IMPLEMENTATION OF VECTOR SPACE MODEL (VSM) FOR ESSAY ANSWER SCORING RECOMMENDATION

Harry Septianto

Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
Email : harryseptianto666@gmail.com

ABSTRACT

Each learning process requires an evaluation form of the exam. Exam can be done in three types, the first of which is a multiple choice exam, short stuffing exam and essay exams. Essay exam is the evaluation of learning in the form of essay questions that have answers more varied than multiple choice questions. Variations of these answers give trouble to teachers in assessing the essay. In this study, the method used for matching words is a method of Vector Space Model (VSM).
Keywords : Vector Space Model, Essay Exam, Scoring Recommendation

1. INTRODUCTION

Each learning process requires an evaluation form of the exam. Exam can be done in three types, the first of which is a multiple choice exam, short stuffing exam and essay exams. Essay exam is the evaluation of learning in the form of essay questions that have answers more varied than multiple choice questions. Variations of these answers give trouble to teachers in assessing the essay.

There have been many studies on automatic correction of essays, one of which is the research conducted by Sahriar Hamzah, M. Budi Santoso Sarosa and Purnomo which uses an algorithm Rabin-Carbs. The level of accuracy of the algorithm Rabin-Krab is 90.31%. In addition to using the algorithm Rabin-Carbs, another string matching algorithm is an algorithm with a level of accuracy Winnowing Winnowing algorithm is 75-80%. In this research to match the word using Vector Space Model (VSM).

Therefore this study is expected to obtain a result of an accurate scoring of VSM.

1.1 Formulation of The Problem

Based on the background described by the authors above, it can be formulated problem is how to match the word and recommending the value of the essay that has included students in the learning media.

1.2 Objective And Purpose

Based on the problems studied, the purpose of this thesis is to implement methods of Vector Space Model (VSM) for matching words and on the value of the essay.

While the objectives to be achieved in this study are as follows:

1 To see the accuracy of this method VSM in matching word.
2 To see how accurate the system by making recommendations to the value of students' answers

1.3 Scope of Problem

There are some limitations problems that can be formulated so that the discussion of the problem can be more focused and detailed, with a view to facilitate the identification and understanding of the application. The limit problems in the implementation of this VSM are :

1 The languages that can be read by system must be in Indonesian good and be in agreement
2 The data was used from Senior High School (SMAN)13 Palembang. Data in the form of a collection of questions and answer that are used by teacher in SMAN 13 Palembang.
3 The case that used is Economy class X (ten). Because in these subject contains many theories compared to other subjects.
4 Using Nazief and Adriani algorithm in the process of stemming and stopword.
5 Using the methods of Vector Space Model (VSM) in the matching word, while the word for weighting method using Term Frequency (TF).
6 Using a percentage of the value of the answers in the recommendation value.
7 Using object-oriented programming.
8 To model the software using the Unified Modeling Language (UML).
9 The system will be built based website.

1.4 Research Methodology

The research methodology used by the author in writing this final report is descriptive methodology, the discussion of methods used to describe the object

to be studied, by locating, collecting, and analyzing the data obtained.

1.4.1 Method of Collecting Data

Data collection methods used in the research is Study Library. Library Studies done is by studying the literature, such as books, articles, e-books, websites, journals, and other sources relating to the method VSM to be built, including artificial intelligence, design, tools and modeling by UML that can help complete the implementation of this method VSM.

1.4.2 Software Development Methods

The method used for software development in this research using Agile Model. This model is a model that provide approaches for the systematic and sequential software developers by Roger S. Pressman [5] is:

a. *Planning*
This stage is modeling using object-oriented programming and applying the method of the VSM system for matching word essay and recommendation scoring.

b. *Design*
This stage is design phase of the construction of an essay answers system will be made to identify and organize the classes in object-oriented concepts.

c. *Coding*
After the stage of planning, the next stage is conversion of the system design into the programing code. The programming language is PHP.

d. *Testing*
System testing is done to ensure that the application is made in accordance with the design and all functions can be used properly without any errors.

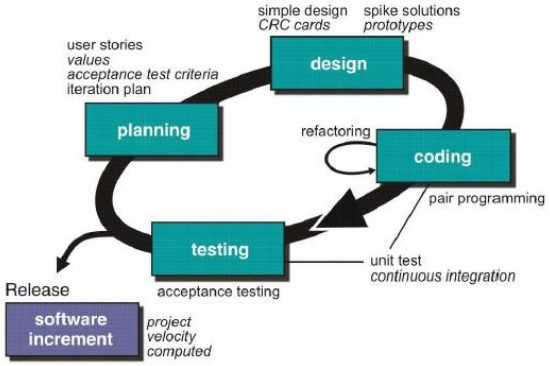


Figure 1. 1 *Agile* Model [5]

2. ISI PENELITIAN

2.1 Vector Space Model (VSM)

Vector space model (VSM) is a representation of the document as a vector in a vector space. VSM is a basic technique in the acquisition of information that can be used for the assessment of the relevance of documents to the search keywords (query) on search engines, document classification and clustering of documents [3]. In the Vector Space Model, a collection of documents represented as a term-document matrix (matrix-term frequency). Each cell in the matrix corresponds to a given weight of a specified term in dokmen. A value of zero means that the term is not present in the document [4].

D1 : Saya mahasiswa Ilmu Komputer
D2 : Saya menimba ilmu di Fakultas Ilmu Komputer
D3 : Mahasiswa Fakultas Ilmu Komputer banyak

	D1	D2	D3
Banyak	0	0	1
Di	0	1	0
Fakultas	0	1	1
Ilmu	1	2	1
Komputer	1	1	1
Mahasiswa	1	0	1
Menimba	0	1	0
Saya	1	1	0

Figure 2. 1 The Example of Document and Matrix Word-Document

Through the vector space model and TF weighting it will get the representation of numerical values that can then be calculated dokummen closeness between documents. The closer the two vectors in a VSM, the more similar the two documents represented vectors. There are four functions to measure the similarity (similarity measure) that can be used for this model:

1. *Cosine distance* / *cosine similarity*
2. *Inner similarity*
3. *Dice similarity*
4. *Jaccard similarity*

One measure of similarity of text that is popular is the cosine similarity. This measure calculates the cosine angle between two vectors. If there are two document vectors d and a query q, and t terms extracted from a collection of documents the cosine value between d and q are defined as follows:

$$sim\,(Di, Q) = \frac{\sum Di * Q}{\sqrt{\sum D\text{^}2} * \sqrt{\sum Q\text{^}2}} \quad (1)$$

2.2 Term Frequency-Inverse Document Frequency (TF-IDF) Weighting

The simplest method of weighting to a term (term weighting) is to use the frequency of occurrence of terms (words) / term frequency (TF) concerned in a document. Inverse Document Frequency (IDF) is the logarithm of the ratio of the

total number of documents processed by the number of documents that have the term concerned. Then Salton experiment to combine both the weighting method, taking into account the frequency of inter-document and intra-document frequency of a term. By using the term in a document the frequency and distribution in the whole document, the appearance of the other documents (IDF). Salton draw conclusions through experiments that the terms for a total frekuensin medium, more useful in retrieval when compared to the terms of the total frequency is too high or too low. The concept of intra-document and inter-document is then known as TF-IDF method**.**

The formula used to express the weight (w) of each document for key words are:

$$Wd,t = tf\ d,t \times \mathrm{IDF}t \qquad (2)$$

Where :

d = document to-d
t = word to-t from keywords
Wd,t = document weight to-d with word to-t

2.3 Nazief and Adriani Stemming Algorithm

Nazief stemming algorithms and Adriani (1996) was developed based on the morphology of Indonesian rule that classifies particle becomes prefix (prefix), inserts (infix), suffix (suffix) and the combined prefix-suffix (confixes). This algorithm uses basic word dictionary, and supports recoding, the rearrangement of words that undergo a process stemming excessive.

Indonesian rule classifying particle morphology into several categories as follows:

1) Inflection suffixes that group suffix that does not alter the basic form of the word. For example, the word "duduk" is given the suffix "-lah" will be a "duduklah".
   The goup is divided iinti two
   a. Particle (P), which included "-lah", "-kah", "tah", and "-pun"
   b. Possessive Pronoun (PP), including "-ku", "-mu", and "-nya".
2) Derivation Suffixes (DS) which is a collection of native Indonesian suffixes are directly added to the basic word are "-i", "-kan", and "-an".
3) Derivation Prefixes (DP) that is set prefix that can be directly given to the word pure basis, or on the basis of words that already have the addition of up to 2 prefix. These include:
   a. Prefix can morphologies ("me","be-", "pe-", and "te-")
   b. Prefix can't morphologies ("di-", "ke-", and "se-")

Rules for beheading word prefix on Nazief and Adiani stemmer algorithm can be seen in the table below.

Table 1 Beheading rules Prefix Stemmer Nazief And Adriani

Aturan	Format Kata	Pemenggalan
1	berV…	ber-V…| be-rV…
2	berCAP…	ber-CAP… dimana C!='r' & P!='er'
3	berCAerV…	ber-CaerV… dimana C!='r'
4	belajar	bel-ajar
5	beC1erC2…	be-C1erC2.. dimana C1!={'r'|'1'}
6	terV…	ter-V… | te-rV…
7	terCerV…	ter-CerV… diaman C!='r'
8	terCP…	ter-CP... dimana C!='r' dan P!='er'
9	teC1erC2...	te-C1erC2... dimana C1!='r'
10	me{l|r|w|y}V...	me-{l|r|w|y}V...
11	mem{b|f|v}...	mem-{b|f|v}...
12	mempe{r|l}...	mem-pe...
13	mem{rV|V}...	me-m{rV|V}... | me-p{rV|V}...
14	men{c|d|j|z}...	men-{c|d|j|z}...
15	menV...	me-nV... | me-tV
16	meng{g|h|q}...	meng-{g|h|q}...
17	mengV...	meng-V... | meng-kV...
18	menyV...	meny-sV…
19	mempV...	mem-pV... dimana V!='e'
20	pe{w|y}V...	pe-{w|y}V...
21	perV...	per-V... | pe-rV...
23	perCAP…	per-CAP... dimana C!='r' dan P!='er'
24	perCAerV...	per-CAerV... dimana C!='r'
25	pem{b|f|V}...	pem-{b|f|V}...
26	pem{rV|V}...	pe-m{rV|V}... | pe-p{rV|V}...
27	pen{c|d|j|z}...	pen-{c|d|j|z}...
28	penV...	pe-nV... | pe-tV...
29	peng{g|h|q}	peng-{g|h|q}...
30	pengV...	peng-V... | peng-kV...
31	penyV...	peny-sV…
32	pelV...	pe-lV... kecuali 'pelajar' yang menghasilkan 'ajar'

Aturan	Format Kata	Pemenggalan
33	peCerV...	per-erV... dimana C!={r|w|y|l|m|n}
34	peCP...	pe-CP... dimana C!={r|w|y|l|m|n} dan P!='e'

Description symbol letters:
C: consonants
V: vowel
A: vowels or consonants
P: particle or fragment of a word, such as "er"

2.4 Morphological Analysis

Morphological Analysis is the process whereby every word stand-alone (individual word) analyzed back to the token forming component and nonword such as punctuation and so separated from the word. The end result of this process is the process of parsing. Parsing is the process of converting a list of words that form sentences into a form that defines the structure unit represented by a list [6]. In the table below can be seen a few characters (token nonword) which must be separated from the word.

Table 2 Character (Token Nonwrod)

Karakter			
!	~	+	/
@	&	+	\
#	*	{	“
$	(	}	‘
%	)	[	:
^	-	]	:
`	_	|	.
,	<	>	?
White space (tab, spasi, enter)			

2.5 Stopword Removal

Stopword removal is a process to eliminate the word 'irrelevant' on the results of parsing a text document by comparing with stoplist. Stoplist contains a set of word 'irrelevant', but often appear in a document. In the table below is a list of stoplist used in the system.

Table 3 Stoplist

Stoplist				
'yang'	‘untuk’	‘ini’	‘telah’	‘begitu’
‘pada’	‘ke’	‘karena’	‘dari’	‘maka’
‘menurut’	‘namun’	‘kepada’	‘di’	‘lagi’
‘antara’	‘dia’	‘oleh’	‘serta’	‘tentang’
‘ia’	‘dua’	‘saat’	‘bagi’	‘demi’
‘seperti’	‘tidak’	‘harus’	‘sekitar’	‘dimana’
‘jika’	‘dan’	‘sementara’	‘kami’	‘kemana’
‘sehingga’	‘kembali’	‘setelah’	‘belum’	‘sampai’
‘sebagai’	‘ada’	‘mereka’	‘anda’	‘sedangkan’
‘masih’	‘juga’	‘sudah’	‘itulah’	‘selagi’
‘hal’	‘akan’	‘saya’	‘daripada’	‘sementara’
‘ketika’	‘dengan’	‘terhadap’	‘yakni’	‘sebelum’
‘adalah’	‘kita’	‘secara’	‘yaitu’	‘tetapi’
‘itu’	‘hanya’	‘agar’	‘kenapa’	‘apakah’
‘dalam’	‘atau’	‘lain’	‘mengapa’	‘supaya’
‘bisa’	‘bahwa’	‘anda’	‘begitu’	‘dll’

2.6 Stemming & Lemmatization

Stemming is a process that aims to reduce the amount of variation in the representation of a word. Risks stemming from the process is the loss of information in the word-stem. This results in a decrease in accuracy or precision. Meanwhile, the advantage is that the process of stemming can improve the ability to do a recall.

The aim of stemming sebearnya is to improve performance and reduce resource usage of the system by reducing the number of unique word that must be accommodated by the system. So, in general, stemming algorithms working on the transformation of a word into a standard representation of morphology (known as stem).

Lemmatization is a process to find the basic form of a word. There is a theory that explains that the lemmatization is a process aimed at normalizing the text or words based on the basic form is the form of his lemma. Normalization here in the sense of defining and removing a prefix and suffix of a word. Lemma is the basic form of a word that has a particular meaning based on dictionary

2.7 Main Process

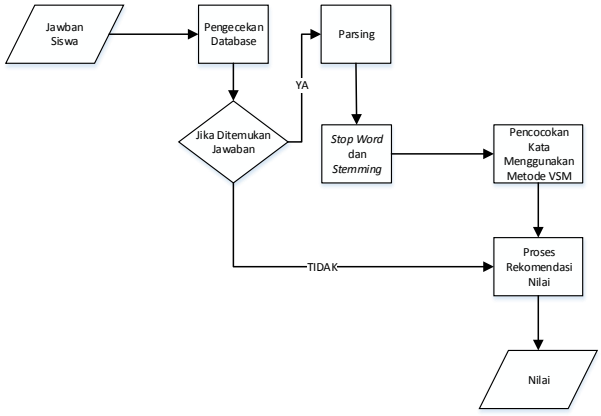


Figure 2.2 Flowchart Main Process Proses

Explanation of figure 2.2 are as follows:

1. Checking Database
   A step where the system checks to the database, any questions that have been answered by the students.
2. Parsing
   Is the process of looking for unique words from the answers that have been submitted by students.
3. Stopword and Stemming
   A search process connecting words, such as: the, or, etc., and returns words to the basic word.
4. Match the word using the VSM
   Is the process of matching words input from the student and answer key contained in the database.
5. Recommended Scoring
   A process to provide recommendations in accordance with the values match between the students' answers with the answer key contained in the database.

2.7.1 Checking Database

A step where the system checks to the database, any questions that have been answered by the students.

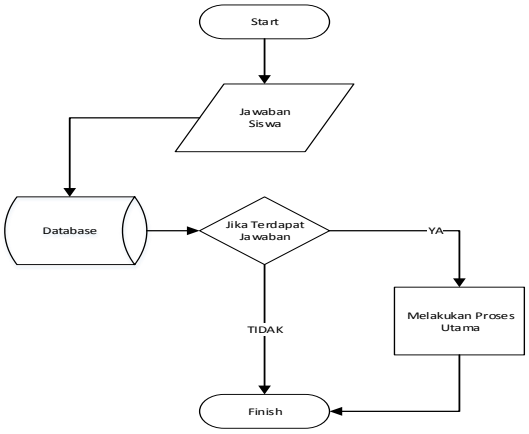


Figure 2.3 *Flowchart* Checking *Database*

2.7.2 Parsing

Is the process of looking for unique words from the answers that have been submitted by students.

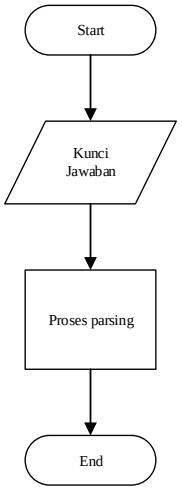


Figure 2.4 Flowchart Parsing Keywords

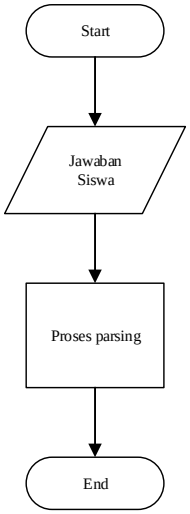


Figure 2.5 Flowchart Parsing Student Answers

2.7.3 Stopword and Stemming

A search process connecting words, such as: the, or, etc., and returns words to the basic word.

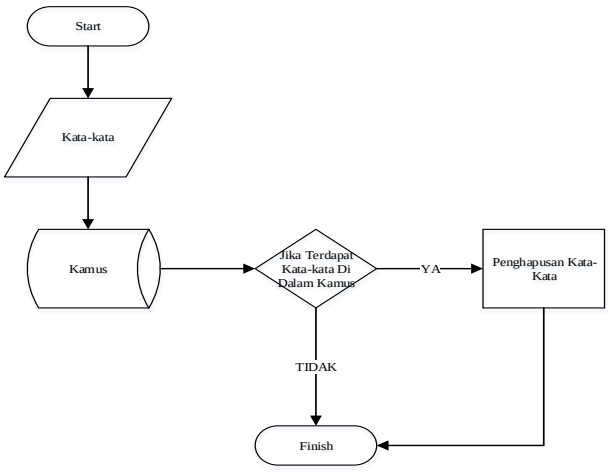


Figure 2.6 Flowchart Clear The Word (*Stopword*)

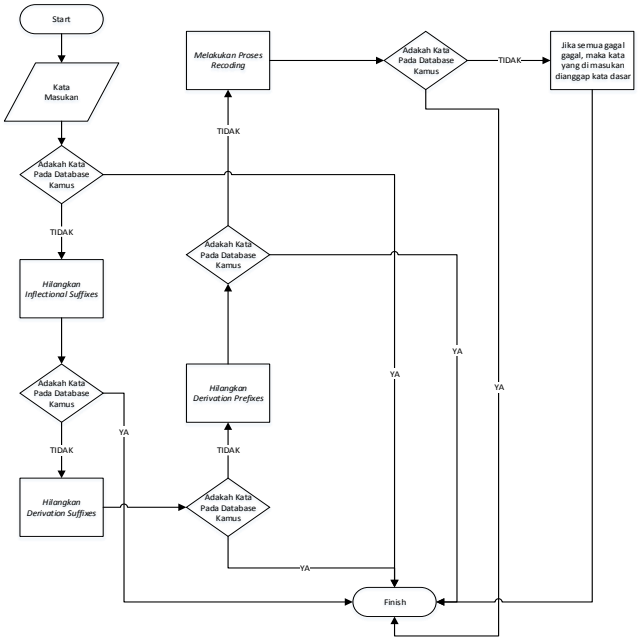


Figure 2. 7 Flowchart Nazief and Adriani Algorithm [7]

2.7.4 Matching Words

The method used in the matching words is a method of Vector Space Model (VSM). Chronology of VSM method can be seen in the image below.

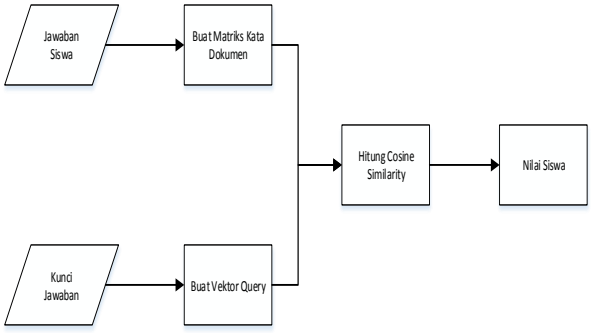


Figure 2.8 Flowchart Main Process of VSM

To calculate the number of words used cosine similarity matching. The formula to calculate it is as follows:

$$sim(Di, Q) = \frac{\sum Di * Q}{\sqrt{\sum D\text{^}2} * \sqrt{\sum Q\text{^}2}}$$

2.7.5 Scoring Recommendation

A process to provide recommendations in accordance with the values match between the students' answers with the answer key contained in the database. How to calculate the following:

$$total\ nilai = \frac{Jumlah\ Nilai\ Benar}{Jumlah\ soal}$$

2.8 ERD

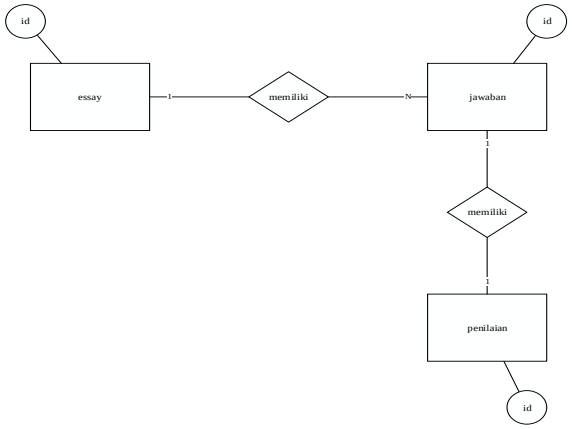


Figure 2.9 ERD

2.9 Relation Scheme

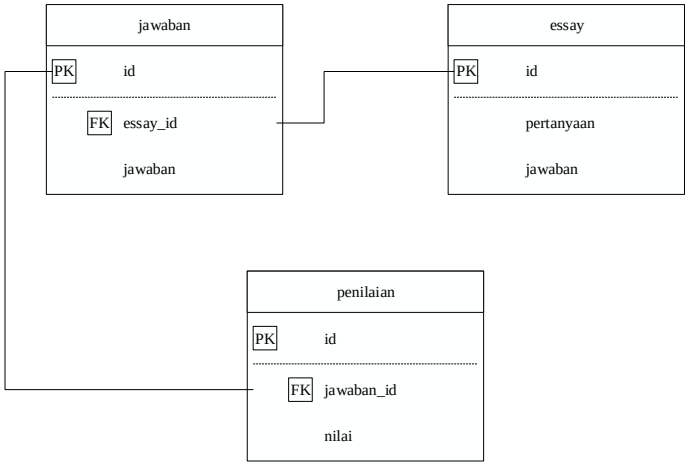
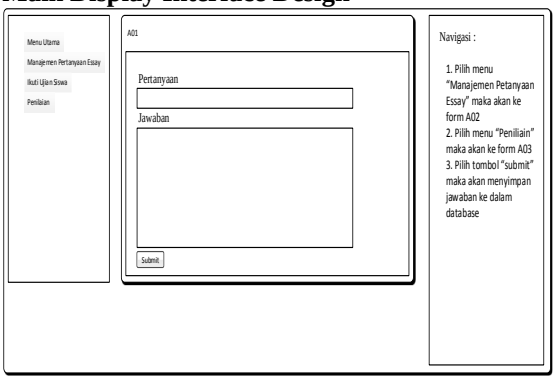


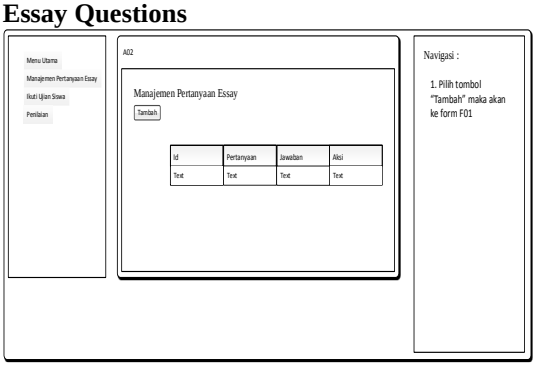
Figure 2.10 Relation Scheme

2.10 Interface Design

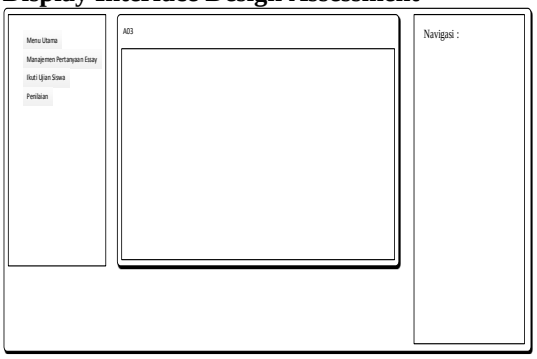
1. Main Display Interface Design



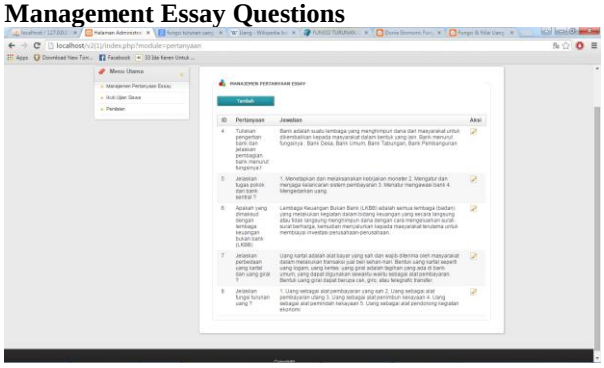
2. **Display Interface Design Management Essay Questions**



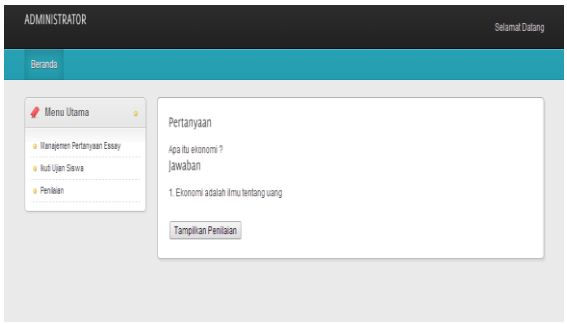
3. **Display Interface Design Assessment**



2. **Display Interface Implementation Management Essay Questions**



3. **Display Interface Implementation Assessment**

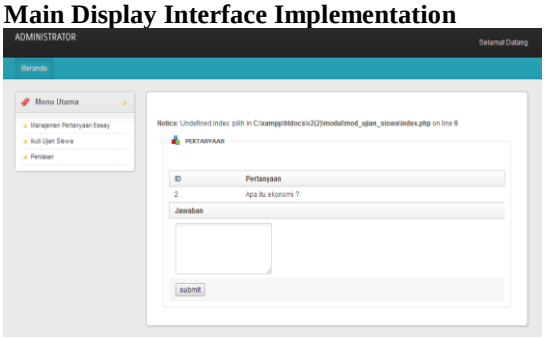


3. TEST RESULT AND IMPLEMENTATION

3.1 Implementation Interface

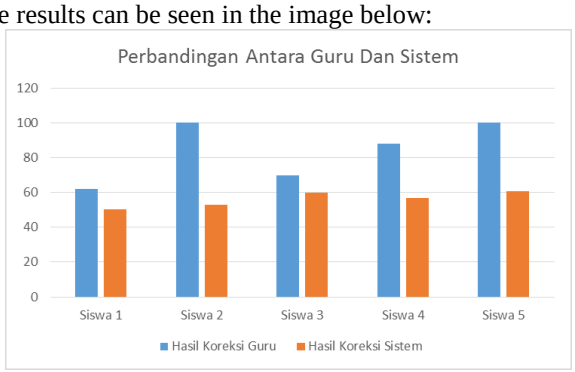
From the design of the interface that has been made in the previous chapter, the next step is to implement it into a display. Implementation of the system interface include:

1. **Main Display Interface Implementation**



3.2 Test Result

Testing accuracy begins with the correction manually, the teacher immediately correct answers have been answered by the students. Then for the next stage using VSM method and system for matching words in the recommendation value. After both processes will get the accuracy of the results of the comparison between the corrections made by the teacher and carried out by the system. In this case the answer sample data taken from five students. The results can be seen in the image below:



4. CONCLUSION

4.1 Conclusion

Based on the test results can be concluded as follows:

1. Method VSM can match the key word answers and answers that have been submitted by students.

2. Obtained the average value recommended by the system is 56.07% and the average value recommended by teachers is 84%, and the difference between the values given by the teacher and the system is 27.93%.
3. The time required by the system to match the word and provide recommendations very old value, because a growing number of students who enter the answer, the more time is needed by the system to match the value of the word and provide recommendations. The average time it takes the system to match the word and provide recommendations for the example above value is 17 seconds.

4.2 Suggestion

The following suggestions can be made to the development of the research that has been done:

1. To improve the accuracy of the system in providing recommendations better value using Natural Language Processing (NLP) NLP assess because not only judge based on common words only, but based on the wording (grammar) of the answers that have been submitted by students.
2. For further research is recommended to use existing methods merger with some other methods to get better results.

BIBLIOGRAPHY

[1] S. Hamza, M. Sarosa and P. B. Santoso, "Sistem Koreksi Soal Essay Otomatis Dengan Menggunakan Metode Rapid Karp," *Jurnal EECCIS*, vol. 7, 2013.

[2] S. Astutik, A. D. Cahyani and M. K. Sophan, "Sistem Penilaian Otomatis Dengan Menggunakan Algoritma Winnowing," *Jurnal Informatika*, vol. 12, pp. 47 - 52, 2014.

[3] H. Septiantri, "Perbandingan Metode Latent Semantic Analysis Dan Vector Space Model Untuk Sistem Penilaian Jawaban Esai Otomatis Bahasa Indonesia," 2009.

[4] Darmawan, Heru Adi; Wurijanto, Tutut; Masturi, Akh;, "Rancang Bangun Aplikasi Search Engine Tafsir Al-Qur'an Menggunakan Teknik Text Mining Dengan Algoritma VSM (Vector Space Model)".

[5] R. S. Pressman and B. R. Maxim, Software Engineering, A Practitioner's Approach Eighth Edition, New York: McGraw-Hill Education, 2015.

[6] W. Budiharto and D. Suhartono, Artificial Intelligence : Konsep dan Penerapannya, Jakarta: Andi, 2014.

[7] Tahitoe, Andita Dwiyoga, "Implementasi Modifikasi Enchanced Confix Stripping Stemmer Untuk Bahasa Indonesia Dengan Metode Corpus Based Stemming," *Jurnal Informatika*, 2010.

[8] S. Dikli, "An Overview Of Automated Scoring Of Essay," *The Journal of Technology, Learning,and Assessment*, Vols. 5, number 1, 2006.

[9] R. A. S. and M. S. , Rekayasa Perangkat Lunak : Terstruktur dan Berorientasi Objek, Bandung: Informatika, 2013.

[10] Fathansyah, Basis Data : Edisi Revisi, Bandung: Informatika, 2012.