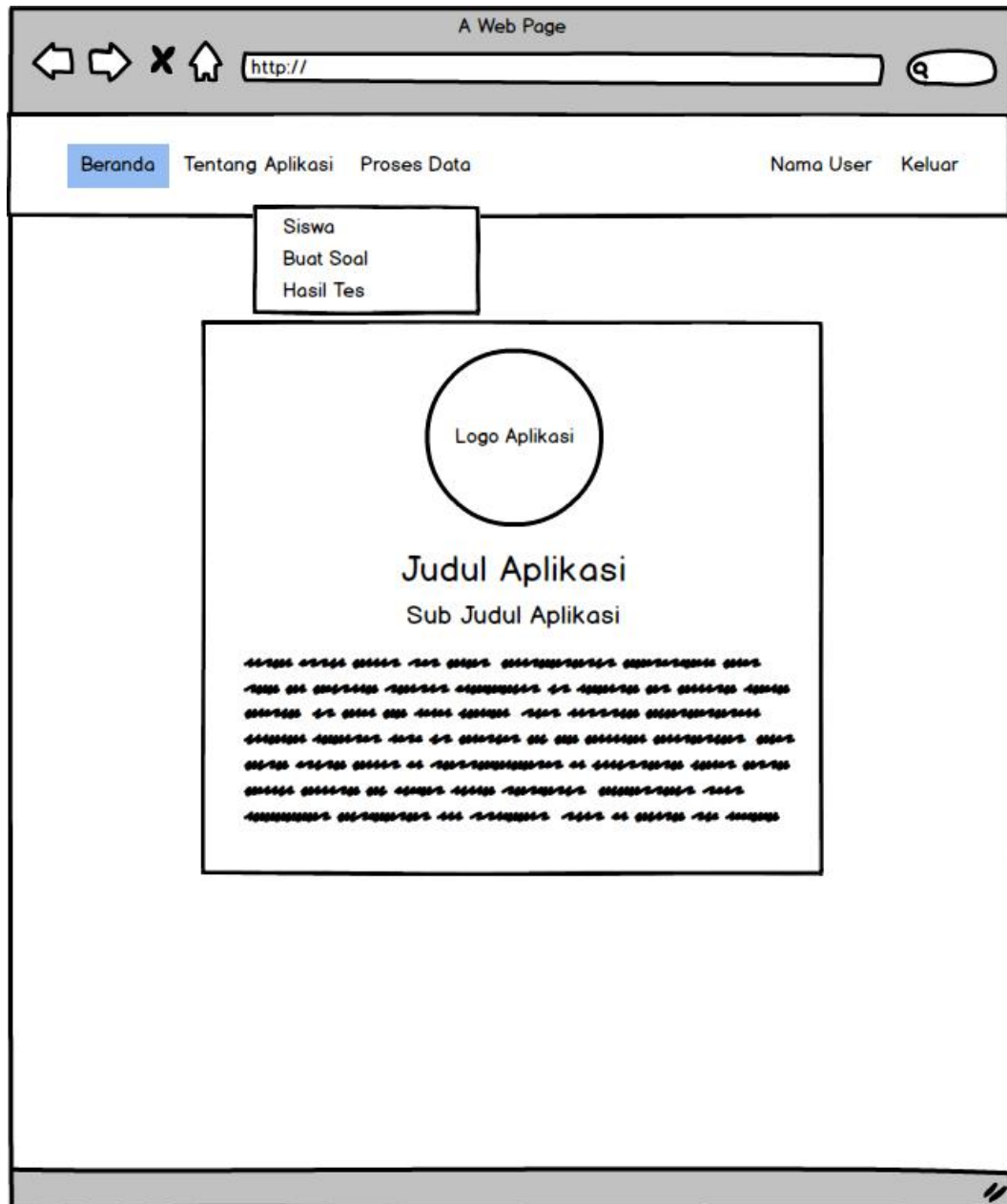
Tampilan masuk guru mencakup formulir masuk standar yang meminta pengguna untuk memasukkan informasi kredensial mereka, seperti kode *user* dan kata sandi. Tujuannya adalah untuk menyediakan pengalaman login yang lancar dan aman bagi para pengguna guru.

The image shows a wireframe of a web browser window. The browser's title bar reads "A Web Page". The address bar contains "http://". The page has a header with two links: "Beranda" and "Tentang Aplikasi", and a blue button labeled "Masuk". The main content area is a box titled "Akses Masuk Guru". Inside this box, there are three input fields: "Kode User", "Kata Sandi", and a button labeled "Masuk". Below the input fields, there are four links: "Daftar Akun Guru", "Bukan Guru?", "Ingin Masuk Sebagai Siswa?", and "Apakah Anda Ingin Kembali Ke Beranda?".

Gambar 3.4 Rancangan Tampilan Masuk Guru

4. Rancangan Tampilan Beranda Guru

Tampilan beranda guru dirancang dengan fokus pada kebutuhan dan aktivitas utama seorang guru dalam sistem. Rancangan ini mencakup ringkasan informasi penting seperti fitur-fitur penambahan data siswa, soal, dan cek hasil ujian.



Gambar 3.5 Rancangan Tampilan Beranda Guru

5. Rancangan Tampilan Kelola Data Siswa

Tampilan ini menyajikan berbagai fungsi untuk pengelolaan data siswa oleh guru. Rancangan ini mencakup fitur-fitur seperti menambah, menghapus, dan memperbarui informasi siswa. Selain itu, terdapat fitur pencarian dan *filter* untuk memudahkan guru dalam menemukan data siswa yang diinginkan.

The screenshot shows a web browser window titled "A Web Page" with a URL bar containing "http://". The application interface has a top navigation bar with links: "Beranda", "Tentang Aplikasi", and "Proses Data" (which is highlighted in blue). To the right of the navigation bar are the labels "Nama User" and "Keluar".

Below the navigation bar, there is a section for adding and managing data. It includes a button labeled "Tambah Siswa". Below this are three buttons for exporting data: "Excel", "PDF", and "Print". To the right of these buttons is a search bar labeled "Search:". Below the search bar is a table with the following columns: "No.", "NIS", "Nama", "Jurusan", "Peminatan", "Jenis Kelamin", "Username", "Action", and "Action". The table contains 4 rows of data, each with placeholder text "xxxxxxx". Each row has two buttons in the "Action" columns: "Ubah" and "Hapus".

At the bottom of the table, there is a status bar that says "Showing 1 to 4 of 4 entries". To the right of the status bar are three buttons: "Previous", "1", and "Next".

Gambar 3.6 Rancangan Tampilan Kelola Data Siswa

6. Rancangan Tampilan Tambah Data Siswa

Tampilan ini akan memberikan formulir yang memungkinkan guru untuk menambahkan data baru untuk siswa ke dalam sistem. Formulir akan mencakup kolom-kolom yang relevan untuk informasi siswa, seperti nama lengkap, jurusan, jenis kelamin dan informasi kontak lainnya. Tujuannya adalah memberikan pengalaman pengguna yang intuitif dan lancar bagi guru ketika menambahkan data siswa ke dalam sistem.

The screenshot shows a web application interface for adding student data. The interface includes a navigation bar with 'Beranda', 'Tentang Aplikasi', and 'Proses Data' (highlighted). A modal window titled 'Add Siswa' is open, showing a form with fields for NIS, Nama, Jurusan, Peminatan, Jenis Kelamin, Username, and Password. The form has 'Simpan' and 'Kembali' buttons at the bottom. In the background, a table with 4 entries is visible, and a sidebar on the right has 'Hapus' buttons for each row.

No.	NIS	Nama	Jurusan	Peminatan	Jenis Kelamin	Username	Password	Action
1	xxxxx							Hapus
2	xxxxx							Hapus
3	xxxxx		xxx					Hapus
4	xxxxx							Hapus

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.7 Rancangan Tampilan Tambah Data Siswa

7. Rancangan Tampilan Ubah Data Siswa

Tampilan ini menampilkan formulir yang memungkinkan guru untuk mengubah informasi yang terkait dengan seorang siswa dalam sistem. Formulir ini akan menampilkan data siswa yang sudah ada dan memungkinkan guru untuk memperbarui atau mengedit informasi tersebut sesuai kebutuhan. Tujuannya adalah memberikan kemudahan bagi guru dalam mengelola informasi siswa dengan mengubah atau memperbarui data yang relevan sesuai kebutuhan.

The screenshot shows a web application interface for managing student data. The main navigation bar includes 'Beranda', 'Tentang Aplikasi', and 'Proses Data' (highlighted). A table on the left displays a list of students with columns 'No.' and 'NIS'. A modal window titled 'Ubah Siswa' is open, allowing users to update student information. The form includes fields for NIS, Nama, Jurusan, Peminatan, Jenis Kelamin, Username, and Password (with a note: 'Diisi jika ingin mengubah password'). The form has 'Simpan' and 'Kembali' buttons. To the right of the form is an 'Action' column with 'Hapus' buttons for each student entry. The bottom of the interface shows 'Showing 1 to 4 of 4 entries' and pagination controls.

No.	NIS
1	xxxxx
2	xxxxx
3	xxxxx
4	xxxxx

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.8 Rancangan Tampilan Ubah Data Siswa

8. Rancangan Tampilan Hapus Data Siswa

Tampilan ini menampilkan konfirmasi untuk menghapus data seorang siswa dari sistem. Sebelum penghapusan dilakukan, akan ada pesan peringatan yang meminta konfirmasi dari pengguna, biasanya dengan pertanyaan apakah pengguna yakin ingin menghapus data siswa tersebut. Setelah konfirmasi diberikan, data siswa akan dihapus dari basis data.

The screenshot shows a web browser window titled "A Web Page" with a URL bar containing "http://". The page has a navigation bar with links: "Beranda", "Tentang Aplikasi", "Proses Data" (highlighted), "Nama User", and "Keluar". A confirmation dialog box is displayed in the center, asking "Yakin Ingin Menghapus Data ?" with "Oke" and "Batal" buttons. Below the dialog is a table with 4 rows of student data. Each row has a "No." column and columns for "NIS", "Nama", "Jurusan", "Peminatan", "Jenis Kelamin", and "Username". The "Action" column contains "Ubah" and "Hapus" buttons. The bottom of the page shows "Showing 1 to 4 of 4 entries" and pagination controls with "Previous", "1" (selected), and "Next" buttons.

No.	NIS	Nama	Jurusan	Peminatan	Jenis Kelamin	Username	Action	Action
1	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	Ubah	Hapus
2	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	Ubah	Hapus
3	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	Ubah	Hapus
4	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx	Ubah	Hapus

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.9 Rancangan Tampilan Hapus Data Siswa

9. Rancangan Tampilan Tambah Data Soal

Tampilan ini akan memberikan formulir yang memungkinkan guru untuk menambahkan data baru untuk soal ke dalam sistem. Formulir akan mencakup kolom-kolom yang relevan untuk informasi soal, seperti judul soal dan kunci jawaban. Tujuannya adalah agar soal-soal tersebut dapat diujikan ke siswa.

The screenshot shows a web application interface for managing question data. The interface includes a navigation bar with 'Beranda', 'Tentang Aplikasi', and 'Proses Data' (highlighted). Below the navigation bar, there is a 'Tambah Soal' button, a search bar, and three buttons: 'Excel', 'PDF', and 'Print'. The main content area features a table with 5 columns: 'No.', 'Soal', 'Kunci Jawaban', 'Action', and 'Action'. The table contains 4 rows of data, each with placeholder text 'xxxxxxx'. Each row has two 'Action' buttons: 'Ubah' and 'Hapus'. At the bottom, there is a pagination bar showing 'Showing 1 to 4 of 4 entries' and buttons for 'Previous', '1', and 'Next'.

No.	Soal	Kunci Jawaban	Action	Action
1	xxxxxxx	xxxxxxx	Ubah	Hapus
2	xxxxxxx	xxxxxxx	Ubah	Hapus
3	xxxxxxx	xxxxxxx	Ubah	Hapus
4	xxxxxxx	xxxxxxx	Ubah	Hapus

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.10 Rancangan Tampilan Tambah Data Soal

10. Rancangan Tampilan Ubah Data Soal

Tampilan ini menampilkan formulir yang memungkinkan guru untuk mengubah informasi yang terkait dengan soal ujian dalam sistem. Formulir ini akan menampilkan data soal yang sudah ada dan memungkinkan guru untuk memperbarui atau mengedit informasi tersebut sesuai kebutuhan. Tujuannya adalah memperbaharui soal apabila terjadi kesalahan dalam pemasukan data.

The screenshot shows a web browser window titled 'A Web Page' with a navigation bar containing 'Beranda', 'Tentang Aplikasi', and 'Proses Data' (highlighted). The main content area displays a table with 4 entries. A modal window titled 'Ubah Soal' is open, showing the details of the selected entry (No. 1). The modal contains input fields for 'No. Soal' (1), 'Soal' (xxxxx), and 'Kunci Jawaban' (xxxx), along with 'Ubah' and 'Kembali' buttons. The table has columns for 'No.', 'Soal', 'Kunci Jawaban', and 'Action'. The 'Action' column has 'Hapus' buttons for each row. The bottom of the page shows 'Showing 1 to 4 of 4 entries' and pagination controls (Previous, 1, Next).

No.	Soal	Kunci Jawaban	Action
1	xxxxx	xxxx	Hapus
2			Hapus
3			Hapus
4			Hapus

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.11 Rancangan Tampilan Ubah Data Soal

11. Rancangan Tampilan Hapus Data Soal

Tampilan ini menampilkan konfirmasi untuk menghapus data soal ujian dari sistem. Sebelum penghapusan dilakukan, akan ada pesan peringatan yang meminta konfirmasi dari pengguna, biasanya dengan pertanyaan apakah pengguna yakin ingin menghapus data soal tersebut. Setelah konfirmasi diberikan, data soal akan dihapus dari basis data.

The screenshot shows a web browser window titled "A Web Page" with a URL bar containing "http://". The page has a navigation bar with links: "Beranda", "Tentang Aplikasi", "Proses Data" (highlighted), "Nama User", and "Keluar". A confirmation dialog box is displayed in the center with the text "Yakin Ingin Menghapus Data ?" and buttons "Oke" and "Batal". Below the dialog is a table with 5 columns: "No.", "Soal", "Kunci Jawaban", "Action", and "Action". The table contains 4 rows of data, each with placeholder text "xxxxxxx". The "Action" column for each row contains two buttons: "Ubah" and "Hapus". At the bottom of the table, there is a pagination bar showing "Showing 1 to 4 of 4 entries" and buttons "Previous", "1" (highlighted), and "Next".

No.	Soal	Kunci Jawaban	Action	Action
1	xxxxxxx	xxxxxxx	Ubah	Hapus
2	xxxxxxx	xxxxxxx	Ubah	Hapus
3	xxxxxxx	xxxxxxx	Ubah	Hapus
4	xxxxxxx	xxxxxxx	Ubah	Hapus

Showing 1 to 4 of 4 entries

Previous 1 Next

Gambar 3.12 Rancangan Tampilan Hapus Data Soal

12. Rancangan Tampilan Lihat Hasil Ujian

Tampilan ini menampilkan hasil ujian siswa secara otomatis yang telah dinilai oleh sistem informasi. Informasi yang ditampilkan dapat berupa nilai akhir siswa yang dinilai secara otomatis dengan algoritma *Text Mining*. Tujuannya adalah menyediakan tampilan yang jelas dan terstruktur untuk memudahkan guru dalam memantau dan mengevaluasi prestasi siswa dalam ujian.

A Web Page

Navigation: Beranda | Tentang Aplikasi | **Proses Data** | Nama User | Keluar

Actions: Excel | PDF | Print | Search:

No.	NIS	Nama Siswa	Nilai Siswa	Action
xx	xxxxxx	xxxxxxxxxx	xx	Cek Hasil
xx	xxxxxx	xxxxxxxxxx	xx	Cek Hasil
xx	xxxxxx	xxxxxxxxxx	xx	Cek Hasil
xx	xxxxxx	xxxxxxxxxx	xx	Cek Hasil

Showing 1 to 4 of 4 entries

Previous | **1** | Next

Gambar 3.13 Rancangan Tampilan Lihat Hasil Ujian

13. Rancangan Tampilan Detail Hasil Ujian

Tampilan ini akan menampilkan detail hasil ujian siswa yang telah diproses menggunakan algoritma *Similarity Text* dan dioptimalkan dengan *Stemmer Sastrawi*. Informasi yang ditampilkan akan mencakup hasil analisis kesamaan teks, pemrosesan kata-kata, dan skor yang diberikan berdasarkan kriteria yang ditetapkan.

A Web Page

http://

Beranda Tentang Aplikasi **Proses Data** Nama User Keluar

NIS
Nama

Excel PDF Print Search:

ID. Soal	Soal	Kunci Jawaban	Jawaban Siswa	Proses Text Mining ...
⊕ xx	xxxx	xxxx	xxxx	xxxx
⊕ xx	xxx>	xxx>	xxx>	xxx>
⊕ xx	xxxx	xxxx	xxxx	xxxx
⊕ xx	xxxx	xxxx	xxxx	xxxx

Showing 1 to 4 of 4 entries

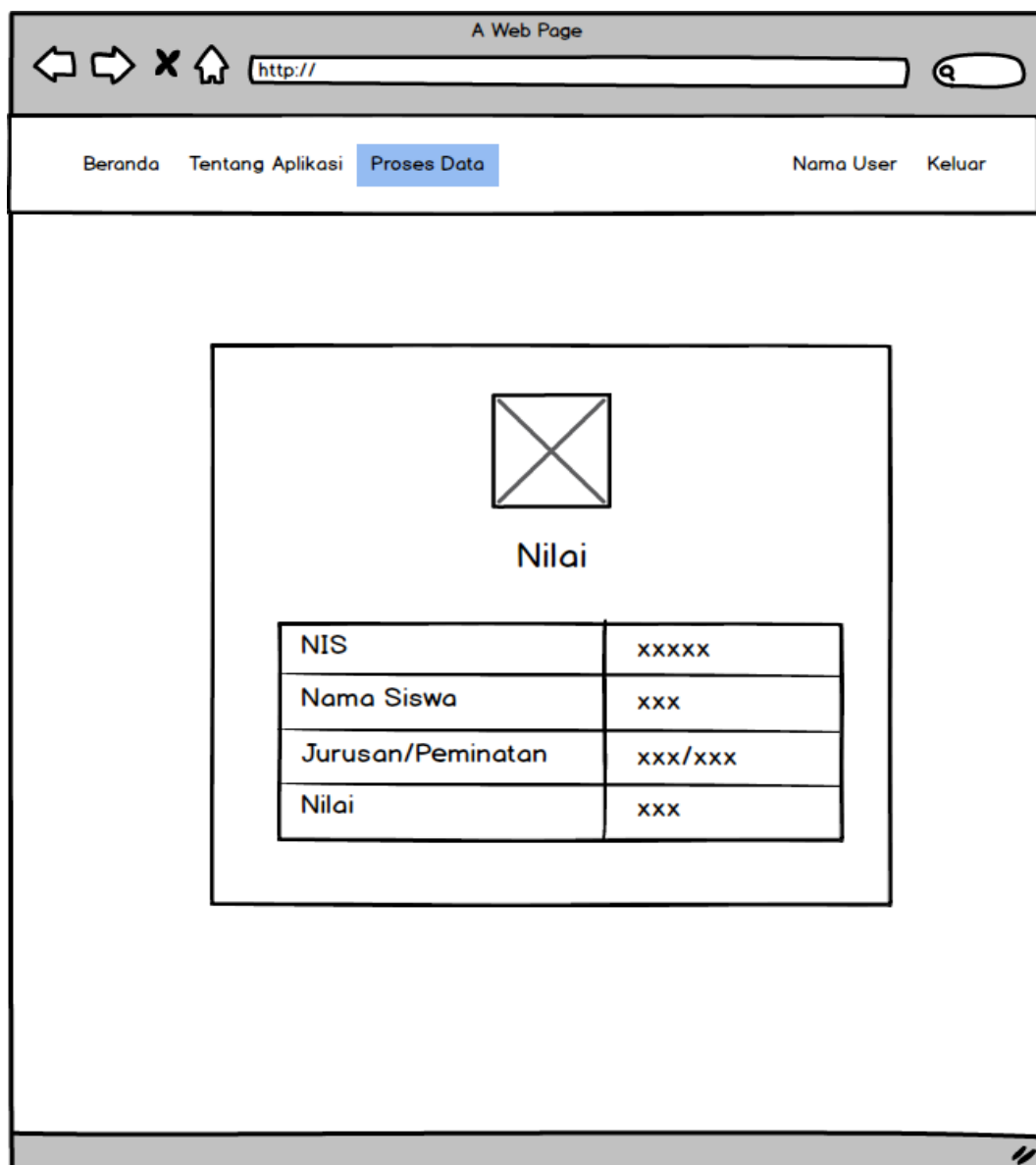
Hitung Hasil Akhir

Previous 1 Next

Gambar 3.14 Rancangan Tampilan Detail Hasil Ujian

14. Rancangan Tampilan Hasil Akhir

Tampilan ini menampilkan hasil akhir dari evaluasi ujian siswa setelah diproses oleh sistem. Hasil akhir ini mencakup skor keseluruhan siswa berdasarkan seluruh bagian ujian yang dinilai. Tampilan ini bertujuan untuk memberikan gambaran menyeluruh tentang prestasi siswa dalam ujian, sehingga guru dapat dengan cepat dan mudah mengevaluasi kinerja siswa secara keseluruhan.

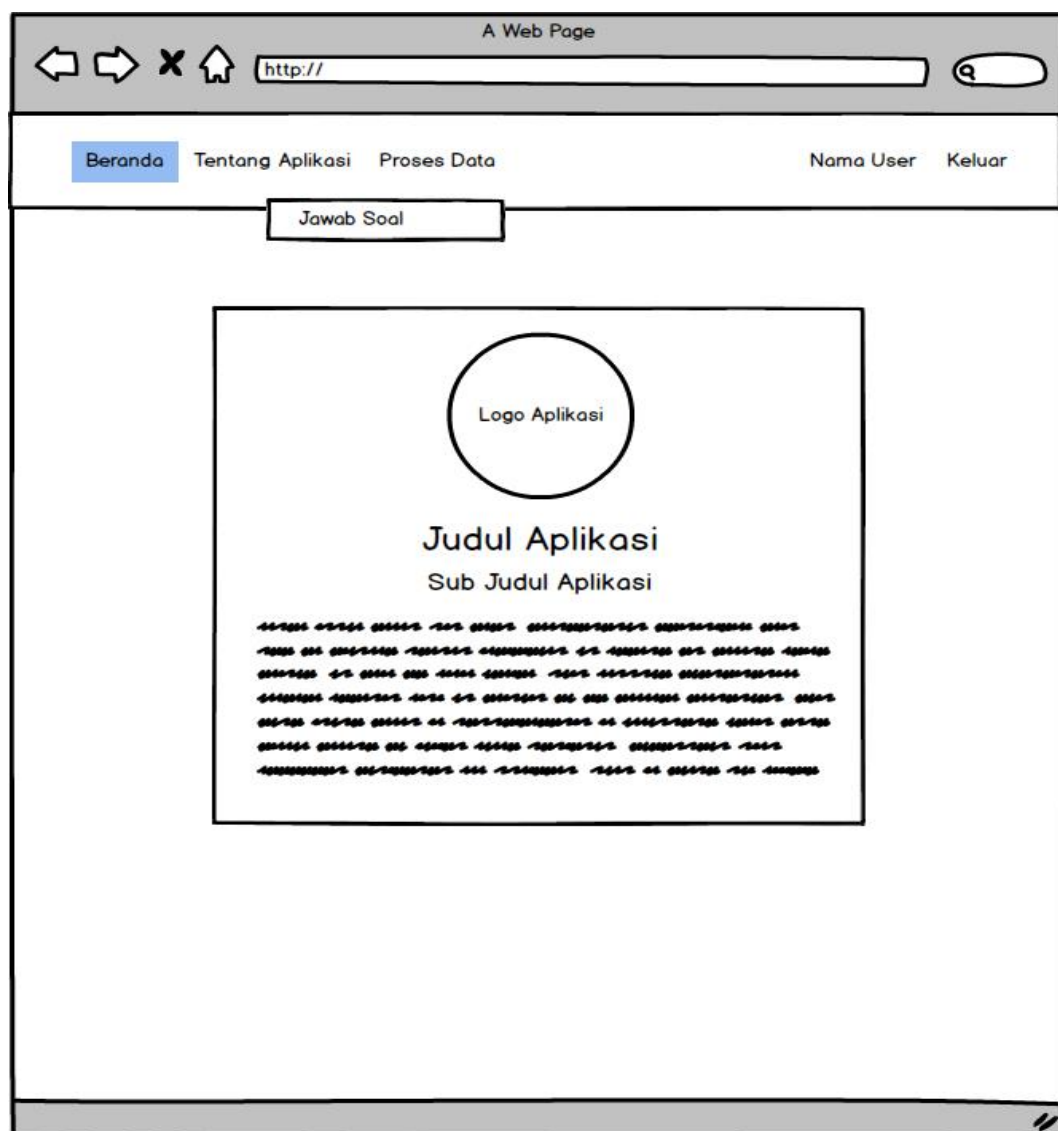


Gambar 3.15 Rancangan Tampilan Hasil Akhir

Berikut ini akan diuraikan keseluruhan rancangan tampilan untuk *user* siswa.

1. Rancangan Tampilan Beranda Siswa

Tampilan beranda siswa dirancang dengan fokus pada kebutuhan dan aktivitas utama seorang siswa dalam sistem. Rancangan ini mencakup ringkasan informasi penting seperti fitur-fitur bagi siswa untuk mengikuti ujian.



Gambar 3.16 Rancangan Tampilan Beranda Siswa

2. Rancangan Tampilan Jawab Soal

Tampilan ini menyajikan antarmuka tempat siswa menjawab soal ujian yang disediakan oleh sistem. Desainnya akan memungkinkan siswa untuk melihat soal dengan jelas, memberikan jawaban mereka, dan melakukan navigasi antara soal-soal yang tersedia. Selain itu, terdapat fitur untuk menyimpan jawaban sementara atau mengubah jawaban sebelum mengirimnya.

The wireframe shows a web browser window titled "A Web Page" with a URL bar containing "http://". The navigation bar includes links for "Beranda", "Tentang Aplikasi", and "Proses Data" (highlighted), along with "Nama User" and "Keluar".

Below the navigation bar is a table with three columns: "No.", "Soal", and "Kunci Jawaban". It contains four rows of placeholder text (xxxxxxx).

Below the table is a "Jawab" button. To its right are buttons for "Excel", "PDF", and "Print", followed by a "Search:" label and a text input field.

Below these elements is a table with three columns: "ID. Soal", "Jawaban", and "Action". It contains four rows of placeholder text (xx, xxxx, xxxx, xxxx). Each row has an "Ubah" button in the "Action" column.

At the bottom, there is a status bar showing "Showing 1 to 4 of 4 entries" and a pagination control with "Previous", "1" (highlighted), and "Next" buttons.

Gambar 3.17 Rancangan Tampilan Jawab Soal

3. Rancangan Tampilan Tambah Jawaban

Tampilan ini menyajikan formulir yang memungkinkan siswa untuk menambah jawaban tambahan dalam sebuah ujian atau latihan. Formulir ini akan berisi opsi untuk memilih nomor soal yang ingin dijawab serta tempat untuk mengisi jawaban yang tepat. Tujuannya adalah menyediakan cara yang jelas dan intuitif bagi siswa untuk menambah jawaban tambahan dalam ujian atau latihan di dalam sistem.

The image shows a web browser window titled "A Web Page". The address bar contains "http://". The page has a navigation bar with links: "Beranda", "Tentang Aplikasi", and "Proses Data" (which is highlighted in blue). To the right of the navigation bar are links for "Nama User" and "Keluar". The main content area displays a modal form titled "Tambah Jawaban" with a close button (X). The form contains the following fields and controls:

- ID Jawaban:** A text input field with the placeholder text "xx".
- Soal:** A dropdown menu with the placeholder text "xxxx".
- Jawaban:** A large text input field for the answer.
- NIS:** A text input field with the placeholder text "xxxx".
- Buttons:** Two buttons at the bottom right: "Simpan" (Save) and "Kembali" (Back).

Gambar 3.18 Rancangan Tampilan Tambah Jawaban

4. Rancangan Tampilan Ubah Jawaban

Tampilan ini menyajikan opsi bagi siswa untuk mengubah jawaban yang sudah mereka berikan dalam sebuah ujian atau latihan. Ketika siswa memilih untuk mengubah jawaban, mereka akan diberikan formulir yang memungkinkan mereka untuk memperbarui jawaban mereka. Tujuannya adalah menyediakan cara yang jelas dan mudah bagi siswa untuk mengubah jawaban mereka jika diperlukan selama ujian di dalam sistem.

The screenshot displays a web application interface for managing exam answers. At the top, there is a navigation bar with tabs: 'Beranda', 'Tentang Aplikasi', and 'Proses Data' (the active tab). To the right of the navigation bar, there is a user profile section with 'Nama User' and a 'Keluar' button. Below the navigation bar, there is a table with three columns: 'No.', 'Soal', and 'Kunci Jawaban'. The table contains 4 entries. A modal window titled 'Ubah Jawaban' is open, allowing the user to edit the answer for the selected entry. The modal contains the following fields and buttons:

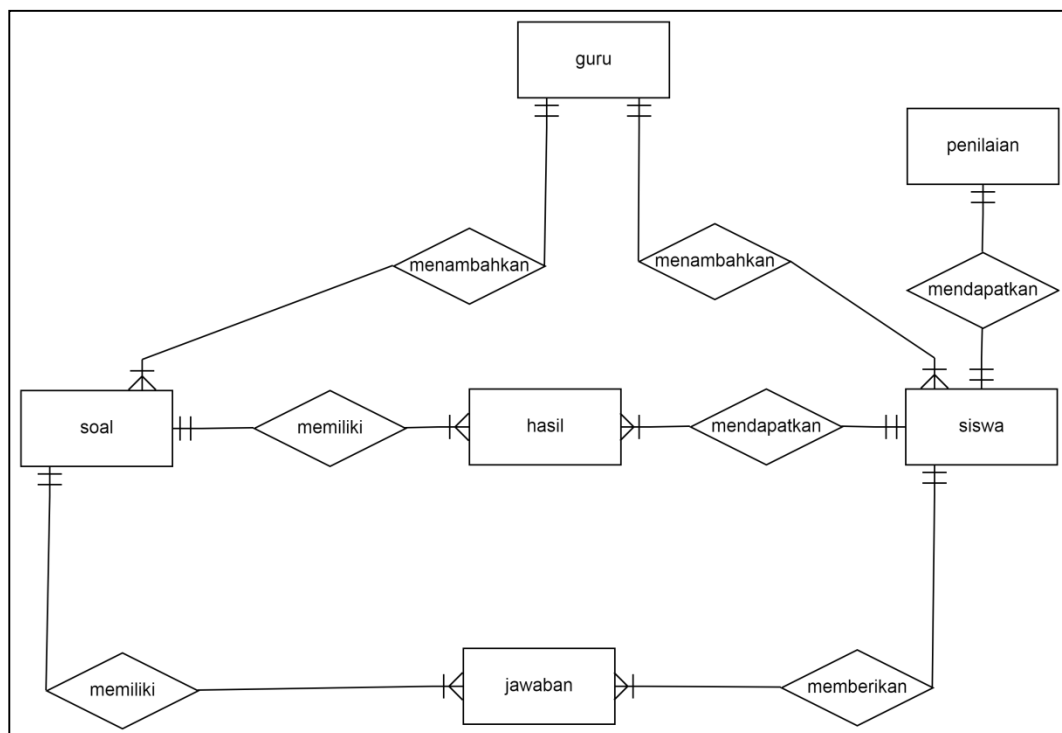
- ID Jawaban:** A text input field with the placeholder 'xx'.
- Soal:** A dropdown menu with the placeholder 'xxxx'.
- Jawaban:** A text input field with the placeholder 'xxxxxx'.
- NIS:** A text input field with the placeholder 'xxxx'.
- Buttons:** 'Ubah' (Update) and 'Kembali' (Back).

At the bottom of the modal, there are 'Jawab' and 'Excel' buttons. Below the table, there is a pagination control showing 'Showing 1 to 4 of 4 entries' and buttons for 'Previous', '1' (the current page), and 'Next'.

Gambar 3.19 Rancangan Tampilan Ubah Jawaban

3.2.2 Perancangan Basis Data

Pada sub bab ini, dibahas mengenai perancangan basis data menggunakan *Entity Relationship Diagram* (ERD). Basis data yang efisien dan terstruktur merupakan pondasi yang penting dalam pengembangan sistem informasi. Pembahasan akan mencakup model data yang direncanakan, hubungan antar entitas, atribut, serta kardinalitas yang terdefinisi dengan baik. Dengan perancangan basis data yang tepat, diharapkan sistem informasi dapat menyimpan dan mengelola data dengan efisien, konsisten, dan sesuai dengan kebutuhan pengguna.



Gambar 3.20 ERD Sistem Penilaian Ujian Otomatis di Sekolah Dharma

Bakti Lubuk Pakam

Berdasarkan Gambar 3.20 diuraikan struktur tabel dari setiap entitas yang ada pada ERD.

Tabel 3.8 Struktur Tabel Guru

Kolom	Jenis	Kosong	Bawaan
nip (Kunci Utama)	int(10)	Tidak	
nama_guru	varchar(50)	Tidak	
jenis_kelamin	enum('L', 'P')	Tidak	
no_telepon	varchar(15)	Tidak	
alamat	varchar(50)	Tidak	
password	varchar(10)	Ya	<i>NULL</i>
username	varchar(10)	Tidak	

Tabel 3.9 Struktur Tabel Hasil

Kolom	Jenis	Kosong	Bawaan
id (Kunci Utama)	int(3)	Tidak	
nim (Kunci <i>Foreign</i>)	int(12)	Tidak	
id_soal (Kunci <i>Foreign</i>)	int(5)	Tidak	
nilai	decimal(6,3)	Tidak	

Tabel 3.10 Struktur Tabel Jawaban

Kolom	Jenis	Kosong	Bawaan
id_jawaban (Kunci Utama)	int(5)	Tidak	
id_soal (Kunci <i>Foreign</i>)	int(5)	Tidak	
jawaban	varchar(400)	Tidak	
nim (Kunci <i>Foreign</i>)	int(12)	Tidak	

Tabel 3.11 Struktur Tabel Siswa

Kolom	Jenis	Kosong	Bawaan
nim (Kunci Utama)	int(12)	Tidak	
nama_siswa	varchar(50)	Tidak	
jurusan	varchar(50)	Tidak	
peminatan	varchar(50)	Tidak	

jenis_kelamin	enum('L', 'P')	Tidak	
password	varchar(10)	Ya	<i>NULL</i>
username	varchar(10)	Tidak	
nip (Kunci <i>Foreign</i>)	int(10)	Tidak	

Tabel 3.12 Struktur Tabel Penilaian

Kolom	Jenis	Kosong	Bawaan
id (Kunci Utama)	int(5)	Tidak	
nim (Kunci <i>Foreign</i>)	int(12)	Tidak	
hasil	decimal(6,3)	Tidak	

Tabel 3.13 Struktur Tabel Soal

Kolom	Jenis	Kosong	Bawaan
id_soal (Kunci Utama)	int(5)	Tidak	
soal_text	varchar(300)	Tidak	
kunci_jawaban	varchar(200)	Tidak	
nip (Kunci <i>Foreign</i>)	int(10)	Tidak	

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

Hasil penelitian berupa dibangunnya sebuah sistem penilaian ujian otomatis berbasis *text mining* pada sekolah Dharma Bakti Lubuk Pakam. Berikut ini akan diuraikan keseluruhan hasil tampilan untuk *user* guru.

1. Tampilan Awal

Pada bagian ini, akan dibahas mengenai rancangan tampilan awal dari sistem penilaian ujian otomatis. Tampilan awal memiliki peran yang penting karena menjadi titik awal interaksi pengguna dengan sistem.

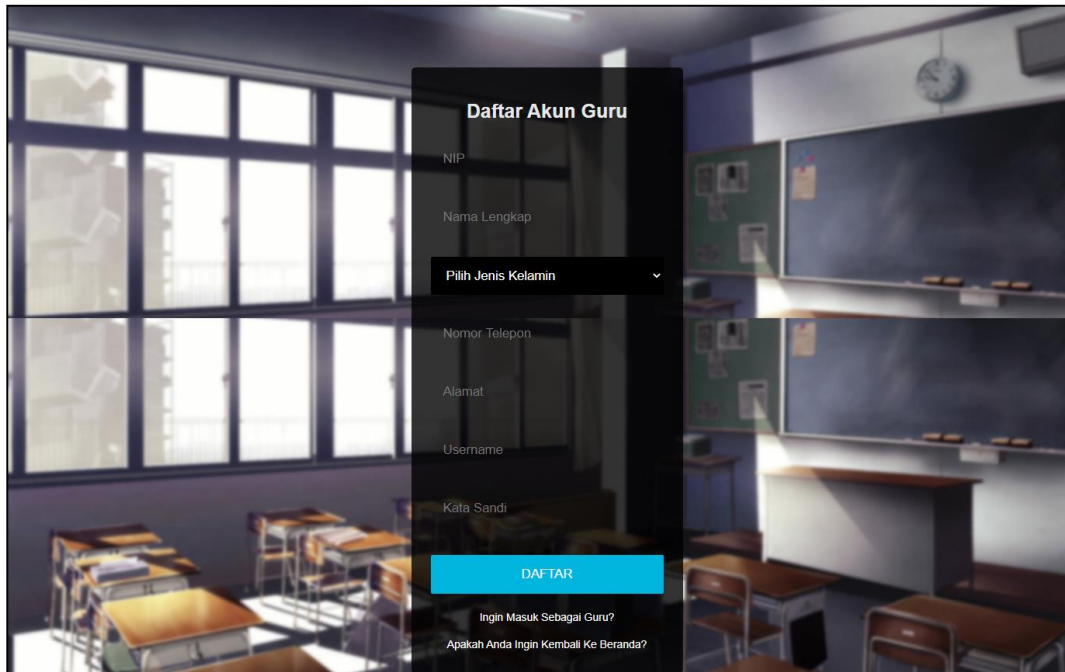


Gambar 4.1 Tampilan Awal

2. Tampilan Daftar Akun Guru

Tampilan daftar akun guru akan mencakup formulir pendaftaran yang meminta informasi yang diperlukan, seperti NIP, nama, jenis kelamin,

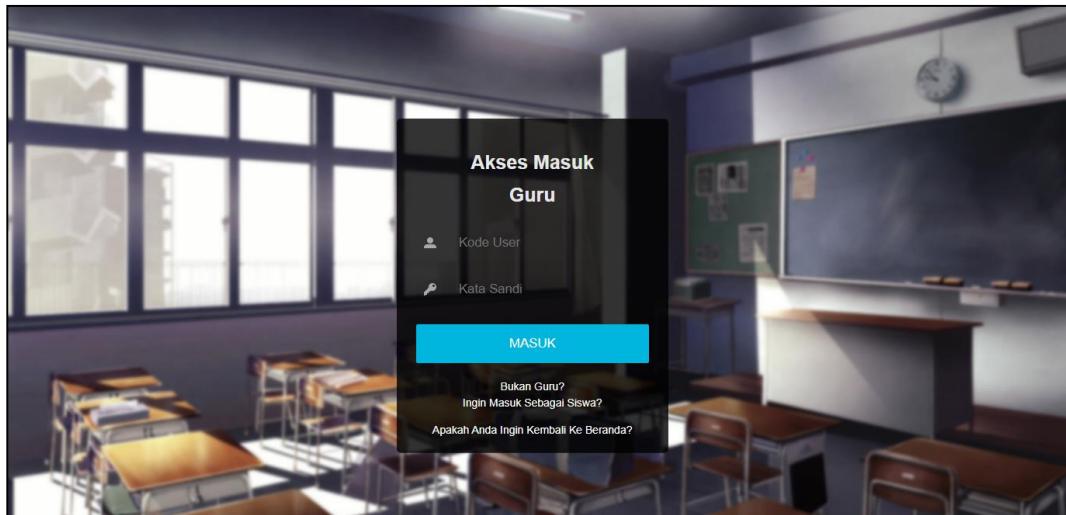
nomor telepon, alamat, *user name*, dan kata sandi. Tujuannya adalah menyediakan proses pendaftaran yang mudah dipahami dan diakses oleh para pengguna guru.



Gambar 4.2 Tampilan Daftar Akun Guru

3. Tampilan Masuk Guru

Tampilan masuk guru mencakup formulir masuk standar yang meminta pengguna untuk memasukkan informasi kredensial mereka, seperti kode *user* dan kata sandi. Tujuannya adalah untuk menyediakan pengalaman login yang lancar dan aman bagi para pengguna guru.



Gambar 4.3 Tampilan Masuk Guru

4. Tampilan Beranda Guru

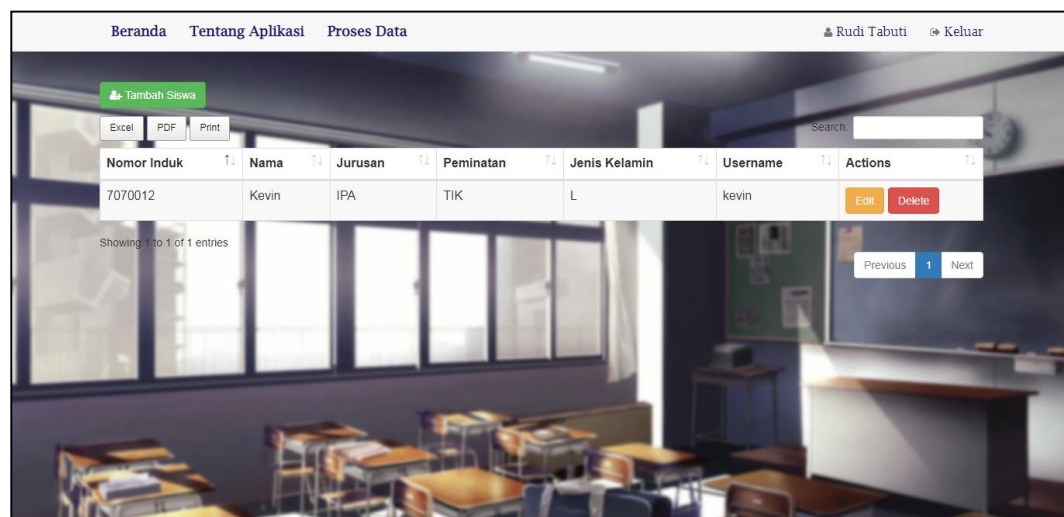
Tampilan beranda guru dirancang dengan fokus pada kebutuhan dan aktivitas utama seorang guru dalam sistem. Rancangan ini mencakup ringkasan informasi penting seperti fitur-fitur penambahan data siswa, soal, dan cek hasil ujian.



Gambar 4.4 Tampilan Beranda Guru

5. Tampilan Kelola Data Siswa

Tampilan ini menyajikan berbagai fungsi untuk pengelolaan data siswa oleh guru. Rancangan ini mencakup fitur-fitur seperti menambah, menghapus, dan memperbarui informasi siswa. Selain itu, terdapat fitur pencarian dan *filter* untuk memudahkan guru dalam menemukan data siswa yang diinginkan.



Gambar 4.5 Tampilan Kelola Data Siswa

6. Tampilan Tambah Data Siswa

Tampilan ini akan memberikan formulir yang memungkinkan guru untuk menambahkan data baru untuk siswa ke dalam sistem. Formulir akan mencakup kolom-kolom yang relevan untuk informasi siswa, seperti nama lengkap, jurusan, jenis kelamin dan informasi kontak lainnya. Tujuannya adalah memberikan pengalaman pengguna yang intuitif dan lancar bagi guru ketika menambahkan data siswa ke dalam sistem.

The screenshot shows a web application interface with a modal window titled "Tambah Siswa" (Add Student). The modal contains the following fields:

- NIM:
- Nama Siswa:
- Jurusan:
- Peminatan:
- Jenis Kelamin:
- Username:
- Password:

A blue "Simpan" (Save) button is located at the bottom of the modal. The background shows a dashboard with a table of students and navigation menus.

Gambar 4.6 Tampilan Tambah Data Siswa

7. Tampilan Ubah Data Siswa

Tampilan ini menampilkan formulir yang memungkinkan guru untuk mengubah informasi yang terkait dengan seorang siswa dalam sistem. Formulir ini akan menampilkan data siswa yang sudah ada dan memungkinkan guru untuk memperbarui atau mengedit informasi tersebut sesuai kebutuhan. Tujuannya adalah memberikan kemudahan bagi guru dalam mengelola informasi siswa dengan mengubah atau memperbarui data yang relevan sesuai kebutuhan.

The screenshot shows a web application interface with a modal window titled "Ubah Siswa" (Edit Student). The modal contains the following fields:

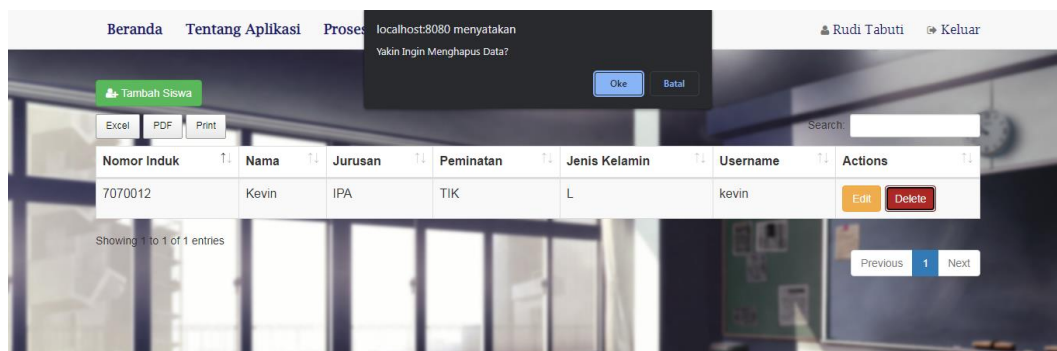
- NIM:
- Nama Siswa:
- Jurusan:
- Peminatan:
- Jenis Kelamin:
- Username:
- Password:

A blue "Ubah" (Edit) button is located at the bottom of the modal. The background shows a dashboard with a table of students and navigation menus.

Gambar 4.7 Tampilan Ubah Data Siswa

8. Tampilan Hapus Data Siswa

Tampilan ini menampilkan konfirmasi untuk menghapus data seorang siswa dari sistem. Sebelum penghapusan dilakukan, akan ada pesan peringatan yang meminta konfirmasi dari pengguna, biasanya dengan pertanyaan apakah pengguna yakin ingin menghapus data siswa tersebut. Setelah konfirmasi diberikan, data siswa akan dihapus dari basis data.



Gambar 4.8 Tampilan Hapus Data Siswa

9. Tampilan Tambah Data Soal

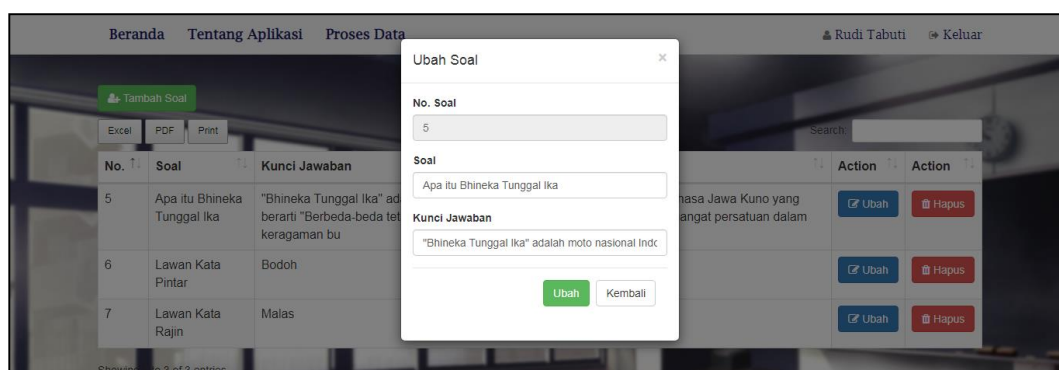
Tampilan ini akan memberikan formulir yang memungkinkan guru untuk menambahkan data baru untuk soal ke dalam sistem. Formulir akan mencakup kolom-kolom yang relevan untuk informasi soal, seperti judul soal dan kunci jawaban. Tujuannya adalah agar soal-soal tersebut dapat diujikan ke siswa.



Gambar 4.9 Tampilan Tambah Data Soal

10. Tampilan Ubah Data Soal

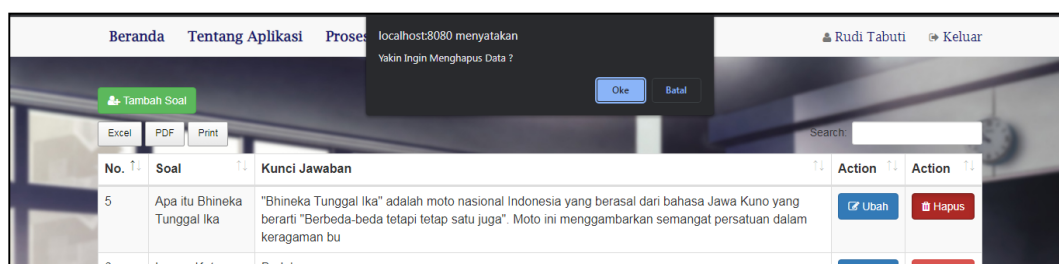
Tampilan ini menampilkan formulir yang memungkinkan guru untuk mengubah informasi yang terkait dengan soal ujian dalam sistem. Formulir ini akan menampilkan data soal yang sudah ada dan memungkinkan guru untuk memperbarui atau mengedit informasi tersebut sesuai kebutuhan. Tujuannya adalah memperbaharui soal apabila terjadi kesalahan dalam pemasukan data.



Gambar 4.10 Tampilan Ubah Data Soal

11. Tampilan Hapus Data Soal

Tampilan ini menampilkan konfirmasi untuk menghapus data soal ujian dari sistem. Sebelum penghapusan dilakukan, akan ada pesan peringatan yang meminta konfirmasi dari pengguna, biasanya dengan pertanyaan apakah pengguna yakin ingin menghapus data soal tersebut. Setelah konfirmasi diberikan, data soal akan dihapus dari basis data.



Gambar 4.11 Tampilan Hapus Data Soal

12. Tampilan Lihat Hasil Ujian

Tampilan ini menampilkan hasil ujian siswa secara otomatis yang telah dinilai oleh sistem informasi. Informasi yang ditampilkan dapat berupa nilai akhir siswa yang dinilai secara otomatis dengan algoritma *Text Mining*. Tujuannya adalah menyediakan tampilan yang jelas dan terstruktur untuk memudahkan guru dalam memantau dan mengevaluasi prestasi siswa dalam ujian.

No.	NIM	Nama Mahasiswa	Nilai Mahasiswa	Action
1	7070012	Kevin	85.897	Cek Hasil

Gambar 4.12 Tampilan Lihat Hasil Ujian

13. Tampilan Detail Hasil Ujian

Tampilan ini akan menampilkan detail hasil ujian siswa yang telah diproses menggunakan algoritma *Similarity Text* dan dioptimalkan dengan *Stemmer Sastrawi*. Informasi yang ditampilkan akan mencakup hasil analisis kesamaan teks, pemrosesan kata-kata, dan skor yang diberikan berdasarkan kriteria yang ditetapkan.

7070012
Kevin

Excel PDF Print

Search:

ID Soal.	Soal	Kunci Jawaban	Jawaban Siswa	Proses Text Mining Jawaban Siswa	Proses Text Mining Kunci Jawaban	Nilai per Soal
5	Apa itu Bhineka Tunggal Ika	"Bhineka Tunggal Ika" adalah moto nasional Indonesia yang berasal dari bahasa Jawa Kuno yang berarti "Berbeda-beda tetapi tetap satu juga". Moto ini menggambarkan semangat persatuan dalam keragaman bu	"Bhineka Tunggal Ika" adalah sebuah pepatah dalam bahasa Jawa Kuno yang bermakna "Berbeda-beda tetapi tetap satu juga". Pepatah ini terkenal di Indonesia dan sering dijadikan sebagai semboyan nasional yang menggambarkan semangat persatuan dalam keberagaman. Ide dasarnya adalah bahwa meskipun masyarakat Indonesia terdiri dari beragam suku, agama, ras, dan adat istiadat, namun mereka dapat hidup be	agam : 2 agama : 1 bahasa : 1 be : 1 beda : 1 bhineka : 1 buah : 1 dasar : 1 gambar : 1 ide : 1 ika : 1 indones : 2 isti : 1 jawa : 1 kuno : 1 makna : 1 masarakat : 1 nal : 1 nasional : 1 ng : 2 pepatah : 2 ras : 1 satu : 2 semangat : 1 semon : 1 suku : 1 t : 1 tetap : 1 tunggal : 1	bahasa : 1 beda : 1 bhineka : 1 bu : 1 gambar : 1 ika : 1 indones : 1 jawa : 1 kuno : 1 moto : 2 nasional : 1 ng : 2 ragam : 1 satu : 2 semangat : 1 tetap : 1 tunggal : 1	57.692
6	Lawan Kata Pintar	Bodoh	Bodoh	bodoh : 1	bodoh : 1	100.000
7	Lawan Kata Rajin	Malas	Malas	malas : 1	malas : 1	100.000

Showing 1 to 3 of 3 entries

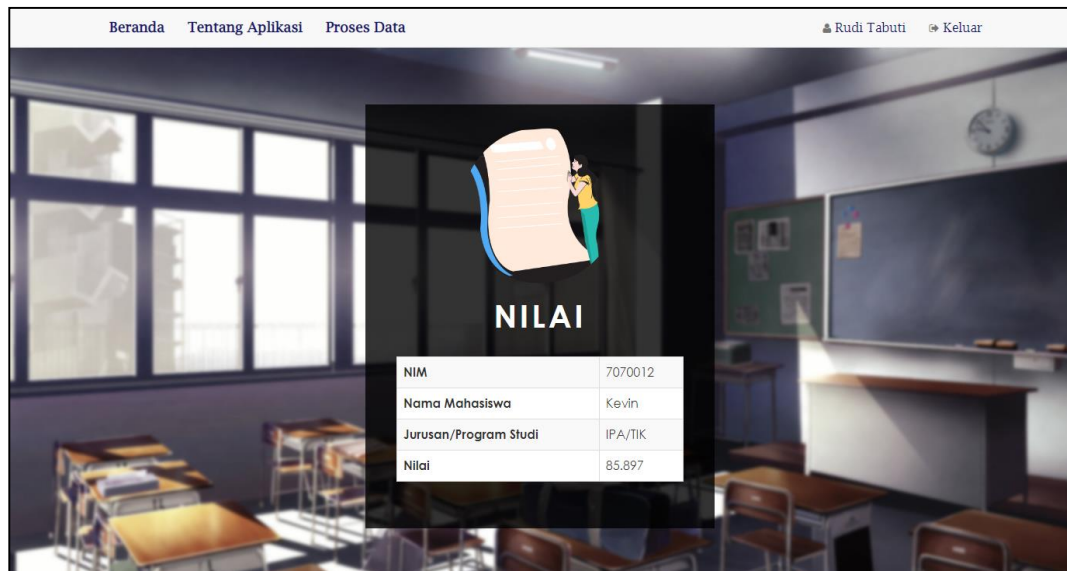
Hitung Hasil Akhir

Previous 1 Next

Gambar 4.13 Tampilan Detail Hasil Ujian

14. Tampilan Hasil Akhir

Tampilan ini menampilkan hasil akhir dari evaluasi ujian siswa setelah diproses oleh sistem. Hasil akhir ini mencakup skor keseluruhan siswa berdasarkan seluruh bagian ujian yang dinilai. Tampilan ini bertujuan untuk memberikan gambaran menyeluruh tentang prestasi siswa dalam ujian, sehingga guru dapat dengan cepat dan mudah mengevaluasi kinerja siswa secara keseluruhan.



Gambar 4.14 Tampilan Hasil Akhir

Berikut ini akan diuraikan keseluruhan hasil tampilan untuk *user* siswa.

1. Tampilan Beranda Siswa

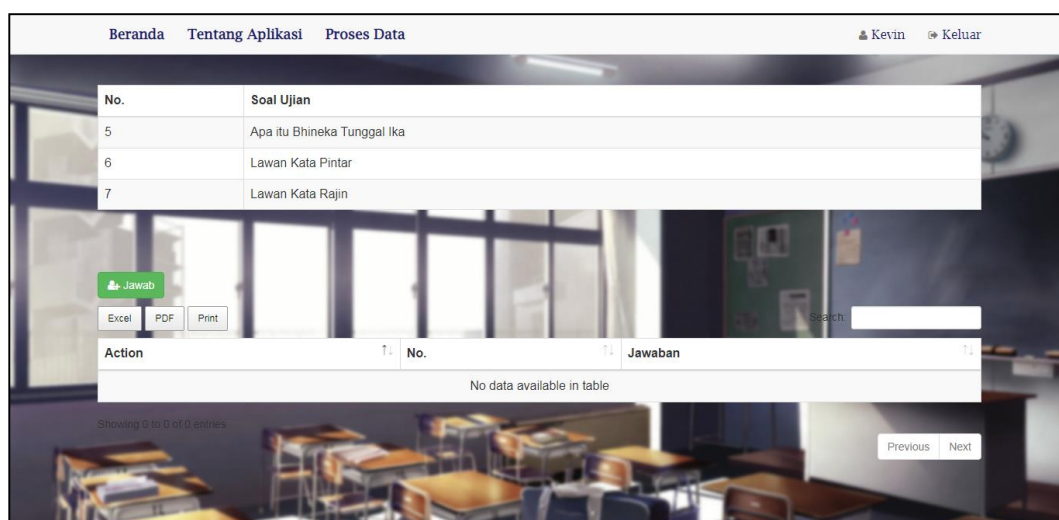
Tampilan beranda siswa dirancang dengan fokus pada kebutuhan dan aktivitas utama seorang siswa dalam sistem. Rancangan ini mencakup ringkasan informasi penting seperti fitur-fitur bagi siswa untuk mengikuti ujian.



Gambar 4.15 Tampilan Beranda Siswa

2. Tampilan Jawab Soal

Tampilan ini menyajikan antarmuka tempat siswa menjawab soal ujian yang disediakan oleh sistem. Desainnya akan memungkinkan siswa untuk melihat soal dengan jelas, memberikan jawaban mereka, dan melakukan navigasi antara soal-soal yang tersedia. Selain itu, terdapat fitur untuk menyimpan jawaban sementara atau mengubah jawaban sebelum mengirimnya.



Gambar 4.16 Tampilan Jawab Soal

3. Tampilan Tambah Jawaban

Tampilan ini menyajikan formulir yang memungkinkan siswa untuk menambah jawaban tambahan dalam sebuah ujian atau latihan. Formulir ini akan berisi opsi untuk memilih nomor soal yang ingin dijawab serta tempat untuk mengisi jawaban yang tepat. Tujuannya adalah menyediakan cara yang jelas dan intuitif bagi siswa untuk menambah jawaban tambahan dalam ujian atau latihan di dalam sistem.

The screenshot shows a web application interface with a background image of a classroom. At the top, there are navigation links: 'Beranda', 'Tentang Aplikasi', and 'Proses Data'. On the right, there is a user profile 'Kevin' and a 'Sign-Out' button. A modal window titled 'Tambah Jawaban' is centered on the screen. It contains the following fields: 'ID Jawaban' with the value '1', 'Soal' with a dropdown menu showing '- Pilih Soal -', 'Jawaban' with an empty text input, and 'NIM' with the value '7070012'. At the bottom of the modal are two buttons: 'Simpan' (green) and 'Kembali' (white).

Gambar 4.17 Tampilan Tambah Jawaban

4. Tampilan Ubah Jawaban

Tampilan ini menyajikan opsi bagi siswa untuk mengubah jawaban yang sudah mereka berikan dalam sebuah ujian atau latihan. Ketika siswa memilih untuk mengubah jawaban, mereka akan diberikan formulir yang memungkinkan mereka untuk memperbarui jawaban mereka. Tujuannya adalah menyediakan cara yang jelas dan mudah bagi siswa untuk mengubah jawaban mereka jika diperlukan selama ujian di dalam sistem.

The screenshot shows the same web application interface as before, but with a different modal window titled 'Ubah Jawaban'. In the background, a table of exam questions is visible. The modal window contains the following fields: 'ID Jawaban' with the value '1', 'Soal' with a dropdown menu showing 'Lawan Kata Pintar', 'Jawaban' with the value 'Bodoh', and 'Nomor Induk' with the value '7070012'. At the bottom of the modal are two buttons: 'Ubah' (green) and 'Kembali' (white). The background table has the following data:

No.	Soal Ujian
5	Apa itu Bhineka Tunggal Ika
6	Lawan Kata Pintar
7	Lawan Kata Rajin

Gambar 4.18 Tampilan Ubah Jawaban

4.2 Pembahasan

Berikut ini diuraikan pembahasan mengenai kelebihan dan kekurangan dari sistem penilaian ujian otomatis berbasis *text mining* pada sekolah Dharma Bakti Lubuk Pakam yang telah selesai dibangun. Berikut ini diuraikan kelebihan dari sistem yang dibangun yaitu:

1. Sistem penilaian ujian otomatis yang dibangun mempermudah proses penilaian jawaban *essay* karena proses perhitung melibatkan metode *text mining* sehingga koreksi nilai dapat dimunculkan dalam beberapa detik saja serta meminimalkan terjadinya kesalahan penilaian dari pihak guru.
2. Sistem penilaian ujian otomatis yang dibangun dapat secara otomatis melakukan penilaian jawaban *essay* dengan cepat dan hasil yang cukup tepat dan akurat.

Kekurangan dari sistem yang dibangun yaitu:

1. Sistem yang dibangun belum memiliki tingkatan kelas dan hanya mencakup tingkatan SMA saja sehingga belum dapat digunakan pada tingkatan kelas SD dan SMP.
2. Penerapan teknik *text mining* yang digunakan masih terbatas pada penerapan algoritma *Similarity Text* dan optimasi dengan *Stemmer Sastrawi*, padahal kombinasi dengan algoritma lainnya masih dapat dilakukan untuk meningkatkan akurasi penilaian.
3. Sistem yang dibangun hanya melibatkan pengajar dan peserta ujian serta belum melibatkan bagian-bagian lainnya yang berkaitan dalam pelaksanaan ujian.

4. Sistem yang dibangun belum memiliki fitur kompleks yang membahas mengenai ujian secara *online*.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Tahapan akhir menguraikan kesimpulan dari hasil penelitian yang didapatkan yaitu:

1. Sistem penilaian ujian otomatis berbasis text dapat mempermudah proses penilaian jawaban essay dengan mengintegrasikan metode *text mining*, yang memungkinkan koreksi nilai di sekolah Dharma Bakti Lubuk Pakam menjadi lebih cepat dan akurat.
2. Algoritma *Similarity Text* dan *Stemmer Sastrawi* yang diterapkan dalam sistem telah terbukti mampu memberikan penilaian yang cukup tepat dan akurat terhadap jawaban *essay*.

5.2 Saran

Adapun saran yang diberikan berkaitan dengan hasil penelitian yang didapatkan yaitu:

1. Disarankan sistem dikembangkan untuk memperluas cakupan penggunaan hingga tingkat pendidikan SD dan SMP. Hal ini akan memperluas manfaat sistem kepada lebih banyak sekolah dan tingkatan pendidikan.
2. Untuk meningkatkan akurasi penilaian, disarankan untuk melakukan kombinasi dengan metode *text mining* lainnya ataupun algoritma optimasi

seperti *Firefly* dan lainnya sehingga akurasi hasil penilaian *essay* menjadi lebih akurat.

3. Mengembangkan sistem agar mencakup lebih banyak *user* yang terlibat dari proses ujian.
4. Mengembangkan sistem agar memiliki fitur yang kompleks sehingga dapat menangani proses ujian secara *online*.

DAFTAR PUSTAKA

- [1] Imam Mufiid, S. Lestanti, and N. Kholila, “Aplikasi Penilaian Jawaban Esai Otomatis Menggunakan Metode Synonym Recognition Dan Cosine Similarity Berbasis Web,” *J. Mnemon.*, vol. 4, no. 2, pp. 31–37, 2021, doi: 10.36040/mnemonic.v4i2.4067.
- [2] A. Bastian and H. Sujadi, “Implementasi Dan Pengujian Aplikasi Penilaian Ujian Essay,” *J-Ensitem*, vol. 5, no. 01, pp. 261–264, 2018, doi: 10.31949/j-ensitem.v5i01.1214.
- [3] L. Ramdani, M. Jamaluddin, N. Yuniarti, A. Rahmani, and J. Hutahaeen, “Aplikasi Penilaian Otomatis Ujian Esai Berbahasa Indonesia Menggunakan Algoritma K-Nearest Neighbor (Studi kasus MAN Cimahi),” *Pros. Ind. Res. Work. Natl. Semin.*, vol. 10, no. August 2019, pp. 314–324, 2020, [Online]. Available: <https://jurnal.polban.ac.id/ojs-3.1.2/proceeding/article/view/1404>.
- [4] N. Chamidah, M. M. Santoni, H. N. Irmanda, and R. Astriratma, “Penilaian Esai Pendek Otomatis dengan Pencocokan Kata Kunci Frasa Nomina Short Essay Autoscoring using Noun Phrase-Keyword Matching,” *Techno.Com*, vol. 20, no. 4, pp. 489–498, 2021.
- [5] Mirda, M. Sarjan, and U. Khairat, “Aplikasi Ujian Essay Koreksi Otomatis Menggunakan Metode Cosine Similarity,” *J. Peqquruang Conf. Ser.*, vol. 4, no. 1, pp. 418–422, 2022.
- [6] Muttaqin *et al.*, *Implementasi Artificial Intelligence (AI) Dalam Kehidupan*. Medan: Yayasan Kita Menulis, 2023.
- [7] D. Purnamasari *et al.*, *Pengantar Metode Analisis Sentimen*. Depok: Gunadarma, 2023.
- [8] Y. Findawati and M. A. Rosid, *Buku Ajar Text Mining*. Sidoarjo: UMSIDA Press, 2020.
- [9] M. R. Faisal, D. Kartini, A. R. Arrahimi, and T. H. Saragih, *Belajar Data Science Text Mining Untuk Pemula I*. Banjarbaru: Scripta Cendekia, 2022.
- [10] A. Prasidhatama and K. M. Suryaningrum, “Perbandingan Algoritma Nazief & Adriani Dengan Algoritma Idris Untuk Pencarian Kata Dasar,” *J. Teknol. dan Manaj. Inform.*, vol. 4, no. 1, pp. 1–4, 2018, doi: 10.26905/jtmi.v4i1.1773.
- [11] D. Mustikasari, I. Widaningrum, R. Arifin, and W. H. E. Putri, “Comparison of Effectiveness of Stemming Algorithms in Indonesian Documents,” *Proc.*

2nd Borobudur Int. Symp. Sci. Technol. (BIS-STE 2020), vol. 203, pp. 154–158, 2021, doi: 10.2991/aer.k.210810.025.

- [12] M. Wali and S. Safrizal, “Similar text sebagai Pengkodean Aplikasi Plagiarisme,” *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 2, no. 1, p. 11, 2018, doi: 10.35870/jtik.v2i1.43.
- [13] N. Fitrianingsih, M. Asfi, D. Prasetyo, R. P. Kusuma, and M. A. Sulhan, “Deteksi Tingkat Kemiripan Judul Menggunakan Algoritma Oliver Pada Sistem Informasi Pengajuan Skripsi,” *J. Media Inform. Budidarma*, vol. 6, no. 4, p. 2148, 2022, doi: 10.30865/mib.v6i4.4409.
- [14] R. A. Sukamto and M. Shalahuddin, “Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek.” Informatika, Bandung, 2018.

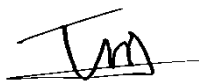
DAFTAR RIWAYAT HIDUP

Nama : Tinis Roy
Tempat/Tanggal Lahir : lubuk pakam, 15 Maret 2003
Jenis Kelamin : Laki - Laki
Agama : Hindu
Alamat : Jln.Sulta Hasauddin
Email : tinisroy17@gmail.com

Pendidikan Formal :

2006 - 2008 : TK Swasta Dharma Bakti Lubuk Pakam
2018 - 2014 : SD Swasta Dharma Bakti Lubuk Pakam
2014 - 2017 : SMP Swasta Dharma Bakti Lubuk Pakam
2017 - 2020 : SMA Swasta Dharma Bakti Lubuk Pakam
2020 – Sekarang : S1 Teknik Informasi STMIK TIME Medan

Medan, 30 Juli 2024



Tinis Roy



SURAT KEPUTUSAN
NO. 095/SK/PI/STMIKTIME/XII/2023

Tentang
PENGANGKATAN DOSEN PEMBIMBING SKRIPSI
PROGRAM STUDI TEKNIK INFORMATIKA
STMIK TIME

1. **Membaca** : a. Hasil pemeriksaan berkas dan kelengkapan mahasiswa.
b. Presentasi Proposal Judul Skripsi mahasiswa tertanggal 09 Oktober – 18 November 2023.
2. **Mengingat** : a. Kurikulum Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK)Time Program Studi Teknik Informatika.
b. Penyesuaian teknis pelaksanaan Skripsi Program Studi Teknik Informatika.
3. **Menimbang** : Bahwa skripsi mahasiswa merupakan wujud aplikasi kristalisasi sikap kemampuan dan ketrampilan mahasiswa/i dalam upaya menerapkan ilmu pengetahuan yang telah dipelajari di bangku kuliah, serta skripsi merupakan salah satu mata kuliah yang memiliki bobot 6 SKS, maka dipandang perlu untuk menetapkan Dosen Pembimbing I dan Dosen Pembimbing II Skripsi.

4. **Menetapkan** :
Pertama :

Mengangkat Saudara :

Pembimbing I	David, S.Kom., M.TI
Pembimbing II	Wilson, S.Kom., M.M

Sebagai Dosen Pembimbing Skripsi atas :

2044068	Tinis Roy	Teknik Informatika
---------	-----------	--------------------

Dengan Judul Skripsi :

Perancangan Sistem Penilaian Ujian Otomatis Berbasis Text Mining Pada Sekolah Dharma Bakti Lubuk Pakam
--

Mahasiswa Program Strata Satu (S1) Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK)Time.

- Kedua : Apabila terdapat kekeliruan di kemudian hari tentang penetapan Dosen Pembimbing Skripsi ini akan diperbaiki sebagaimana mestinya
- Ketiga : Surat Keputusan ini berlaku sejak tanggal ditetapkan 07 Desember 2023.

Ditetapkan di
Medan, 07 Desember 2023



Hendri, M.Kom
Pembantu Ketua I STMIK TIME



STMIK TIME

SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER - TIME

FORMULIR BIMBINGAN SKRIPSI

NAMA : TINIS ROY
 NIM : 2044068
 JUDUL SKRIPSI : PERANCANGAN SISTEM PENILAIAN UJIAN OTOMATIS DENGAN ALGORITMA
 SIMILARITY TEXT DAN OPTIMASI DENGAN ALGORITMA STEMMER
 SASTRAWI PADA SEKOLAH DHARMA BAKTI LUBUK PAKAM
 NAMA DOSEN : DAVID, M.TI
 DOSEN PEMBIMBING : ☒ I (SATU) ☐ II (DUA)

TANGGAL BIMBINGAN	CATATAN BIMBINGAN	PARAF
19 Januari 2024	Bab 1 : Acc	
21 Februari 2024	Bab 2 : Acc	
15 April 2024	Bab 3 : Revisi	
29 April 2024	Bab 3 : Acc	
28 Juni 2024	Bab 2 : Isikan halaman 25 Bab 3 : Rapikan Tabel 3.8	
29 Juni 2024	Bab 2 : Acc Bab 3 : Acc	
01 Juli 2024	Bab 4 : Acc	
02 Juli 2024	Bab 5 : Acc	

Ketentuan :

1. Bimbingan dengan dosen pembimbing minimal 8 (delapan) kali
2. Formulir Bimbingan wajib diisi dan ditunjukkan ke dosen pembimbing tiap bimbingan
3. Formulir Bimbingan Skripsi ini dianggap sah, apabila telah mendapatkan Pengesahan dari dosen pembimbing
4. Jika tidak memenuhi ketentuan di atas, skripsi dapat **DIBATALKAN**

Medan, Juli 2024
 Dosen Pembimbing,

Mahasiswa,



David, M.TI

Tinis Roy



STMIK TIME

SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER - TIME

FORMULIR BIMBINGAN SKRIPSI

NAMA : Tinis Roy
 NIM : 2044068
 JUDUL SKRIPSI : PERANCANGAN SISTEM PENILAIAN UJIAN OTOMATIS DENGAN ALGORITMA
 SIMILARITY TEXT DAN OPTIMASI DENGAN ALGORITMA STEMMER
 SASTRAWI PADA SEKOLAH DHARMA BAKTI LUBUK PAKAM
 NAMA DOSEN : WILSON, S.KOM, M.M
 DOSEN PEMBIMBING : ☐ I (SATU) ☒ II (DUA)

TANGGAL BIMBINGAN	CATATAN BIMBINGAN	PARAF
12 Mar 2024	Bab 1 - Revisi Typo - Tambahkan singkatan untuk beberapa kata	
13 Mar 2024	Bab 1 - Acc Bab 2 - Cek Kembali kata asing untuk cetak miring	
29 April 2024	Bab 2 dan 3 Acc	
20 Juli 2024	Bab 4 - Revisi typo - Sesuaikan margin sub judul	
22 Juli 2024	Bab 4 - Tampilan pada daftar gambar bermasalah	
24 Juli 2024	Bab 4 Acc	
27 Juli 2024	Bab 5 - Sesuaikan margin penomoran	
28 Juli 2024	Bab 5 Acc	

Ketentuan :

1. Bimbingan dengan dosen pembimbing minimal 8 (delapan) kali
2. Formulir Bimbingan wajib diisi dan ditunjukkan ke dosen pembimbing tiap bimbingan
3. Formulir Bimbingan Skripsi ini dianggap sah, apabila telah mendapatkan Pengesahan dari dosen pembimbing
4. Jika tidak memenuhi ketentuan di atas, skripsi dapat **DIBATALKAN**

Medan, 29 April 2024

Dosen Pembimbing,



Wilson, S.Kom.MM

Mahasiswa,

Tinis Roy

LISTING PROGRAM

```
-- phpMyAdmin SQL Dump
-- version 4.3.11
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: 21 Jun 2024 pada 06.24
-- Versi Server: 5.6.24
-- PHP Version: 5.6.8

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `sdblp-essay`
--
```

--

-- Struktur dari tabel `dosen`

--

```
CREATE TABLE IF NOT EXISTS `dosen` (
  `nip` int(10) NOT NULL,
  `nama_dosen` varchar(50) NOT NULL,
  `jenis_kelamin` enum('L','P') NOT NULL,
  `no_telepon` varchar(15) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `Password` varchar(10) DEFAULT NULL,
  `username` varchar(10) NOT NULL
) ENGINE=InnoDB AUTO_INCREMENT=21123 DEFAULT
CHARSET=utf8mb4;
```

--

-- Dumping data untuk tabel `dosen`

--

```
INSERT INTO `dosen` (`nip`, `nama_dosen`, `jenis_kelamin`, `no_telepon`,
`alamat`, `Password`, `username`) VALUES
(21121, 'Tinis Roy', 'L', '081721972112', 'Jl. Rajawali Ampat V', 'tinisroy',
'tinisroy'),
(21122, 'Rudi Tabuti', 'L', '082170770511', 'Jln. Selam IV No. 101', '123456',
'ruditabuti');
```

```
-----

--

-- Struktur dari tabel `hasil`

--

CREATE TABLE IF NOT EXISTS `hasil` (
  `id` int(3) NOT NULL,
  `nim` int(12) NOT NULL,
  `id_soal` int(5) NOT NULL,
  `nilai` decimal(6,3) NOT NULL
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=utf8mb4;

--

-- Dumping data untuk tabel `hasil`

--

INSERT INTO `hasil` (`id`, `nim`, `id_soal`, `nilai`) VALUES
(10, 7070012, 6, '100.000'),
(11, 7070012, 7, '100.000'),
(12, 7070012, 5, '57.692');

-----

--
```

```
-- Struktur dari tabel `jawaban`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `jawaban` (
```

```
  `id_jawaban` int(5) NOT NULL,
```

```
  `id_soal` int(5) NOT NULL,
```

```
  `jawaban` varchar(400) NOT NULL,
```

```
  `nim` int(12) NOT NULL
```

```
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4;
```

```
--
```

```
-- Dumping data untuk tabel `jawaban`
```

```
--
```

```
INSERT INTO `jawaban` (`id_jawaban`, `id_soal`, `jawaban`, `nim`) VALUES
```

```
(1, 6, 'Bodoh', 7070012),
```

```
(2, 7, 'Malas', 7070012),
```

```
(3, 5, '"Bhineka Tunggal Ika" adalah sebuah pepatah dalam bahasa Jawa Kuno yang bermakna "Berbeda-beda tetapi tetap satu juga". Pepatah ini terkenal di Indonesia dan sering dijadikan sebagai semboyan nasional yang menggambarkan semangat persatuan dalam keberagaman. Ide dasarnya adalah bahwa meskipun masyarakat Indonesia terdiri dari beragam suku, agama, ras, dan adat istiadat, namun mereka dapat hidup be', 7070012);
```

```
-- -----
```

```
--
```

```
-- Struktur dari tabel `mahasiswa`
```

--

```
CREATE TABLE IF NOT EXISTS `mahasiswa` (
  `nim` int(12) NOT NULL,
  `nama_mahasiswa` varchar(50) NOT NULL,
  `jurusan` varchar(50) NOT NULL,
  `program_studi` varchar(50) NOT NULL,
  `jenis_kelamin` enum('L','P') NOT NULL,
  `password` varchar(10) DEFAULT NULL,
  `username` varchar(10) NOT NULL,
  `nip` varchar(11) NOT NULL,
  `is_ujian` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

--

-- Dumping data untuk tabel `mahasiswa`

--

```
INSERT INTO `mahasiswa` (`nim`, `nama_mahasiswa`, `jurusan`,
`program_studi`, `jenis_kelamin`, `password`, `username`, `nip`, `is_ujian`)
VALUES
(7070012, 'Kevin', 'IPA', 'TIK', 'L', '123456', 'kevin', '21122', 1),
(7070013, 'Alyani Zhafarina', 'IPA', 'TIK', 'P', 'alyani', 'zha', '21121', 0),
(7070031, 'Fadhil Baskoro Utomo', 'IPA', 'TIK', 'L', '123456', 'fadhil', '21121', 0),
(7070044, 'Fahmia Amelia', 'IPA', 'TIK', 'P', 'miaw', 'miaw', '21121', 0);
```

```
--
```

```
-- Struktur dari tabel `penilaian`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `penilaian` (
  `id` int(5) NOT NULL,
  `nim` int(12) NOT NULL,
  `hasil` decimal(6,3) NOT NULL
) ENGINE=InnoDB AUTO_INCREMENT=105 DEFAULT
CHARSET=utf8mb4;
```

```
--
```

```
-- Dumping data untuk tabel `penilaian`
```

```
--
```

```
INSERT INTO `penilaian` (`id`, `nim`, `hasil`) VALUES
(104, 7070012, '85.897');
```

```
-- -----
```

```
--
```

```
-- Struktur dari tabel `soal`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `soal` (
```

```

`id_soal` int(5) NOT NULL,
`nip` varchar(10) NOT NULL,
`soal_text` varchar(300) NOT NULL,
`kunci_jawaban` varchar(200) NOT NULL
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8mb4;

--

-- Dumping data untuk tabel `soal`

--

INSERT INTO `soal` (`id_soal`, `nip`, `soal_text`, `kunci_jawaban`) VALUES
(1, '21121', 'Apa fungsi eloquent?', 'berfungsi untuk create update delete pada laravel'),
(2, '21121', '"..... DATABASE buku" berfungsi untuk membuat database baru dengan nama buku', 'CREATE'),
(3, '21121', 'Apa kepanjangan dari MVC', 'merupakan model view controller'),
(4, '21121', 'Apakah peran sebuah framework ', 'mudah cepat stabil dalam kerja laravel'),
(5, '21122', 'Apa itu Bhineka Tunggal Ika', '"Bhineka Tunggal Ika" adalah moto nasional Indonesia yang berasal dari bahasa Jawa Kuno yang berarti "Berbeda-beda tetapi tetap satu juga". Moto ini menggambarkan semangat persatuan dalam keragaman bu'),
(6, '21122', 'Lawan Kata Pintar', 'Bodoh'),
(7, '21122', 'Lawan Kata Rajin', 'Malas');

--

-- Indexes for dumped tables

--

```

```
--  
  
-- Indexes for table `dosen`  
  
--  
  
ALTER TABLE `dosen`  
  ADD PRIMARY KEY (`nip`);  
  
--  
  
-- Indexes for table `hasil`  
  
--  
  
ALTER TABLE `hasil`  
  ADD PRIMARY KEY (`id`);  
  
--  
  
-- Indexes for table `jawaban`  
  
--  
  
ALTER TABLE `jawaban`  
  ADD PRIMARY KEY (`id_jawaban`), ADD KEY `nim` (`nim`), ADD KEY  
  `id_soal` (`id_soal`);  
  
--  
  
-- Indexes for table `mahasiswa`  
  
--  
  
ALTER TABLE `mahasiswa`  
  ADD PRIMARY KEY (`nim`);
```

```

--
-- Indexes for table `penilaian`
--
ALTER TABLE `penilaian`
  ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `nim_2` (`nim`), ADD KEY
  `nim` (`nim`);

--
-- Indexes for table `soal`
--
ALTER TABLE `soal`
  ADD PRIMARY KEY (`id_soal`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `dosen`
--
ALTER TABLE `dosen`
  MODIFY `nip` int(10) NOT NULL
  AUTO_INCREMENT,AUTO_INCREMENT=21123;

--
-- AUTO_INCREMENT for table `hasil`
--
ALTER TABLE `hasil`

```

```

    MODIFY          `id`          int(3)          NOT          NULL
    AUTO_INCREMENT,AUTO_INCREMENT=13;

--

-- AUTO_INCREMENT for table `jawaban`

--

ALTER TABLE `jawaban`

    MODIFY          `id_jawaban`    int(5)          NOT          NULL
    AUTO_INCREMENT,AUTO_INCREMENT=4;

--

-- AUTO_INCREMENT for table `penilaian`

--

ALTER TABLE `penilaian`

    MODIFY          `id`          int(5)          NOT          NULL
    AUTO_INCREMENT,AUTO_INCREMENT=105;

--

-- AUTO_INCREMENT for table `soal`

--

ALTER TABLE `soal`

    MODIFY          `id_soal`      int(5)          NOT          NULL
    AUTO_INCREMENT,AUTO_INCREMENT=8;

--

-- Ketidakleluasaan untuk tabel pelimpahan (Dumped Tables)

--

--

-- Ketidakleluasaan untuk tabel `jawaban`

--

```

```

ALTER TABLE `jawaban`

ADD CONSTRAINT `jawaban_ibfk_1` FOREIGN KEY (`nim`) REFERENCES
`mahasiswa` (`nim`),

ADD CONSTRAINT `jawaban_ibfk_2` FOREIGN KEY (`id_soal`)
REFERENCES `soal` (`id_soal`);

--

-- Ketidakeleluasaan untuk tabel `penilaian`

--

ALTER TABLE `penilaian`

ADD CONSTRAINT `penilaian_ibfk_1` FOREIGN KEY (`nim`) REFERENCES
`mahasiswa` (`nim`);

/*!40101 SET
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET
COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```