

**PERANCANGAN SISTEM PENILAIAN JAWABAN ESAI
OTOMATIS MENGGUNAKAN METODE *SYNONYM
RECOGNITION* DAN *COSINE SIMILARITY***

PROPOSAL

**Oleh :
IMAM MUFIID
NIM. 17104410005**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ISLAM BLITAR
2021**

DAFTAR ISI

Sampul Proposal	
Daftar isi.....	ii
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah	4
1.3. Tujuan Penelitian.....	4
1.4. Manfaat Penelitian.....	4
1.5. Batasan Masalah.....	5
1.6. Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA.....	6
2.1. Landasan Teori	6
2.2. Kajian Penelitian	11
BAB III METODE PENELITIAN.....	19
3.1. Ruang Lingkup Penelitian untuk Rancangan	19
3.2. Metode Penelitian Rancangan	19
3.3. Metode Analisis Sintensis	20
3.4. Skema Metode Perancangan.....	21
Daftar Rujukan.....	47

BAB I

PENDAHULUAN

1.1.Latar Belakang

Ujian merupakan salah satu alat evaluasi dalam proses belajar mengajar. Ujian dilakukan secara keseluruhan untuk menentukan kualitas pembelajaran dan merupakan suatu alat ukur untuk mengukur performa peserta didik. Namun disisi lain ujian merupakan suatu pekerjaan rumah dan kesibukan tambahan yang harus dilakukan oleh pengajar terutama dalam proses koreksi.

Proses koreksi soal pilihan ganda mungkin tidak terlalu susah karena soal berjenis pilihan ganda memiliki sifat *true false*, namun tidak untuk soal esai, ujian dengan sistem esai merupakan bentuk evaluasi dimana pilihan jawaban tidak disediakan, dan peserta didik harus menjawab dengan kalimat sehingga dapat melatih peserta didik dalam menyampaikan sesuatu informasi secara verbal, selain itu ujian esai juga menuntut pemahaman yang lebih baik akan suatu ilmu dan dapat digunakan untuk mengukur tingkat pemahaman manusia akan suatu ilmu secara lebih mendalam.

Jawaban soal uraian akan menghasilkan beragam kemungkinan jawaban sesuai dengan pemahaman masing-masing peserta didik. Hasil dari jawaban essay bukan hanya benar atau salah tapi juga hampir benar. Sehingga skor nilai untuk jawaban essay bisa beragam (Bastian & Sujadi, 2018).

Soal dengan tipe esai tersebut membutuhkan banyak waktu untuk mencocokkan hasil jawaban peserta didik yang berbeda-beda dengan jawaban yang benar, pengajar harus membaca satu persatu jawaban peserta didik dan terkadang terdapat tulisan peserta didik yang sulit dibaca. Salim (2017) mengatakan salah satu kelemahan jawaban dalam bentuk essay adalah sulitnya menilai jawaban yang membutuhkan banyak waktu. Namun 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 (Smrti, et al., 2019).

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.

Penyelesaian koreksi hasil ujian dengan cepat dan tepat diharapkan mampu menyelesaikan permasalahan subjektifitas pengajar yang terjadi pada saat proses penilaian jawaban ujian esai secara manual. Untuk dapat mengoreksi jawaban dengan cepat dan tepat dibutuhkan aplikasi analisis jawaban esai, oleh karena itu dengan adanya aplikasi analisis jawaban esai diharapkan dapat membantu pengajar untuk mengoreksi jawaban dari soal esai dengan cepat dan mudah.

Metode penilaian jawaban esai otomatis menggunakan beberapa metode seperti TF-IDF, *Rabin Karp*, *Cosine Similarity* telah banyak diusulkan oleh banyak peneliti. TF-IDF merupakan sebuah metode pembobotan teks dengan memanfaatkan index dari teks, metode ini menghitung kemunculan kata yang ada dalam sebuah teks (Bahri & Wajhillah, 2020), namun metode TF-IDF memiliki kelemahan yaitu jika suatu dokumen jawaban memiliki kata yang diulang-ulang maka nilai kemiripan semakin tinggi meskipun dokumen jawaban tidak sama dengan dokumen kunci jawaban.

Rabin karp merupakan algoritma pencarian string yang menggunakan fungsi hashing untuk membandingkan string yang dicari (m) dengan string yang dibandingkan (n) (Suryati, Wibisono, & Wihardi, 2018). Namun letak kalimat sangat mempengaruhi nilai similarity algoritma Rabin-Karp, dimana algoritma ini menghasilkan nilai similarity yang berbeda satu sama lainnya ketika diuji dengan Data-Uji yang dihapus/dipotong pada tempat yang berbeda (di awal, akhir dan acak) (Yusuf, Vivianie, Marsya, & Sofyan, 2019).

Cosine Similarity merupakan sebuah metode yang digunakan untuk menghitung tingkat kesamaan antara dua objek. Secara umum perhitungan metode ini didasarkan pada vector space similarity antara dua objek (misalkan dokumen 1 dan dokumen 2) yang dinyatakan dalam dua buah vector dengan menggunakan kata kunci dari sebuah dokumen sebagai ukuran (Cipto & Irfan,

2020). Aplikasi penilaian jawaban esai otomatis dapat membantu pekerjaan pengajar dalam melakukan penilaian ujian esai, menggunakan *machine learning* (Ramdani, Jamaludin, Yuniarti, Rahmani, & Hutahean, 2019). Adapun metode yang digunakan untuk permasalahan ini, yaitu ada *Synonym Recognition*, yaitu pendekatan semantik terhadap dokumen teks. Pendekatan ini memanfaatkan kesamaan makna dalam kata yang kemungkinan banyak terjadi (Yusuf, Fauzi, & Brata, 2018) dan *Cosine Similarity*, yaitu metode untuk menghitung kesamaan atau kemiripan dari dua buah dokumen (Fataruba, 2018).

Berdasarkan masalah tersebut penelitian ini mengusulkan sebuah perancangan sistem penilaian jawaban esai secara otomatis. Penilaian akan dilakukan secara komputerisasi dengan mendeteksi kata-kata yang mengandung sinonim dan memperhitungkan nilai kedekatan teks antara jawaban peserta didik dengan kunci jawaban. Perancangan system penilaian jawaban esai secara otomatis ini diterapkan pada soal esai Bahasa Indonesia, diharapkan mampu menyelesaikan permasalahan subjektivitas pengajar yang terjadi pada saat proses penilaian jawaban ujian esai secara manual. Hasil dari sistem penilaian ini adalah berupa grade atau nilai kedekatan antara jawaban dengan kunci jawaban.

Penelitian ini akan mengimplementasikan proses *Text Mining* menggunakan metode *Synonym Recognition* dan *Cosine Similarity*. Smrti, et al., (2019) mengatakan bahwa *text mining* dapat didefinisikan secara luas sebagai suatu proses menggali informasi dimana seorang user berinteraksi dengan sekumpulan dokumen menggunakan tools analisis yang merupakan komponen-komponen dalam data mining yang salah satunya adalah kategorisasi. *Text Mining* juga memiliki definisi yaitu menambang data yang berupa teks dimana sumber data biasanya didapatkan dari dokumen, dan tujuannya adalah mencari kata-kata yang dapat mewakili isi dari dokumen sehingga dapat dilakukan analisa implementasi *text mining* pada penilaian otomatis. *Text mining* bisa dianggap subjek riset yang tergolong baru. Text mining dapat memberikan solusi dari permasalahan seperti pemrosesan, pengorganisasian/ pengelompokkan dan menganalisa *unstructured text* dalam jumlah besar (Smrti, et al., 2019)

1.2.Rumusan Masalah

Berdasarkan uraian latar belakang diatas, maka rumusan masalah yang diajukan dalam penelitian adalah sebagai berikut “*Bagaimana merancang sistem penilaian jawaban esai otomatis?*”

1.3.Tujuan Penelitian

Tujuan dari penelitian ini adalah merancang sistem penilaian jawaban esai otomatis.

1.4.Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah :

1. Bagi Pengajar

Aplikasi ini dapat digunakan untuk pengajar untuk membuat soal esai secara online dan juga dapat digunakan untuk mengoreksi jawaban masing-masing siswa secara otomatis dengan waktu yang relative singkat.

2. Bagi Universitas

Penelitian ini diharapkan dapat memberikan kontribusi penambahan ilmu pengetahuan, khususnya bagi Manajemen Sumber daya Manusia serta menjadi bahan bacaan di perpustakaan universitas dan dapat memberikan referensi bagi mahasiswa lain.

3. Bagi Pembaca

Pengguna mendapatkan informasi mengenai penilaian jawaban esai otomatis menerapkan metode *Synonym Recognition* dan *Cosine Similarity*

4. Bagi Peneliti

Manfaat yang didapat bagi peneliti adalah mengembangkan ilmu yang didapat dari perkuliahan serta ilmu dan pengetahuan baru yang tidak didapat dari perkuliahan.

1.5. Batasan Masalah

Pembatasan suatu masalah digunakan untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah agar penelitian lebih terarah, maka penulis memberikan batasan masalah sebagai berikut :

1. Penelitian ini hanya berfokus pada perancangan system penilaian jawaban esai otomatis.
2. Perancangan system penilaian jawaban esai otomatis yang dibuat untuk system berbasis website.
3. Penilaian jawaban esai otomatis bisa digunakan untuk jawaban berbasis teks Bahasa Indonesia.
4. Metode yang digunakan adalah *Synonym Recognition* dan *Cosine Similarity*

1.6. Sistematika Penulisan

Dalam memudahkan penulisan proposal skripsi ini, penulis menyusun proposal dengan suatu sistematika penulisan yang terdiri dari sebagai berikut:

BAB 1: Pendahuluan

Pada bab ini merupakan bab dimana penulis menjelaskan pendahuluan dalam penelitian ini, penulis membuat latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian serta data-data yang dibutuhkan.

BAB 2: Landasan Teori

Pada bab ini merupakan bab dimana penulis menjelaskan mengenai teori-teori singkat tentang hal-hal yang berhubungan dengan judul penelitian antara lain adalah penelitian terdahulu yang dapat dijadikan rekomendasi, sistem, penilaian, metode *synonym recognition*, metode *cosine similarity*, *Entity Relational Diagram*, *Data Flow Diagram*, *flowchart*, dan basis data yang digunakan penulis.

BAB 3: Metode Penelitian

Pada bab ini merupakan bab dimana penulis menuliskan menjelaskan mengenai waktu dan tempat penelitian, pengumpulan data (observasi, survey, wawancara), perancangan sistem(flowchart, DFD, dan ERD), serta alat atau bahan lain yang dibutuhkan demi menunjang perancangan sistem

BAB II

TINJAUAN PUSTAKA

2.1. Landasan Teori

2.1.1. Sistem

Sistem bisa ditafsirkan sebagai kesatuan elemen yang memiliki keterkaitan. Beberapa elemen dapat digabung menjadi suatu unit, kelompok atau komponen sistem dengan fungsi tertentu. Komponen sistem ini bisa dilihat, dianggap atau memang dirancang untuk berfungsi mandiri sebagai modul system (Ariantini, Lumenta, & Jacobus, 2016).

2.1.2. Penilaian

Menurut Permendikbud No.23 Tahun 2016, penilaian adalah proses pengumpulan dan pengolahan informasi untuk mengukur pencapaian hasil belajar peserta didik. Kegiatan Penilaian memerlukan instrumen penilaian dan teknik penilaian. Penilaian tidak hanya difokuskan pada hasil belajar tetapi juga pada proses belajar. Menurut Salim (2017) penilaian merupakan alih bahasa dari istilah assesment bukan merupakan evaluasi karena sering disalah artikan.

2.1.3. Text Mining

Menurut Joane, Sinsuw, & Jacobus (2017) *text mining* merupakan salah satu bidang khusus dari data mining. *Text mining* lebih berfokus pada teknik dan metodologi dalam lingkup temu kembali sistem informasi. Text mining berfungsi untuk mendapatkan informasi dari sebuah dokumen teks. dalam penggalian informasi terdapat tahapan *Text Preprocessing*. *Text preprocessing* merupakan tahap awal dari Text mining dimana data text akan dibersihkan sehingga text menjadi lebih terstruktur sebelum masuk ketahap berikutnya untuk diolah lebih lanjut. Metode Text Mining diantaranya *Case Folding* dan *Tokenizing, Filtering, Filtering*, dan *Stemming* (Bastian & Sujadi, 2018).

a. Case folding

Kurniawati & Pradnya (2020) mengatakan *Case folding* adalah proses mengkonversi semua huruf yang ada di dalam dokumen menjadi huruf kecil

atau lower case. Case folding digunakan untuk mempermudah pencarian. Dalam case folding huruf yang diterima adalah dari “a” sampai “z”. karakter selain huruf tersebut dihilangkan dan dianggap sebagai delimiter.

b. *Tokenizing*

Proses tokenizing adalah proses pemotongan string masukan berdasarkan tiap kata yang terdapat pada kalimat. Setiap huruf input akan diubah menjadi huruf kecil. Semua tanda baca dan tanda hubung akan dihapuskan, termasuk semua karakter selain huruf alfabet (Yusuf, Fauzi, & Brata, 2018).

c. *Filtering*

Pada tahap filtering dimana akan dilakukan penghapusan stopwords untuk mengeliminasi kata-kata yang tidak deskriptif dan yang sering muncul (Joane, Sinsuw, & Jacobus, 2017)

d. *Stemming*

Stemming adalah proses untuk menggabungkan atau memecahkan setiap varian-varian suatu kata menjadi kata dasar. Proses stemming pada kata Bahasa Indonesia berbeda dengan stemming bahasa Inggris. Proses stemming pada kata bahasa inggris adalah proses untuk mengeliminasi sufiks pada kata, sementara proses stemming bahasa Indonesia adalah proses untuk mengeliminasi sufiks, prefiks, dan konfik (Salim, 2017)

2.1.4. *Synonym Recognition*

Synonym Recognition atau pengenalan kata bersinonim adalah teknik yang digunakan untuk mengenali kata dengan penulisan berbeda namun memiliki makna yang sama. Pada proses pencocokan sinonim, kaitannya antara isi dokumen dengan kamus sinonim yang terdapat pada database. Jika kata tidak tersedia pada kamus sinonim, maka tidak akan melakukan proses synonym recognition. Seluruh isi dokumen akan discan dan dicocokkan dengan kata yang ada dalam kamus, kemudian akan diubah berdasarkan isi kamus sinonim tersebut (Yusuf, Fauzi, & Brata, 2018).

2.1.5. *Cosine Similarity*

Cosine Similarity adalah ukuran kesamaan antara dua buah vektor dalam sebuah ruang dimensi yang didapat dari nilai cosinus sudut dari perkalian dua buah vektor yang dibandingkan karena cosinus dari 0° adalah 1 dan kurang dari 1 untuk nilai sudut yang lain, maka nilai similarity dari dua buah vektor dikatakan mirip ketika nilai dari *Cosine Similarity* adalah 1 (Ariantini, Lumenta, & Jacobus, 2016). Menurut Fataruba (2018) *Cosine similarity* digunakan dalam ruang positif, dimana hasilnya dibatasi antara nilai 0 dan 1. Kalau nilainya 0 maka dokumen tersebut dikatakan mirip jika hasilnya 1 maka nilai tersebut dikatakan tidak mirip. Menurut Zizka, Darena, & Svoboda (2019) *cosine similarity* bias hanya dihitung sebagai nilai rentang cosine similarity dari 0 hingga 1 sebagai sudut antara vector dapat berada di antara 0 sampai 90 derajat, semakin mirip dokumennya maka semakin kecil sudutnya di antara vector.

Berikut adalah rumus *cosine similarity* :

$$similarity = \cos(\emptyset) = \frac{A.B}{||A||.||B||} \dots\dots\dots \text{Rumus 2.1}$$

$$\cos(\emptyset) = \frac{\sum_{i=0}^n A.B}{\sqrt{\sum_{i=0}^n (A_i)^2} \cdot \sqrt{\sum_{i=0}^n (B_i)^2}} \dots\dots\dots \text{Rumus 2.2}$$

Keterangan :

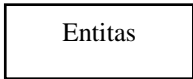
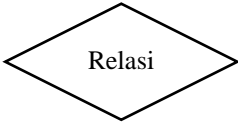


- A : Vector dokumen A
- B : Vector dokumen B
- A_i : Bobot kata I dalam blok A_i
- B_i : Bobot kata I dalam blok B_i
- i : Jumlah kata dalam vector
- n : Jumlah vector

2.1.6. *Entity Relational Diagram (ERD)*

ERD atau *Entity Relational Diagram* adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas- entitas dan menentukan hubungan antar entitas. Dengan penggunaan Entity Relationship Diagram dalam bentuk gambar dapat mempermudah dalam menganalisa kebutuhan suatu basis data dalam sebuah sistem yang akan dibangun dengan lebih cepat dan mudah (Rifai & Yuniar, 2019). Sedangkan menurut Nugraha &

Pramukasari (2017) *Entity Relational Diagram* adalah salah satu metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual untuk jenis atau model data sematik system.

Tabel 2.1 simbol ERD

Notasi	Keterangan
	Entitas , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai
	Relasi , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda
	Atribut , berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key diberi garis bawah)
	Garis , sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut

(Sumber : Irawan, 2020)

2.1.7. Diagram Konteks

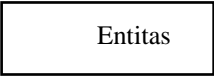
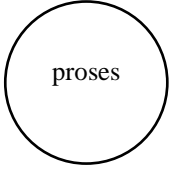
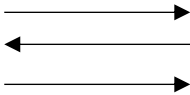

Menurut Noviantoro (2020) diagram konteks adalah diagram tingkat tinggi dari Diagram Alir Data yang merupakan gambaran global dari sistem informasi yang menggambarkan aliran-aliran data ke dalam maupun keluar sistem dan merupakan alat yang digunakan untuk melihat batasan antara sistem dengan entitas eksternal.

2.1.8. Data Flow Diagram (DFD)

Menurut Purnamasari, Pudjiantoro, & Nursantika (2017) *Data Flow Diagram* atau DFD adalah alat pembuatan model yang memungkinkan sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data. Sedangkan menurut Muslihudin & Oktafianto (2016) *Data Flow Diagram* (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan ke mana tujuan data yang keluar dari

sistem, di mana data tersimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data tersimpan dan proses yang dikenakan pada data tersebut.


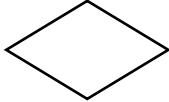
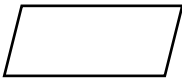
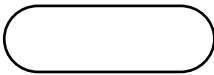

Tabel 2.2 simbol *Data Flow Diagram*

Simbol	Keterangan
	Entitas eksternal dapat berupa orang/unit terkait berinteraksi dengan system tetapi di luar system
	Orang/unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi
	Aliran data dengan arah khusus dari sumber ke tujuan
	Penyimpanan data atau tempat data dilihat oleh proses

2.1.9. Flowchart

Menurut Sukarsa (2019) bagan alir program (flowchart) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Sedangkan menurut Ramlah (2017) flowchart adalah media pembelajaran yang berupa bagan yang disusun secara sistematis dengan menggunakan simbol yang mudah dipahami oleh peserta didik sehingga mudah memahami materi yang diajarkan. Flowchart berisi serangkaian symbol dan hubungan diantara mereka, dapat melukiskan sejumlah kegiatan pengolahan informasi yang berkisar dari konfigurasi computer sampai tahap-tahap terinci sebuah program. Sebuah diagram dapat melukiskan sejumlah pengolahan informasi yang berkisar dari konfigurasi computer sampai tahap-tahap terinci sebuah program.

Tabel 2.3 simbol *flowchart*

Simbol atau Gambar	Penggunaan	Keterangan
	Proses atau langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir
	Titik keputusan	Proses atau langkah di mana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selal ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Masukkan atau keluaran data	Digunakan untuk mewakili data masuk atau data keluar
	Terminasi	Menunjukkan awal atau akhir sebuah proses
	Garis alir	Menunjukkan arah aliran proses atau algoritma

(Sumber : Irawan, 2020)

2.2.Kajian Penelitian

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu yang pernah dilakukan sebagai bahan perbandingan dan kajian. Adapun hasil-hasil penelitian yang dijadikan perbandingan tidak terlepas dari topik penelitian yaitu mengenai sistem penilaian jawaban esai otomatis menggunakan metode *synonym recognition* dan *cosine similarity*.

Berdasarkan penelitian yang pernah dilakukan oleh Rinarta (2017) penilaian hasil ujian secara manual terkadang memiliki kelemahan yaitu terdapat penilaian yang berbeda terhadap peserta-peserta ajar dengan jawaban yang hampir sama. Pemberian nilai otomatis dilakukan dengan menggunakan stemming Porter yang dikombinasikan dengan *Cosine similarity*. Algoritma Porter digunakan untuk mencari kata dasar dari kata-kata yang digunakan dalam kunci jawaban. Hasil dari ekstraksi kata dasar pada kunci jawaban dan jawaban siswa akan dibandingkan dengan menggunakan *Cosine similarity*. Pemberian nilai otomatis dilakukan dengan menggunakan *stemming* Porter yang dikombinasikan dengan *Cosine similarity*. Hasil dari ekstraksi kata dasar pada kunci jawaban dan jawaban siswa akan dibandingkan dengan menggunakan *Cosine similarity*.

Penelitian Fataruba (2018) menjelaskan bahwa pengajar memerlukan waktu yang banyak untuk memeriksa jawaban essay, semakin banyak jumlah ujian dan banyaknya jumlah pelajar yang mengikuti ujian, maka semakin banyak jumlah ujian yang dikoreksi oleh pengajar . Penelitian tersebut menggunakan metode *Cosine Similarity* dan mendapatkan hasil akurasi system dalam memberika nilai siswa mencapai 80% berdasarkan data yang dihitung menggunakan *confision matrix*.

Menurut Salim (2017) yang menjadi masalah pada soal berbentuk esai adalah siswa harus menjawab dengan bentuk kalimat. Dengan begitu jawaban dapat bervariasi berdasarkan pemikiran mereka. Salah satu kelemahan jawaban dalam bentuk esai adalah sulitnya menilai jawaban yang membutuhkan banyak waktu. Penelitian yang dilakukan Salim diperoleh bahwa tidak terdapat beda yang besar antara mean nilai siswa koreksi otomatis metode *cosine similarity* dengan koreksi manual yaitu sebesar 1,6 dengan mean koreksi otomatis sebagai variabel lebih besar.

Proses *preprocessing* merupakan tahap awal sebelum melakukan pencocokan dua buah dokumen menggunakan metode *synonym recognition* dan *cosine similarity*, data yang digunakan merupakan jawaban siswa dan kunci jawaban. Tahap *preprocessing* merupakan bagian dari tahap di dalam Text Mining. Menurut Jo (2019) pada buku nya menjelaskan bahwa *text mining* didefinisikan sebagai proses mengekstraksi pengetahuan implisit dari data tekstual. Sedangkan *preprocessing* merupakan proses mengubah teks menjadi daftar kata-kata.

Menurut Bahri & Wajhillah (2020) pada penilitiannya tahap *preprocessing* menghilangkan tanda baca dan merubah kalimat menjadi *lower case (Cleaning)*, Pada tahap ini, corpus jawaban harus dibersihkan dengan cara menghilangkan tanda baca. Tujuannya adalah agar corpus jawaban yang akan dijadikan sebagai acuan terbebas dari kata-kata atau simbol lain yang tidak memiliki manfaat substansi dari kalimat. Masih pada tahap ini juga, dilakukan prose *cleaning* corpus jawaban berupa penyetaraan jenis huruf menjadi huruf kecil semua.

Sejalan dengan peneilitian Bahri & Wajhillah (2020), Arnawa (2019) pada penilitiannya dalam melakukan preprocessing, dimana tahap pertama dilakukan tokenizing yaitu tahap pemotongan string input berdasarkan tiap kata yang menyusunnya. Tahap kedua dilakukan filtering yaitu mengambil kata – kata penting dari hasil tokenizing. Tahap selanjutnya dilakukan stemming dimana tahap ini dilakukan untuk menemukan kata dasar dari sebuah kata.

Jo (2019) pada bukunya menjelaskan bahwa ada tiga langkah dasar dalam *preprocessing*. Langkah pertama adalah tokenisasi yang merupakan proses segmentasi teks dengan spasi atau tanda baca menjadi token. Langkah kedua adalah stemming yang merupakan proses mengubah setiap token menjadi bentuk rootnya sendiri menggunakan aturan tata bahasa. Langkah terakhir adalah penghapusan kata-kata yang merupakan proses menghapus kata-kata gramatikal seperti artikel, konjungsi, dan preposisi. Tokenisasi adalah prasyarat untuk dua langkah berikutnya, tetapi stemming dan penghapusan stop-word dapat ditukar satu sama lain, tergantung pada situasi yang diberikan.

Penelitian yang dilakukan oleh Ariantini, Lumenta, & Jacobus dengan judul “Pengukuran Kemiripan Dokumen Teks Bahasa Indonesia Menggunakan Metode Cosine Similarity” pengukuran nilai kemiripan dua buah dokumen menggunakan metode *cosine similarity*, yang dimulai dari *upload file* dokumen tugas mahasiswa kemudian disimpan kedalam basis data kemudian sistem preproses teks untuk menghapus tanda baca dalam dokumen, menghilangkan titik, koma, dan sebagainya kemudian sistem menghitung kemiripan TF yaitu term frekuensi, adalah proses untuk mengambil setiap kata dari dokumen. Kemudian nilai dari TF dimasukkan kedalam rumus *cosine similarity* untuk menghitung nilai kemiripan adalah proses untuk mencari kemiripan antar dokumen.

Penelitian yang berjudul “Sistem Temu Kembali Informasi Pasal-Pasal KUHP (Kitab Undang- Undang Hukum Pidana) Berbasis Android Menggunakan Metode Synonym Recognition dan Cosine Similarity” oleh Yusuf, Fauzi, & Brata (2018). Penelitian ini menggunakan *synonym recognition* untuk mengenali kata dengan penulisan berbeda namun memiliki makna yang sama. Pada proses

pencocokan sinonim seluruh isi dokumen akan discan dan dicocokkan dengan kata yang ada dalam kamus, kemudian akan diubah berdasarkan isi kamus sinonim tersebut. Adapun contoh kinerja sistem untuk synonym recognition dengan kalimat “komputer dipakai untuk mengolah data”. Dan hasil pendeteksian sinonimnya adalah sebagai berikut : **Dipakai=Digunakan**. Kata “dipakai” akan diganti dengan kata “digunakan”. Sehingga kalimat yang semula “komputer dipakai untuk mengolah data” diganti dengan kata “komputer digunakan untuk mengolah data”.

Penelitian selanjutnya dengan judul “*A Comparative Analysis of Euclidean Distance and Cosine Similarity Measure for Automated Essay-Type Grading*” oleh Oduntan, Adeyanju, Falohun, & Obe (2018) data diambil dari data jawaban siswa dan data jawaban penguji dengan 50 jawaban siswa yang di periksa dengan format *softcopy* txt. Penelitian tersebut menghasilkan ukuran cosine similarity memiliki korelasi positif yang tinggi ketika skor mesin yang dihasilkan dibandingkan dengan penguji manusia.

Penelitian yang dilakukan oleh Rymmai & JS (2017) dengan judul “*Book Recommendation using Cosine Similarity*” menggunakan dua buah rekomendasi pencarian rekomendasi buku, yang pertama berdasarkan *search query*, yang kedua berdasarkan *book search*. Masing-masing memiliki prosedur yang sama yaitu melakukan *text mining* pada skenario pertama dan mendapatkan hasil *tokenization*. Hasil *tokenization* tersebut membantu menunjukkan keberadaan istilah dalam dokumen tertentu, dan dilanjutkan menormalisasi term, yang membantu memastikan bahwa frekuensi term dan ukuran dokumen tidak mempengaruhi bobot term. Setelah proses tersebut selesai dilanjutkan melakukan perhitungan cosine similarity. Penelitian ini mendapatkan hasil bahwa buku dengan kemiripan cosine tertinggi yang ditemukan akan direkomendasikan kepada pengguna.

Penelitian yang dilakukan oleh Gunawan, Sembiring & Budiman (2018) dengan judul “*The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents*” impelentasi untuk mencari relevansi kedekatan antar dokumen dimulai dengan *preprocessing text*. *Preprocessing text* atau pemrosesan awal teks dimulai dengan penghapusan tanda baca. Kemudian

menerapkan case folding untuk mengatur semua huruf menjadi huruf kecil. Setelah proses case folding selesai, selanjutnya membuat token teks menjadi token terpisah. Beberapa token ini akan dihapus jika istilah tersebut tercantum dalam daftar *stopword*. Setelah token bersih tanpa ada istilah yang tanpa arti, maka langkah selanjutnya adalah *stemming*. Proses *stemming* dilakukan untuk mendapatkan *root word* yang digunakan untuk menghitung relevansi teks antar dokumen. Hasil dari proses *stemming* adalah ekstraksi kata kunci dari dokumen. Kata kunci ditulis dalam *root word* dan dicari banyaknya kata kunci dalam dokumen, dan selanjutnya dilakukan pembobotan kata kunci.

Relevansi teks dihitung dengan menggunakan fungsi *cosine similarity*. Perhitungannya melibatkan kata kunci yang sama dari kedua dokumen untuk mendapatkan bobot. Persamaan cosine similarity akan menghasilkan nilai antara 0 sampai 1. Jika nilainya mendekati 1, berarti dua buah dokumen tersebut sangat relevan, namun sebaliknya jika nilainya mendekati 0, maka dua buah dokumen tersebut tidak relevan.

Tabel 2.4 Daftar Penelitian Terdahulu

Peneliti	Obyek	Metode	Tools	Hasil
Komang Rinartha (2017)	Sekolah	<i>Text Stemming</i> dan <i>Cosine Similarity</i>		Pemodelan penilaian essay otomatis <i>realtime</i>
Firmansyah Fataruba (2018)	SMA Negeri 1 Ambon	<i>Cosine Similarity</i>	Website	Akurasi system dalam memberikan nilai mencapai 80% yang dihitung dengan <i>Confision Matrix</i>
Mohammad Agus Salim (2017)	SMKN Buduran	3 Algoritma Nazief dan Adriani dan <i>Cosine Similarity</i>	HTML, PHP, jQuery, Bootstrap, MySQL	Mean nilai siswa menggunakan Koreksi otomatis menggunakan cosine similarity dengan koreksi manual tidak

Peneliti	Obyek	Metode	Tools	Hasil
				terdapat perbedaan, yaitu sebesar 1,6 dengan mean koreksi otomatis sebagai variable besar
Saeful Bahri & Rusda Wajhillah (2020)		Rabin Karp & TF-IDF	Website	penerapan TF-IDF pada proses pengelompokan kalimat dan hitung tingkat kemiripan dengan algoritma rabinkarp mengalami rata-rata peningkatan sebesar 11,81%.
Ida Bagus Ketut Surya Arnawa (2019)	STIKOM BALI	<i>Rabin Karp</i>		Algoritma Rabin Karp ini dapat digunakan untuk mengkoreksi otomatis jawaban ujian essay dengan mencocokkan kunci jawaban dengan jawaban mahasiswa
Dewa Ayu Rai Ariantini, Arie S. M. Lumenta, Agustinus Jacobus (2016)		<i>Cosine Similarity</i>	Website	Penggunaan data dummy pada sistem deteksi kemiripan dokumen sudah berhasil diterapkan dan sudah bisa digunakan untuk mengukur tingkat kemiripan dokumen

Peneliti	Obyek	Metode	Tools	Hasil
Safier Yusuf , M. Ali Fauzi , Komang Candra Brata (2018)	Kitab Undang- undang Hukum Pidana (KUHP)	<i>Synonym</i> <i>Recognition dan</i> <i>Cosine</i> <i>Similarity</i>	Android, SQLite	Tingkat precision, recall dan f.measure mempunyai nilai yang optimal pada threshold 15 dengan nilai precision 0,393, recall 0,594 dan f.measure sebesar 0,404.
Odunayo E. Oduntan, Ibrahim A. Adeyanju, Adeleye S. Falohun, Olumide O. Obe (2018)	Matrix Laboratory 8.1	<i>Euclidean</i> dan <i>Cosine</i> <i>Similarity</i>		Hasil yang didapatkan bahwa <i>Cosine similarity</i> kedekatannya lebih tinggi daripada <i>Euclidean Distance</i>
Rofeca Giri Rymmai dan Saleema JS (2017)		<i>Cosine</i> <i>Similarity</i>		Penelitian ini mendapatkan hasil bahwa buku dengan kemiripan cosine tertinggi yang ditemukan akan direkomendasikan kepada pengguna
D. Gunawan, C.A Sembiring, M.A Budiman (2017)	Dokumen	<i>Cosine</i> <i>similarity</i>		Dalam penelitian ini relevansi teks dihitung dengan menggunakan cosine similarity, dimana nilainya yang mendekati 1 dianggap lebih

Peneliti	Obyek	Metode	Tools	Hasil
				relevan, begitu juga sebaliknya.

BAB III

METODE PENELITIAN

3.1. Ruang Lingkup Penelitian untuk Rancangan

Berdasarkan latar belakang diatas, maka penulis membatasi permasalahan dalam penelitian ini sesuai dengan judul yang diajukan, penelitian ini hanya berkaitan dengan perancangan system penilaian jawaban esai otomatis. Penelitian ini dilaksanakan di SMAN Alam Al-Ghifari yang berlokasi di JL. Sumba GG III, Karangtengah, Kec. Sananwetan, Kota Blitar yang dimulai pada bulan Februari - Juni 2021.

3.2. Metode Penelitian Rancangan

Metode penelitian digunakan sebagai suatu pedoman dalam menentukan langkah, gambaran, prosedur, waktu dan tempat pengambilan data. Penelitian ini menggunakan metode penelitian deskriptif kualitatif. Penelitian deskriptif kualitatif merupakan sebuah metode penelitian yang memanfaatkan data kualitatif dan dijabarkan secara deskriptif. Metode penelitian deskriptif kualitatif ini didukung oleh metode pengembangan perangkat lunak dengan model *prototype*. Adapun metode penelitian yang rencana penulis terapkan di Perancangan Sistem Penilaian Jawaban Esai Otomatis adalah sebagai berikut :

3.2.1. Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Data bisa memiliki berbagai wujud mulai dari gambar, suara, huruf, angka, bahasa, symbol bahkan keadaan. Teknik Pengumpulan data yang dilakukan dalam penelitian di antaranya sebagai berikut :

3.2.1.1. Wawancara

Metode ini dilakukan untuk mendapatkan informasi kebutuhan sistem yang diperlukan oleh pengguna. Dalam pelaksanaannya dilakukan dengan tanya jawab secara langsung di beberapa kesempatan.

Tabel 3.1 Tabel daftar pertanyaan wawancara

No	Pertanyaan	Jawaban
1	Bagaimana proses koreksi dan penilaian jawaban pada pelajaran yang Bapak/Ibu ajar? Khususnya jawaban dengan tipe esai?	Untuk penilaian masih manual
2	Apakah pernah terjadi kendala ketika proses koreksi jawaban?	Ada kendala selama proses koreksi jawaban esai
3	Jika ada, kendala apa itu? Bisa dijelaskan?	Koreksi jawaban siswa butuh waktu yang lama
4	Bagaimana cara Bapak/Ibu menilai jawaban esai?	Untuk penilaian soal esai, pakai system poin, semakin sesuai kunci jawaban poinnya akan maksimal, namun jika tidak sesuai maka poinnya juga tidak akan maksimal
6	Berapa lama waktu Bapak/Ibu perlukan dalam mengoreksi jawaban setiap satu kelas?	Perkelas biasanya butuh 2-3 hari untuk mengoreksi sampai selesai
7	Data apa saja yang diperlukan untuk menilai jawaban dengan tipe esai?	Data yang digunakan yang jelas soal ujian, jawaban siswa, dan kunci jawaban yang sudah di siapkan terlebih dahulu

3.2.1.2. Observasi

Metode observasi adalah suatu metode yang digunakan dengan cara pengamatan dan pencatatan data secara sistematis terhadap fenomena-fenomena yang diselidiki.

3.2.1.3. Kepustakaan

Metode ini dilakukan dengan mempelajari buku-buku atau jurnal serta pemikiran-pemikiran ahli. Informasi yang didapat dari metode ini dapat digunakan untuk mengetahui bagaimana cara merancang system penilaian jawaban esai otomatis.

3.3. Metode Analisis Sintesis

Menurut Cipto & Irfan (2020) pada penelitiannya, ujian merupakan salah satu cara untuk mengevaluasi proses belajar. Dalam dunia Pendidikan, ujian dimaksudkan untuk mengukur taraf pencapaian suatu tujuan pengajaran oleh siswa atau siswi sebagai peserta didik. Namun proses koreksi ujian berbasis esai memerlukan banyak waktu bagi pengajar. Semakin banyak jumlah ujian dan banyaknya jumlah pelajar yang mengikuti ujian, maka semakin banyak jumlah ujian yang dikoreksi oleh pengajar (Fataruba, 2018). Sejumlah 90% pengajar

menyatakan bahwa kelelahan akan mengakibatkan inkonsistensi dalam pemberian skor (*human error*) (Ramdani, Jamaludin, Yuniarti, Rahmani, & Hutahean, 2019).

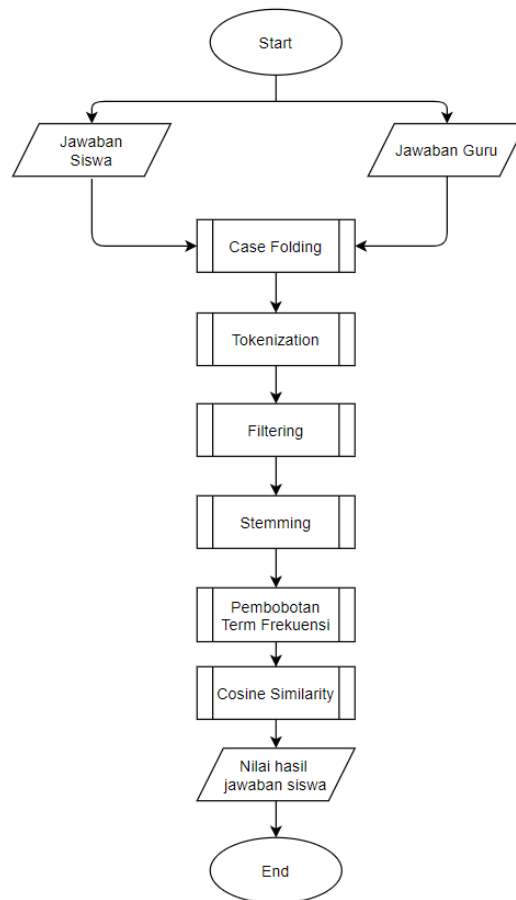
Objektifitas dan konsistensi dari instruktur sangat penting ketika harus memutuskan sebuah jawaban benar atau salah karena akan berpengaruh terhadap hasil penilaian, sehingga dibutuhkan sebuah solusi yang mampu mempermudah penerapan metode penilaian jawaban berupa uraian singkat dengan tidak mengesampingkan objektifitas penilaian. Hasil dari jawaban essay bukan hanya benar atau salah tapi juga hampir benar. Sehingga skor nilai untuk jawaban essay bisa beragam (Bastian & Sujadi, 2018). Dosen dituntut untuk konsisten dalam memberikan skor nilai jawaban essay semua mahasiswa. Selain konsistensi, Dosen juga memerlukan waktu lebih banyak untuk pemeriksaan jawaban ujian essay (Fauzi, Utomo, Setiawan, & Pramukantoro, 2016).

Dengan permasalahan tersebut diatas penus mempunyai gagasan untuk membuat sebuah Perancangan Sistem Aplikasi yang mampu menilai jawaban esai secara otomatis yang diharapkan mampu menyelesaikan permasalahan subjektifitas pengajar yang terjadi pada saat proses penilaian jawaban ujian esai secara manual.

3.4. Skema Metode Perancangan

3.4.1. *Flowchart* Sistem

Berdasarkan hasil analisa, maka perlunya perancangan system penilaian jawaban esai otomatis sehingga dapat membantu pengajar/guru dalam menilai jawaban esai secara otomatis. Berikut Flowchart sistem yang digunakan dalam sistem ditampilkan pada gambar 3.1:



Gambar 3.1 *flowchart* system

Keterangan :

- a. Mulai
- b. *Input* jawaban siswa dan kunci jawaban.

Dimulai dari dua inputan dari buah dokumen, sebuah kunci jawaban dimasukkan ke dalam system oleh guru bersamaan ketika guru memasukkan soal-soal yang akan di berikan kepada siswa. Di sisi lain ketika soal sudah di distribusikan ke siswa maka siswa menginputkan jawabannya di masing-masing soal. Berikut contoh dari kunci jawaban yang di input guru dan jawaban yang di input siswa :

Tabel 3.2 Table data *training*

Soal	Kunci jawaban	Jawaban siswa
Apakah fungsi dari motherboard?	Menghubungkan seluruh komponen penyusun sebuah komputer	Menghubungkan komponen penyusun yang ada di dalam sebuah komputer

c. Case Folding

Kemudian dari dua buah dokumen tersebut masing akan masuk ke tahap *preprocessing* yang pertama yaitu case folding, maka hasil dari tahap case folding akan seperti pada table 3. :

Tabel 3.3 Table hasil *case folding*

Kunci jawaban			Jawaban siswa		
menghubungkan	seluruh	komponen	menghubungkan	komponen	komponen
penyusun	sebuah	komputer	penyusun	yang ada di	dalam sebuah komputer

d. Tokenization

Ketika proses case folding telah selesai maka selanjutnya masuk ke tahap *preprocessing* yang kedua yaitu *tokenizing*. Proses ini memotong string input berdasarkan tiap kata yang menyusunnya. Hasil dari tahap *tokenization* akan seperti table 3. :

Tabel 3.4 Table hasil *tokenization*

Kunci Jawaban	Jawaban siswa
menghubungkan	menghubungkan
seluruh	komponen
komponen	komponen
penyusun	penyusun
sebuah	yang
komputer	ada
	di
	dalam
	sebuah
	computer

e. Filtering

Selanjutnya dokumen yang sudah selesai di tahap *tokenization* maka akan masuk ke dalam tahap filtering. Tahap ini menghilangkan kata-kata yang kurang penting (*stopword*). Contoh *stopword* adalah “yang”, “dan”, “di”, “dari”, dan lain-lain. Berikut hasil dari tahap *filtering* pada table 3. :

Tabel 3.5 Table hasil *filtering*

Kunci Jawaban	Jawaban siswa
menghubungkan	menghubungkan
komponen	komponen
penyusun	komponen
komputer	penyusun
	komputer

f. *Stemming*

Jika proses filtering sudah selesai maka dokumen tersebut akan masuk ke tahap yang terakhir dari *preprocessing* yaitu tahap *stemming*. Tahap ini mengubah suatu kata/term di dalam dokumen ke dalam bentuk kata dasar, berikut hasil dari tahap *stemming* pada table 3. :

Tabel 3.6 Table hasil *stemming*

Kunci Jawaban	Jawaban siswa
hubung	hubung
komponen	komponen
susun	komponen
komputer	susun
	komputer

g. *Synonym Recognition*

Tahap preprocessing sudah selesai selanjutnya dari hasil *stemming* maka setiap term atau kata akan masuk ke dalam tahap *synonym recognition*. Dokumen akan discan setipa kata, kata tersebut di scan ke dalam database sinonim jika memiliki sinonim maka kata atau term tersebut diubah ke dalam bentuk sinonimnya. Namun jika tidak ada di dalam database sinonim maka kata tersebut tidak akan diganti. Hasil dari proses synonym recognition seperti table 3. :

Tabel 3.7 Tabel hasil *synonym recognition*

Kunci Jawaban	Jawaban siswa
hubung	hubung
anggota	anggota
susun	anggota
komputer	susun
	komputer

h. *Cosine Similarity*

Selanjutnya masuk ke dalam tahap perhitungan kemiripan dokumen dengan cosine similarity. Langkah pertama yang dilakukan adalah menghitung term frekuensi dari masing-masing dokumen. Hasil dari term frekuensi tersebut maka akan dimasukkan ke dalam rumus *cosine similarity*. Rumus cosine similarity bisa dilihat di rumus 2.1 dan 2.2. Berikut hasil dari term frekuensi masing-masing dokumen :

Tabel 3.8 Table hasil term frekuensi

Term	Term Dokumen Kunci Jawaban / Term A	Term Dokumen Jawaban Siswa / Term B
Hubung	1	1
Anggota	1	2
Susun	1	1
komputer	1	1

Term Frequency adalah untuk menghitung kemunculan kata kemudian menyimpannya kedalam basis data tabel indeks. Vektor A dan Vektor B mewakili setiap term A dan term B untuk melihat berapa banyak nilai yang diperoleh dalam setiap kata di dalam dokumen. Setelah itu setiap term dalam vector akan dihitung menggunakan cosine similarity.

Dari hasil term frekuensi dapat kita ambil nilai vector sebagai berikut :

Vektor A : (1, 1, 1, 1)

Vektor B : (1, 2, 1, 1)

Kemudian dari masing-masing vector tersebut dimasukkan ke dalam rumus cosine similarity seperti berikut :

$$\text{Similarity}(A, B) = \frac{\sum_{i=0}^n A_i \cdot B_i}{\sqrt{\sum_{i=0}^n (A_i)^2} \cdot \sqrt{\sum_{i=0}^n (B_i)^2}} \dots \dots \dots \text{Rumus 3.1}$$

$$A \cdot B = \sum_{i=0}^n A_i \cdot B_i = (1 * 1) + (1 * 2) + (1 * 1) + (1 * 1) \dots \dots \dots \text{Rumus 3.2}$$

$$|A| = \sqrt{\sum_{i=0}^n (A_i)^2} = \sqrt{1^2 + 1^2 + 1^2 + 1^2} \dots \dots \dots \text{Rumus 3.3}$$

$$|B| = \sqrt{\sum_{i=0}^n (B_i)^2} = \sqrt{1^2 + 2^2 + 1^2 + 1^2} \dots \dots \dots \text{Rumus 3.4}$$

$$= \frac{5}{\sqrt{4} \cdot \sqrt{7}} \dots \dots \dots \text{Rumus 3.5}$$

$$= \frac{5}{2 * 2.645} \dots \dots \dots \text{Rumus 3.6}$$

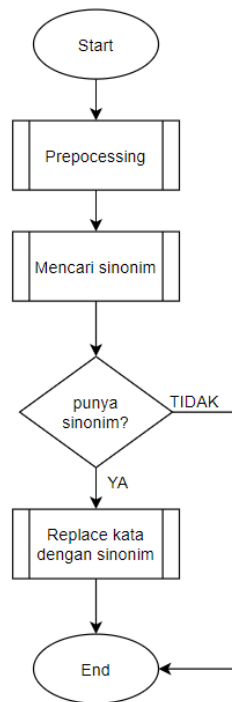
$$= 0.945$$

Hasil dari perhitungan kesamaan dua buah dokumen mendapatkan nilai 0,945.

i. Selesai.

3.4.2. Flowchart *Synonym Recognition*

Untuk menggambarkan proses dari metode synonym recognition, maka penulis menambahkan flowchart Synonym Recognition seperti gambar 3.2 :



Gambar 3.2 Flowchart *Synonym Recognition*

Keterangan :

- a. Mulai
- b. *Preprocessing*

Dimulai dari tahap preprocessing sebuah dokumen, dokumen akan masuk kedalam tahap preprocessing terlebih dahulu. Tahap preprocessing meliputi case folding, tokenization, filtering, dan stemming. Penjelasan tahap *preprocessing* sudah dijelaskan pada tahap sebelumnya. Berikut tabel memperlihatkan dua buah dokumen sebelum dan sesudah dilakukan *preprocessing*.

Tabel 3.9 Tabel dokumen sebelum *preprocessing*

Dokumen A	Dokumen B
Menghubungkan seluruh komponen penyusun sebuah komputer	Menghubungkan komponen komponen penyusun yang ada di dalam sebuah komputer

Tabel 3.10 Tabel dokumen sesudah *preprocessing*

Dokumen A	Dokumen B
hubung	hubung
komponen	komponen
susun	komponen
komputer	susun
	komputer

c. Mencari Sinonim

Selanjutnya hasil dari preprocessing akan dilanjutkan untuk mencari sinonim dari masing-masing term atau kata di dalam database sinonim. Jika kata memiliki sinonim di dalam database sinonim maka kata tersebut diubah ke dalam sinonimnya. Namun jika kata tersebut tidak memiliki sinonimnya maka kata tersebut tidak diubah sama sekali. Maka hasil dari synonym recognition sebagai berikut :

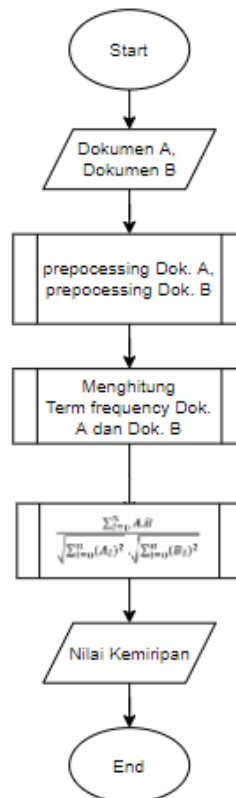
Tabel 3.11 Hasil *synonym recognition*

Kunci Jawaban	Jawaban siswa
hubung	hubung
anggota	anggota
susun	anggota
komputer	susun
	komputer

d. Selesai.

3.4.3. Flowchart Cosine Similarity

Untuk menggambarkan proses perhitungan menggunakan metode cosine similarity, penulis menambahkan *flowchart Cosine Similarity* seperti pada gambar 3.3 :



Gambar 3.3 Flowchart Cosine Similarity

Keterangan :

- a. Mulai
- b. Input dokumen

Dimulai dari input dua buah dokumen berbeda. Pada tahap ini pengguna(guru) menginputkan dokumen kunci jawaban ke dalam database system, dan pengguna(siswa) menginputkan jawaban siswa ke dalam database system. Berikut contoh dokumen kunci jawaban dan jawaban siswa pada tabel 3. .

Tabel 3.12 Dokumen kunci jawaban dan jawaban siswa

Kunci jawaban	Jawaban siswa
Menghubungkan seluruh komponen penyusun sebuah komputer	Menghubungkan komponen komponen penyusun yang ada di dalam sebuah komputer

c. *Preprocessing*

Selanjutnya masing-masing dokumen tersebut akan masuk ke dalam tahap *preprocessing*. Hasil dari tahap *preprocessing* akan seperti pada tabel 3. .

Tabel 3.13 Tabel hasil *preprocessing*

Dokumen A	Dokumen B
hubung	hubung
komponen	komponen
susun	komponen
komputer	susun
	komputer

d. *Term Frekuensi*

Jika proses preprocessing selesai maka selanjutnya masing-masing dokumen tersebut di cari term frekuensinya. Term yang di hitung adalah term yang dimiliki oleh dua buah dokumen. Jika term hanya ditemukan pada salah satu dokumen maka term tersebut tidak dihitung frekuensinya. Berikut tabel hasil term frekuensi.

Tabel 3.14 Table hasil term frekuensi

Term	Term Dokumen Kunci Jawaban / Term A	Term Dokumen Jawaban Siswa / Term B
Hubung	1	1
Anggota	1	2
Susun	1	1
komputer	1	1

e. *Perhitungan Cosine Similarity*

Selanjutnya dari hasil term frekuensi maka akan dimasukkan ke dalam rumus cosine similarity. Cosine similarity akan memproses nilai kemiripan dari dua buah dokumen tersebut. Dari hasil term frekuensi dapat kita ambil nilai vector sebagai berikut :

Vektor A : (1, 1, 1, 1)

Vektor B : (1, 2, 1, 1)

Kemudian dari masing-masing vector tersebut dimasukkan ke dalam rumus cosine similarity seperti berikut :

$$Similarity(A, B) = \frac{\sum_{i=0}^n A_i \cdot B_i}{\sqrt{\sum_{i=0}^n (A_i)^2} \cdot \sqrt{\sum_{i=0}^n (B_i)^2}} \dots \dots \dots \text{Rumus 3.7}$$

$$A \cdot B = \sum_{i=0}^n A_i \cdot B_i = (1 * 1) + (1 * 2) + (1 * 1) + (1 * 1) \dots \dots \dots \text{Rumus 3.8}$$

$$|A| = \sqrt{\sum_{i=0}^n (A_i)^2} = \sqrt{1^2 + 1^2 + 1^2 + 1^2} \dots \dots \dots \text{Rumus 3.9}$$

$$|B| = \sqrt{\sum_{i=0}^n (B_i)^2} = \sqrt{1^2 + 2^2 + 1^2 + 1^2} \dots \dots \dots \text{Rumus 3.10}$$

$$= \frac{5}{\sqrt{4} \cdot \sqrt{7}} \dots \dots \dots \text{Rumus 3.11}$$

$$= \frac{5}{2 * 2.645} \dots \dots \dots \text{Rumus 3.12}$$

$$= 0.945$$

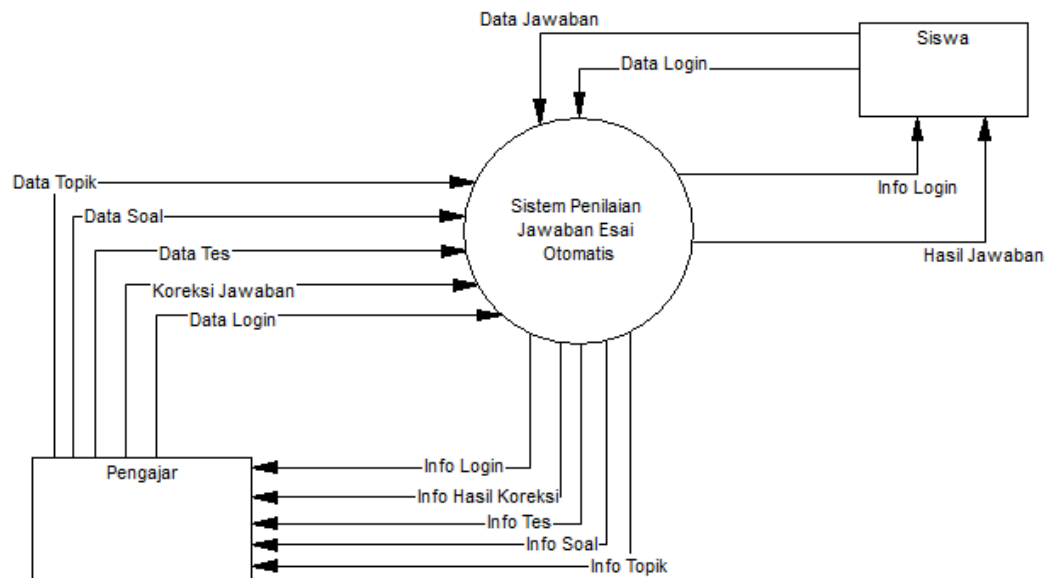
f. Menampilkan nilai kemiripan

Kemudian nilai dari perhitungan *cosine similarity* ditampilkan. Hasil dari nilai kemiripan dua buah dokumen menggunakan perhitungan *cosine similarity* diatas adalah 0.945.

g. Selesai.

3.4.4. Diagram Konteks

Untuk menggambarkan perancangan proses, pengguna dapat melakukan login dalam sistem ini mempunyai kewenangan untuk mengatur dan mengelolah data-data serta proses yang ada di dalam sistem, yaitu melakukan proses manipulasi seluruh data yang tersedia, seperti mengedit dan menghapus data. Adapun diagram konteks dari penilaian jawaban esai otomatis yang akan dirancang dapat dilihat pada Gambar 3.4 :

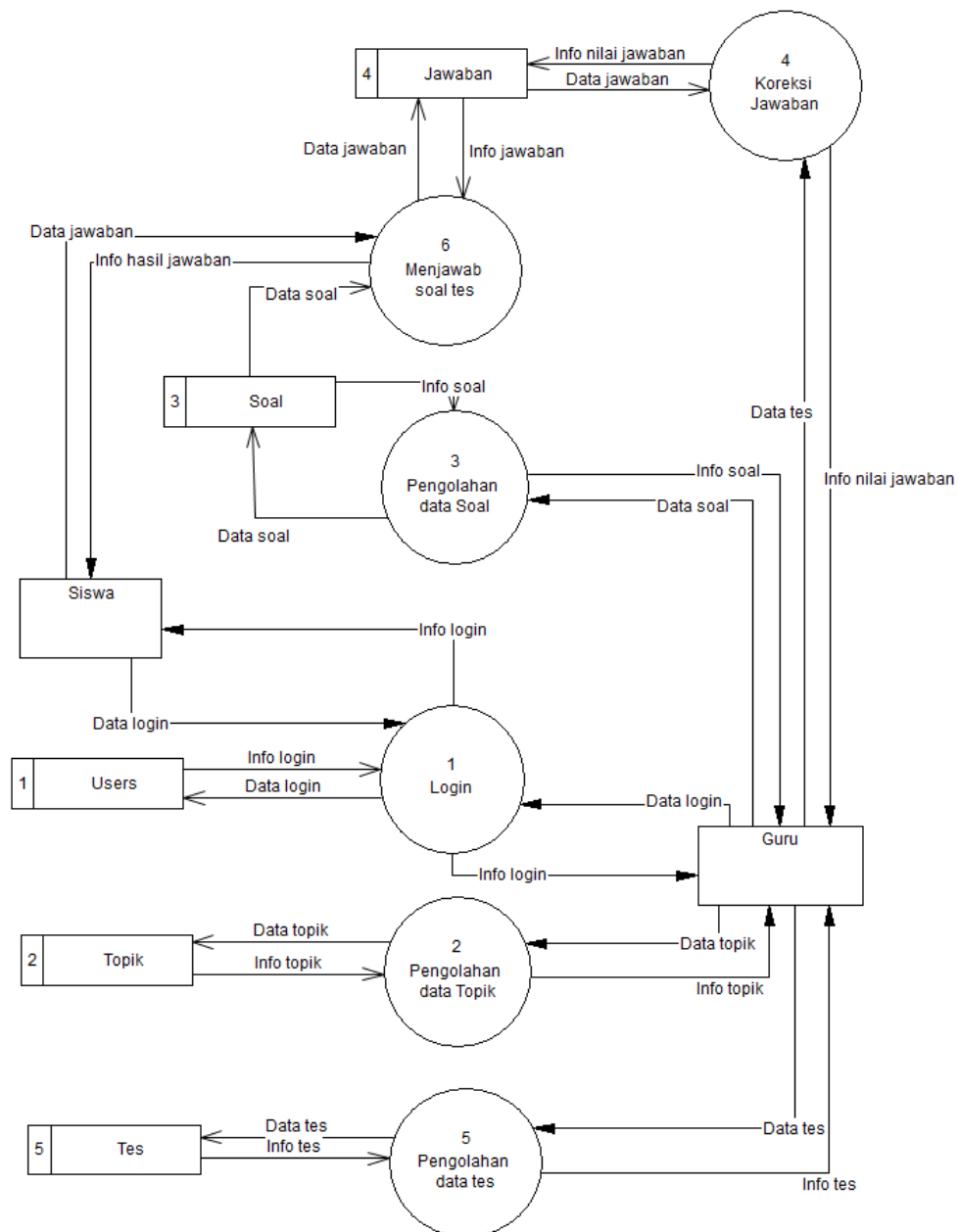


Gambar 3.4 Diagram Konteks

3.4.5. Data Flow Diagram (DFD)

3.4.5.1. DFD Level 1 Sistem

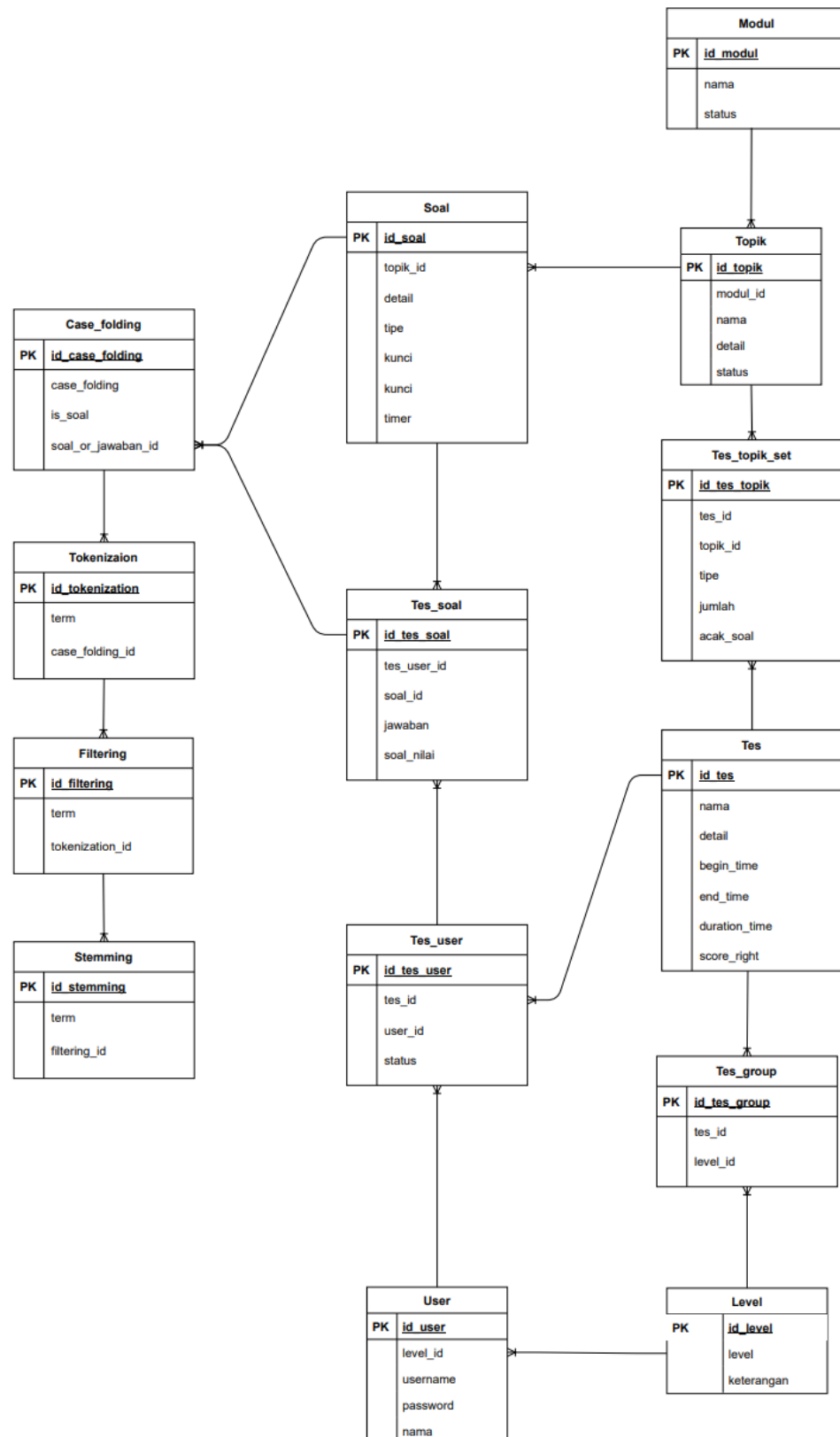
Menjelaskan proses yang ada di dalam sistem yang di bangun berdasarkan diagram konteks, adapun proses tersebut di bagi menjadi 6 (enam), yaitu proses login, proses pengelolaan data topik, pengelolaan data soal, pengelolaan data tes, koreksi jawaban, dan proses menjawab soal tes. DFD level 1 dapat dilihat pada gambar 3.5 :



Gambar 3.5 Data Flow Diagram Level 1

3.4.6. Entity Relational Diagram (ERD)

ERD atau *Entity Relational Diagram* adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas- entitas dan menentukan hubungan antar entitas.



Gambar 3.6 *Entity Relationship Diagram*

3.4.7. Perancangan Basis Data

Untuk membuat database dalam system penilaian jawaban esai otomatis ini terdiri dari 14 tabel, yaitu table Users, tabel Level, tabel Tes_group, tabel Tes, tabel Tes_user, tabel Tes_soal, tabel Tes_topik_set, tabel Topik, tabel Modul, table Soal, table Case_folding, table Tokenization, table Filtering, dan tabel Stemming.

Berikut adalah contoh struktur table database yang ada pada table 3.2 Sampai 3.15 :

1. Tabel User

Tabel ini digunakan untuk menyimpan data user/pengguna. Ditunjukkan pada table 3.2.

Tabel 3.15 tabel user

Field	Type	Null	Key
Id	Int	No	PK
Level_id	Int	No	FK
username	Varchar	No	
Password	Varchar(256)	No	
Nama	Varchar(100)	No	

2. Tabel Level

Tabel ini digunakan untuk menyimpan data level user. Ditunjukkan pada table 3.3.

Tabel 3.16 tabel level

Field	Type	Null	Key
Id	Int	No	PK
Level	Varchar	No	
keterangan	Varchar	No	

3. Tabel Tes_group

Tabel ini digunakan untuk menyimpan data group siswa. Ditunjukkan pada table 3.4.

Tabel 3.17 tabel *tes_group*

Field	Type	Null	Key
Id	Int	No	PK
Tes_id	Int	No	FK
Level_id	Int	No	FK

4. Table Tes

Tabel ini digunakan untuk menyimpan data tes. Ditunjukkan pada table 3.5.

Tabel 3.18 tabel tes

Field	Type	Null	Key
Id	Int	No	PK
Nama	Varchar	No	
Detail	Varchar	Null	
Begin_time	Datetime	No	
End_time	Datetime	No	
Duration_time	Varchar	Null	
Score_hight	Int	Null	

5. Tabel Tes_user

Tabel ini digunakan untuk menyimpan data tes user siswa. Ditunjukkan pada table 3.6.

Tabel 3.19 tabel tes_user

Field	Type	Null	Key
Id	Int	No	PK
Tes_id	Int	No	FK
user_id	Int	No	FK
Status	Tinyint	No	

6. Tabel Tes_soal

Tabel ini digunakan untuk menyimpan data soal, jawaban dan nilai siswa tiap soal. Ditunjukkan pada table 3.7.

Tabel 3.20 tabel tes_soal

Field	Type	Null	Key
Id	Int	No	PK
Tes_user_id	Int	No	FK
soal_id	Int	No	FK
Jawaban	Text	Null	
Soal_nilai	Int	Null	

7. Tabel Tes_topik_set

Tabel ini digunakan untuk menyimpan data pengaturan tes. Ditunjukkan pada table 3.8.

Tabel 3.21 tabel tes_topik_set

Field	Type	Null	Key
Id	Int	No	PK

Field	Type	Null	Key
Tes_id	Int	No	FK
topik_id	Int	No	FK
Tipe	Int	No	
Jumlah	Int	Null	
Acak_soal	Tinyint	No	

8. Tabel Topik

Tabel ini digunakan untuk menyimpan data topik soal. Ditunjukkan pada table 3.9.

Tabel 3.22 tabel topik

Field	Type	Null	Key
Id	Int	No	PK
modul_id	Int	No	FK
Nama	Varchar	No	
Detail	Varchar	Null	
Status	Tinyint	No	

9. Tabel Modul

Tabel ini digunakan untuk menyimpan data modul. Ditunjukkan pada table 3.10.

Tabel 3.23 tabel modul

Field	Type	Null	Key
Id	Int	No	PK
Nama	Varchar	No	
Status	Tinyint	Null	

10. Tabel Soal

Tabel ini digunakan untuk menyimpan data soal. Ditunjukkan pada table 3.11.

Tabel 3.24 tabel soal

Field	Type	Null	Key
Id	Int	No	PK
Topik_id	Int	No	FK
Detail	Varchar	Null	
Tipe	Int	No	
Kunci	Text	No	
Timer	Tinyint	Null	

11. Tabel Case_folding

Tabel ini digunakan untuk menyimpan data hasil case folding. Ditunjukkan pada table 3.12.

Tabel 3.25 tabel case_folding

Field	Type	Null	Key
Id	Int	No	PK
Is_soal	Tinyint	No	
Case_folding	Text	No	
Soal_or_jawaban_id	Int	No	FK

12. Tabel Tokenization

Tabel ini digunakan untuk menyimpan data hasil *tokenization*. Ditunjukkan pada table 3.13.

Tabel 3.26 tabel tokenization

Field	Type	Null	Key
Id	Int	No	PK
Term	Text	No	
Case_folding_id	Int	No	FK

13. Tabel Filtering

Tabel ini digunakan untuk menyimpan data filtering. Ditunjukkan pada table 3.14.

Tabel 3.27 tabel filtering

Field	Type	Null	Key
Id	Int	No	PK
Term	Text	No	
Tokenization_id	Int	No	FK

14. Tabel Stemming

Tabel ini digunakan untuk menyimpan data stemming. Ditunjukkan pada table 3.15.

Tabel 3.28 tabel stemming

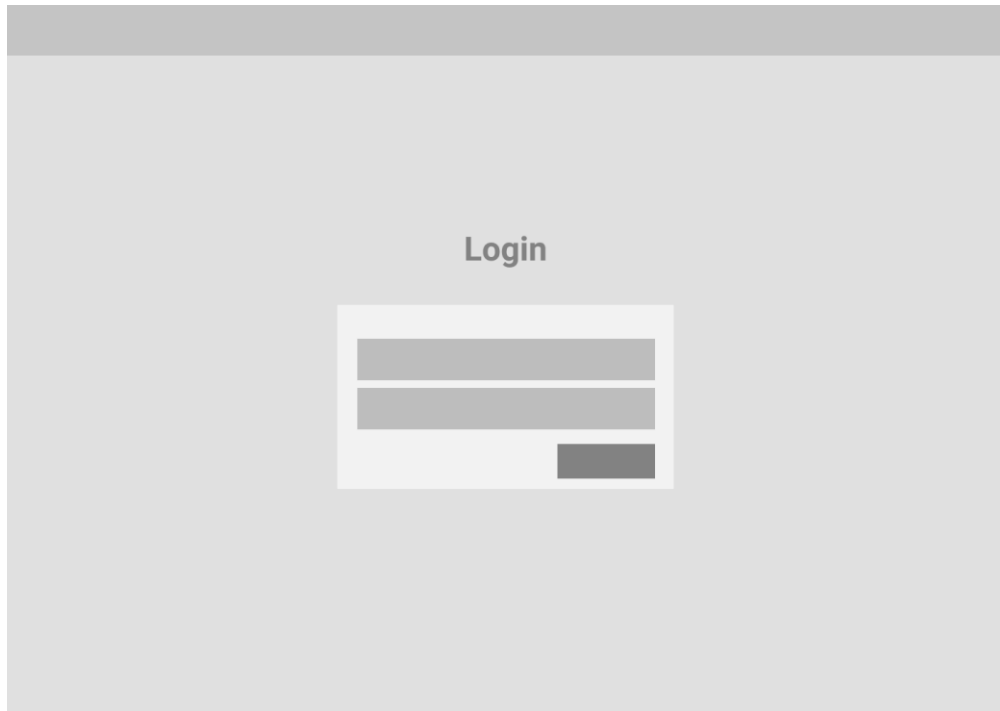
Field	Type	Null	Key
Id	Int	No	PK
Term	Text	No	
filtering_id	Int	No	FK

3.4.8. Desain Antarmuka

Dalam pembuatan sistem penilaian jawaban esai otomatis ini, dibuat suatu desain standar untuk tampilan antarmuka secara garis besar. Berikut adalah gambaran desain tampilan antarmuka yang telah dirancang oleh peneliti:

1. Halaman Login

Halaman *login* dimana halaman ini merupakan sebuah gerbang pada aplikasi, pengguna harus memasukkan *username* dan *password* untuk bisa menggunakan aplikasi. Halaman *login* dapat dilihat pada gambar 3.7 :

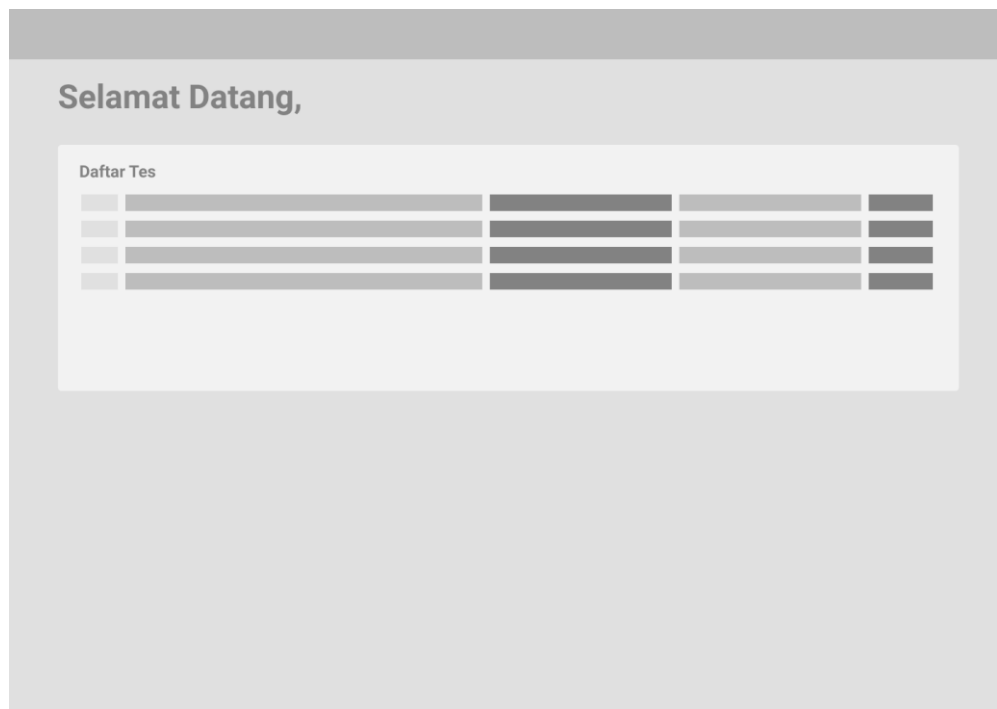


Gambar 3.7 Halaman *login*

Halaman *login* memiliki dua field dan satu tombol, field tersebut adalah *username* dan *password*. Sedangkan tombol yang ada adalah untuk melakukan proses login setelah field terisi. Pengguna wajib melakukan proses login untuk masuk ke halaman aplikasi sesuai level user-nya.

2. Halaman Utama Siswa

Setelah pengguna masuk, pengguna akan diarahkan ke halaman dashboard. Jika pengguna adalah siswa maka desain tampilan seperti gambar 3.8 :



Gambar 3.8 Halaman *utama siswa*

Halaman utama siswa menampilkan daftar tes yang sudah maupun belum dikerjakan siswa. Siswa dapat melihat daftar tes yang sudah maupun belum dikerjakan. Dalam daftar tes terdapat tombol yang digunakan siswa untuk mengerjakan tes, maka nanti akan tampil halaman ujian untuk menjawab soal tes.

3. Halaman Ujian Siswa

Ketika siswa menekan tombol untuk mengerjakan tes maka akan tampil tampilan seperti gambar 3.9 :

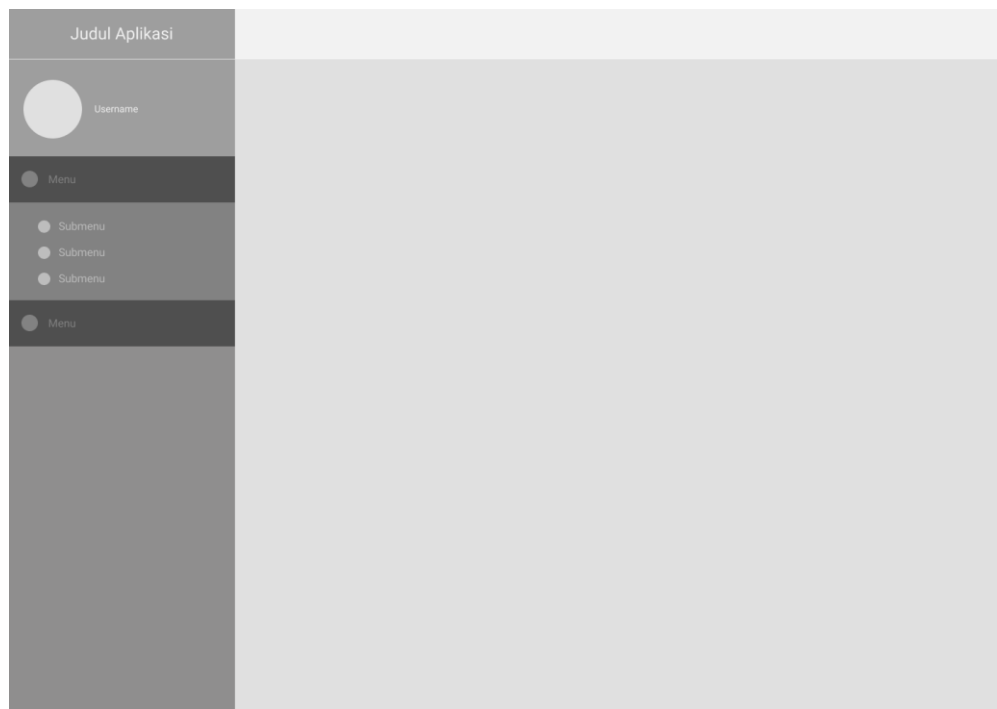


Gambar 3.9 Halaman ujian siswa

Halaman ini digunakan siswa untuk menjawab pertanyaan yang sudah disiapkan oleh guru/pengajar. Siswa dapat melihat serta menjawab soal pada ujian tersebut. Jika siswa sudah selesai mengerjakan terdapat tombol untuk mengakhiri ujian, tombol tersebut digunakan untuk mengakhiri ujian dan jawaban siswa akan tersimpan ke dalam database system.

4. Halaman Utama Guru

Selanjutnya dibawah ini tampilan halaman dashboard untuk guru/pengajar, bisa dilihat pada gambar 3.10 :

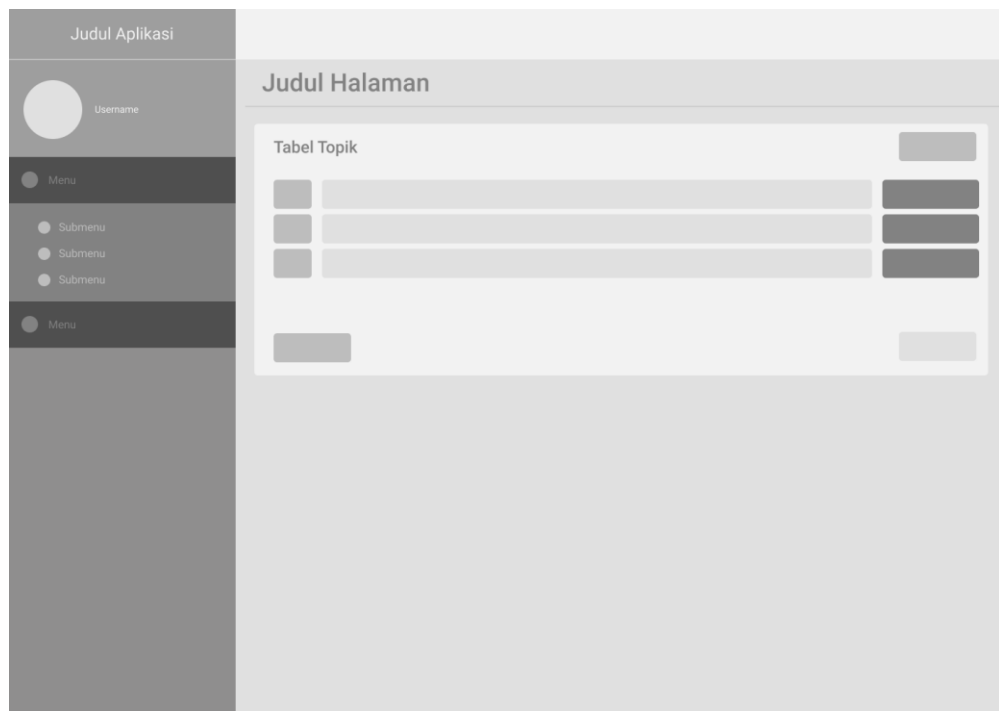


Gambar 3.10 Halaman utama guru

Halaman utama guru adalah halaman pertama muncul ketika guru berhasil melakukan proses login. Halaman ini hanya menampilkan teks selamat datang pada aplikasi, serta guru juga dapat melihat dan mengakses beberapa menu dan submenu pada *sidebar* aplikasi.

5. Halaman Topik

Ketika tombol menu topik ditekan, maka halaman yang akan tampil dapat dilihat pada gambar 3.11 :



Gambar 3.11 Halaman topik

Halaman topik digunakan oleh guru untuk menambahkan topik/jenis tes yang akan di kerjakan oleh siswa. Guru dapat melihat daftar dari topik yang pernah guru tambahkan sebelumnya. Halaman ini guru dapat menambahkan, mengubah serta menghapus topik ke dalam aplikasi,

6. Halaman Tambah Soal

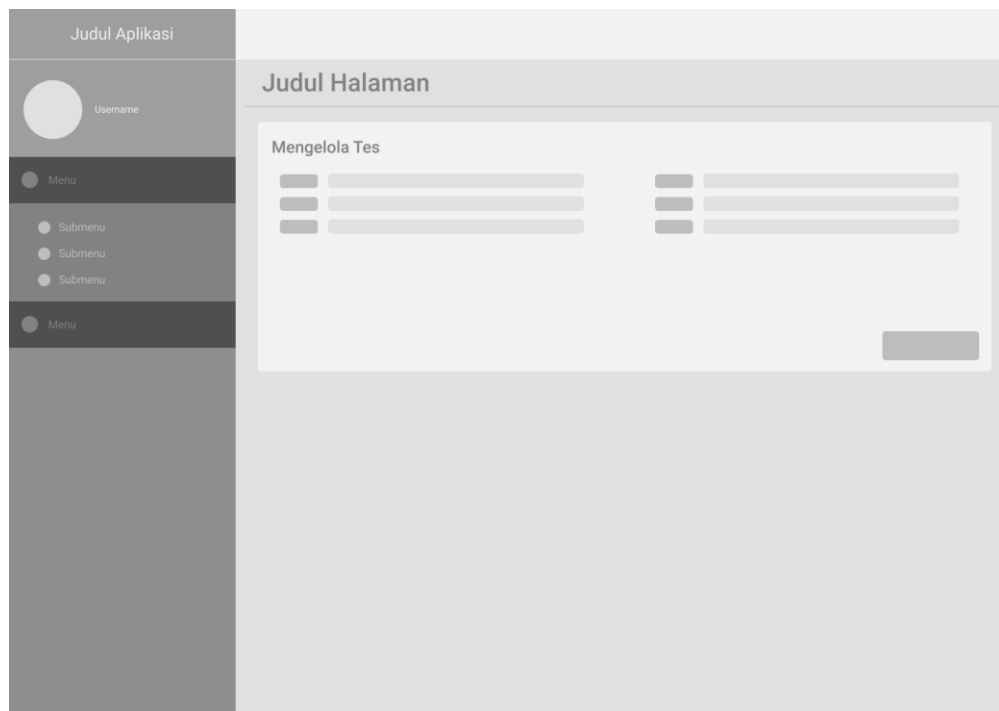
Kemudian pengguna(guru) diminta untuk menambahkan soal berserta kunci jawaban, sesuai topik yang telah ditambahkan oleh guru. Berikut tampilan dari halaman tambah soal pada gambar 3.12 :

Gambar 3.12 Halaman tambah soal

Halaman ini pengguna(guru) juga dapat melihat daftar soal yang sudah ditambahkan oleh guru. Tidak hanya menambahkan soal, guru juga dapat melihat, menghapus maupun merubah pertanyaan maupun kunci jawaban yang sudah ditambahkan.

7. Halaman Tes

Ketika pengguna(guru) menekan tombol atau menu tes, maka akan tampil halaman tes seperti gambar 3.13 :

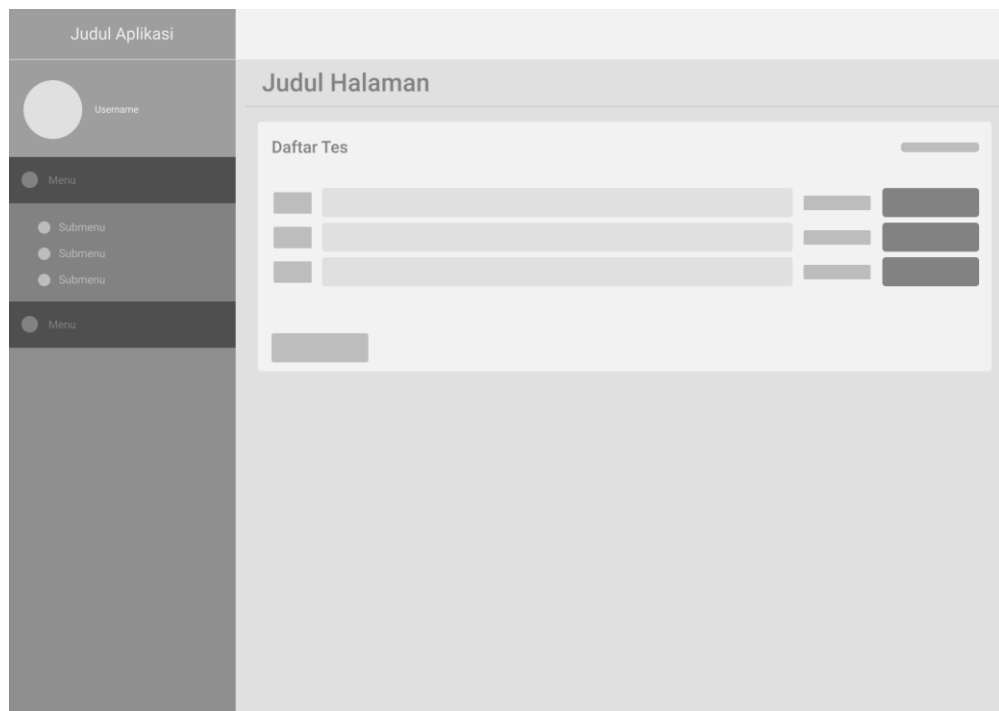


Gambar 3.13 Halaman tes

Halaman ini digunakan oleh pengguna(guru) untuk menambahkan nama tes yang akan dilakukan. Terdapat nama tes, deskripsi tes, waktu tes, kelas mana yang melakukan tes, memasukkan topik soal mana yang akan dimasukkan ke dalam tes dan sebagainya. Guru hanya dapat menambahkan tes saja pada halaman ini.

8. Halaman Daftar Tes

Selanjutnya ketika pengguna(guru) menekan tombol menu daftar tes, maka akan tampil halaman daftar tes seperti gambar 3.14 :



Gambar 3.14 Halaman daftar tes

Halaman daftar tes berisi daftar tes yang sudah ditambahkan oleh pengguna(guru). Pengguna(guru) dapat melihat daftar tes yang pernah ditambahkan ke dalam aplikasi. Pengguna(guru) juga dapat melihat detail, mengubah dan menghapus tes.

9. Halaman Evaluasi Tes

Setelah itu jika pengguna(guru) menekan tombol evaluasi tes maka akan tampil halaman evaluasi tes seperti gambar 3.15 :

DAFTAR RUJUKAN

- Ariantini, D. A., Lumenta, A. S., & Jacobus, A. (2016). PENGUKURAN KEMIRIPAN DOKUMEN TEKS BAHASA INDONESIA MENGGUNAKAN METODE COSINE SIMILARITY. *E-Journal Teknik Informatika*, Vol. 9.
- Arnawa, I. B. (2019). KOREKSI UJIAN ESSAY OTOMATIS DENGAN TEXT MINING PADA E-LEARNING STIKOM BALI. *Jurnal Teknologi Informasi dan Komputer*, 228-233.
- Bahri, S., & Wajhillah, R. (2020). Optimalisasi Algoritma Rabin Karp menggunakan TF-IDF Dalam Pencocokan Text Pada Penilaian Ujian Essay Otomatis. *JURNAL NASIONAL INFORMATIKA DAN TEKNOLOGI JARINGAN*, Vol. 2 103-105.
- Bastian, A., & Sujadi, H. (2018). IMPLEMENTASI DAN PENGUJIAN APLIKASI PENILAIAN UJIAN ESSAY. *Jurnal J-Ensitem*, Vol. 05.
- Cipto, L. R., & Irfan, P. (2020). Aplikasi Ujian Online dengan Penilaian Otomatis Menggunakan Algoritma Cosine Similarity Pada SMAN 7 Mataram. *Jurnal BITE*, Vol. 2 Hal. 57-65.
- Fataruba, F. (2018). Penerapan Metode Cosine Similarity Untuk Pengecekan Kemiripan Jawaban Ujian Siswa. *Jurnal Mahasiswa Teknik Informatika*, Vol. 2.
- Fauzi, M. A., Utomo, D. C., Setiawan, B. D., & Pramukantoro, S. E. (2016). AUTOMATIC ESSAY SCORING SYSTEM USING N-GRAM.
- Gunawan, D., Sembiring, C., & Budiman, M. (2018). The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents. *International Conference on Computing and Applied Informatics*.
- Irawan, R. B. (2020). Prediksi Jumlah Penjualan Jajanan Khas Daerah Menggunakan Metode Least Square Berbasis Multiplatform.
- Jo, T. (2019). *Text Mining Concept, Implementation and Big Data Challenge*. Seoul, Korea: Springer International Publishing AG.
- Joane, Y. L., Sinsuw, A., & Jacobus, A. (2017). Rancang Bangun Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma Ratcliff/Obershelp. *E-Journal Teknik Informatika*, Vol. 11.
- Kurniawati, F. E., & Pradnya, W. M. (2020). Implementasi Algoritma Winnowing Pada Sistem Penilaian Otomatis Jawaban Esai Pada Ujian Online Berbasis Web. *Jurnal Teknik Komputer AMIK BSI*, Vol. 6.

- Muslihudin, M., & Oktafianto. (2016). *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: Penerbit Andi.
- Noviantoro, R. W. (2020). *PENENTUAN TARIF JASA LUKIS DENGAN MENERAPKAN METODE ANALYTICAL HIEARARCHY PROCESS*.
- Nugraha, A. R., & Pramukasari, G. (2017). SISTEM INFORMASI AKADEMIK SEKOLAH BERBASIS WEB DI SEKOLAH MENENGAH PERTAMA NEGERI 11 TASEK MALAYA. *JURNAL MANAJEMEN INFORMATIKA*, Vol. 4.
- Oduntan, O. E., Adeyanju, I. A., Falohun, A. S., & Obe, O. O. (2018). A Comparative Analysis of Euclidean Distance and Cosine Similarity Measure for Automated Essay-Type Grading. *Journal of Engineering and Applied Sciences*, Hal. 4198-4204.
- Purnamasari, Y., Pudjiantoro, T. H., & Nursantika, D. (2017). SISTEM PENILAIAN KINERJA DOSEN TELADAN MENGGUNAKAN METODE SIMPLE MULTY ATTRIBUTE RATING TECHNIQUE (SMART). *Jurnal Teknologi Elektro, Universitas Mercu Buana*, Vol. 8.
- Ramdani, L., Jamaludin, M., Yuniarti, N., Rahmani, A., & Hutahean, J. (2019). Aplikasi Penilaian Otomatis Ujian Esai Berbahasa Indonesia. 314-324.
- Ramlah. (2017). *PENGEMBANGAN MEDIA PEMBELAJARAN FLOWCHART BERBASIS DRILL PADA PEMBELAJARAN BIOLOGI POKOK BAHASAN SISTEM PERNAPASAN SISWA KELAS VIII MTs NEGERI BALANG-BALANG KAB.GOWA*. Makasar: UIN Allaudin Makasar.
- Rifai, A., & Yuniar, Y. P. (2019). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Ujian Pada SMK Indonesia Global Berbasis Web. *JURNAL KHATULISTIWA INFORMATIKA*, Vol. VII.
- Rinartha, K. (2017). Pemodelan Penilaian Essay Otomatis Secara Realtime Menggunakan Kombinasi Text Stemming Dan Cosine Similarity. *Konferensi Nasional Sistem & Informatika*, 322-327.
- Rymmai, R. G., & JS, S. (2017). Book Recommendation using Cosine Similarity. *International Journal of Advanced Research in Computer Science*, Hal. 276-281.
- Salim, A. M. (2017). PENGEMBANGAN APLIKASI PENILAIAN UJIAN ESSAY BERBASIS ONLINE MENGGUNAKAN ALGORITMA NAZIEF DAN ADRIANI DENGAN METODE COSINE SIMILARITY. *Jurnal IT-EDU*, Vol. 02.

- Smrti, N. N., Saputra, A., Zulfachmi, Setiawan, R. A., Putra, I. P., & Andisana, I. P. (2019). Aplikasi Penilaian Otomatis Jawaban Soal Essay dengan Menggunakan Metode K-Nearest Neighbors. Tanjung Benoa-Bali.
- Sukarsa, I. M. (2019). APLIKASI KONVERSI FLOWCHART KE KODE PROGRAM BAHASA PEMROGRAMAN PL/SQL MYSQL. *Teknologi Elektro*, Hal. 44-53.
- Suryati, T., Wibisono, Y., & Wihardi, Y. (2018). Aplikasi Deteksi Plagiarisme Dokumen Skripsi . *dengan Algoritma Rabin-Karp*, Vol. 2.
- Yusuf, B., Vivianie, S., Marsya, J. M., & Sofyan, Z. (2019). Analisis Perbandingan Algoritma Rabin-Karp dan Ratcliff/Obershelp untuk Menghitung Kesamaan Teks dalam Bahasa Indonesia. *Seminar Nasional APTIKOM (SEMNASTIK)*.
- Yusuf, S., Fauzi, M. A., & Brata, K. C. (2018). Sistem Temu Kembali Informasi Pasal-Pasal KUHP (Kitab Undang-Undang Hukum Pidana) Berbasis Android Menggunakan Metode Synonym Recognition dan Cosine Similarity. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol. 2.
- Zizka, J., Darena, F., & Svoboda, A. (2019). *Text Mining Machine Learning Principle and Techniques*. Taylor & Francis Group.