

## BAB 3

### ANALISIS DAN PERANCANGAN

#### 3.1. Analisis Masalah

Salah satu cara penilaian dalam pembelajaran di kelas adalah dengan ujian. Jenis ujian yang biasa digunakan dalam setiap proses belajar mengajar adalah tipe pilihan ganda (*multiple choice*), benar atau salah (*true or false*), serta uraian (*essay*) [2]. Setiap jawaban akan langsung otomatis dinilai dan setiap jawaban akan mempengaruhi nilai akhir. Misalnya, pada ujian bertipe *multiple choice* jenis soal pilihan ganda ini mudah dalam memeriksanya [27], seperti penelitian yang dilakukan oleh Diana Sari [3] dan Azizul Hakim Prabowo dan Asahar Johar [28].

Pada ujian online bertipe uraian (*essay*) membutuhkan jawaban text yang teliti dalam menjawabnya tidak boleh terjadi kesalahan ketik (*typo*) karena akan mengakibatkan berkurangnya nilai atau *scoring*. Kesalahan dalam penulisan dapat disebabkan oleh kesalahan dalam pengetikan yang tidak sesuai dengan ejaan kata sehingga dapat mengubah kata menjadi tidak bermakna [4] atau karena kecepatan dalam menulis ataupun kelengahan dalam mengetikkan jari-jari di atas keyboard komputer [5]. Kekurangan salah satu huruf, kelebihan, atau kesalahan penempatan [5]. Oleh karena itu masalah *typo* dalam menjawab soal esai ini tidak bisa dibiarkan karena akan mengakibatkan akurasi penilaian atau *scoring* akan berkurang.

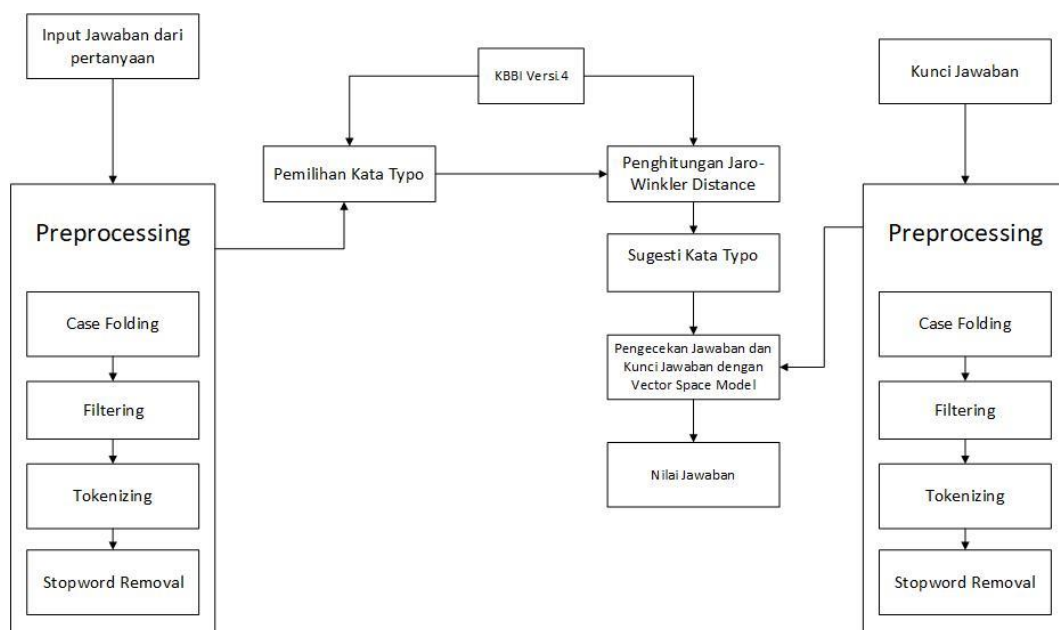
Berikut ini adalah beberapa contoh dalam kesalahan pengetikan yang mungkin terjadi, misalnya dalam menjawab soal pertanyaan berikut : “ Apa nama ibu kota Indonesia ?”. Dijawab dengan mengetikkan “Jakrata, Jakartaa, dan Jakart”, sedangkan ejaan yang benar adalah “Jakarta”. Kesalahan pertama pada kata “Jakrata” adalah jenis kesalahan ejaan transposisi huruf karena pada huruf “r” dan “a” tertukar posisinya. Kesalahan kedua pada kata “Jakartaa” adalah jenis kesalahan ejaan duplikasi huruf karena kelebihan huruf “a”. Ketiga pada kata “Jakart” adalah jenis kesalahan ejaan ketertinggalan huruf karena kurangnya huruf “a”.

Untuk mengatasi masalah kesalahan ejaan (*typo*) yang sering terjadi dalam menjawab pertanyaan ini diperlukan suatu metode untuk mengatasi koreksi

kesalahan ejaan dalam menjawab soal esai. Dalam penelitian kali ini akan diajukan salah satu metode koreksi kesalahan ejaan. Metode yang akan diajukan adalah metode *Jaro-Winkler distance*. Seperti metode yang digunakan oleh Fairiy Okta'mal *et al* [9], menggunakan metode Jaro-Winkler dengan akurasi sebesar 96%-100% untuk deteksi dini hama dan penyakit padi.

### 3.2. Analisis Sistem

Sistem yang akan dibangun mempunyai beberapa tahapan. Tahapan-tahapan tersebut yaitu proses memasukan kalimat jawaban, proses *preprocessing*, proses *case folding*, proses *filtering*, proses *tokenizing* proses *stopword removal*, kemudian melakukan proses pilih kata yang *typo* atau ejaan yang salah setelah itu diproses dengan metode *jaro winkler* hingga muncul segesti kata yang benar. Untuk lebih jelasnya dapat melihat gambar 3.1 dibawah ini.



**Gambar 3. 1 Alur Umum Sistem**

Berikut penjelasan alur umum sistem yang akan dibangun:

Input jawaban dari pertanyaan adalah proses untuk memasukan jawaban dari pertanyaan soal esai. Jawaban berupa kalimat definitif (penjelasan). Kemudian jawaban dari pertanyaan tersebut dilakukan proses *preprocessing* untuk mendapatkan suatu kata yang tepat untuk dilakukan proses berikutnya. Seperti *Case*

*Folding, Filtering, Tokenizing, dan Stopword Removal.* Pada proses *case folding* ini dilakukan untuk menyeragamkan huruf menjadi huruf kecil yaitu pada jawaban pertanyaan. Kemudian proses *Filtering* yaitu proses membuang simbol-simbol pada jawaban pertanyaan seperti tanda baca, titik, koma, tanda tanya, kutip, tanda seru dan lain-lain yang dianggap sebagai kata tidak penting dan tidak berpengaruh terhadap makna kata. Lalu dilakukan proses *tokenizing* sebagai proses pemecah sekumpulan karakter dalam suatu teks ke dalam satuan kata. Berikutnya dilakukan proses *stopword removal* yaitu untuk menghilangkan kata yang ‘tidak relevan’ pada hasil token sebuah jawaban pertanyaan dengan cara membandingkannya dengan stoplist. Stoplist berisi sekumpulan kata yang ‘tidak relevan’, namun sering sekali muncul dalam sebuah dokumen.

Selanjutnya memilih kata *typo* yaitu memilih kata yang memiliki kesalahan ejaan dengan cara mencocokkan kata hasil token dengan kata yang ada didalam kamus. Proses pencocokan dilakukan satu persatu. Jika kata yang dicari memiliki *typo* dan tidak ditemukan didalam kamus kata maka dianggap *typo*. Lalu dilakukan proses perhitungan *Jaro winkler distance* pada kata yang *typo* untuk mendapatkan nilai kata tertinggi. Apabila nilai *jaro winkler distance* 0 menandakan tidak ada kesamaan sedangkan nilai 1 memiliki kesamaan yang sama persis. Proses berikutnya dilakukan sugesti kata *typo* untuk mengambil kata yang memiliki nilai *jaro winkler distance* tertinggi. Kata yang memiliki nilai tertinggi akan dijadikan sebagai kata sugesti untuk mengkoreksi kata yang *typo*. Kemudian yang terakhir, setelah kata sudah benar dilakukan koreksi akan dilakukan pengecekan jawaban pertanyaan dan kunci jawaban dengan menggunakan vector space model untuk mendapatkan nilai jawaban pertanyaan yang telah diinputkan.

### **3.2.1. Analisis Data Masukan**

Data masukan yang dipakai dalam penelitian ini adalah daftar pertanyaan dan jawaban berupa kalimat definitif (penjelasan). Penjelasan data masukan akan dijelaskan dibawah ini.

### 3.2.1.1. Daftar Pertanyaan dan Jawaban

Data masukan yang dibutuhkan adalah daftar pertanyaan dan jawaban. Daftar jawaban berupa kalimat pendefinisian. Berikut adalah sebagai contoh daftar pertanyaan dan jawaban.

**Tabel 3. 1 Daftar Pertanyaan Soal Esai**

No	Soal Esai
1	Apa yang dimaksud dengan Algoritma ?
2	Apa pengertian Komputer ?
3	Apa pengertian Handphone ?

Daftar pertanyaan diatas dibuat dalam *database*.

**Tabel 3. 2 Daftar Kunci Jawaban**

No	Kunci Jawaban
1	Langkah-langkah atau metode untuk menyelesaikan suatu masalah
2	Suatu alat elektronik yang mampu melakukan beberapa tugas seperti menerima input, memproses, input, menyimpan perintah-perintah dan menyediakan output dalam bentuk informasi.
3	Alat komunikasi melalui suara dan pesan singkat

Daftar pertanyaan dan jawaban diatas dijadikan sebagai data uji untuk kasus perbaikan kesalahan ejaan pada sistem *essay scoring*. Kali ini peneliti hanya mengambil satu pertanyaan dan jawaban sebagai perhitungan mencari kata *typo* dengan menggunakan metode *Jaro Winkler Distance*.

### 3.2.1.2. Data Uji

Berikut adalah data uji yang digunakan untuk menyelesaikan perbaikan kesalahan ejaan pada sistem *essay scoring*. Berikut satu pertanyaan dan satu jawaban dalam bentuk tabel 3.3.

**Tabel 3. 3 Pertanyaan Esai**

No	Soal Esai
1	Apa yang dimaksud dengan Algoritma ?

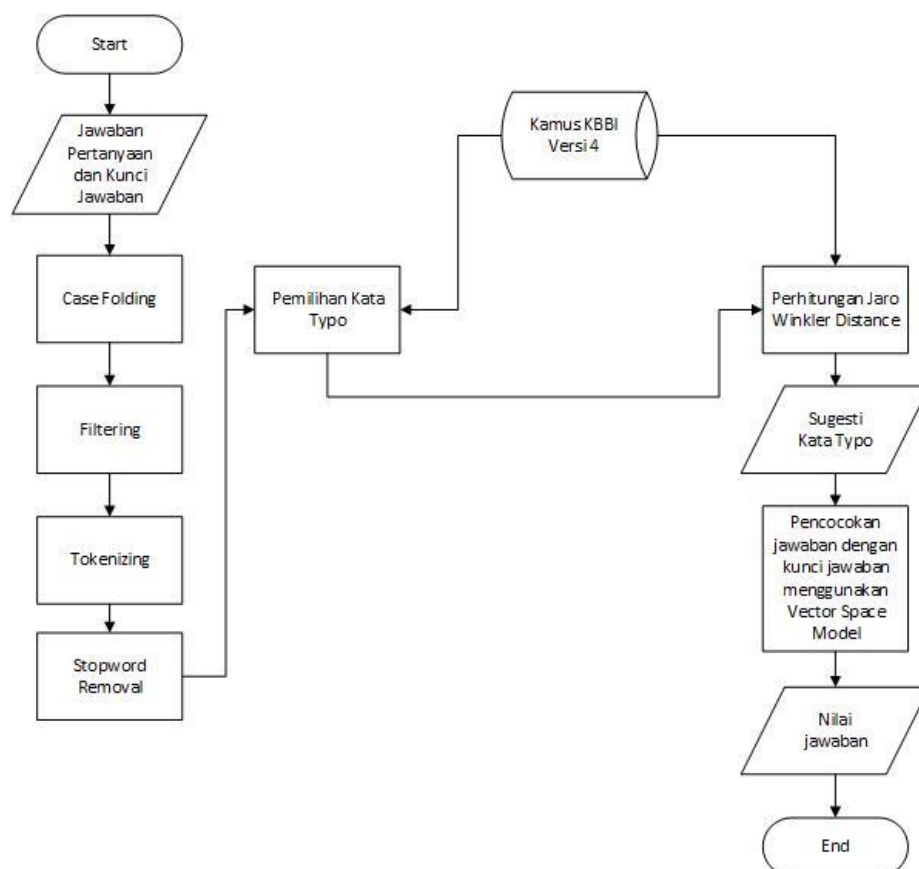
**Tabel 3. 4 Jawaban Pertanyaan**

No	Jawaban Pertanyaan
1	Langkah-langkah atau metode untuk menyelesaikan suatu <u>maslaah</u>

Pada jawaban diatas terdapat ada satu buah kesalahan ejaan yaitu pada kata “Maslaah” (harusnya “Masalah”). Kata ‘masalah’ sengaja ditulis menjadi kata ‘maslaah’ untuk jadikan kata yang memiliki typo atau kata yang bertransposisi. Dari kata ‘maslaah’ akan dilakukan proses untuk diperbaiki dengan kata yang benar.

### 3.3. Analisis Proses

Analisis proses yang digunakan adalah *preprocessing*, *case folding*, *filtering*, *tokenizing*, *stopword removal*, *memilih kata typo*, proses *jaro winkler*, *sampai sugesti kata* dan proses pengecekan jawaban dan kunci jawaban dengan vector space model (VSM) atau *scoring*. Berikut adalah gambar 3.2 flowchart analisis proses mengatasi kesalahan ejaan.

**Gambar 3. 2 Flowchart Analisis Proses**

### 3.3.1. Case Folding

*Case folding* adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf ‘a’ sampai dengan ‘z’ yang diterima. Berikut hasilnya *case folding* pada tabel 3.5.

**Tabel 3. 5 Proses Case Folding**

Sebelum	Langkah-langkah atau metode untuk menyelesaikan suatu masalah
Sesudah	langkah-langkah atau metode untuk menyelesaikan suatu masalah

Huruf “L” yang digarisbawahi adalah huruf yang telah ubah menjadi huruf kecil.

### 3.3.2. Filtering

*Filtering* merupakan proses membuang kata-kata serta tanda-tanda yang tidak bermakna secara signifikan, yang tersisa hanya huruf a..z dan angka 1..9. Berikut hasilnya *filtering* pada tabel 3.6.

**Tabel 3. 6 Proses Filtering**

Sebelum	langkah-langkah atau metode untuk menyelesaikan suatu masalah
Sesudah	langkahlangkah atau metode untuk menyelesaikan suatu masalah

Pada kata “langkah-langkah” dihilangkan tanda strip (-) dihilangkan karena simbol tersebut tidak penting.

### 3.3.3. Tokenizing

*Tokenizing* adalah proses pemecah sekumpulan karakter dalam suatu teks ke dalam satuan kata. Proses untuk membagi teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi *token-token* atau bagian-bagian tertentu dengan delimeter spasi. Berikut hasilnya *tokenizing* pada tabel 3.7.

**Tabel 3. 7 Proses Tokenizing**

Sebelum	langkahlangkah atau metode untuk menyelesaikan suatu masalah
---------	--

Sesudah
langkahlangkah
atau
metode
untuk
menyelesaikan
suatu
<u>maslaah</u>

### 3.3.4. Stopword Removal

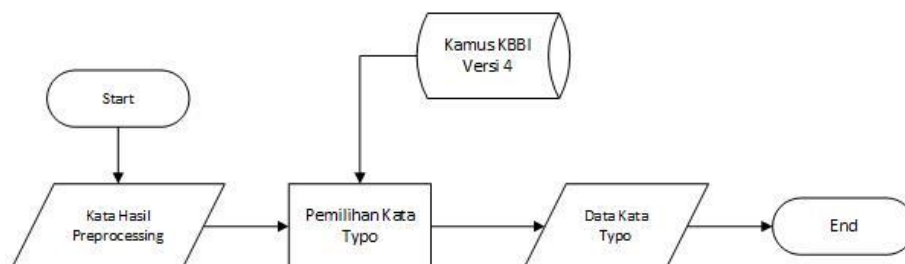
Stopword Removal adalah sebuah proses untuk menghilangkan kata yang ‘tidak relevan’ pada hasil token sebuah dokumen teks dengan cara membandingkannya dengan stoplist. Stoplist berisi sekumpulan kata yang ‘tidak relevan’, namun sering sekali muncul dalam sebuah dokumen. Daftar stoplist bisa lihat pada tabel 2.1 pada halaman 10 bab 2 [14]. Berikut hasilnya *stopword removal* pada tabel 3.8.

**Tabel 3. 8 Proses Stopword Removal**

Sebelum	Sesudah
langkahlangkah	langkahlangkah
atau	metode
metode	menyelesaikan
untuk	suatu
menyelesaikan	<u>maslaah</u>
suatu	
<u>maslaah</u>	

### 3.4. Pemilihan Kata *Typo*

Dibawah ini adalah gambar 3.3 pemilihan kata *typo*.



**Gambar 3. 3 Pemilihan Kata *Typo***

Proses memilih kata *typo* dilakukan dengan cara sebagai berikut.

- a. Setiap kata hasil preprocessing dicocokkan dengan kata yang ada didalam kamus untuk memilih kata yang *typo*. Yaitu dengan cara memilih kata yang tepat didalam kamus.
- b. Proses memilih kata *typo* dilakukan dengan cara mencocokkan setiap karakter kata yang *typo* dengan karakter kata yang ada didalam kamus kata sampai karakter-karakter kata yang dicocokkan (kata *typo*) mempunyai kemiripan dengan karakter-karakter kata yang ada didalam kamus kata.
- c. Setelah mendapatkan karakter kata (kata *typo*) yang memiliki kesamaan didalam kamus kata. Kata yang memiliki kemiripan akan dibandingkan yang kemudian dilakukan proses *jaro winkler distance*.

### 3.5. Hasil Deteksi

Berikut adalah hasil deteksi kata *typo*. Bisa dilihat pada tabel 3.9 dibawah ini.

**Tabel 3. 9 Hasil Deteksi**

No	Kata Typo
1	maslaah
2	langkahlanguk

### 3.6. Tahapan Untuk Koreksi Kata *Typo*

Berikut adalah tahapan koreksi kata *typo* pada aplikasi perbaikan kesalahan ejaan pada sistem *essay scoring*.

- a. Dari hasil proses pemilihan kata *typo* kata yang sudah pilih sebagai kata *typo* selanjutnya dilakukan proses koreksi kata *typo*.
- b. Proses koreksi kata *typo* dilakukan dengan melakukan perhitungan kata *typo* dengan menggunakan *jaro winkler distance* yang dibandingkan dengan kata yang ada di dalam kamus kata untuk menghasilkan nilai kata tertinggi. Nilai *jaro winkler distance* antara 0-1. Semakin nilai tinggi yaitu 1 menandakan kesamaan string yang dibandingkan mencapai seratus persen atau sama persis.



- c. Sehingga koreksi kata *typo* dilakukan dengan cara memilih kata yang memiliki nilai tertinggi sebagai kata yang benar untuk mengkoreksi kata yang *typo*.

Berikut adalah proses perhitungan *jaro winkler distance* untuk menemukan kata yang tepat dan benar sebagai koreksi kata yang *typo*.

**Tabel 3. 10 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

			0
			a
0	m		
1	a		✓
2	s		
3	l		
4	a		
5	a		
6	h		

→ Kata yang dijadikan acuan (kamus kata KBBI)

Pada tabel 3.10 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘a’ diatas tidak ada kata yang memiliki *prefix* yang sama persis maka tidak ada transposisi huruf.

$|s1|$  = Panjang string = 1 (a),  $|s2|$  = Panjang string = 7 (maslaah), karakter yang sama persis  $m = 1$ , huruf yang bertransposisi  $t = 0$ , dan prefix length  $l = 0$ .

Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{1}{1} + \frac{1}{7} + \frac{1-0}{1} \right) = \frac{1}{3} \left( \frac{7}{7} + \frac{1}{7} + \frac{7}{7} \right) = \frac{1}{3} \left( \frac{15}{7} \right) = \left( \frac{15}{63} \right) = 0.238$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.238 + (0 \times 0(1 - 0.238)) = 0.238 + (0 \times 0.762) = 0.238$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dengan “a” adalah 0.238.

**Tabel 3. 11 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

		0	1
		a	b
0	m		
1	a	✓	
2	s		
3	l		
4	a		
5	a		
6	h		

Pada tabel 3.11 di atas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘ab’ di atas tidak ada kata yang memiliki *prefix* yang sama persis maka tidak ada transposisi huruf.

$|s_1|$  = Panjang string = 2 (ab),  $|s_2|$  = Panjang string = 7 (maslaah), karakter yang sama persis  $m = 1$ , huruf yang bertransposisi  $t = 0$ , dan prefix length  $l = 0$ . Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{1}{1} + \frac{1}{7} + \frac{1-0}{1} \right) = \frac{1}{3} \left( \frac{7}{7} + \frac{1}{7} + \frac{7}{7} \right) = \frac{1}{3} \left( \frac{15}{21} \right) = \left( \frac{15}{63} \right) = 0.238$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.238 + (0 \times 0(1 - 0.238)) = 0.238 + (0 \times 0.762) = 0.238$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dengan “ab” adalah 0.238.

**Tabel 3. 12 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

		0	1	2
		a	b	a
0	m			
1	a	✓		
2	s			
3	l			
4	a			✓
5	a			
6	h			

Pada tabel 3.12 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘aba’ diatas tidak ada kata yang memiliki *prefix* yang sama persis maka tidak ada transposisi huruf.

$|s1|$  = Panjang string = 2 (aba),  $|s2|$  = Panjang string = 7 (maslaah), karakter yang sama persis  $m = 2$ , huruf yang bertransposisi  $t = 0$ , dan prefix length  $l = 0$ .

Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{2}{2} + \frac{2}{7} + \frac{2-0}{2} \right) = \frac{1}{3} \left( \frac{14}{14} + \frac{4}{14} + \frac{14}{14} \right) = \frac{1}{3} \left( \frac{32}{14} \right) = \left( \frac{32}{42} \right) = 0.761$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.761 + (0 \times 0(1 - 0.761)) = 0.761 + (0 \times 0.239) = 0.761$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dengan “aba” adalah 0.761.

**Tabel 3. 13 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

		0	1	2	3	4
		a	b	a	d	i
0	m					
1	a	✓				
2	s					
3	l					
4	a			✓		
5	a					
6	h					

Pada tabel 3.13 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘abadi’ diatas tidak ada kata yang memiliki *prefix* yang sama persis maka tidak ada transposisi huruf.

$|s1|$  = Panjang string = 5 (abadi),  $|s2|$  = Panjang string =7 (maslaah), karakter yang sama persis  $m = 2$ , huruf yang bertransposisi  $t = 0$ , dan prefix length  $l = 0$ .

Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{2}{7} + \frac{2}{5} + \frac{2-1}{2} \right) = \frac{1}{3} \left( \frac{20}{70} + \frac{28}{70} + \frac{35}{70} \right) = \frac{1}{3} \left( \frac{83}{70} \right) = \left( \frac{83}{210} \right) = 0.395$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.395 + (0 \times 0.1(1 - 0.395)) = 0.395 + (0 \times 0.605) = 0.395$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dengan “abadi” adalah 0.395.

**Tabel 3. 14 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

		0	1	2	3	4	5	6
		m	a	j	a	l	a	h
0	m	✓						
1	a		✓					
2	s							
3	l					✓		
4	a				✓			
5	a						✓	
6	h							✓

Pada tabel 3.14 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘majalah’ diatas ada kata yang memiliki posisi huruf yang sama persis dan memiliki transposisi huruf.  $|s1|$  = panjang string = 7 (majalah),  $|s2|$  = panjang string = 7 (maslaah), karakter yang sama persis  $m = 6$ , karakter yang bertransposisi yaitu ‘L’ dan ‘A’ maka nilai transposisi  $t = 1$ , dan prefix length  $l = 2$ . Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{6}{7} + \frac{6}{7} + \frac{6-1}{6} \right) = \frac{1}{3} \left( \frac{36}{42} + \frac{36}{42} + \frac{35}{42} \right) = \frac{1}{3} \left( \frac{107}{42} \right) = \left( \frac{107}{126} \right) = 0.849$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.849 + (2 \times 0.1(1 - 0.849)) = 0.849 + (0.2 \times 0.151) = 0.879$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dan “majalah” adalah 0.879.

**Tabel 3. 15 Penentuan Transposisi dan Karakter Ejaan Yang Salah**

		0	1	2	3	4	5	6
		m	a	s	a	l	a	h
0	m	✓						
1	a		✓					
2	s			✓				
3	l					✓		
4	a				✓			
5	a						✓	
6	h							✓

Pada tabel 3.15 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘maslaah’ dan ‘masalah’ diatas ada kata yang memiliki posisi huruf yang sama persis dan memiliki transposisi huruf.  $|s1|$  = panjang string = 7 (masalah),  $|s2|$  = panjang string (maslaah), dapat dilihat bahwa karakter yang sama persis  $m = 7$ , pada tabel diatas ada karakter yang bertransposisi yaitu ‘L’ dan ‘A’ maka nilai transposisi  $t = 1$ , dan prefix length  $l = 3$ . Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{7}{7} + \frac{7}{7} + \frac{7-1}{7} \right) = \frac{1}{3} \left( \frac{7}{7} + \frac{7}{7} + \frac{6}{7} \right) = \frac{1}{3} \left( \frac{20}{7} \right) = \left( \frac{20}{21} \right) = 0.952$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.952 + (3 \times 0.1(1 - 0.952)) = 0.952 + (0.3 \times 0.048) = 0.967$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “maslaah” dan “masalah” adalah 0.967.

**Tabel 3. 16 Penentuan Duplikasi Huruf dan Karakter Ejaan Yang Salah**

		0	1	2	3	4	5	6
		m	a	s	a	l	a	h
0	m	✓						
1	a		✓					
2	s			✓				
3	a				✓			
4	l					✓		
5	a						✓	
6	h							✓
7	h							

Pada tabel 3.16 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘masalahh’ dan ‘masalah’ diatas ada kata yang memiliki posisi huruf yang sama persis dan memiliki transposisi huruf.  $|s1|$  = panjang string = 7 (masalah),  $|s2|$  = panjang string = 8 (masalahh), dapat dilihat bahwa karakter yang sama persis  $m = 7$ , pada tabel diatas tidak ada karakter yang bertransposisi tetapi kelebihan huruf “h” maka nilai transposisi  $t = 0$ , dan prefix length  $l = 4$ .

Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{7}{7} + \frac{7}{8} + \frac{7-0}{7} \right) = \frac{1}{3} \left( \frac{56}{56} + \frac{49}{56} + \frac{56}{56} \right) = \frac{1}{3} \left( \frac{161}{56} \right) = \left( \frac{161}{168} \right) = 0.958$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.958 + (4 \times 0.1(1 - 0.958)) = 0.958 + (0.4 \times (0.042)) = 1$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “masalahh” dengan “masalah” adalah 1. Kata yang memiliki kemiripan dengan kata “masalahh” adalah kata “masalah” oleh karena itu kata yang tepat untuk mengkoreksi kata “masalahh” adalah “masalah”.

**Tabel 3. 17 Penentuan Ketertinggalan Huruf dan Karakter Ejaan Yang Salah**

		0	1	2	3	4	5	6
		m	a	s	a	l	a	h
0	m	✓						
1	a		✓					
2	s			✓				
3	a				✓			
4	l					✓		
5	h							✓

Pada tabel 3.17 diatas tanda ceklis (✓) digunakan untuk memberi tanda pada kedua kata yang memiliki huruf yang sama persis dan untuk mengetahui transposisi huruf. Pada kata ‘masalahh’ dan ‘masalah’ diatas ada kata yang memiliki posisi huruf yang sama persis dan memiliki transposisi huruf.  $|s1|$  = panjang string = 7 (masalah),  $|s2|$  = panjang string = 6 (masalh), dapat dilihat bahwa karakter yang sama persis  $m = 6$ , pada tabel diatas tidak ada karakter yang bertransposisi tetapi ketertinggalan huruf “a” maka nilai transposisi  $t = 0$ , dan prefix length  $l = 4$ .

Perhitungan nilai Jaro *distance* dihitung dengan persamaan 2.1 berikut.

$$dj = \frac{1}{3} \left( \frac{6}{7} + \frac{6}{6} + \frac{6-0}{6} \right) = \frac{1}{3} \left( \frac{36}{42} + \frac{42}{42} + \frac{42}{42} \right) = \frac{1}{3} \left( \frac{120}{42} \right) = \left( \frac{120}{126} \right) = 0.952$$

Perhitungan nilai Winkler *distance* dihitung dengan persamaan 2.3 berikut.

$$dw = 0.952 + (4 \times 0.1(1 - 0.952)) = 0.952 + (0.4 \times (0.048)) = 0.971$$

Dari hasil perhitungan, didapat bahwa nilai jaro winkler distance dari perbandingan kata “masalh” dengan “masalah” adalah 0. Kata yang memiliki kemiripan dengan kata “masalh” adalah kata “masalah” oleh karena itu kata yang tepat untuk mengkoreksi kata “masalahh” adalah “masalah”.

### 3.7. Sugesti Kata *Typo*

Pada proses ini kata yang telah dihitung dengan *jaro winkler distance* dijadikan kata sugesti. Kata yang dijadikan sugesti adalah kata yang memiliki nilai *jaro winkler distance* tertinggi. Berikut tabel sugesti kata.

**Tabel 3. 18 Sugesti Kata Typo dan Nilai Jaro winkler distance**

Token	Kamus	<i>Jaro winkler distance (dw)</i>
maslaah masalahh masalh	a	0.238
	ab	0.238
	aba	0.761
	...	...
	...	...
	abadi	0.395
	...	...
	majalah	0.849
	masalah	0.967
	masalah	1
	masalah	0.971

Tabel diatas adalah daftar kata hasil perhitungan *jaro winkler* untuk kata “maslaah”. Kata yang bernilai 1 menandakan kata tersebut tidak ada yang salah atau sama persis. Sedangkan kata yang bernilai kurang dari 1 artinya kata tersebut memiliki kesalahan ejaan atau *typo*. Tabel diatas kata ‘masalah’ memiliki nilai sebesar 0.967 maka kata ‘maslaah’ tepat jika digantikan dengan kata ‘masalah’ karena memiliki nilai yang mendekati nilai 1 yaitu 0.967. Pada kata “masalahh” dan “masalh” mempunyai kemiripan kata dengan kata “masalah” oleh karena itu kedua kata yang memiliki kesalahan duplikasi dan ketertinggalan huruf tepat jika dikoreksi dengan kata “masalah”.

### 3.8. Analisis Rencana Pengujian

Untuk pengujian peneliti menggunakan metode *precision* dan *recall*. Berikut penjelasan pengujiannya.

**Tabel 3. 19 Hasil Deteksi dan Koreksi Kesalahan Kata Data Uji**

No	Kata(tidak relevan) Yang Akan Dikoreksi	Hasil Deteksi		Sugesti Koreksi	
		Sistem	Manual	Sistem	Manual
1	maslaah	✓	✓	masalah	masalah
2	masalahh	✓	✓	masalah	masalah
3	masalh	✓	✓	masalah	masalah



4	komunikassi	✓	✓	komunikasi	komunikasi
5	singhatt	✓	✓	singkat	singkat
6	mkan	✓	✓	kan	makan

**Keterangan (Deteksi):** [✓] = Benar (Deteksi Benar)  
[×] = Salah (Deteksi Salah)

Sehingga untuk kasus perbaikan kesalahan ejaan diatas dapat dituliskan sebagai berikut:

**Tabel 3. 20 Recall and Precision Deteksi**

		Nilai Sebenarnya	
		TRUE	FALSE
Nilai Prediksi	TRUE	TP=6	FP=0
	FALSE	FN=0	TN=0

$$precision = \frac{TP}{TP + FP} = \frac{6}{6 + 0} = 1 = 100\%$$

$$recall = \frac{TP}{TP + FN} = \frac{6}{6 + 0} = 1 = 100\%$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{6 + 0}{6 + 0 + 0 + 0} = \frac{6}{6} = 1 = 100\%$$

**Tabel 3. 21 Recall and Precision Koreksi**

		Nilai Sebenarnya	
		TRUE	FALSE
Nilai Prediksi	TRUE	TP=5	FP=1
	FALSE	FN=0	TN=0

$$precision = \frac{TP}{TP + FP} = \frac{5}{5 + 1} = 0.83 = 83\%$$

$$recall = \frac{TP}{TP + FN} = \frac{5}{5 + 0} = 1 = 100\%$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{5 + 0}{5 + 0 + 1 + 0} = \frac{5}{6} = 0.83 = 83\%$$

Dapat disimpulkan bahwa pengujian yang dilakukan dari 6 kata yang memiliki kesalahan ejaan, sistem dapat mendeteksi 6 kata yang salah dan mengkoreksi 5 kata yang relevan. Sehingga nilai *precision* (kepersisan atau kecocokan) untuk deteksi dan koreksi mendapatkan 100% dan 83% *Recall* (*penemuan-kembali*) mendapatkan 100% dan 83%, dan *Accuracy* 100% dan 83%.

### 3.9. Menghitung Nilai Jawaban Esai Dengan *Vector Space Model*

Untuk melakukan analisis ini di butuhkan beberapa sampel data uji. Data uji yang digunakan berupa jawaban anak dan kunci jawaban. Ambil contoh 3 jawaban anak yang akan digunakan.

**Tabel 3. 22 Jawaban Anak**

Jawaban Anak
D1 : Langkah-langkah atau metode untuk menyelesaikan suatu masalah
D2 : Prosedur yang tersusun sistematis untuk menyelesaikan masalah
D3 : Tehnik untuk menyelesaikan suatu masalah

Jadi total jawaban anak ada 3 yang akan dilakukan penilaian jawaban anak dengan kunci jawaban.

**Tabel 3. 23 Kunci Jawaban**

Kunci Jawaban
Q : Langkah-langkah atau metode untuk menyelesaikan suatu masalah

Apakah jawaban siswa sudah relevan dengan kunci jawaban yang ada ?

Untuk menjawab pertanyaan di atas perlu dilakukan suatu proses perhitungan pembobotan atau perankingan dari masing-masing jawaban anak tersebut sehingga dari hasil pembobotan atau perankingan tersebut akan terlihat jawaban anak yang mana paling benar. Di dalam penerapan metode *Vector Space Model* ada beberapa tahapan proses pengolahan data terlebih dahulu antara lain:

**Tabel 3. 24 Perhitungan Vector Sapce Model (Jumlah  $tf=tf_{ij}$ )**

<i>Terms</i>	Jumlah $tf=tf_{ij}$				<i>df<sub>i</sub></i>
	<b>q</b>	<b>d1</b>	<b>d2</b>	<b>d3</b>	
langkahlangkah	1	1	0	0	2
metode	1	1	0	0	2

menyelesaikan	1	1	1	1	4
masalah	1	1	1	1	4
prosedur	0	0	1	0	1
tersusun	0	0	1	0	1
sistematis	0	0	1	0	1
teknik	0	0	0	1	1

Tabel diatas kata jawaban(d) dengan kata kunci jawaban(q). Jika dalam jawaban terdapat kata yang sama dalam kunci jawaban maka jawaban akan diberi nilai satu. Dan akan dijumlahkan nilai setiap jawaban dengan kata kunci. Dengan  $tf$  adalah *term frequency*, dan  $tf_{ij}$  adalah banyaknya kemunculan term  $t_i$  dalam dokumen  $d_j$ , *Term frequency* (tf) dihitung dengan menghitung banyaknya kemunculan *term*  $t_i$  dalam dokumen  $d_j$ .

**Tabel 3. 25 Perhitungan Vector Sapce Model**

		$idf_i = \log \frac{N}{df_i}$
<i>Terms</i>	$N/df_i$	$idf_i$
langkahlangkah	$4/2 = 2$	0.3010
metode	$4/2 = 2$	0.3010
menyelesaikan	$4/4 = 1$	0
masalah	$4/4 = 1$	0
prosedur	$4/1 = 4$	0.6021
tersusun	$4/1 = 4$	0.6021
sistematis	$4/1 = 4$	0.6021
teknik	$4/1 = 4$	0.6021

Dengan  $idf_i$  adalah *inverse document frequency*, N adalah jumlah dokumen yang diambil oleh sistem, dan  $df_i$  adalah banyaknya dokumen dalam koleksi dimana term  $t_i$  muncul di dalamnya, maka Perhitungan  $idf_i$  digunakan untuk mengetahui banyaknya term yang dicari ( $df_i$ ) yang muncul dalam dokumen lain yang ada pada database.

**Tabel 3. 26 Perhitungan Vector Sapce Model**

<i>Terms</i>	<b>Bobot</b>			
	$W_{ij} = tf_{ij} \cdot \log \frac{N}{df_i}$			
	q	d1	d2	d3

langkahlangkah	0.3010	0.3010	0	0
metode	0.3010	0.3010	0	0
menyelesaikan	0	0	0	0
masalah	0	0	0	0
prosedur	0	0	0.6021	0
tersusun	0	0	0.6021	0
sistematis	0	0	0.6021	0
tehnik	0	0	0	0.6021

Dengan  $W_{ij}$  adalah bobot dokumen,  $N$  adalah Jumlah dokumen yang terambil oleh sistem,  $tf_{ij}$  adalah banyaknya kemunculan term  $t_i$  pada dokumen  $d_j$ , dan  $df_i$  adalah banyaknya dokumen dalam koleksi dimana term  $t_i$  muncul di dalamnya. Bobot dokumen ( $W_{ij}$ ) dihitung untuk didapatkannya suatu bobot hasil perkalian atau kombinasi antara *term frequency* ( $tf_{ij}$ ) dan *Inverse Document Frequency* ( $idf$ ). Dari tabel di atas persamaan 2.4, persamaan 2.5 dan persamaan 2.6:

### 3.9.1. Menghitung *Similarity Analysis*

Perhitungan Jarak Dokumen dengan menggunakan persamaan 28:

$$|d1| = \sqrt{0.3010^2 + 0.3010^2} = \sqrt{0.181202} = 0.4257$$

$$|d2| = \sqrt{0.6021^2 + 0.6021^2 + 0.6021^2} = \sqrt{1.08757323} = 1.0429$$

$$|d3| = \sqrt{0.6021^2} = \sqrt{0.36252441} = 0.6021$$

Perhitungan Jarak *Query* dengan menggunakan persamaan 2.7.

$$\|q\| = \sqrt{0.3010^2 + 0.3010^2} = \sqrt{0.181202} = 0.4257$$

Menghitung *index terms* dari dokumen dan *query* menggunakan persamaan 2.9.

$$q * q1 = q1d11 + q2d12 + q3d13 + q4d14 + q5d15 + q6d16 + q7d17 + q8d18$$

$$= 0.3010*0.3010 + 0.3010*0.3010 + 0*0 + 0*0 + 0*0 + 0*0 + 0*0 + 0*0$$

$$= 0.1812$$

$$\begin{aligned}
 q * d2 &= q1d21 + q2d22 + q3d23 + q4d24 + q5d25 + q6d26 + \\
 & q7d27 + q8d28 \\
 &= 0.3010*0+0.3010*0+0*0 +0*0+0*0.6021+0*0.6021+0*0.6021+0*0 \\
 &= 0
 \end{aligned}$$

$$\begin{aligned}
 q * d3 &= q1d31 + q2d32 + q3d33 + q4d34 + q5d35 + q6d36 + \\
 & q7d37 + q8d38 \\
 &= 0.3010*0+0.3010*0+0*0 +0*0+0*0 +0*0+0*0+0*0.6021= 0
 \end{aligned}$$

Menghitung *similaritas* dokumen dan meranking menggunakan persamaan 2.10.

$$sim(q, d) \frac{q*dj}{||q||*||dj||} = \frac{0.1812}{0.4257* 0.4257} = 0.9999$$

$$sim(q, d) \frac{q*d2}{||q||*||d2||} = \frac{0}{0.4257* 1.0429} = 0$$

$$sim(q, d) \frac{q*d3}{||q||*||d3||} = \frac{0}{0.4771* 0.6021} = 0$$

Dari proses similaritas di atas dapat diambil ranking dari setiap dokumen yaitu :

**Tabel 3. 27 Perankingan**

Rank 1 : Dokumen(jawaban anak) 1 = 0.9999
Rank 2 : Dokumen(jawaban anak) 2 = 0
Rank 3 : Dokumen(jawaban anak) 3 = 0

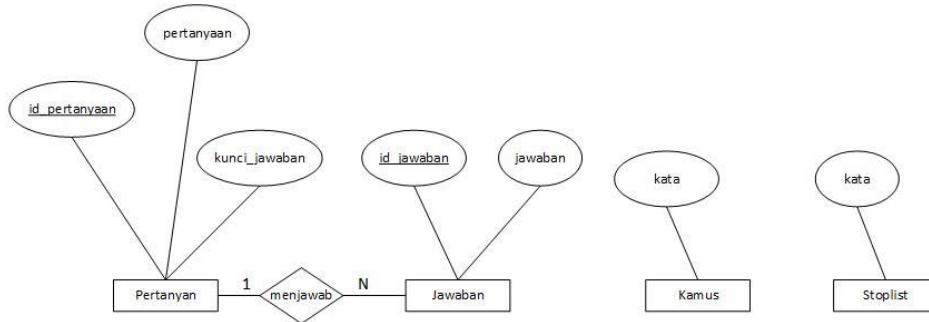
Bisa dilihat dari hasil ranking di atas yang mempunyai ranking tertinggi adalah jawaban anak nomer satu yaitu dengan 0.9999 dan dari hasil jawaban anak mempunyai kemiripan dengan kunci jawaban sehingga bisa dikatakan bahwa jawaban anak tersebut bisa diberikan skor atau penilaian.

### 3.10. Analisis Basis Data

Analisis basis data merupakan cara untuk menggambarkan data yang digunakan pada sistem yang akan dibangun dengan menggambarkan ERD dan menjelaskan dalam kamus data ERD.

### 3.10.1. Entity Relationship Diagram

Berikut gambaran sistem yang dibangun pada skema *Entity Relationship Diagram* (ERD).



**Gambar 3. 4 Entity Relationship Diagram (ERD)**

#### 3.10.1.1. Kamus Data ERD

Tabel 3.28 merupakan penjelasan gambar ERD di atas terdapat pada Kamus Data ERD dibawah ini.

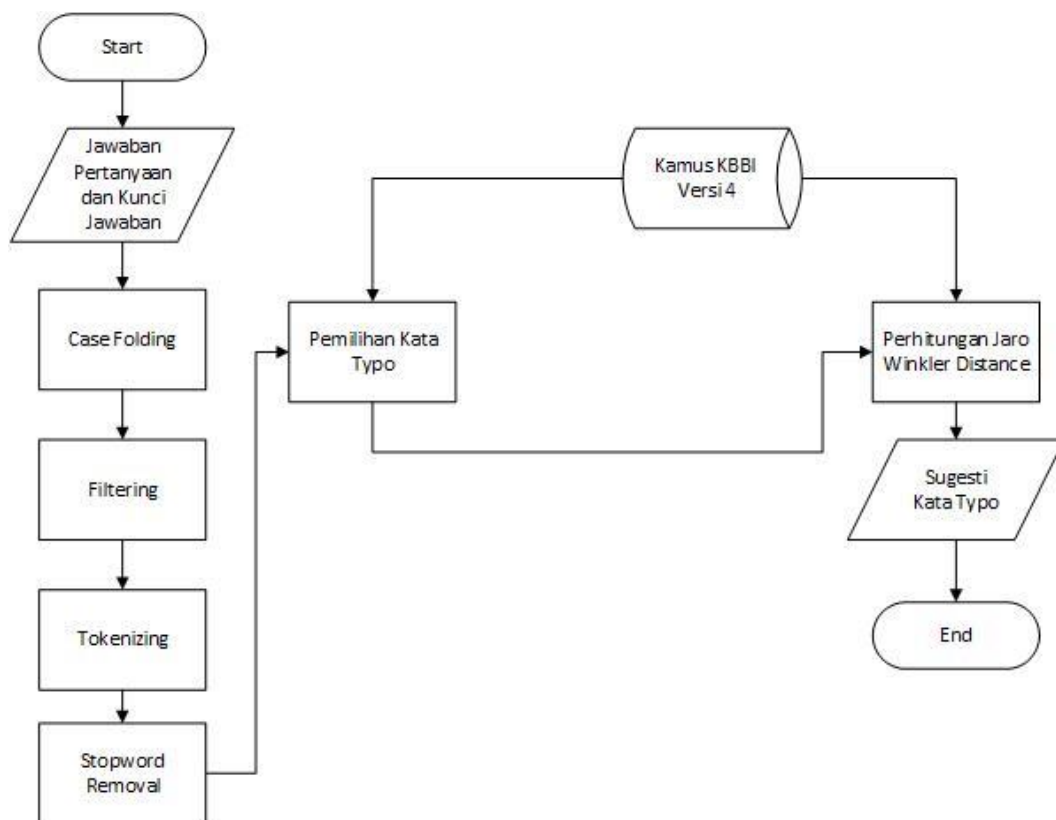
**Tabel 3. 28 Kamus Data ERD**

Entitas	Atribut	Uraian	Tipe	PK	FK
Pertanyaan	id_pertanyaan, pertanyaan, kunci_jawaban	id_pertanyaan, kumpulan pertanyaan	int(11), varchar(500)	<u>id_pertanyaan</u>	-
Jawaban	id_jawaban, jawaban, id_pertanyaan	id_jawaban, kumpulan jawaban dari pertanyaan	int(11) varchar(500)	<u>id_jawaban</u>	id_pertanyaan
Kamus	Kata	Kumpulan kata dalam kamus	varchar(30)	kata	-
Stoplist	Kata	Berisi kata-kata stoplist	varchar(20)	kata	-

### 3.11. Perancangan Prosedural

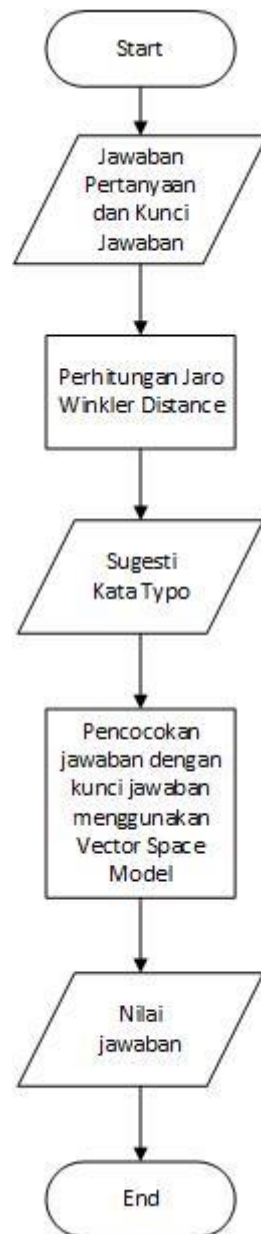
Perancangan Prosedural ini berisi urutan langkah-langkah penyelesaian masalah. Ini berarti algoritma adalah proses yang prosedural. Bahasa pemrograman yang mendukung pembuatan program sebagai kumpulan prosedur [29].

Berikut adalah gambar 3.5 proses mengatasi *typo* dengan jaro winkler distance.



**Gambar 3. 5** Proses Mengatasi *Typo*

Berikut adalah gambar 3.6 proses *essay scoring* dengan vector space model.



**Gambar 3. 6 Proses *Essay Scoring***

### 3.11.1. *Sourcode*

*Sourcode* adalah bahasa yang digunakan untuk menyederhanakan penulisan algoritma. *Sourcode* bukan bahasa pemrograman. *Sourcode* notasi yang menyerupai bahasa pemrograman tingkat tinggi. Keuntungan menggunakan notasi *sourcode* adalah kemudahan mengkonversinya lebih tepat yang disebut



mentranslasi ke notasi bahasa pemrograman, karena terdapat korespondensi antara setiap sourcode dengan notasi bahasa pemrograman [30].

**Tabel 3. 29 Sourcode Distance Jaro Winkler Dan Kata Yang Sama**

```
function getCommonCharacters(str1,str2:string)
DEKLARASI
a,b,I,j : integer
distance : integer
temp_str : string
commonchr : string
nomatch : boolean
ALGORITMA
a <- strlen(str1)
b <- strlen(str2)
temp_str <- str2
distance <- floor(min(a,b)/2.0)
{jarak yang dianggap matching dari teori Winkler}
{mencari karakter yang sama dari kedua buah string}
  for I <- 0 to a do
    nomatch <- true
    for j <- max (0,i-distance) to nomatch and
      min (i+distance+1,b)
      if temp_str[j] = str1[i]
        nomatch <- false
        commonchr <- commonchr,str1[i]
        temp_str <- delete(temp_str[j])
      endif
    endfor
  endfor
endfunction
```

Berikut adalah sourcode rumus mencari  $t =$  transposisi, nilai jaro, dan prefix pada tabel 3.30.

**Tabel 3. 30 Sourcode Mencari  $t =$  Transposisi, Nilai Jaro, Dan Prefix**

```
mencari t
t=0;
for(i=0;i<min(strlen(s1),strlen(s2));i++){
```

```

if(s1[i]!=s2[i]){
    t++;
t=t/2; hasil['t']= t;
//hitung nilai jaro
jaro=((s1len/k1len)+( s2len/k2len)+((s1len-t)/s1len))/3); hasil['jaro']=jaro;
//cari prefix
pref=0;
for(i=0;i<min(k1len,k2len);i++){
    if(kata1[i]==kata2[i]){
        pref++;
    }
    else break;
}

```

Berikut adalah sourcode menghitung nilai winkler distance dan function cari kata yang sama pada tabel 3.31.

**Tabel 3. 31 Sourcode Hitung Nilai Winkler Distance**

```

//hitung nilai winkler
winkler=jaro+(pref*0.1*(1-jaro)); hasil['winkler']=winkler;
hasil;
carikarsama(str1, str2, distance){
    str2len=strlen(str2);
    karsama="";
    if(!empty(str1)){
        foreach (str_split(str1) as i => char) {
            cari=strpos(str2,char,i <= distance ? 0 : min(i-distance,str2len));
            if(cari!=false && cari<=i+distance+1){
                karsama.=;
            }
        }
    }
    karsama;
}

```

Berikut adalah sourcode perhitungan Vector Space Model pada tabel 3.32.

**Tabel 3. 32 Sourcode Perhitungan Vector Space Model**

Perhitungan Vector Space Model

```

term=array();
foreach (ktoken2 as kt2) {
    if(!in_array(kt2, term)){
        term[]=kt2;
    }
}

```

```

foreach (token3 as tt) {
  foreach (tt as t) {
    if(!in_array(t, term)){
      $term[]=$t;

tf=array();
i=0;
foreach (term as t) {
  if(in_array(t, ktoken2)){
    if(tf[i][0]) tf[i][0]+=1;
    else tf[i][0]=1;

    i++;

i=0;
foreach (term as t) {
  j=1;
  foreach (token3 as tt) {
    if(in_array(t, tt)){
      if(tf[i][j]) tf[i][j]+=1;
      else tf[i][j]=1;

      j++;

      i++;

foreach (tf as $t) {
  dfi[]=array_sum($t);

```

Jumlah  $tf=tf$   
 Bobot  $W * \log N/dfi$

Berikut adalah sourcode hitung similarity analysis pada tabel 3.33.

**Tabel 3. 33 Sourcode Similarity Analysis dan Perankingan**

Menghitung Similarity Analysis

```

Perhitungan Jarak Dokumen
jdok=array();

for(i=1;i<=count(token3);i++){
  echo "|d".(i)." = &radic;<span style='text-
str=";
  jdok[i]=0;

```

```

for(j=0;j<count(wij);j++){
  if(wij[j][i]!=0) {
    str.="(."wij[j][i].")<sup>2</sup> +
    jdok[i]+=pow(wij[j][i], 2);
  }
}

echo rtrim(str," +");

echo "|q| = &radic;<span style='text-

echo "Menghitung index terms dari dokumen dan

Menghitung similaritas dokumen
for(i=1;i<=count(token3);i++){

```

Perhitungan Jarak Query

query

Perangkingan  
Dokumen(jawaban anak)

### 3.11.2. Skema Relasi

Sebuah database relasional terdiri dari koleksi dari tabel-tabel, yang masing-masing diberikan nama yang unik. Sebuah baris dalam tabel merepresentasikan sebuah keterhubungan atau relationship dari beberapa nilai yang ada [31].



**Gambar 3. 7 Skema Relasi**

### 3.11.3. Analisis Kebutuhan Non Fungsional

Analisis Kebutuhan Non Fungsional digunakan untuk membantu kebutuhan di luar sistem yang dibutuhkan untuk memperingankan sistem yang akan dibangun. Analisis yang dipakai diantaranya Analisis Kebutuhan Pengguna, Analisis Kebutuhan Perangkat Keras, dan Analisis kebutuhan Perangkat Lunak.

#### 3.11.3.1. Analisis Kebutuhan Pengguna

Pengguna sistem ini yaitu penguji yang dengan melakukan pengujian terhadap hasil koreksi kesalahan ejaan pada sistem essay scoring yang kemudian sistem menampilkan koreksi kesalahan ejaan. Agar sistem dapat berjalan dengan baik pengguna harus memiliki kemampuan sebagai berikut.

**Tabel 3. 34 Analisis Kebutuhan Pengguna**

No	Kebutuhan
1	Mengoperasikan Komputer
2	Mampu menggunakan mouse dan keyboard
3	Tidak buta huruf

#### 3.11.3.2. Analisis Kebutuhan Perangkat Keras

Spesifikasi perangkat keras yang diperlukan untuk menjalankan sistem yaitu:

**Tabel 3. 35 Spesifikasi Minimum Perangkat Keras**

Perangkat Keras	Spesifikasi Minimum
Prosesor	AMD atau Intel setara Pentium 4 3.0 Ghz
RAM	Minimal RAM 1 GB
Hardisk	Minimal 120 GB
Monitor	Resolusi minimal 1024 x 768 pixel

#### 3.11.3.3. Analisis Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang diperlukan agar sistem dapat berfungsi dengan baik sebagai berikut.

**Tabel 3. 36 Spesifikasi Minimum Perangkat Lunak**

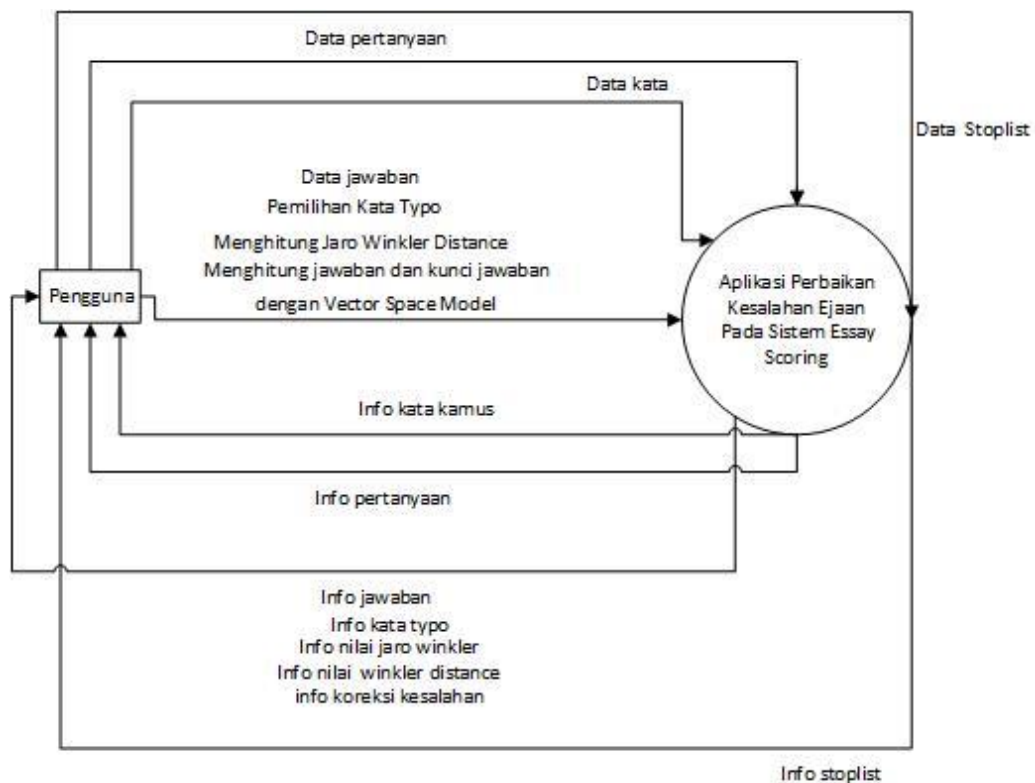
Perangkat Lunak	Spesifikasi Minimum
Sistem Operasi	AMD atau Intel setara Pentium 4 3.0 Ghz
Web Browser	Google Chrome versi 46
Webserver	XAMPP for Windows Version 1.7.5

### 3.11.4. Analisis Kebutuhan Fungsional

Analisis ini dilakukan untuk menganalisis proses-proses yang akan diterapkan dalam aplikasi kesalahan ejaan pada sistem *essay scoring* yang akan dibangun menggunakan Diagram Konteks, DFD, dan Spesifikasi Proses.

#### 3.11.4.1. Diagram Konteks

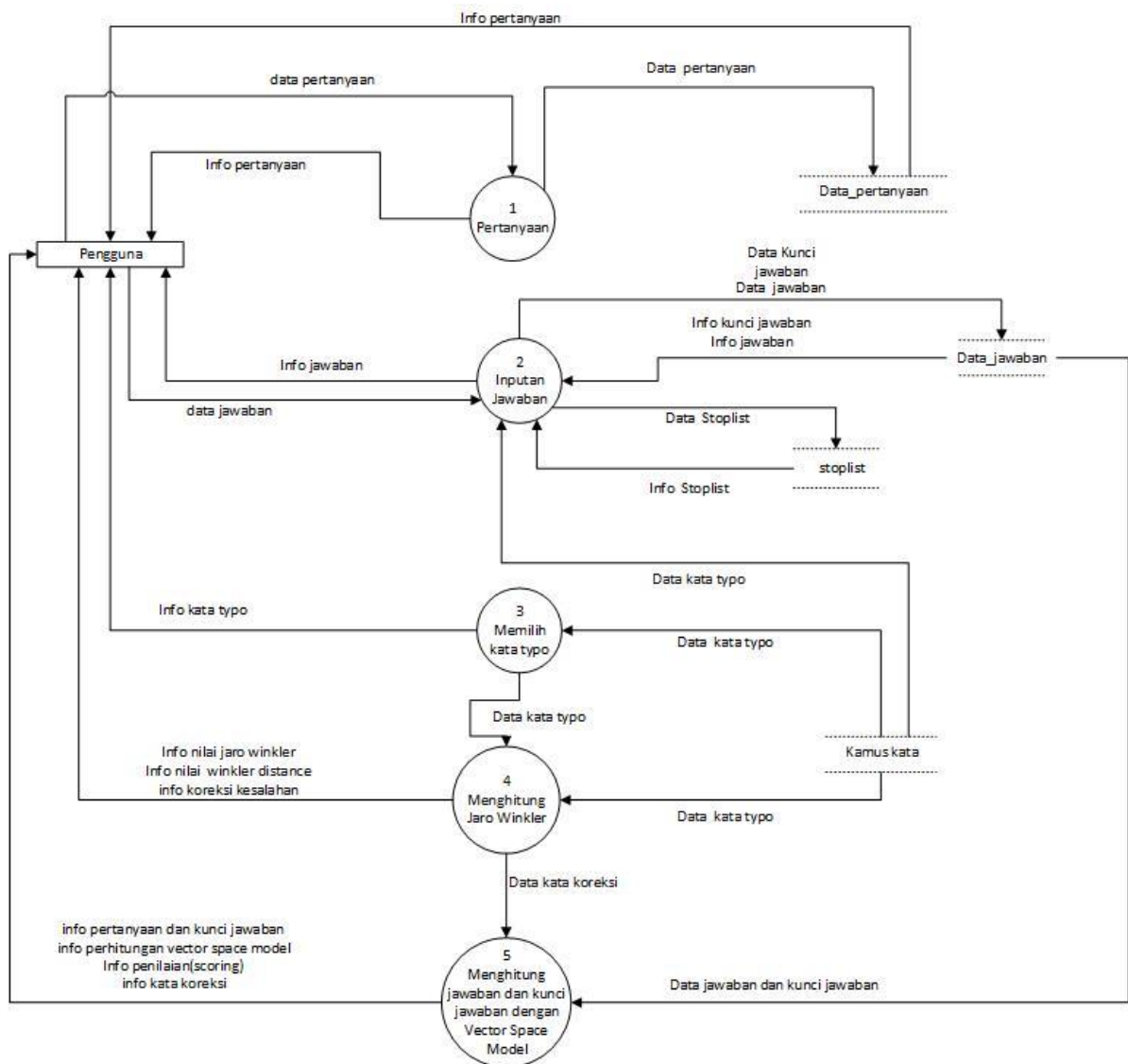
Diagram Konteks ini menggambarkan ruang lingkup pembangunan aplikasi kesalahan ejaan pada sistem *essay scoring* dengan Pengguna sebagai entitas luar atau pengguna sistem. Berikut gambar diagram konteks pada gambar 3.8.



Gambar 3. 8 Diagram Konteks

### 3.11.4.2. Data Flow Diagram (DFD) Level 1

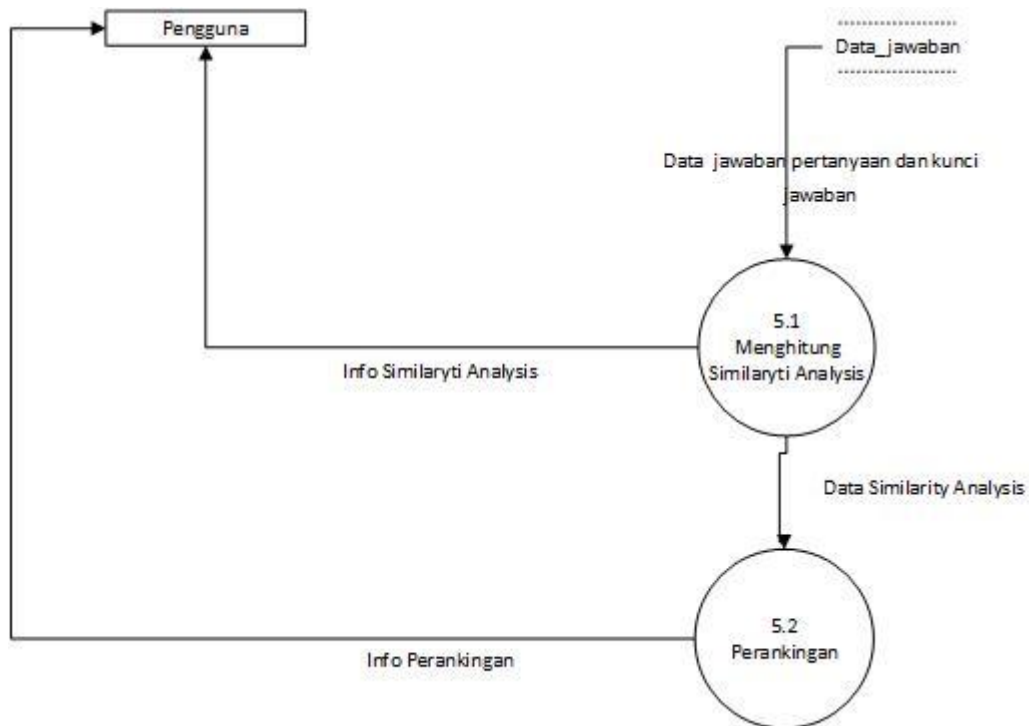
Data Flow Diagram Level 1, menggambarkan aliran data yang merupakan turunan dari DFD Konteks. Berisi proses aliran data teks pertanyaan, jawaban, kamus kata, dan pertanyaan dan jawaban. Dan menggambarkan aliran data yang ada pada aplikasi seperti proses Deteksi dan Pengkoreksian Kesalahan Ejaan serta Scoring. Berikut adalah gambar 3.9 DFD level 1.



Gambar 3. 9 DFD Level 1

### 3.11.4.3. Data Flow Diagram Level 2 Menghitung Jawaban dan Kunci Jawaban dengan *Vector Space Model*

Pada DFD tahap ini menggambarkan aliran data proses menghitung jawaban dan kunci jawaban dengan *vector space* model. Berikut adalah gambar DFD 3.10.



**Gambar 3. 10 DFD Level 2 Proses Menghitung Jawaban dan Kunci Jawaban dengan (VSM).**

### 3.11.4.4. Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan proses aliran yang ada pada DFD.

**Tabel 3. 37 Spesifikasi Proses**

No	Proses	Keterangan
1.	No. Proses	1
	Nama Proses	Pertanyaan
	Source (Sumber)	Pengguna
	Input	Data inputan pertanyaan
	Output	Info teks uji
	Logika Proses	1. Pengguna memasukan pertanyaan ke dalam sistem.



No	Proses	Keterangan
		2. Sistem menyimpannya data pertanyaan di <i>database</i> . 3. Keluaran berupa info pertanyaan
No	Proses	Keterangan
2.	No. Proses	2
	Nama Proses	Inputan Jawaban
	Source (Sumber)	Pengguna
	Input	Data inputan jawaban
	Output	Info jawaban
	Logika Proses	1. Pengguna menginputkan data jawaban ke dalam sistem. 2. Sistem menyimpannya data teks jawaban di <i>database</i> . 3. Keluaran info jawaban
No	Proses	Keterangan
3.	No. Proses	3
	Nama Proses	Memilih Kata Typo
	Source (Sumber)	Proses Pemilihan kata <i>typo</i>
	Input	Data inputan jawaban
	Output	Data kata <i>typo</i>
	Logika Proses	1. Proses pemilihan kata typo dengan mencocokkan kata hasil token dengan kata yang ada didalam kamus kata 2. Keluaran berupa info kata typo
No	Proses	Keterangan
4.	No. Proses	4
	Nama Proses	Menghitung Jaro Winkler
	Source (Sumber)	Proses Perhitungna jaro winkler distance
	Input	Data Inputan Jawaban
	Output	info nilai winkler distance, info koreksi kesalahan
	Logika Proses	1. Proses perhitungang dengan cara menghitung setiap kata yang dicocokkan dalam kamus dengan jaro winkler distance kemudian diberikan nilai setiap kata. 2. Keluaran berupa info nilai winkler distance, info koreksi kesalahan
No	Proses	Keterangan
5.	No. Proses	5
	Nama Proses	Menghitung jawaban dan kunci jawaban dengan vektor space model
	Source (Sumber)	Proses mencocokkan pertanyaan dan jawaban untuk memberikan penilaian

No	Proses	Keterangan
	<i>Input</i>	Pertanyann dan jawaban
	<i>Output</i>	Info pertanyaan dan kunci jawaban, info nilai winkler distance, info koreksi kesalahan
	Logika Proses	<ol style="list-style-type: none"> <li>1. Proses perhitungan dilakukan dengan menggunakan metode vsm</li> <li>2. Setiap jawaban dan kunci jawaban di cocokan lali dihitung menggunakakn vsm</li> <li>3. Keluaran berupa info nilai jaro winkler, info nilai winkler distance, info koreksi kesalahan koreksi</li> </ol>

#### 3.11.4.5. Kamus Data DFD

Kamus data digunakan untuk mendeskripsikan data-data yang terdapat dalam data flow diagram. Berikut kamus data pada aplikasi kesalahan ejaan pada sistem *essay scoring*.

**Tabel 3. 38 Kamus Data DFD**

No	Detail	Keterangan
1	Nama	Data Pertanyaan
	Digunakan pada saat	Proses 1 Pertanyaan
	Deskripsi	Data awal berupa pertanyaan yang dijadikan inputan
	Struktur data	Kalimat
	Penjelasan struktur data: Kalimat	[0..9   A..Z   a..z   symbol]
No	Detail	Keterangan
2	Nama	Data Jawaban
	Digunakan pada saat	Proses 2 Inputan jawaban
	Deskripsi	Dokumen awal berupa jawaban
	Struktur data	Kalimat
	Penjelasan struktur data: Kalimat	[0..9   A..Z   a..z   symbol]
No	Detail	Keterangan
3	Nama	Kamus Kata
	Digunakan pada saat	Proses 3 memilih kata typo Proses 4 menghitung jaro winkler distance
	Deskripsi	Proses pencocokan kata yang memiliki <i>typo</i> dengan kamus kata
	Struktur data	Kalimat
	Penjelasan struktur data: Kalimat	[a..z   symbol]
No	Detail	Keterangan

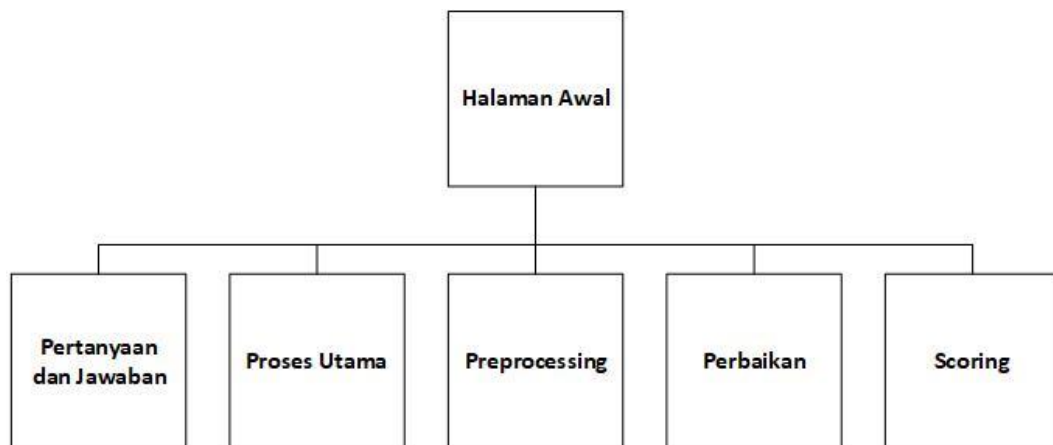
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
4	Nama	Stoplist
	Digunakan pada saat	Proses 2 inputan jawaban yang telah di stopword removal
	Deskripsi	Untuk menghilangkan kata yang tidak penting
	Struktur data	Kata
	Penjelasan struktur data: Kata	[ a..z   symbol]
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
5	Nama	Pemilihan kata typo
	Digunakan pada saat	Proses 3 memilih kata typo
	Deskripsi	Proses pemilihan kata typo dari jawaban yang telah di inputkan
	Struktur data	Kata
	Penjelasan struktur data: Kata	[a..z   symbol]
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
6	Nama	Menghitung jaro winkler distance
	Digunakan pada saat	Proses 5 menghitung jaro winkler
	Deskripsi	Proses menghitung kata yang memiliki typo dengan algoritma jaro winkler distance
	Struktur data	Kata
	Penjelasan struktur data: Kata	[a..z   symbol]
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
7	Nama	Menghitung jawaban dan kunci jawaban dengan vector space model
	Digunakan pada saat	Proses 5 Menghitung jawaban dan kunci jawaban dengan vector space model
	Deskripsi	Proses menghitung jawaban dan kunci jawaban yang akan diberi scoring
	Struktur data	Kata
	Penjelasan struktur data: Kata	[a..z   space]
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
8	Nama	Daftar Pertanyaan
	Digunakan pada saat	Proses 1 Pertanyaan
	Deskripsi	Daftar pertanyaan yang telah dimasukkan kedalam sistem
	Struktur data	Kata
	Penjelasan struktur data: Kata	[0..9   A..Z   a..z   symbol]
<b>No</b>	<b>Detail</b>	<b>Keterangan</b>
9	Nama	Daftar Jawaban
	Digunakan pada saat	Proses 2 Inputan Jawaban

No	Detail	Keterangan
	Deskripsi	Kata jawaban yang telah dimasukan kedalam sistem
	Struktur data	Kata
	Penjelasan struktur data: Kata	[0..9   A..Z   a..z   symbol]
No	Detail	Keterangan
10	Nama	Kunci Jawaban
	Digunakan pada saat	Proses 2 Inputan Jawaban
	Deskripsi	Data yang digunakan sebagai kunci jawaban untuk acuan jawaban-jawaban siswa
	Struktur data	Kata
	Penjelasan struktur data: Kata	[a..z   space]
No	Detail	Keterangan
11	Nama	Hitung kata typo
	Digunakan pada saat	Proses 4 Menghitung Jaro Winkler
	Deskripsi	Porses aliran data untuk menghitung kata typo
	Struktur data	Kata
	Penjelasan struktur data: Kata	[0..9   A..Z   a..z   symbol]
No	Detail	Keterangan
12	Nama	Info nilai jaro winkler, info winkler distance, dan info koreksi kesalahan
	Digunakan pada saat	Proses 4 Menghitung Jaro Winkler
	Deskripsi	Info berupa nilai hasil dari perhitungan dan hasil koreksi kata yang typo dengan kata yang benar
	Struktur data	Nilai Kata
	Penjelasan struktur data: Nilai Kata	[0..9] [a..z]
No	Detail	Keterangan
13	Nama	Menghitung jawaban pertanyaan dan kunci jawaban, hitung simlaryti analysis, perangkingan
	Digunakan pada saat	Proses 5 Menghitung jawaban pertanyaan dan kunci jawaban dengan VSM
	Deskripsi	Proses aliran data untuk menuju ke proses 5, proses perhitungan simlaryti analysis, dan perangkingan(scoring)
	Struktur data	Kata

No	Detail	Keterangan
		Skor
	Penjelasan struktur data: Teks	[a..z] [0..9]

### 3.11.5. Perancangan Struktur Menu

Perancangan Struktur Menu ini menggambarkan menu proses kerja pada aplikasi perbaikan kesalahan ejaan pada sistem essay scoring. Berikut gambar perancangan struktur menu pada gambar 3.11.



**Gambar 3. 11 Perancangan Menu Proses**

### 3.11.6. Perancangan Antarmuka

Perancangan antarmuka adalah tampilan aplikasi yang dibuat yang akan digunakan oleh pengguna sebagai bentuk interaksi pengguna dengan sistem yang akan dibangun. Berikut perancangan antarmuka yang akan dibuat pada aplikasi kesalahan ejaan pada sistem *essay scoring*.

Nama Tampilan: L01	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Pilih Pertanyaan:</p> <div style="border: 1px solid black; padding: 5px; display: inline-block;">Pilih Pertanyaan... ▼</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin: 10px 0;">Proses</div>	<p>Keterangan:</p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 12 Antarmuka Halaman Home**

Nama Tampilan: L02	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Pilih Pertanyaan:</p> <div style="border: 1px solid black; padding: 5px; display: inline-block;">Pilih Pertanyaan... ▼</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin: 10px 0;">Proses</div> <p style="text-align: center;"> <a href="#">Pertanyaan dan Jawaban</a>   <a href="#">Preprocessing</a>   <a href="#">Perbaikan</a>   <a href="#">Scoring</a> </p> <p style="text-align: center;">Pertanyaan dan Jawaban</p> <p>Apa yang dimaksud dengan Algoritma ?</p> <p>Q: Langkah-langkah atau metode untuk meyelesaikan suatu masalah</p> <p>D1: Prosedur yang tersusun sistematis untuk menyelesaikan masalah</p> <p>D2: ....</p> <p>D3: ....</p>	<p>Keterangan:</p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 13 Antarmuka Halaman Pertanyaan dan Jawaban**

Nama Tampilan: L03

Perbaiki Kesalahan Ejaan Pada Esay Scoring [Pertanyaan dan Jawaban](#) [Proses Utama](#)

Pilih Pertanyaan:

Pilih Pertanyaan... ▼

**Proses**

[Jawaban dan Pertanyaan](#) [Preprocessing](#) [Perbaikan](#) [Scoring](#)

Preprocessing

Case Folding

Filtering

Tokenizing

Stopword Removal

Keterangan:

1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01
2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)
3. Pilih menu Preprocessing maka akan menuju ke tampilan L03
4. Klik menu Perbaikan maka akan menuju ke tampilan L04

Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih

**Gambar 3. 14** Antarmuka Halaman Preprocessing

Nama Tampilan: L04

Perbaiki Kesalahan Ejaan Pada Esay Scoring [Pertanyaan dan Jawaban](#) [Proses Utama](#)

Pilih Pertanyaan:

Pilih Pertanyaan... ▼

**Proses**

[Jawaban dan Pertanyaan](#) [Preprocessing](#) [Perbaikan](#) [Scoring](#)

Proses Perbaikan

langkahlangkah atau metode untuk menyelesaikan suatu masalah

Token	Panjang Kata 1	Koreksi	Panjang Kata 2	Jumlah Karakter Sama	Jumlah Transpos	dj	p	wd
ma	7	ma	7	7	7	0.	3	0.
sla		sal				95		96
ah		ah				2		67

Keterangan:

1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01
2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)
3. Pilih menu Preprocessing maka akan menuju ke tampilan L03
4. Klik menu Perbaikan maka akan menuju ke tampilan L04
5. Klik menu Scoring maka akan menuju ke tampilan L05

Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih

**Gambar 3. 15** Antarmuka Halaman Proses Perbaikan

Nama Tampilan: L05	
Perbaiki Kesalahan Ejaan Pada Essay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Pilih Pertanyaan:</p> <p>Pilih Pertanyaan... ▼</p> <p><b>Proses</b></p> <p><a href="#">Jawaban dan Pertanyaan</a> <a href="#">Preprocessing</a> <a href="#">Perbaikan</a> <a href="#">Scoring</a></p> <p>Essay Scoring</p> <p>Jawaban &amp; Kunci Jawaban</p> <p>Perhitungan Space Model</p> <p>Menghitung Similarity Analysis</p> <p>Perangkingan</p>	<p>Keterangan:</p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> <li>4. Klik menu Perbaikan maka akan menuju ke tampilan L04</li> <li>5. Klik menu Scoring maka akan menuju ke tampilan L05</li> <li>6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 16** Antarmuka Halaman *Essay Scoring*

Nama Tampilan: L02	
Perbaiki Kesalahan Ejaan Pada Essay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Pilih Pertanyaan:</p> <p>Pilih Pertanyaan... ▼</p> <p><b>Proses</b></p> <p><a href="#">Pertanyaan dan Jawaban</a> <a href="#">Preprocessing</a> <a href="#">Perbaikan</a> <a href="#">Scoring</a></p> <p>Pertanyaan dan Jawaban</p> <p>Apa yang dimaksud dengan Algoritma ?  Q: Langkah-langkah atau metode untuk meyelesaikan suatu masalah  D1: Prosedur yang tersusun sistematis untuk menyelesaikan masalah  D2: ....  D3: ....</p>	<p>Keterangan:</p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 17** Antarmuka Halaman Kelola Pertanyaan dan Jawaban



Nama Tampilan: L07	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Kelola Pertanyaan dan Jawaban</p> <div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Tambah Pertanyaan</p> <p>Pertanyaan</p> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Pilih Pertanyaan... ▼</div> <p>Jawaban</p> <div style="border: 1px solid black; height: 20px; margin-bottom: 5px;"></div> <div style="display: flex; justify-content: space-around;"> <span>Simpan</span> <span>Kembali</span> </div> </div>	<p><b>Keterangan:</b></p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> <li>4. Klik menu Perbaiki maka akan menuju ke tampilan L04</li> <li>5. Klik menu Scoring maka akan menuju ke tampilan L05</li> <li>6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06</li> <li>7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07</li> <li>8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 18 Antarmuka Tambah Pertanyaan**

Nama Tampilan: L08	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Kelola Pertanyaan dan Jawaban</p> <div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Ubah Pertanyaan</p> <p>Pertanyaan</p> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Apa yang dimaksud.... ?</div> <p>Jawaban</p> <div style="border: 1px solid black; height: 20px; margin-bottom: 5px;">.....</div> <div style="display: flex; justify-content: space-around;"> <span>Simpan</span> <span>Kembali</span> </div> </div>	<p><b>Keterangan:</b></p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> <li>4. Klik menu Perbaiki maka akan menuju ke tampilan L04</li> <li>5. Klik menu Scoring maka akan menuju ke tampilan L05</li> <li>6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06</li> <li>7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07</li> <li>8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi</li> <li>9. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Delete Pertanyaan L09 lalu klik ubah pada menu Aksi</li> </ol> <p>Ukuran layar: 1366x768 Font: Arial Warna Background: Putih</p>

**Gambar 3. 19 Antarmuka Halaman Ubah Pertanyaan**

Nama Tampilan: L09

Perbaiki Kesalahan Ejaan Pada Esay Scoring [Pertanyaan dan Jawaban](#) [Proses Utama](#)

**Perhatian!**

Anda yakin menghapus data ini?

[Pertanyaan](#) [Jawaban](#)

**Daftar Pertanyaan**

No	Jawaban	Pertanyaan	Aksi
1	.....	.....	<a href="#">Ubah   Hapus</a>

**Keterangan:**

1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01
2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)
3. Pilih menu Preprocessing maka akan menuju ke tampilan L03
4. Klik menu Perbaiki maka akan menuju ke tampilan L04
5. Klik menu Scoring maka akan menuju ke tampilan L05
6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06
7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07
8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi
9. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Delete Pertanyaan L09 lalu klik ubah pada menu Aksi
10. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Kelola Jawaban L10

Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih

**Gambar 3. 20 Antarmuka Halaman Delete Pertanyaan**

Nama Tampilan: L10

Perbaiki Kesalahan Ejaan Pada Esay Scoring [Pertanyaan dan Jawaban](#) [Proses Utama](#)

**Kelola Pertanyaan dan Jawaban**

[Pertanyaan](#) [Jawaban](#)

**Daftar Jawaban**

No	Jawaban	Pertanyaan	Aksi
1	.....	.....	<a href="#">Ubah   Hapus</a>

**Keterangan:**

1. Buka web browser dan ketik: localhost/typoessay maka akan tampilan halaman L01
2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)
3. Pilih menu Preprocessing maka akan menuju ke tampilan L03
4. Klik menu Perbaiki maka akan menuju ke tampilan L04
5. Klik menu Scoring maka akan menuju ke tampilan L05
6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06
7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07
8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi
9. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Delete Pertanyaan L09 lalu klik ubah pada menu Aksi
10. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Kelola Jawaban L10
11. Pilih menu ubah jawaban untuk menuju ke tampilan L11

Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih

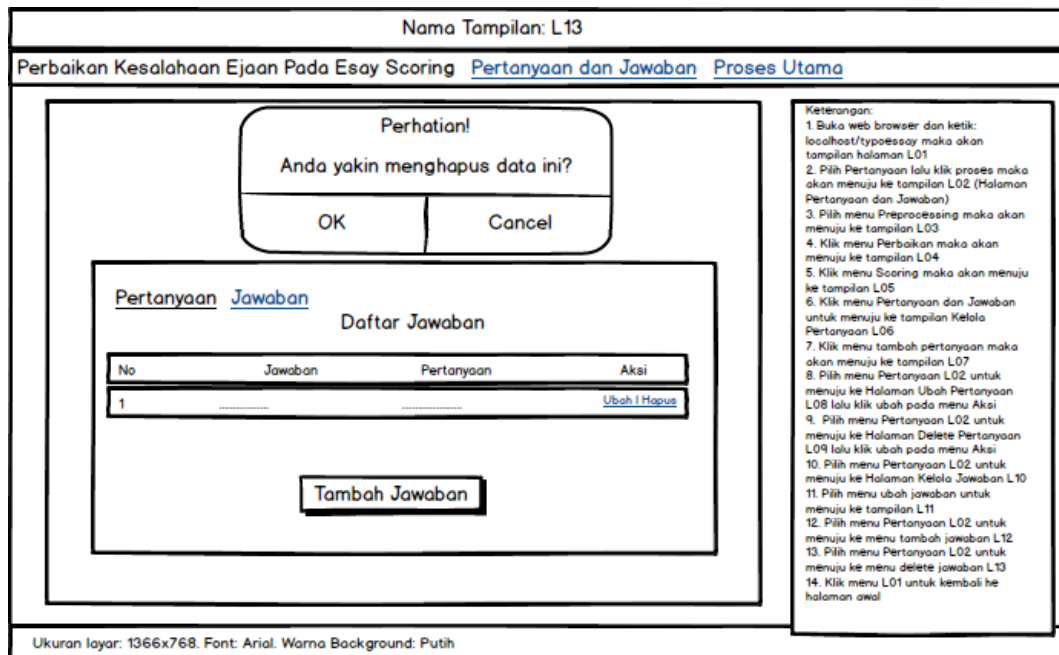
**Gambar 3. 21 Antarmuka Halaman Kelola Jawaban**

Nama Tampilan: L11	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Kelola Pertanyaan dan Jawaban</p> <div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center;"><b>Ubah Jawaban</b></p> <p>Pertanyaan</p> <p><input type="text" value="Apa yang dimaksud.... ?"/></p> <p>Jawaban</p> <p><input type="text" value="....."/></p> <p style="text-align: center;"><input type="button" value="Simpan"/> <input type="button" value="Kembali"/></p> </div>	<p><b>Keterangan:</b></p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> <li>4. Klik menu Perbaiki maka akan menuju ke tampilan L04</li> <li>5. Klik menu Scoring maka akan menuju ke tampilan L05</li> <li>6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06</li> <li>7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07</li> <li>8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi</li> <li>9. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Delete Pertanyaan L09 lalu klik ubah pada menu Aksi</li> <li>10. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Kelola Jawaban L10</li> <li>11. Pilih menu ubah jawaban untuk menuju ke tampilan L11</li> <li>12. Pilih menu Pertanyaan L02 untuk menuju ke menu tambah jawaban L12</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 22 Antarmuka Halaman Ubah Jawaban**

Nama Tampilan: L12	
Perbaiki Kesalahan Ejaan Pada Esay Scoring <a href="#">Pertanyaan dan Jawaban</a> <a href="#">Proses Utama</a>	
<p>Kelola Pertanyaan dan Jawaban</p> <div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center;"><b>Tambah Jawaban</b></p> <p>Pertanyaan</p> <p><input type="text" value="Pilih Pertanyaan..."/></p> <p>Jawaban</p> <p><input type="text" value="....."/></p> <p style="text-align: center;"><input type="button" value="Simpan"/> <input type="button" value="Kembali"/></p> </div>	<p><b>Keterangan:</b></p> <ol style="list-style-type: none"> <li>1. Buka web browser dan ketik: localhost/typessay maka akan tampilan halaman L01</li> <li>2. Pilih Pertanyaan lalu klik proses maka akan menuju ke tampilan L02 (Halaman Pertanyaan dan Jawaban)</li> <li>3. Pilih menu Preprocessing maka akan menuju ke tampilan L03</li> <li>4. Klik menu Perbaiki maka akan menuju ke tampilan L04</li> <li>5. Klik menu Scoring maka akan menuju ke tampilan L05</li> <li>6. Klik menu Pertanyaan dan Jawaban untuk menuju ke tampilan Kelola Pertanyaan L06</li> <li>7. Klik menu tambah pertanyaan maka akan menuju ke tampilan L07</li> <li>8. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Ubah Pertanyaan L08 lalu klik ubah pada menu Aksi</li> <li>9. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Delete Pertanyaan L09 lalu klik ubah pada menu Aksi</li> <li>10. Pilih menu Pertanyaan L02 untuk menuju ke Halaman Kelola Jawaban L10</li> <li>11. Pilih menu ubah jawaban untuk menuju ke tampilan L11</li> <li>12. Pilih menu Pertanyaan L02 untuk menuju ke menu tambah jawaban L12</li> <li>13. Pilih menu Pertanyaan L02 untuk menuju ke menu delete jawaban L13</li> </ol>
Ukuran layar: 1366x768. Font: Arial. Warna Background: Putih	

**Gambar 3. 23 Antarmuka Halaman Tambah Jawaban**

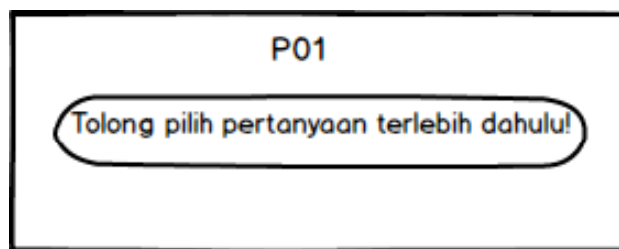


**Gambar 3. 24 Antarmuka Halaman Delete Jawaban**

### 3.11.7. Perancangan Pesan

Perancangan pesan adalah pesan peringatan yang keluar dalam bentuk pesan ketika ada proses yang dijalankan.

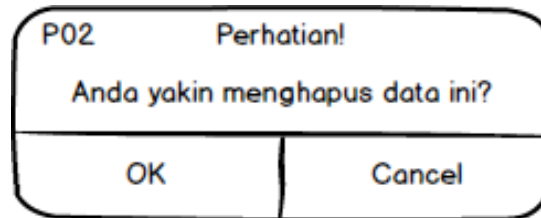
1. Pesan ketika ingin pilih pertanyaan. Pesan ini akan muncul ketika pengguna tidak men-ceklis pertanyaan yang diinginkan. Pesan ini mengharuskan pengguna memilih pertanyaan dengan cara men-ceklis.



**Gambar 3. 25 Perancangan Pesan P01**

2. Pesan ketika ingin pilih jawaban.

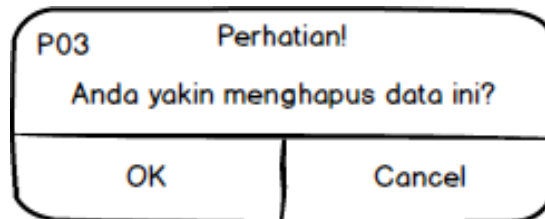
Pesan ini akan muncul ketika pengguna ingin menghapus pertanyaan yang diinginkan.



**Gambar 3. 26 Perancangan Pesan P02**

3. Pesan ketika ingin pilih jawaban.

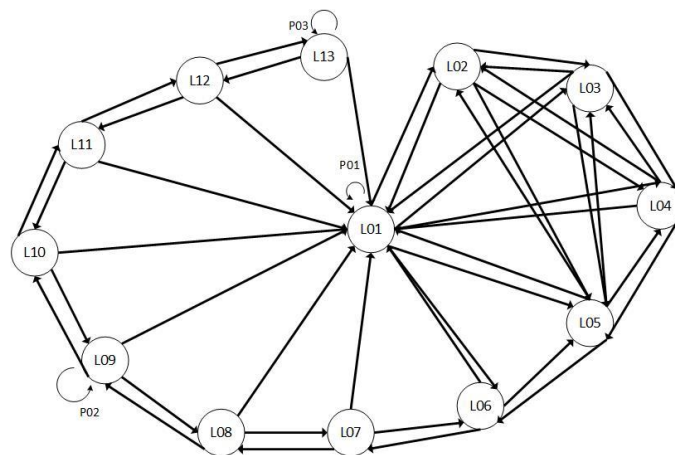
Pesan ini akan muncul ketika pengguna ingin menghapus jawaban yang diinginkan.



**Gambar 3. 27 Perancangan Pesan P03**

### 3.11.8. Perancangan Jaringan Semantik

Jaringan semantik adalah jaringan penghubung yang menghubungkan dari navigasi menu ke menu-menu yang lain. Perancangan jaringan semantik dalam aplikasi kesalahan ejaan pada sistem *essay scoring* bisa dilihat pada gambar 3. 28 sebagai berikut.



**Gambar 3. 28 Jaringan Semantik**