TOPIC	:	AUTOMATED ESSAY GRADING SYSTEM
NAME	:	CHEONG TAU PAU
MATRIC NO.	:	WEK020031
SUPERVISOR	:	PN. NORISMA IDRIS
MODERATOR 1	:	A. P. DR. SYED MALEK FAKAR DUANI
MODERATOR 2	:	DR. RUKAINI
DURATION	:	28/06/2004 – 27/03/2005

Abstract

The aim of this project is to build an automated essay grading system which can aid in the current manual essay grading process. The core objective of the project is to develop the system for grading essay written in Malay Language for the History subject. The secondary objective is to compare the efficiency of the nearest neighbor algorithm and the vector space model. Reviews on existing system and algorithms are done to obtain more information about developing the system.

Automated essay grading system is a web based system that is used by the teachers and students. The system could aid the teacher in assessing the essay submitted by the student in electronic form. The students could use the system to practice their essay writing skills. The teachers and students can view the results of the essay submitted. The student can only view their own results where the teachers could view all the students' result. The system should be available online 24 hours a day throughout the year except for unexpected reasons.

The Waterfall Methodology with Prototyping was chosen to develop the automated essay grading system. The web based application will be served using Microsoft IIS and it will be developed using ASP.Net with VB.Net and Javascript as backend. Microsoft® SQL Server 2000 will be use as the database system.

Acknowledgement

The development of this Automated Essay Grading System was carried out with many advices, assistance, guidance, contributions and ideas from many individual.

First and foremost, I would like to express my utmost gratitude to my supervisor, Pn. Norisma Idris for her guidance throughout the development of my project. The advices and suggestions from her are very useful throughout the project. At the same time, I would like to express my hearty thanks to Dr.Rukaini and Assoc. Prof. Dr. Syed Malek Fakar Duani, my project moderators for spending precious time to assess this project.

Special thanks also go to my fellow course mates, Mr. Chong Theng Hui and friends for sharing their time and knowledge with me. Their supports and motivations are deeply appreciated. I feel grateful to have their encouragements whenever I encounter difficulty.

Table of Content

List of Figure

List ofTable

CHAPTER 1

1.0 Introduction

Automated essay grading system is a system built to assess essay using the computer. The system has a built in engine to assess essay by having the students responses as the input. During the process of the assessment, the students responses are parsed into the engine and then intelligently matched against a given marking scheme template. This is to determine the similarity of the students responses and the prepared model answer. A document which has a high similarity will gain a good grade as output and vice versa.

1.1 Problem Definition

Writing or composing an essay could show the level of a persons understanding about certain topic. It had been considered by many lecturers as the most useful tool to assess how well the students understand the lectures given. Unfortunately, essay grading is a tedious and difficult job. There are a few constrains in marking essay. The essays' length, number of essay, time to finish marking all the essays and physical endurance and mental stability can be affected after a long duration of marking task.

Consider the following scenarios. An essay exam question was given by a lecturer to the students and then all the students write the essays and submit it. In the manual document evaluation system, the answers could be graded in the following ways:

(i) the lecturer who grades the essays manually according to the answers in the lecturer's mind.

(ii) Assistant(s) who uses a marking scheme or a fixed grading standard, or

(iii) Combining the both mentioned method by having the essays grade by the lecturer and the assistant(s).

From the above manual essay grading approaches, we could conclude that all the manual grading approaches will need a model answer or a fixed marking standard. These answer schemes might be prepared by the lecturer, or by the discussion between the lecturer and the assistant, or by certain examining board.

The grading process is actually a repeating process of comparing the model answer with the essays submitted by students. This is the part of the process that is tedious, time-consuming and error prone. If this part of the essay grading process could be automated by letting the computer mark the essay with the provided answer scheme, then this will benefit the lecturers or teachers. First, they could save time on reading and assessing a large number of similar essays. This will give them more time to do more beneficial activity such as preparing for class. Further more, the students could get a faster or instantaneous feedback on the essay they submit. By having the feedback immediately, the student could quickly learn their weakness in the essay writing and improve themselves instead of getting the result after one week or more than one week. At that time, the students might forget or hardly recall about their own essay.

With the above mentioned scenario, it is clear that the grading process which is dependant on a marking standard could be potentially computerized. In order to aid in the problems stated above, some computer-based essay marking systems have been developed. The essay grading system is a technology developed to grade essays with prepared model answers.

1.2 Project Motivations

The current advancement in the technology and the prices of hardware that decrease over time encourages many schools to use computer to aid in teachings. For the time being, the computers are use to store and retrieve information, teaching practical computer classes, prepare teaching materials and other office related activities. An automated essay grading system will utilize more of the computing power available.

Currently, a standard high school class in the urban areas will be consisting around fifty students. If a teacher is giving homework of writing an essay for 500 words, for each class the teacher will have to assess 50 essays of the same title. Not only is the grading process physically endurance taxing, but it also affects the mental. Imagine if the teacher is teaching 5 classes and have to grade 250 essays with the same title. Even if the teacher could finish assessing 25 essays in an hour, it would take 10 hours to grade all the essays. If there is a system to aid the teacher in this situation, the teacher could spent the 10 hours to do more quality works.

Besides that, there are also many attempts to have e-learning courses online. Most of these courses could only provide multiple choice questions or short answers question, because assessing an essay will be harder to implement. All these education courses will certainly benefit from an automated essay grading system. As the system is capable of grading the essay and less man-power and time will be needed in that area.

1.3 Project Objectives

The main objective of the project is to develop a program/system capable of grading essay type answers based on given model answers. Our aim is to create a system that could mark history essay answers that are written in Malay language.

To achieve this aim, the objectives are set out as follows:

i) Incorporating the essay grading function.

This system should be able to grade an essay with a provided model answer.

ii) Grade essays in a shorter time compare to human essay graders.

The system proposed above should give the grade in a short time compare to a human essay graders. By having the system grading the essay faster than a human grader, the system will be useful and help the essay graders to save time for other quality works.

iii) To grade History essay written in the Malay Language.

The system will focus on grading essay which is written in the Malay Language and belongs to a factual subject which is the History. This is because the system will be focusing on grading an essay for its content.

iv) Compare the Vector Space Model and Nearest Neighbor Algorithm

For research purposes, this system will include two essay grading methods. This is to compare the efficiency and the accuracy of the two methods. This is a secondary objective that aims to provide the result of comparison of the two methods for future researches, references or developments.

1.4 Domain/ Scope

The scope of the project is to grade essays of a specific subject. The chosen subject to be graded by the system is History. The system is proposed to grade essays written in the Malay Language. The system will need a model answer for grading the essay.

There are two targeted user domain for the system. First, the system was aimed to aid the essays grader. This is to levitate their tedious and time consuming essay grading process. Second, the system could be used by the students to improve their essay writing skills.

1.5 Project Limitation

Most of the limitations are due to current weakness in our approach to grade the essays using a mathematical approach instead of a natural language processing approach. The approaches to the system are used by many other essays grading system as well. Although these approaches which are the document similarity comparison are sufficient to grade an essay with a certain level of accuracy, they still have many limitations.

Firstly, the system is aimed to mark and grade essays of the factual subject and not literature subject. Thus, the system will mark and grade essays for content and not for style, because the former stress on the style of writing to make the essays interesting. The latter will refer more to the fact included in the essays, where the model answer could be prepared easier. So if there are two essays with same facts, but even one wrote with very good language and make it interesting and the other doesn't then the grade they get would be very close. The reason is that the system grade solely by comparing the similarity of the two documents.

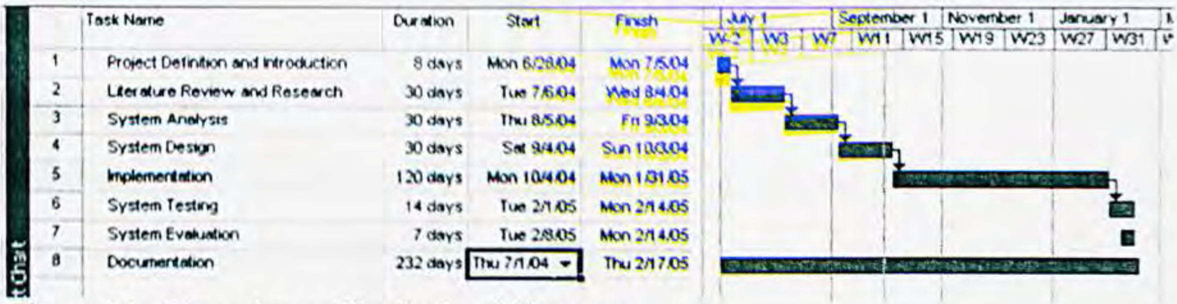
Another limitation is that the system grades the essay referring to a synonym dictionary. Anything which is not defined in the synonym dictionary will not be considered even though they have the same meaning in the real life. This is due to the grading process which is truly mathematical, so the system actually does not understands the words.

Even though the system could grade an essay, but it does not understand the essays at all. This is because the grading process is just a function to compare the similarity of the essay answer and the model answer.

Lastly, the system depends on the model answer given. This limits the system so that it could only recognize answers that are provided. This is because the system has no built in knowledge about the subject itself. So the model answer must be prepared with all the fact needed to answer the question.

1.6 Project Schedule

This project would be carried out starting from 28/6/2004 to 14/2/2004. The duration of each development phase is stated. This is an estimation of the project time line that shows the software development cycle. The project had been divided into two parts. The first part of the project development period involves literature review, requirement analysis and initial system design. The second part of the development period will focus on system module coding, implementation, and testing. In addition, documentation will be performed throughout the whole development period.



	Task Name	Duration	Start	Finish
1	Project Definition and Introduction	8 days	Mon 6/28/04	Mon 7/5/04
2	Literature Review and Research	30 days	Tue 7/6/04	Wed 8/4/04
3	System Analysis	30 days	Thu 8/5/04	Fri 9/3/04
4	System Design	30 days	Sat 9/4/04	Sun 10/3/04
5	Implementation	120 days	Mon 10/4/04	Mon 1/31/05
6	System Testing	14 days	Tue 2/1/05	Mon 2/14/05
7	System Evaluation	7 days	Tue 2/8/05	Mon 2/14/05
8	Documentation	232 days	Thu 7/1/04	Thu 2/17/05

Figure 1.1: Gantt chart of the Project Milestones

1.7 Chapter Summary

Introduction focuses on the big picture of Essay Grading System. It covers the fundamental concepts and philosophies for developing the system. This chapter also specifies the objectives, scopes and schedule for developing the system.

CHAPTER 2

2.0 Literature Review

Literature review was conducted to gather materials related to the proposed system which is the automated essay grading system. The literature review will be focus on reviewing existing systems and some algorithm applied in essay grading.

2.1 Automated Essay Grading Systems

Automated essay grading system had been around for about forty years. We will review on the first automated essay grading system and see the revolution of the capabilities of the consequence essay grading systems.

2.1.1 Origins of Automated Essay Grading Systems

Essay grading is a time-consuming and tedious process for the grader. Besides that, human inaccuracy including emotion, endurance and other factors will affect the essay evaluating process. Realizing the importance of automated essay grading, Page describes an automated document evaluation system in his paper [38]. And then earliest implementation of automated essay grading named Project Essay Grader (PEG) was done by Page.

Soon, with the rapid advancement in the computer technologies, many other automated essay grading system were built on different dimension of text grading and different technologies. Some were built to evaluate essays primarily for linguistic style. Other essay assessing systems emphasize on grading primarily for subject matter. In his paper, Page made a distinction about grading for content and grading for style.

According to Page, "Content" refers loosely to what the essay says, and "style" refers to syntax and mechanics and diction and other aspects of the way it is said. [38]

Besides the distinction of grading dimensions, many new techniques and approaches were introduce in the latter systems, including Latent Semantic Analysis, Bayes' Theorem and other techniques or hybrid of above mentioned techniques.

2.1.2 Evolutions of Automated Essay Grading Systems

As stated above, the automated essay grading systems starts with the Project Essay Grader which was built by Page in the 1960's. However, the system uses a method which assess the 'surface features' of an essay [38]. This makes the system performs remarkably well at grading the style or writing quality of an essay. However, the system cannot grade on the essay's content because of the mentioned method would not recognize the content of the essay.

In the late nineties, Thomas Landauer and Peter Foltz developed the Intelligent Essay Assessor which employed the Latent Semantic Analysis technique [6]. LSA is a machine-learning technology originally designed for document indexing and text retrieving purposes. Unlike other information retrieval algorithms, this technique uses a much deeper mathematical analysis to correlate different words and contexts. A function called singular value decomposition is used and this could often exhibit near-human comprehension of new words or passages [28]. This is also where the name latent semantic came from. The Intelligent Essay Assessor is most functional in grading short essays for its factual content, where style of writing does not matter very much in the grading.

With the existence of an essay system that could grade fact but not style and an essay system that could grade style but not fact, the American Educational Testing Service decided to develop a system capable of both evaluating methods. The E-rater extends the concepts of both PEG and IEA by using a combination of statistical and NLP techniques to extract linguistic features from the essay to be graded. [9] Essays are evaluated against a benchmark set of human graded essays. The system include the writing-style proxies of PEG, a content analysis function and more detailed algorithms for grading structure, and even select and weight predictive features for essay scoring. E-rater appears to address the deficiencies of both IEA and PEG, and consistently produces accurate essay grades.

From the above mentioned systems, we could see that the essay grading system is evolving to reach a wider coverage of capabilities. This is because human essay grader will consider both factual and writing style when they grade the essays. Although in different type of essays, they emphasize on different part. As the system evolves, more features and rules will be adapted into the system so that the system will be more capable and ready to evaluate all types of essays.

2.1.3 Limitations of Automated Essay Grading Systems

The most significant limitation of current automated essay grading system is that they must be 'trained' with large volumes of pre-graded essays before being able to produce accurate results. Some of the systems also need manual intervention in the training process to define rules and classify data [20].

Another limitation of available automated essay grading system is that they cannot deal with tabular and graphical content in essays.[9] There is some form of essay questions that requires the answer to be drawn. For example a biology question might ask the student to draw certain organ and explain about the organ. Because current system cannot deal with graphical content, these types of questions must be avoided.

Students also need suitable computer facilities to write their essays and submit it. Especially if the system is web based, internet access will be needed. Since the cost to provide every student with a computer is very high, an alternative is to provide suitable computer facilities that could generate the students' essays in machine readable form.

2.1.4 Future Research of Automated Essay Grading Systems

The models for automated essay evaluating system had been evolved over the years since 1966. From the model that do surface analysis only, to content specific analysis, to a combination of both. Different statistical techniques are used in these systems. All these system uses training sets of essays that were graded by humans. But currently, there are no intentions of taking away the human grader from the system.

According to Bramanis, in The Evolution of Automated Essay Grading, Instead of removing the human grader from the system, research is progressing towards providing student feedback for improving essays, detecting incoherent passages of text, co-reference resolution, and the grading of short-answer tests [20].

2.2 Document Similarity

Most approaches to essay grading are to compare the essays with a set of model answers using the information retrieval technique. The similarity of the essays and the model answers are determined by the mentioned methods and the essay will be graded depending on its similarity with the model answers. Following are some methods used to determine the similarity of two documents. These methods are reviewed according to the level of dimension it used to find the similarity between documents. The Nearest Neighbor Algorithm uses a one dimension distance to determine similarity. Whereas, the Vector Space Model uses a two dimensional approach which is calculate the cosine of the angle between two vectors. Lastly, the Latent Semantic Analysis uses a multi-dimensional space to analyze the similarity of documents.

2.2.1 Nearest Neighbor Algorithm

A Nearest Neighbor method is one of the simplest algorithms. It is a statistical method using historical data of measurements outcome variables to get the value of the outcome variable of a sample based on its measurements. In another word, it gives an output by comparing the distance of two data. This methods measure the similarity of two documents in one dimension. Following is a nearest neighbor algorithm by Sobhan Raj Hota. [16]

Algorithm:

- For each feature in the input case
- Find the corresponding feature in the stored case
- Compare the two values to each other and compute the degree of match

- Multiply by the weight coefficient representing the importance of the feature to the match
- Add the results to derive the match score.

The Equation used in Nearest Neighborhood algorithm is as follows:

$$\frac{\sum_{i=1}^{i=n} w_i \times sim(T_i \, . \, S_i)}{\sum_{i=1}^{i=n} w_i}$$

Where

W_i is the weighting factor of each individual feature i

S_i is the stored cases of each individual feature i

T_i is the target case of the feature i

The similarity of two features is determined by the following function.

$$sim(T_i \, . \, S_i) = \begin{cases} 0. & if \quad T_i \neq S_i \\ 1. & if \quad T_i = S_i \end{cases}$$

2.2.2 Vector Space Model

The Vector Space Model is originally a classical information retrieval method. It is a ranking model that ranks individual documents against a query. The Vector Space Model has been involved in a large part of the Information Retrieval research. It was originally introduced by Gerald Salton and his associates. This model proposes a framework in which partial matching is possible [42].

All terms in a document will be assigned a weight. This set of terms defines an n-dimensional space where each document, including the query, is represented as a vector. The vector is defined as the all of the terms weights in the document. Then, a measure of similarity between document d and query q is done by calculating the cosine of the angle that is formed between the two vectors. By computing the cosine of the angle in between the two vectors, this method actually measures the similarity of two documents in two dimensions. Below is an equation of the similarity between vector d and vector q.[37]

$$sim(d.q) = \frac{d \cdot q}{\|d\| \cdot \|q\|}.$$

When we look at the array representations of these vectors, $d = [w_{0d}, w_{1d}, \ldots, w_{n-1,d}]$ and q= $[w_{0q}, w_{1q}, \ldots, w_{n-1,q}]$, where the weight w_{ij} represents the presence of term i in document j (coordinate of vector in the i-th dimension), the inner product becomes a normalized sum of weight products [19]:

$$sim(d.q) = \frac{\sum_{i=0}^{n-1} w_{id} w_{iq}}{\sqrt{\sum_{i=0}^{n-1} w_{id}^2 \sum_{i=0}^{n-1} w_{iq}^2}}.$$

There are two types of weights that could be assigned to the weight w_{ij}. First, it could be a simple binary coordinate. For example, 1 will be assigned as the weight if term *i* appears in document *j* and 0 otherwise, or it could be adjusted following some specific weighting scheme. A common approach is to consider the product [37]

$w_{ij} = tf_{ij} * idf_i$

where

tf_{ij} = term frequency of term *i* in document *j*

idf_i = inverse document frequency showing how rare is term *i* in the entire collection of documents.

The inverse document frequency can be computed as

$$idf_i = \log_2(N/n_i)$$

where

N = number of documents in the collection

n_i = number of documents containing term *i*.

In the information retrieval area, the vector space model are used in the search engine, where the amount of calculations at query run time has to be minimized to ensure fast query processing. In that case, similarities between the query vector and each document vector in the collection are calculated as the dot product [19]

$$sim(d,q)=\sum_{i=0}^{n-1}\left(\frac{w_{id}}{\sqrt{\sum_{i=0}^{n-1}w_{id}^{2}}}\frac{w_{iq}}{\sqrt{\sum_{i=0}^{n-1}w_{iq}^{2}}}\right);$$

where the normalized term weights $\frac{w_{id}}{\sqrt{\sum_{i=0}^{n-1}w_{id}^{2}}}$ are computed for each term-document combination beforehand and stored in a lookup table. The normalization denominators $\sqrt{\sum_{i=0}^{n-1}w_{id}^{2}}$ are equal within each document *d*. The normalized weights are usually stored in a lookup file called the *inverted file* [37]. This consists of two tables, called the *dictionary* and the *postings*.

Binary weights

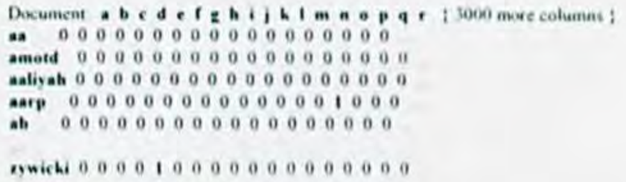
The second weighting method is to use a binary weight which consists of 0 or 1 only. This is a method suggested by Salton and Buckley. In their paper [42], they provide some tailoring guidelines for the weighting scheme. They suggested that the query weight, w_{iq} should be reduced to tf_{iq} for long queries that contains multiple occurrences of terms. They also suggested that in the case of collections with short documents or collections using controlled vocabulary, a binary weighting should be used.

This is because the weighting method mentioned above promotes the use of terms that are rare in the entire collection. It also promotes multiple occurrences of a term within the same document. This is good when use in information retrieval because in that case, it could be assumed that the document would be more relevant to a query. But in the case of grading essay, multiple occurrences of certain term included in the correct answer should not make a student's answer more correct. Or else, the student could write certain term repeatedly and gain a remarkable grade. Besides that, the weighting formula also promotes the use of terms that are rare in the entire collection. The weighting function gives less weight to a common word. But in grading exam documents, we should take into considerations that the question might be easy and most of the students got it right. We want a high similarity between the correct answer and the student answer in that case. Or else, everyone might be getting a low grade when all of them got the answer right [19].

2.2.3 Latent Semantic Analysis

Latent Semantic Analysis is an algebraic model [37]. Latent Semantic Analysis examines the document collection as a whole, and considers documents that have many words in common to be semantically close. It could extract and represent the contextual-usage meaning of words by statistical computations applied to a large corpus of text [26]. There are two important parts in Latent Semantic Analysis. The first part is to arrange all the frequency of each term from all documents into the term document matrix. Below is an example of a term document matrix from www.nitle.org/ :

Table 2-1 Term Document Matrix.



Document	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	{3000 more columns}
aa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
amotd	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
aaliyah	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
aarp	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
ab	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
zywicki	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	

after assigning weights to different terms in the term document matrix, it will go through a normalization process to normalize the matrix. Then it will be send to the second main part of Latent Semantic Analysis which is the Singular Value Decomposition function [28]. After applying this function, the dimension of the term document matrix will be scale down. The result after this function will be a matrix that could show the latent semantic meaning of terms. Thus, this algorithm was given the name Latent Semantic Analysis. Latent Semantic Analysis could work very well when use in essay grading system since it could find document with similar concepts. However, the latent semantic analysis requires a large set of documents to do the analyzing. When it is used in an automated essay grading system, it needs to be trained using a large set of sample answer. This is the reason that this method is not used in the system. But we should maintain its consideration for future efforts.

2.3 Chapter Summary

In this chapter, previous systems are reviewed. Then important algorithms that are use in the grading system are reviewed. After reviewing previous systems and important algorithms, readers can have a better understanding of an automated essay grading system and the methods employed to grade essays.

CHAPTER 3

3.0 Methodology

To develop an information system or software, certain procedure and guidelines should be followed. The procedure and guidelines should be well organized, so that it could lead the software development to success. These set of guidelines and procedure are called methodology. A methodology includes a sequence set of tasks involving activities, constraints, deliverables and resources that produce and maintain most or all information systems and software.

Various methodologies will have different phase of software development life cycle. Nevertheless, every methodology will have its complete software development life cycle. Each methodology will navigate through important phases in software development and have its phase products including structure charts, stimulus-respond threads, and state transition. Different methodologies will give a different solution in the given environment of the multi-layered finite space consisting of the analysis, design, implementation, and testing plane, starting with the root represented by the problem statement and ending with the goal represented by the system acceptance test. Below are some of the most popular methodology models:

- Waterfall Model
- Iterative-and-incremental Model
- Rapid-prototyping Model
- Extreme Programming
- Spiral Model

- V-Model
- Code-and-fix Model
- Synchronize-and-stabilize Model

Ranging from the classic life cycle models to the innovative evolutionary models – they provide adequate analysis on project duration, budget and requirements to software developers. This is crucial for developing software that delivers on time, meets the user requirements, keeps the errors and faults to minimum and easy to maintain.

In the process of developing a full-fledged software system, it is crucial for us to have a specific methodology. This is important to ensure the software product delivered on time, meet user requirements whilst errors and faults are as minimal as possible.

3.1 Methodology Consideration

3.1.1 Importance of Good Methodology

A methodology should be chosen depending on the project. A good methodology is a methodology that suite the project and has clear beginning and end at which milestones can be determined. Without a good methodology, software development risks might occurs.

Following are some benefits expected from a good methodology:

- Give a clearer overview of the system requirements.
- Provide a guideline for the developers so that they know when they need to complete a certain milestones and thus the system could be developed consistently.
- Facilitate and enhance the planned process towards greater effectiveness,

efficiency and reliability.

- Errors, inconsistencies and discrepancies could be identified during the development. This could avoid making more errors or mistakes.
- Increase the system quality by having prototype so that the user could verify the prototype built. Thus, system could be developed accordingly.
- Facilitates planning and controlling of the project.

3.1.2 Conclusion on Development Methodology

The methodology used in the development of automated essay grading system is the modified waterfall model with prototyping. In this case, the strength of both the waterfall model and prototyping model can be combined in a single project in which waterfall model support interactive design while the prototyping model helps to gain user requirement.

Waterfall Model with Prototyping

Many problems cause by the traditional model could be solved by implementing the prototyping into the waterfall model. That is, a software development will evolves lots of iteration processes as listed from the traditional waterfall model. The waterfall model will serves as the base for the whole development in the project development as its steps are similar to the generic steps of software engineering paradigms. On the other hand, prototyping involved in the earl stage of the development where there was a high degree of uncertainty in user requirement. Additionally, the modified waterfall model has the same breakdown of development tasks, but has the additional feature of feedback to make things better. Therefore, users are allowed to go back to the previous stage to fix problem that occurs. Consequently, this approach can handle more dynamic projects.

The modified waterfall model with prototyping has gathered the advantages of both traditional waterfall model and prototyping model. It improves the quality of the software life circle process and guarantees the quality of the final delivery system.

Advantages:

- Allows all or part of the system to be constructed quickly to understand or clarify the requirement
- Understands the feasibility of a design or approach
- Reduces risk and uncertainty

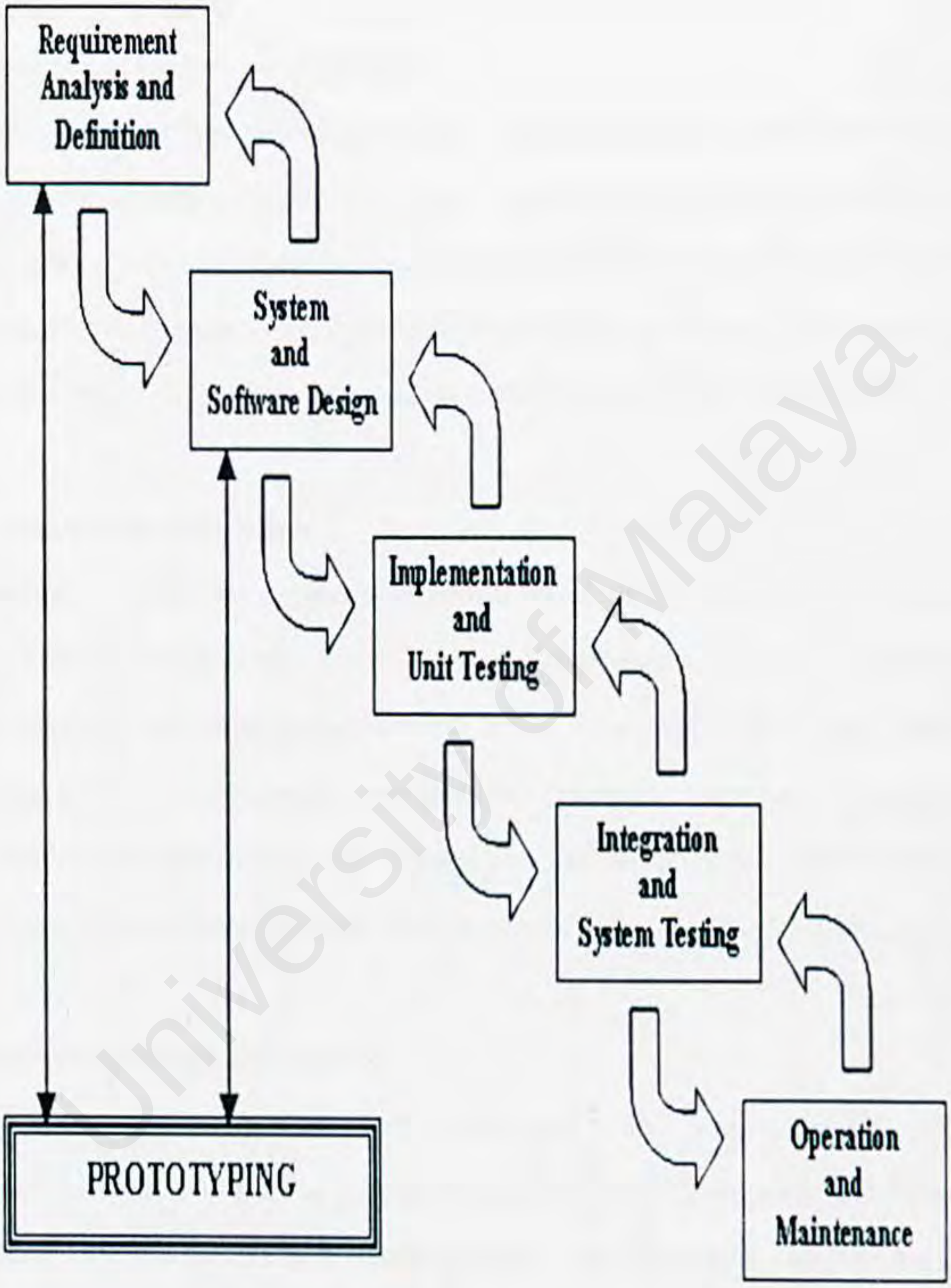

Requirement Analysis and Definition
System and Software Design
Implementation and Unit Testing
Integration and System Testing
Operation and Maintenance
PROTOTYPING

Figure 3-1: Waterfall Model with Prototyping

Stages in Waterfall Model with Prototyping

Requirement analysis and definition

In this stage, the requirement of the automated essay grading system was collected, despite the possibilities of fulfilling all these requirements. System services, constraints and goals are established by gaining information through various techniques. In this stage, feasibility study is involved and the deliverable at the end of this stage is a requirement document that states what need to be built. This is the "WHAT" phase.

System and Software Design

Reaching this stage, the system design process partitions the requirements of hardware and software. The architecture of the automated essay grading system is established in this stage and the requirement document in the earlier stage will be map into the architecture. This architecture describes the functions, subsystems, components, interfaces and behaviors of the system. The deliverable design document is the architecture that defines the implement requirement. This is the "HOW" phase.

Implementation and Unit Testing

The automated essay grading system will be built in this phase where the software design is released as a set of programs or program units. Components and functions should be built according to the design document and requirement document from the previous stages. The design document and the requirement document should give guidance for the development. At the end of this state, unit testing is performed to verify that each unit meets it specification. In shorts, this stage deals with issues of quality, performance, baselines, libraries and debugging. The deliverable of this stage is the product it self.

Integration and System Testing

In this stage, the testing of the completeness of the automated essay grading system will be done. This is to ensure the quality of the system. At this stage, all the programs or program units of automated essay grading system are integrate and test as a complete system to determine if the solution as constructed meets the requirements. Once the testing of the complete system is successful, the product is considered finished and is delivered to the user.

Operation and Maintenance

This is usually the longest life-cycle phase. At this stage, the system is put into practical use. Should there be any errors, and then maintenance will take place after this. This is the part where system may have errors in it. Therefore, maintenance involved correcting these errors, which were not discovered earlier, and enhancement to the system can be done.

3.1.3 Justification of Methodology

The modified waterfall model with prototyping was chosen for the following reasons:

- This development methodology would help the developers to learn about the system and gain better understanding of the entire system. Gaining more understanding about the entire system will ensure that the developer builds the right system according to the specification and verification checks the quality of the implementation. It also enables developers to develop more accurate system according to the user's discretion.
- With prototyping, users' participations are involved where user feedbacks can be gathered. It helps the developers to ensure that the requirements are feasible and practical. Furthermore, it helps to evaluate alternative design strategies and decide which is best for the project.
- This is simple and easy to understand development methodology. It helps the developers to lay out what they need to do easily. Therefore in the process, there are no necessaries to burden the developers with the tasks in upcoming stage. In addition, the developers can have a better understanding and clearer guideline on what he or she should do during the development process.
- Bi-directional arrows are used in between adjacent stages. The Bi-directional arrows mean that there is feedback between adjacent stages. If a problem is found in one stage, developers can return to the previous stages so that suitable correction can be taken. There is a cascading effect where developers can go back further and further up the waterfall until the problem can be corrected properly.

- Because this methodology is simple and easy to understand, it will be easier to present or explain to the users, especially those who are not familiar with software development methodologies. Therefore, developers can give the users or customers a clearer view on what is going on.

3.2 Requirement Elicitation Techniques

Requirements elicitation techniques are essential for all system analysts. This process is used to collect information about system problems, opportunities, solution, requirement and priorities. These are important step in building a complete system. These skills help to view the big picture of the system and lay down a strong groundwork.

Different methods could be applied in this phase as different system might need different types of methods of requirements elicitation. In this project, five methods have been used to gather information and elicit requirements. The five main methods are Internet surfing, reading materials, observation of existing system, interview and reviewing past researches.

- Internet surfing

  Many published papers and information could be found by using the Internet. By utilizing the Internet using search engines such as Google, and Yahoo, relevant articles, research papers and web sites could be collected as the resource especially for literature reviews. Detailed explanation about technology issues could be found and these are very useful in choosing approach for the automated essay grading system.

- Reading Materials

  In order to gain more understanding on developing the automated essay grading system, reading materials is crucial. Materials that are gathered by using the Internet, reference books, journals and other printing materials had been read to learn about the previous approach to the system. Reference books, theses and journals in Faculty Computer Science & Information Technology Thesis room and University of Malaya library are good resources to review on previous work and methods of building the system. Ideas from these readings are adopted to build the automated essay grading system.

- Observation of existing system

Some existing automated essay grading systems are tried and specifications of such system are observed. This is done by trying the system which are available online and also by reading relevant article about the systems. Reviews of existing systems are discussed in Chapter 2.

- Past Research

Previous theses done by seniors which stored in the document room of Faculty Computer Science & Information Technology are very good references even though not all of these theses are relevant to the automated essay grading system. They are good references because they provided general idea of what the theses should covers. By reading past research done by seniors, the basic guideline and idea on how to generate a report could be grasped.

3.3 Conclusion on Tools and Technology

After reviewing and analyzing on all the tools and technologies, the most appropriate tools and technologies are chosen to develop this project.

Selected System Architecture

- The client-server architecture will be applied in designing the automated essay grading system, as this architecture is suitable in developing this interactive system. There are several advantages of using this architecture. One of the advantages of client-server architecture is the implementation is easier to organize and it allow different tiers to be developed in different languages. It also provides more flexible resource allocation and performance balancing. The complexity of deploying and supporting underlying services and network communications are hidden from the user.

Selected Application Platform

- Windows XP is chosen to be the project application platform among other operating system. This is because it supports all the tools and technologies that will be use in developing the automated essay grading system. Besides that, Window XP also offers significant graphic user interface enhancements and it has built-in support for compressed files. A more stable and improved troubleshooting tools are available too. It also provides advanced sorting options.

Selected Web Server

- Microsoft Internet Information Server (IIS) is chosen as the system web server, because it comes free with Microsoft Windows XP and includes a set of program for building and administrating web site. It also supports the .NET framework very well because it was designed to support Microsoft ASP.NET technology and web based application that access database. IIS has indexing, performance and security enhancements over its old versions. And it also provides well integrated server administration tools which is easy to configure. Besides that, the IIS also supports window based web authoring and development tool

Selected Programming Language

- For this project, ASP.NET is chosen as it is the next generation of Microsoft's Active Server Page (ASP), a feature of Microsoft's Internet Information Server (IIS). It allow dynamic web page to be developed by inserting queries to a relational database in the web page. ASP.NET is designed to write web application and it has many built in function that handles web based activity. After being compiled the first time, ASP.NET application will run fast unless the codes are changed.

- **Visual Basic.Net** - Visual Basic.Net is the next generation of Visual Basic. VB.Net could be use as code-behind to ASP.net. Besides that, it is essential to the database connection and data reader. The authoring environment also provides the developers with some very easy functions such as drag and drop buttons and forms. This eased the process of creating web forms.

- **Javascript** - Javascript will be use as the client-side language. It is useful when some client-side function, such as message box and alert pop-up windows are needed. This is to complement the lack of such functionality of Visual Basic.Net.

Selected Database Management System

- **Microsoft SQL Server 2000** is the suitable choice for the development of e-veterinary system as it works well with databases of any size. Additionally, Microsoft SQL Server 2000 is the most robust database for the windows family. Advantages of Microsoft SQL Server 2000 are that it is able to support large-scale database and allows future expansion. It has High scalability, availability and reliability. It is also very easy to installation, deployment and use. It works well with other Microsoft's component and it can be queried and updated via web browsers through integration with IIS.

3.4 Chapter Summary

This chapter discuss about the methodology of the system development. Waterfall methodology with prototyping has been chosen as the framework to develop the system. Various techniques of information gathering such as online surfing, reading materials, observation, discussion and questionnaire have been practiced. This chapter also includes the conclusion on tools and technology

Chapter 4

4.0 System Analysis

System analysis for the automated essay grading system is an attempt to determine clearly of all requirements before proceeding into the following phase. These requirements include the functional requirements and non-functional requirements. The functional requirements are functions or subsystems that are mandatory to the system. Non-functional requirements are essential definition of the system properties and constrains under which a system must operate.

4.1 System Requirement Analysis

System requirement analysis determines the functional and non-functional requirement of the proposed system – Automated Essay Grading System. It is an important feature of the system that makes the system capable to operate and fulfills the system's intention.

4.1.1 Functional Requirements

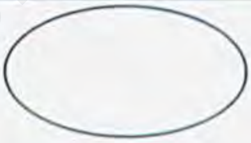
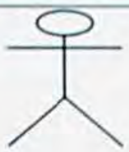
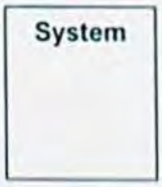
A functional requirement is a function or subsystem that must be included in the system to satisfy the need of the system. Functional requirements describe the functions the system should provide and also corresponding user. Use case diagram is used here to show the functional requirements.

Use Case Diagram

A use case models an interaction between the software product itself and the users of that software product (actors).

A use case diagram is used to model the interaction between the system and the users of the system. It shows a set of use cases, actors (normally the users) and their relationships. With a use case diagram, the system could be easily visualize because it shows clearly all the functions that certain user could gain access to. Use case diagrams are also essential for specifying and documenting the behavior of a function, for testing executable systems through forward engineering, and for comprehending executable system through reverse engineering. Table 4-1 show the notation used in a use case diagram.

Table 4-1: Notation used in a Use Case Diagram

Elements / Relationships	Notation	Description
Use case		a sequence of actions that provide something of measurable value to an actor
Actor		a person, organization, or external system that plays a role in one or more interactions with the system
System Boundary		To indicates the scope of the system where anything within the box represents functionality that is in scope and anything outside the box is not.

Association	────────	lines connecting use cases and actors to one another, with an optional arrowhead on one end of the line

The use case below shows the functional requirement of automated essay grading system.

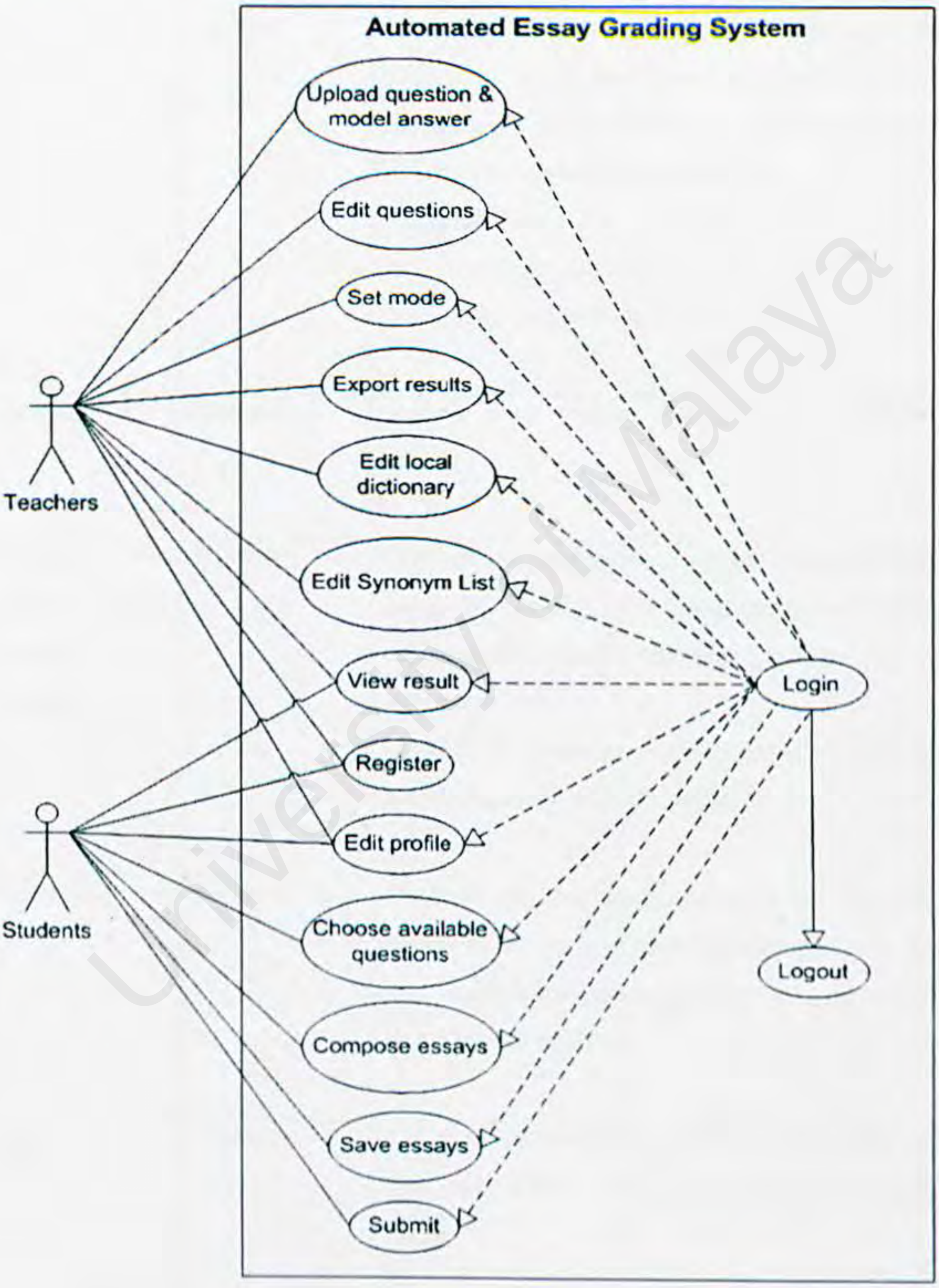


Figure 4-1: Use Case Diagram of Automated Essay Grading System

Table 4-2: Description for Automated Essay Grading System Use Case Diagram

Use Case	Actor	Description
Log –in	Teachers, Students	This function verify and authenticate the user and determine the level of the user when login to the system. A valid user name and password will be needed to login. According to the level of the user, different function will be available. Alternate flow: If not being authorized, error message will be displayed. User will not be able to log in
Registration	Student	Student must register before they could use the system.
Upload questions and model answers.	Teachers	Teachers can upload the questions and model answers using this function. The question and corresponding model answer must be uploaded together. Alternate flow: If either the question or model answer is not chosen, error message will be displayed.
Edit question	Teachers	Teachers can use this function to set the uploaded questions to be available for the students. Besides that, teachers can choose to delete the question once it is not needed anymore.
Set mode	Teachers	Teachers can set the mode of the system to be a practice or a test. If it is a test, the teacher can set the timer.
Export Results	Teacher	Teachers can use the function to save the results of

		the student in a file.
View Results	Teachers, Students	Teachers can view all the students result and the students can only view their own results. The submitted essay could be view in the view result page as well.
Edit local dictionary	Teachers	Teachers can edit the local dictionary to avoid some word to be over-stemmed.
Edit synonym list	Teachers	Teachers can edit the synonym list to add or delete synonyms and also adjust the weight if certain synonyms are used.
Edit Profile	Teachers, Students	This function enables the users to edit their profiles, update their details and change passwords.
Choose available question	Students	Students can choose the question from the available questions that the teachers enabled.
Compose essay	Students	Students can write or compose their essay.
Save essay	Students	Students can save their essay before summiting them.
Submit	Students	Students use this function to submit their essay after finishing them.
Log-out	Teachers, Students	The users can logout after they finished using the system.

4.1.2 Non-Functional Requirements

A non-functional Requirement is a description of features, characteristic and attributes of the system. It also decrypts the constraints that may limit the boundaries of the proposed system. Below are some of the non-functional requirements for Automated Essay Grading System.

- **Consistency**

  With the same set of inputs, the system should generate a set of consistent output.

- **Reliability**

  The system should be able to handle error and give a correct respond in its daily operations. It should be able to perform the daily functions and operations reliably. For example, when the teacher uploads a question and answer, the database should be updated in the corresponding fields.

- **Security**

  The system will be implemented by using user password and authentication manner. This will provide efficient security to allow information could be view by the rightful user only. For example: when a student login, the student can only access functions available to students.

- **Efficiency**

  The system should be more efficient in grading essay compared to a human grader. The system should produce the outcome at a speed which is faster than a human grader needs.

- **Manageability and Maintainability**

  The system should be easy to manage. There are functions available to maintain the users profile and the teachers can manage the uploaded files using the edit question function.

- **Expandability**

  The system should be able to be extended to accommodate more functionality in the future. Because the system are built using web forms, additional function could be accommodate easily by adding new button or web form for the function.

- **Database Maintenance**

  Database maintenance is a vital operation of every database system. The administrator should perform the backup operation frequently to avoid unexpected events. Should the database be damaged or lost in any way, it could be restored by using the backup copy of the database.

4.2 Tools and Technology Proposed

To choose the most suitable tools and technology for this project – Automated Essay Grading System, reviews on the system and discussions had been done. After all the discussions and consideration, the proposed tools and technology are as shown in the table below:

Table 4-3: Tools and Technology Proposed

Development Model	Waterfall Model with Prototyping
System Architecture	Client-Server Architecture
Application Platform	Windows XP Professional
Web Server	Internet Information Server 5.1 with .NET framework
Web Browser	Internet Explorer 6.0
Programming Language	ASP.net, VisualBasic.net, Javascript
Authoring Tools	Microsoft Visual Studio .NET 2003
Database Management System	Microsoft® SQL Server 2000

4.3 Hardware and Software Requirements

Hardware and software requirements describe the constraints on computers and peripheral equipments. Hardware and software requirements need to be decided to determine the performance requirements' feasibility. Both hardware and software requirements are divided into runtime and development requirements.

Table 4-4: Hardware and Software Requirements

	Run Time	Development
Hardware Requirements	• 500 MHz Pentium / higher microprocessor / or equivalent • Random Access Memory : 128 MB and above (512MB recommended) • Hard disk : 40 GB and above • Standard input and output • Others standard computer peripherals	• Pentium IV 1.7 Gigabyte • Random Access Memory : 512 MB • Hard disk : 40GB • Display : VGC display card • Others standard computer peripherals
Software Requirements	• Windows 2000 server (Windows XP Professional is recommended) • Any web browser (Internet Explorer 6.0 is recommended)	• Windows XP • Internet Explorer 6.1 • Internet Information Server 5.0 with .NET framework • Microsoft® SQL Server 2000

4.5 Chapter Summary

This chapter starts with discussing about the evaluation of the requirement analysis, which consists of functional requirements and non-functional requirements. The functional requirements are shown using the use case diagram which describes the functionality and the services that the system is expected to provide. The non-functional requirements will affect the overall quality and performance of the system. Then summary of hardware and software is presented in this chapter. It is divided into run-time requirements and development requirements. A list of tools and technology proposed is also included.

Chapter 5

5.0 System Design

System design is an essential step because it is the transition of the analysis phase to the system development phase. Despites the development model or standard being used, system design is applied. In the system design, the requirements analyzed in the previous phase will be translate into the representation of the software. This will act as a 'blue print' for constructing the system. The information collected earlier is used to accomplish the logical design of the system. It involves designing accurate data entry procedures and provides accurate input to the system being developed by using techniques of good form and screen design.

Under this chapter, the system design will be discuss in the following categories:

- System Architecture
- System Functionality Design
- Database Design
- Interface Design

5.1 System Architecture

Figure 5.1 is a diagram that shows how the components in the automated essay grading system interact and work together to achieve the total system goals. This system consists of two main modules. These modules are the indexing module and the grading module. This report will focus mainly on the grading module.

This is how the system works, the upload document which is also the model answer used to grade the student's essay will be indexed. The indexed version of the document is stored so that the system doesn't need to repeat the indexing process every time it grades a student's essay. The student essay will go through an indexing process before sending it to the grading process. After that, both indexed document will be send to the grading module to process the grade. The result will be stored in the database.

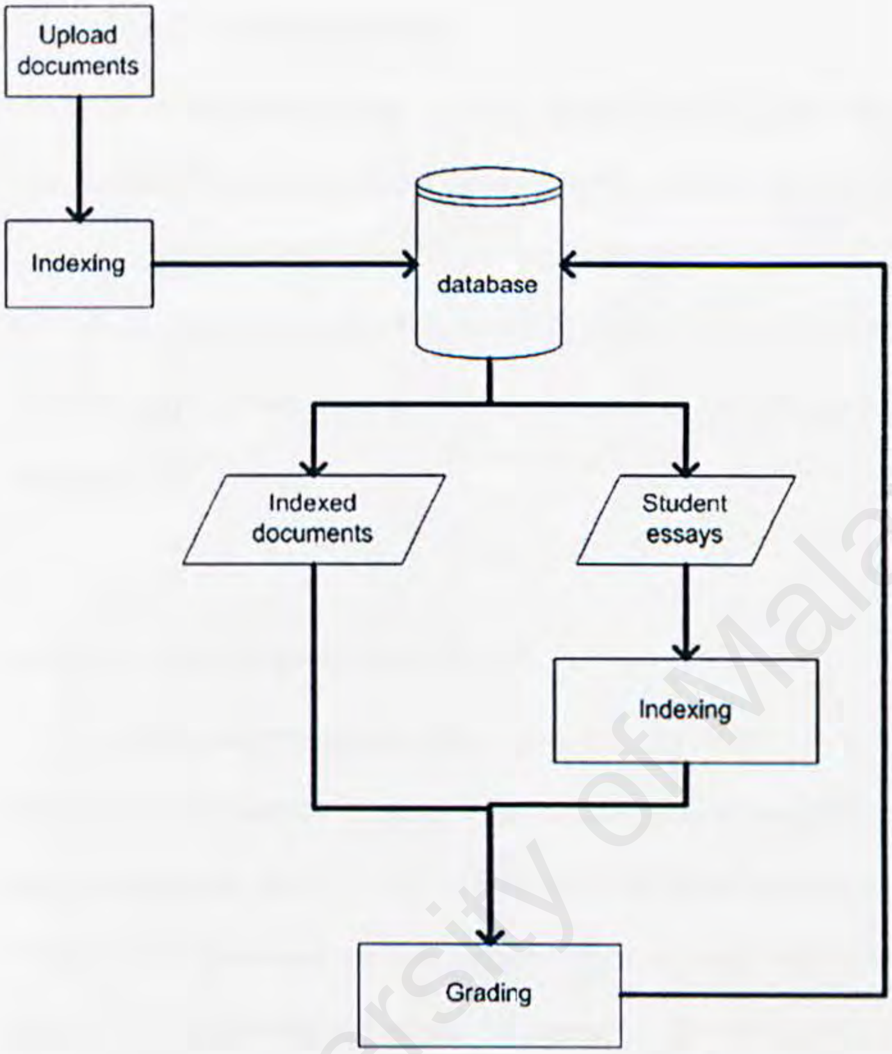


Figure 5.1: The Framework of Automated Essay Grading System

5.2 System Functionality Design

In the automated essay grading system, there are two main modules which are the Indexing Module and the Grading Module. System functionality design explains how the modules work and how it interacts with the other module. This report will focus only on the grading module. There are two grading modules presented here. These two modules will be designed to be interchangeable for gathering results for research purposes.

5.2.1 The Vector Space Model Module

After going through the indexing module, the result will be a list of terms with corresponding frequency of occurrences. Before going through the grading calculation, some preparations must be done with the result from the indexing process. The terms will be sorted to ease the matching process. First, a weight will be assigned to each term. Since the binary weight will be use, for each term, if the frequency is larger than 0 then the weight will be set to 1. Otherwise, the weight will be set to 0. Then the non-zero weights will be normalized.

After the above mentioned steps are done, we proceed to calculate the product of the model answer vector and submitted answer vector. To do this, we will search for a pair of matching terms and once a pair of matching terms is found, we calculate the product of their weights. This process will be repeated until all pairs of matching terms are found and have calculated product of their corresponding weights. When matching the terms, a synonym dictionary will be use to match terms with same meaning. A multiplier could be assign in the synonym dictionary to give a smaller weight once a synonym is used. After calculating all the product of matched terms, the results are added to get the sum. This sum will be stored in a variable *S*.

Then we proceed with calculating the square for all the terms weight. The squared terms weight for the model answer and submitted answer will be sum respectively. After that, the square root of each result will be computed and store in variable *d* and *q*.

By dividing variable *S* with the product of variable *d* and *q*, we will get the cosine of the angle between two vectors of the model answer and submitted answer. This result will be use to find the final grade of the submitted answer. Since the result will be ranging from 0 to 1, it is necessary to go through another function to transform it into a grade. With a result closer to 0 brings the meaning that both essays are not similar, it will be assigned a worse grade. A result closer to 1 will get a better result.

The following chart shows all the process of this module:

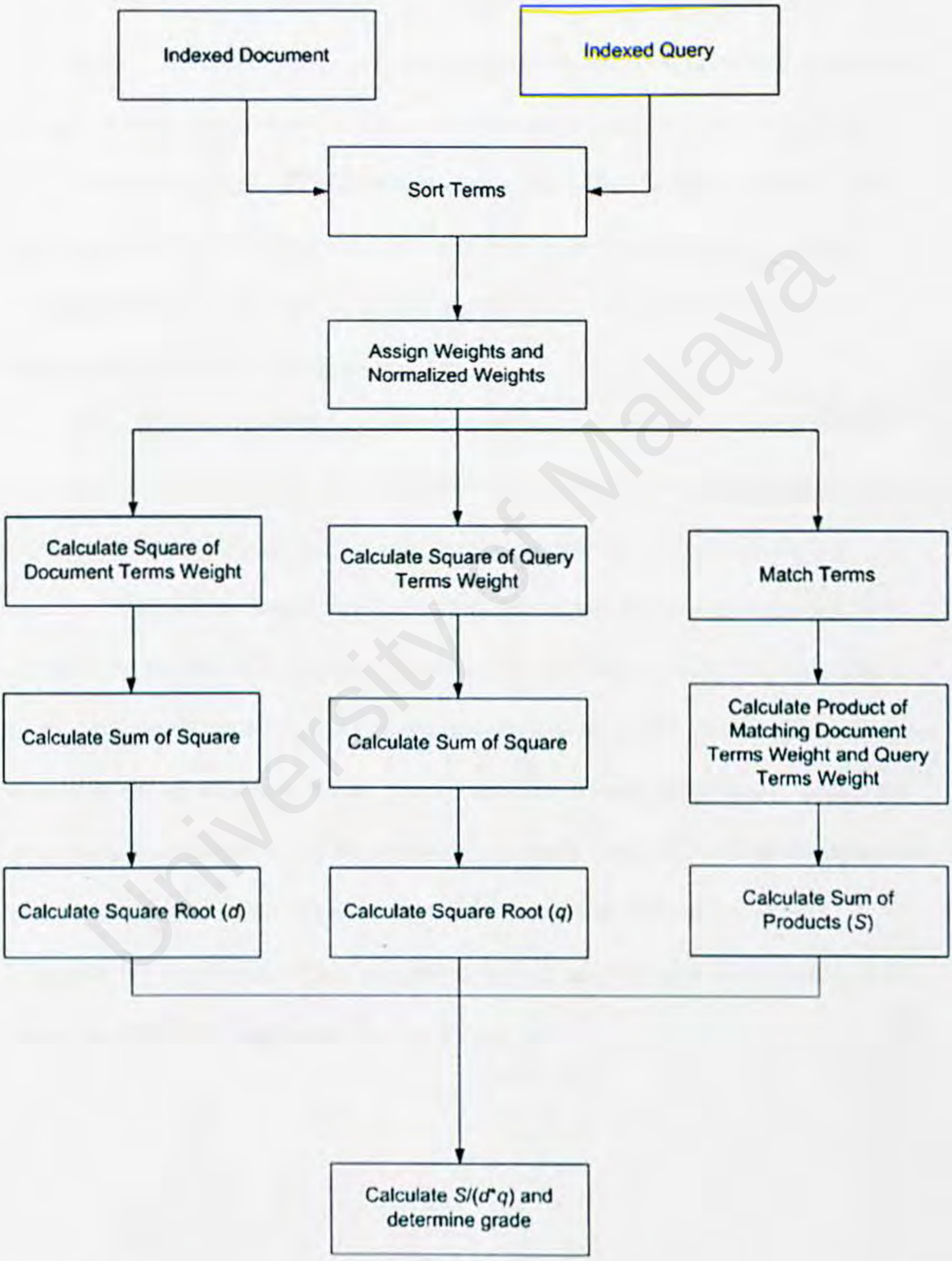

Indexed Document
Indexed Query
Sort Terms
Assign Weights and Normalized Weights
Calculate Square of Document Terms Weight
Calculate Square of Query Terms Weight
Match Terms
Calculate Sum of Square
Calculate Sum of Square
Calculate Product of Matching Document Terms Weight and Query Terms Weight
Calculate Square Root (d)
Calculate Square Root (q)
Calculate Sum of Products (S)
Calculate S/(d*q) and determine grade

Figure 5.2 Vector Space Model Module

5.2.2 The Nearest Neighbor Algorithm Module

The results from the indexing module will be match to find matching pair of terms. Before the matching process starts, the terms will be sorted to ease the matching process. Then the frequencies of the terms will be normalized. A synonym dictionary will be use to match terms with same meaning. A multiplier can be set so that a lower weight will be given when that particular synonym is used. By using the overlap distance function [34], all matching terms will be assigned a weight of 0 and non-matching terms will be assigned a weight of 1.

Then the product of the normalized terms frequencies and the weight will be calculated. With all matching terms assigned to a weight of 0, we actually get the sum of the normalized terms frequencies of non-matching terms and store the result in a variable *s*. Then, all the normalized terms frequencies, whether they are matched or not, will be added together and store in a variable *S*. By dividing variable *s* with variable *S*, we obtained the distance between two documents. The result will go through a transform function to change the result into an essay grade. The distance between two documents will determine the similarity of the documents. A result closer to 1 will be meaning that both documents are not similar and a result closer to 0 will determine that both documents are very similar. Thus, a higher grade will be given to a result close to 0 and a worse result will be assigned to the result close to 1.

The following chart shows the process of this module:

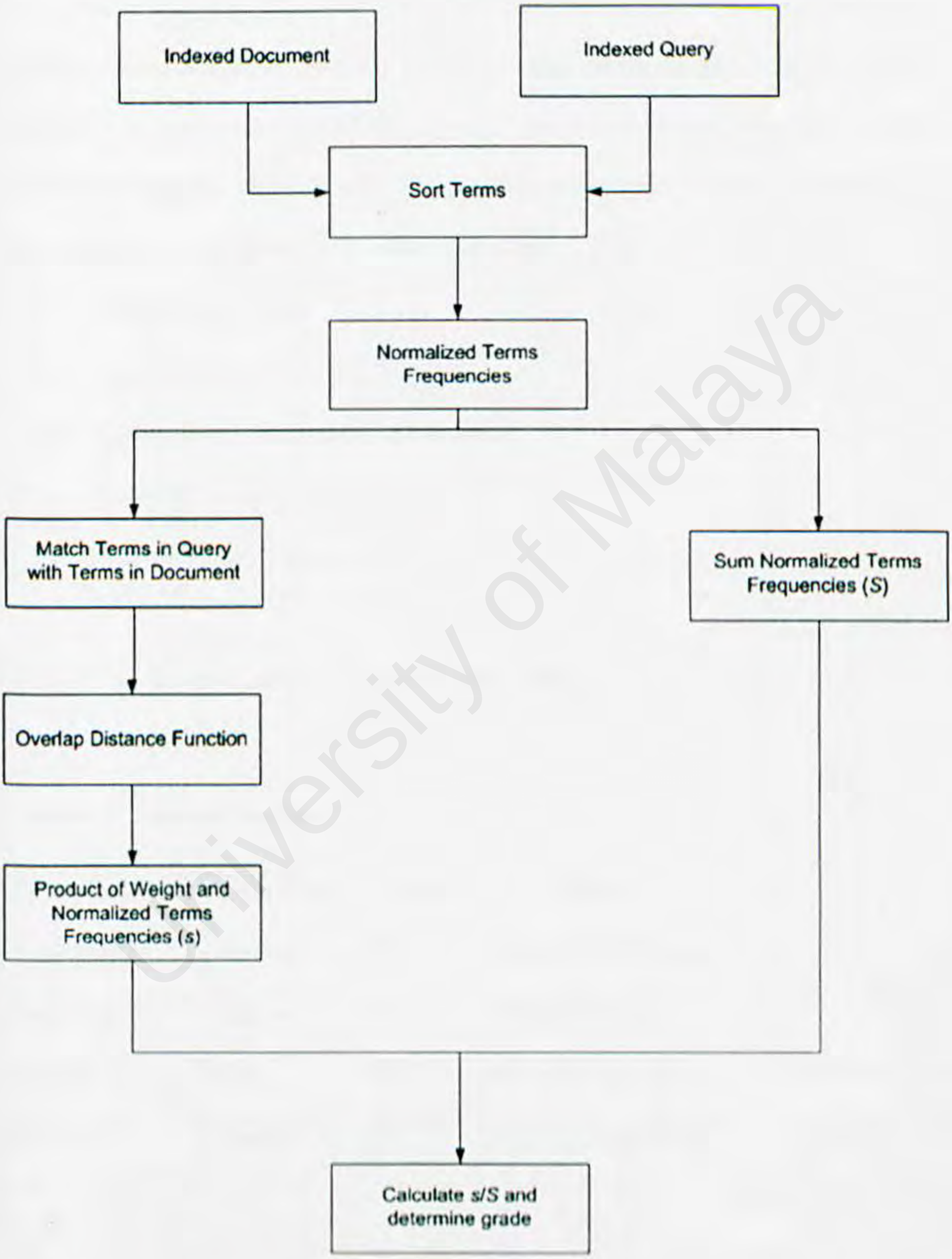

Indexed Document
Indexed Query
Sort Terms
Normalized Terms Frequencies
Match Terms in Query with Terms in Document
Sum Normalized Terms Frequencies (*S*)
Overlap Distance Function
Product of Weight and Normalized Terms Frequencies (*s*)
Calculate *s*/*S* and determine grade

Figure 5.3 Nearest Neighbor Algorithm Module

5.3 Database Design

Data storage is important in a system. The data storage in the automated essay grading system is needed especially to store the path where the model answers and the students essays are save. Without data storage, this information might lose and we could not retrieve desired data anymore. The database is constructed using Microsoft SQL Server 2000. The objectives of database design are:

- Efficient data storage
- Data availability
- Data accuracy, consistency and integrity
- Purposeful information retrieval
- Efficient updating and retrieval

Below is the tables of automated essay grading system

*** : primary key**

Table 5-1: Table of Teacher

Field Name	Data Type	Length	Description
*TeacherID	varchar	9	Teacher's ID number
TeacherName	varchar	50	Teacher's name
UserName	char	20	Teacher's user name
Password	varchar	20	Teacher's password

Table 5-2: Table of Student

Field Name	Data Type	Length	Description
*StudentID	varchar	9	Student's ID number
StudentName	varchar	50	Student's name
UserName	varcahr	20	Student's user name
Password	varchar	20	Student's password
TeacherID	varchar	9	Student's teacher

Table 5-3: Table of EssayQuestions

Field Name	Data Type	Length	Description
*QuestionID	nvarchar	10	Question ID
TeacherID	char	9	Teacher's ID
Availability	nvarchar	1	Determine whether the question is available to the students or not.
Mode	nvarchar	10	Determine whether it is a test, quiz or exercise.
TableName	nvarchar	100	Table name of the model answer

Table 5-4: Table of Essay Answers

Field Name	Data Type	Length	Description
*StudentID	Nvarchar	9	Student's ID
QuestionID	char	10	Question ID
EssayGrade	nvarchar	10	Grade of the essay
FullAnswer	text		Full answer of the student essay

5.4 Prototype User Interface Design

A Graphical User Interface (GUI) is a graphical user interface rather than purely textual that is used to accommodate interaction between machines and the user. Its main task is to communicating information from the machine to the user, and communicating information from the user to the machine.

Automated essay grading system is a web based system; therefore the user interfaces are designed according to web page style. Typically the user interface is to help users to navigate through web pages and make request.

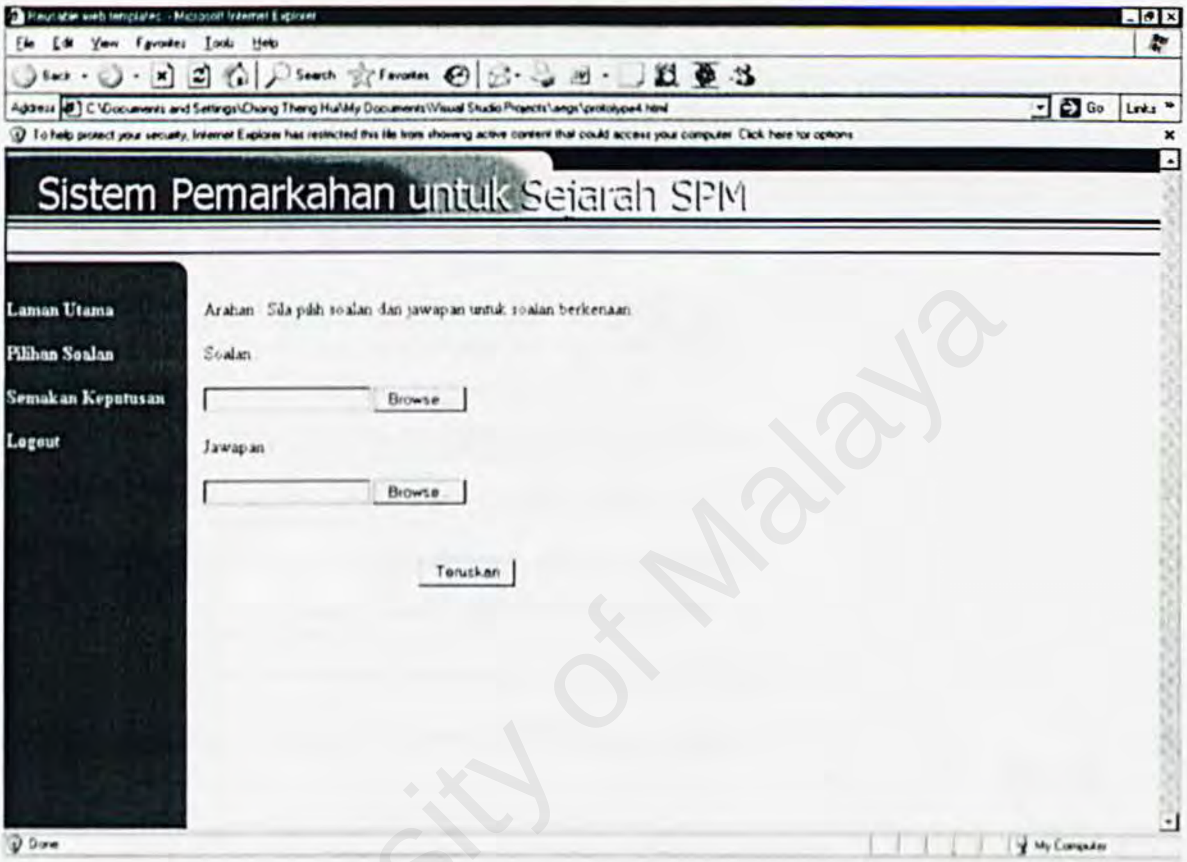
The automated essay grading system interfaces are divided into two different categories:

i. Teacher interface

ii. Student interface

Teacher interface

i. Upload question and model answer page



Reusable web templates - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address C:\Documents and Settings\Chang Theng Hui\My Documents\Visual Studio Projects\angs\prototype4.html
Go Links
To help protect your security, Internet Explorer has restricted this file from showing active content that could access your computer. Click here for options
Laman Utama
Pilihan Soalan
Semakan Keputusan
Logout
Browse
Browse
Teruskan
Done
My Computer

Sistem Pemarkahan untuk Sejarah SPM

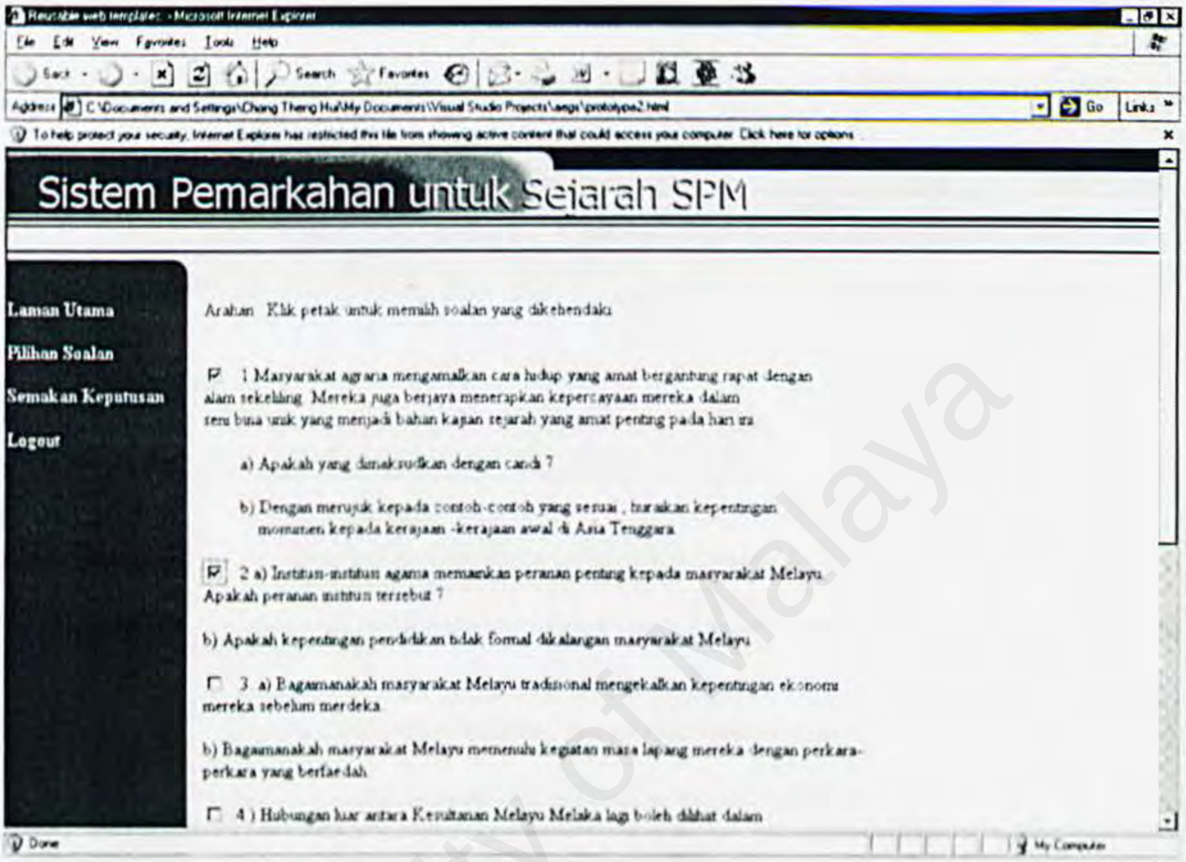
Arahan : Sila pilih soalan dan jawapan untuk soalan berkenaan.

Soalan :

Jawapan :

Figure 5-4: Prototype Upload question and model answer page

ii. Edit question page


Reusable web templates - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address C:\Documents and Settings\Chang Theng Hui\My Documents\Visual Studio Projects\aego\prototype2.html
Go Links
To help protect your security, Internet Explorer has restricted this file from showing active content that could access your computer. Click here for options
Sistem Pemarkahan untuk Sejarah SPM
Laman Utama
Pilihan Soalan
Semakan Keputusan
Logout
Done
My Computer

Arahan Klik petak untuk memilih soalan yang dikehendaki

☑ 1 Masyarakat agraria mengamalkan cara hidup yang amat bergantung rapat dengan alam sekeliling. Mereka juga berjaya menerapkan kepercayaan mereka dalam seni bina unik yang menjadi bahan kajian sejarah yang amat penting pada hari ini

a) Apakah yang dimaksudkan dengan candi ?

b) Dengan merujuk kepada contoh-contoh yang sesuai , huraikan kepentingan monumen kepada kerajaan -kerajaan awal di Asia Tenggara

☑ 2 a) Institusi-institusi agama memainkan peranan penting kepada masyarakat Melayu. Apakah peranan institusi tersebut ?

b) Apakah kepentingan pendidikan tidak formal dikalangan masyarakat Melayu

□ 3 a) Bagaimanakah masyarakat Melayu tradisional mengekalkan kepentingan ekonomi mereka sebelum merdeka

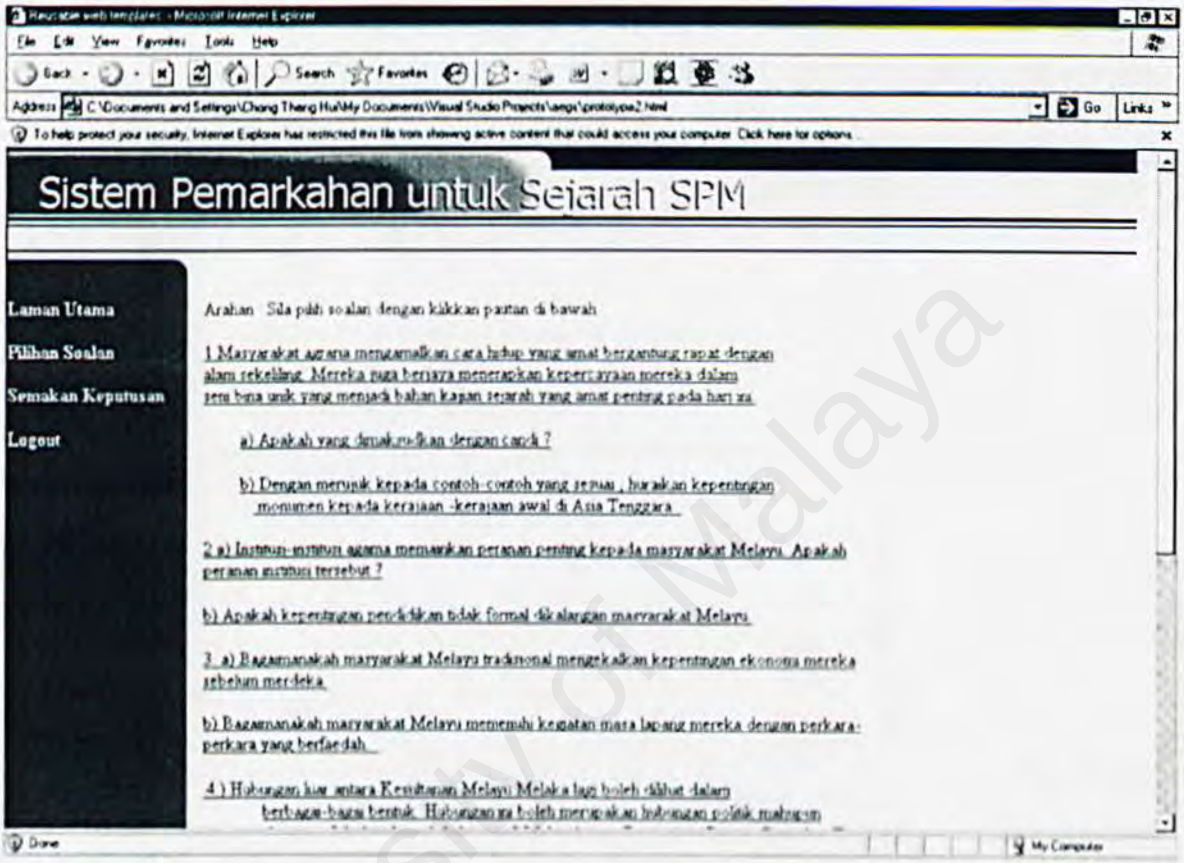
b) Bagaimanakah masyarakat Melayu memenuhi kegiatan masa lapang mereka dengan perkara-perkara yang berfaedah

□ 4) Hubungan luar antara Kesultanan Melayu Melaka lagi boleh dilihat dalam

Figure 5-5: Prototype Edit question page

Student Interface

i. Choosing available question page



Arahan Sila pilih soalan dengan klikkan pautan di bawah

1 Masyarakat agraria mengamalkan cara hidup yang amat bergantung rapat dengan alam sekeliling. Mereka juga berjaya menerapkan kepercayaan mereka dalam seni bina unik yang menjadi bahan kajian sejarah yang amat penting pada hari ini

a) Apakah yang dimaksudkan dengan candi ?

b) Dengan merujuk kepada contoh-contoh yang sesuai , huraikan kepentingan monumen kepada kerajaan -kerajaan awal di Asia Tenggara

2 a) Institusi-institusi agama memainkan peranan penting kepada masyarakat Melayu. Apakah peranan institusi tersebut ?

b) Apakah kepentingan pendidikan tidak formal dikalangan masyarakat Melayu

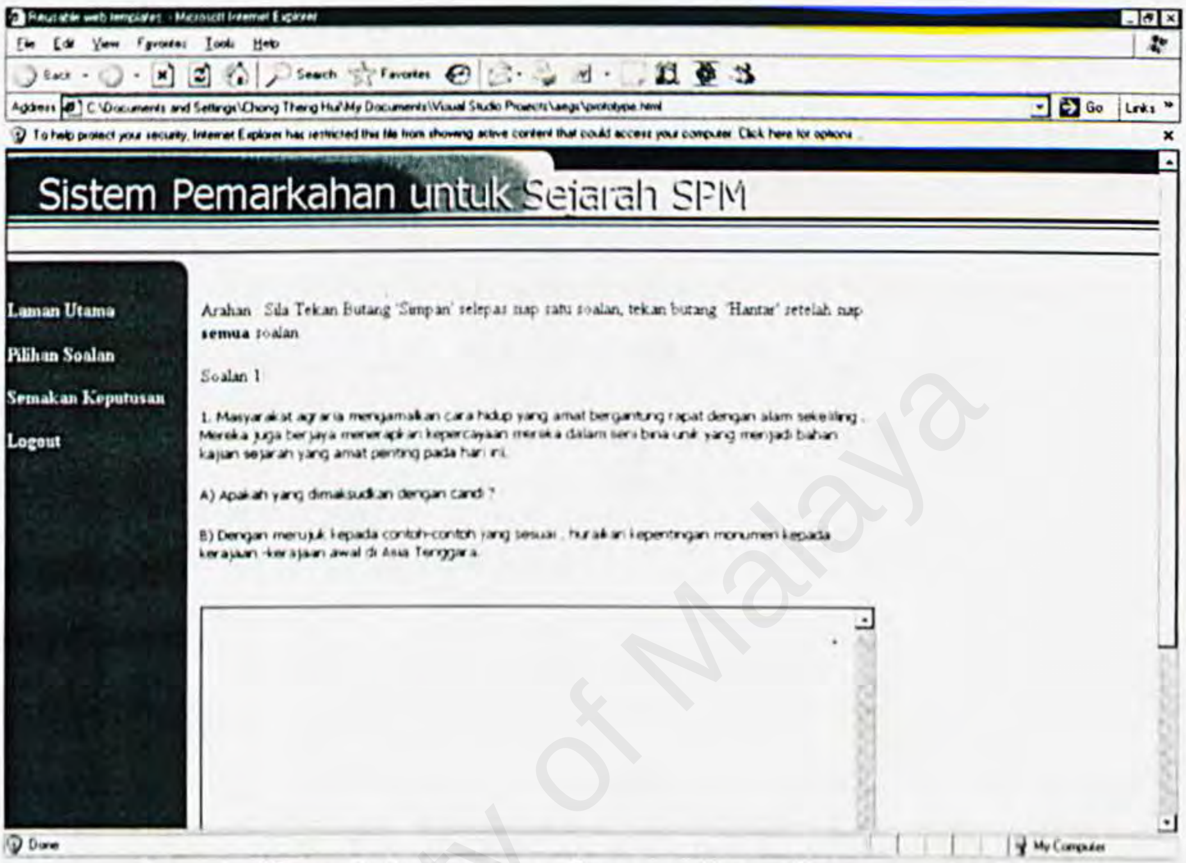
3 a) Bagaimanakah masyarakat Melayu tradisional mengekalkan kepentingan ekonomi mereka sebelum merdeka

b) Bagaimanakah masyarakat Melayu memenuhi kegiatan masa lapang mereka dengan perkara-perkara yang berfaedah

4) Hubungan luar antara Kesultanan Melayu Melaka lagi boleh dilihat dalam berbagai-bagai bentuk. Hubungan ini boleh merupakan hubungan politik maupun

Figure 5-6: Prototype Choose available question page

ii. Compose essay page


Reusable web templates - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address C:\Documents and Settings\Chong Theng Hui\My Documents\Visual Studio Projects\aegs\prototype.html
Go Links
To help protect your security, Internet Explorer has restricted this file from showing active content that could access your computer. Click here for options...
Laman Utama
Pilihan Soalan
Semakan Keputusan
Logout
Done
My Computer

Sistem Pemarkahan untuk Sejarah SPM

Arahan : Sila Tekan Butang 'Simpan' selepas siap satu soalan, tekan butang 'Hantar' setelah siap **semua** soalan

Soalan 1

1. Masyarakat agraria mengamalkan cara hidup yang amat bergantung rapat dengan alam sekeliling . Mereka juga berjaya menerapkan kepercayaan mereka dalam seni bina unik yang menjadi bahan kajian sejarah yang amat penting pada hari ini.

A) Apakah yang dimaksudkan dengan candi ?

B) Dengan merujuk kepada contoh-contoh yang sesuai , huraikan kepentingan monumen kepada kerajaan -kerajaan awal di Asia Tenggara.

Figure 5-7: Prototype Compose Essay Page

5.5 Chapter Summary

Chapter Five present the system architecture, system functionality design, database design and interface design. The main system consist two main modules and this report focuses on the grading module. System functionality design explains the process in automated essay grading system where two different approach of developing the module were explained. For database design, 4 tables in the data dictionary that act as the description of the database structure and contents. Some proposed interface designs were included for the user interface design.

Chapter 6

6.0 System Implementation

In this phase, the design of the Automated Essay Grading System will be implemented and built. The system implementation process involved converting the software representation design into a form that could execute on a computer. The initial stage of system implementation involves setting up the development environment. This includes setting up development tools to facilitate the system implementation. After that, the system implementation phase will includes the coding of the program by using a suitable programming language and coding approach, testing of the system to make sure that the functions in the program will work as planned and debugging the program, which is a process to discover and correct error or bug within the program.

6.1 Development Environment

The initial stage of system implementation involves setting up the development environment. Development environment is very important to the development of a system as suitable hardware and software will determine the success of the project.

6.1.1 Hardware

Below is the hardware configuration that used to develop this system:

- AMD Athlon 1.7GHz
- 512 MB RAM
- 80GB hard disk with approximately 20GB of fee space

- 17" Flatscreen monitor
- Other standard peripherals

6.1.2 Software

The software tools that were used to develop this system are as below:

- Microsoft Windows XP Professional with Service Pack 2
- Microsoft Internet Information System 5.1
- Microsoft Internet Explorer 6.0
- Microsoft .Net Framework 1.1
- Microsoft Visual Studio.Net 2003
- Microsoft SQL Server 2000
- Adobe Photoshop CS
- Macromedia Flash MX 2004

6.2 Program Development

During program development, program is written, user interface is being developed and database is initialized with data.

6.2.1 Program Development Process

The automated essay grading system is following a program development process that consists of 5 steps:

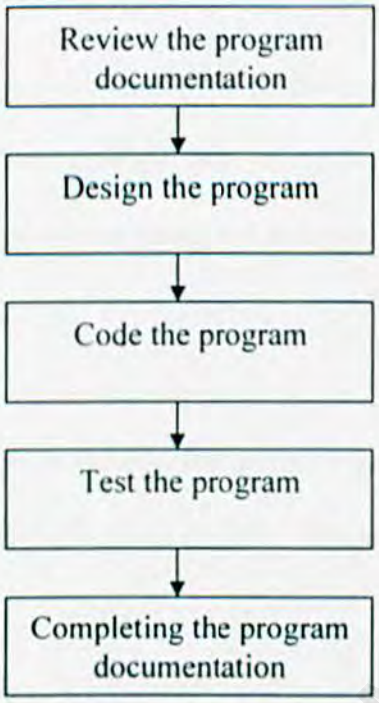

Review the program documentation
Design the program
Code the program
Test the program
Completing the program documentation

Figure 6-1: Program Development Process

i. Review the Program Documentation

The program documentation that was prepared during the early phases needs to be reviewed. This documentation helps to determine the work that need to be covered during the coding phase.

ii. Design the Program

After reviewing the documentation, the next level is to complete a program design during the system development where the ability of the system should be determined. This is the process designing what the system must do by developing a logical solution to the programming problem.

iii. Code the Program

Coding is a process of writing the program instruction where the instruction implements the program design. The coding step translates the design specification to a programming language and further compiled it into a machine-readable format.

iv. Test the Program

During the level of program testing, the program processes actual data and produces information on which user will be relying on. The testing involved are unit testing and integration testing.

v. Completing the Program Documentation

Completing the program is essential for the successful operation and maintenance of the system. This documentation includes the system's user manual that may be needed by most of the users as well as the system administrator.

6.2.2 Coding Approach

This system is developed modularly using top-down approach. This top-down approach allows the higher-level modules to be coded first before the lower-level modules. The codes in the lower modules contain only an entry and an exit. In shorts, this approach look at the large picture of the system first, and then exploding into smaller part.

6.2.3 Coding Principle Applied

There are a few principles need to apply when coding the program.

Readability

Readability is essential for future enhancement. Coding style and convention applied may strongly affect the readability. Codes need to be formatted to enhance understanding.

Reusability

Reusability is an important principle. It can be considered as a method for improving product quality throughout the system development process. In addition, it also reduces the coding time as well as the testing and documentation time.

Modularity

Software with effective modularity is easier to develop because function may be compartmentalized and interfaces are simplified. Independent modules are easier to maintain because secondary effects caused by design or code modification are limited, error propagation is reduced, and reusable module are possible.

6.2.4 Style Adopted

The coding paradigm adopted by the system is oriented at giving reliability and performance a balance.

Naming Convention

Naming convention provides easy identification for the programmer. The naming convention is created with coding consistency and standardization in mind. This will also make understanding an object easier by looking at the prefix of the name. For example:

Automated Essay Grading System Object Naming Conventions

ASP.NET Server Input Controls	**Instances**
Button	**btn**ObjectName
Calendar	**cal**ObjectName
Check Box	**chk**ObjectName
Check Box List	**chl**ObjectName
Data Grid	**dtg**ObjectName
Data List	**dtl**ObjectName
Drop Down List	**lst**ObjectName
Hyper Link	**lnk**ObjectName
Image Button	**but**ObjectName
Image	**img**ObjectName
Label	**lbl**ObjectName
Link Button	**lnk**ObjectName
Panel	**pnl**ObjectName
Radio Button	**rdb**ObjectName
Radio Button List	**rdl**ObjectName
Repeater	**rpt**ObjectName
Table	**tbl**ObjectName
Table Row	**row**ObjectName
Table Cell	**cel**ObjectName
Text Box	**txt**ObjectName

Besides the naming convention for each objects, there are also naming convention for stored procedures and tables in the SQL server.

Automated Essay Grading System Stored Procedure Naming Convention

1. Stored procedure name:
   usp_Type_ StoredProcName
   e.g. usp_Mst_Student
   usp_Stud_Result
   usp_Teacher_Question

2. Function name:
   func_FunctionName
   e.g. func_GetDocketNum

3. SQL statements:

```
if exists ( Condition )
BEGIN
<4 spaces>SELECT Attribute1
<4 spaces>FROM TableName
<4 spaces>WHERE Condition
END
```

.:. Uses 4 spaces, instead of tab, for each inner statement.

Automated Essay Grading System SQL Table Naming Convention

1. Table name

   a) Master table:
   Mst_TableName
   e.g. Mst_Customer

   b) View table:
   View_TableName
   e.g. View_Warehouse

   c) Module table:
   Module_TableName
   e.g. Stud_Result
   Teacher_Soalan

2. **FndMaster**
   - for dropdownlist control

3. For each created Table, add **10 attributes** for future maintenance.

Indentation and Spacing

The main purpose of indentation and spacing is to ease reading and tracing of code. They make the coding looks neat and tidy.

Program Comments

The comments explain the logic of the certain code, the purpose of a particular program block or other descriptive label. For Example:

```
--Start Matching
IF EXISTS(SELECT a.ID FROM ' + @aTableName + ' a INNER JOIN ' + @sTableName + ' b
ON a.ansWord = b.ansWord WHERE a.ID=@i)
```

The comment in the code above indicates the starting of the matching function.

6.3 Implementation of Essay Grading Algorithms

First, a class in the Visual Basic is needed to call upon the stored procedure that contains the main algorithm for grading essay. This class is responsible to encapsulate all the information that needs to call the stored procedure. The code is presented as follow:

```
Namespace Engine
    Public Class clsGrading
        '//declare strConn
        Private strConn As String =
System.Configuration.ConfigurationSettings.AppSettings.Get("ConnectionString")
        Private objSqlHelper As SqlHelper

        Public Sub startGrading(ByVal aTableName As String, ByVal sTableName As String, ByRef
dr As SqlDataReader)
            Dim objcls As New clsSqlHelper
            Dim objParams As SqlParameter()
            objcls.getParameterSet(objParams, "usp_GRADING_Grade_Essay")
            objParams(0).Value = aTableName
            objParams(1).Value = sTableName
            objcls.execDataReader(dr, "usp_GRADING_Grade_Essay", objParams)
        End Sub

    End Class
   End Namespace
```

Essay Grading Stored Procedure

The essay grading stored procedure is the part where the grading algorithm is implemented into the system. This stored procedure has five main parts. The first part is to declare and initializes all required variables. Then, it goes into a loop that calculates required calculations for matched terms. After that, it will start to find matches by using the synonym list. These are done separately so that the original word will have a higher priority to be matched. After all the matching processes, the final calculation for both Nearest Neighbour Algorithm and the Vector Space Model will be done. By using the result of the Vector Space Model, the last part will decide the grade of the essay. The result of the two algorithms and the grade will be returned in a table.

```
CREATE      proc usp_GRADING_Grade_Essay
  @aTableName varChar(30),
  @sTableName varchar(30)
as
SET NOCOUNT ON

DECLARE @SQL varChar(8000)
set @SQL = '
DECLARE @SumOfProductVSM FLOAT
DECLARE @SumOfProductNN FLOAT
DECLARE @i INTEGER
DECLARE @iMAX INTEGER
DECLARE @processWord VARCHAR(50)
DECLARE @synonymWord VARCHAR(50)
DECLARE @weight FLOAT
DECLARE @tmpScmVSM FLOAT
DECLARE @tmpScmNN FLOAT
DECLARE @tmpAnsVSM FLOAT
DECLARE @tmpAnsNN FLOAT
SET @i = 1
SET @SumOfProductVSM = 0
SET @SumOfProductNN = 0
-- Initialize the matched terms to 0
update ' + @aTableName + '
set att1 = 0
update ' + @sTableName + '
set att1 = 0

-- starts matching terms
```

```
WHILE @i <= ( SELECT MAX(a.ID) FROM ' + @aTableName + ' a INNER JOIN ' + @sTableName + '
b ON a.ansWord = b.ansWord )
BEGIN
   -- Find Matched term
   IF EXISTS(SELECT a.ID FROM ' + @aTableName + ' a INNER JOIN ' + @sTableName + ' b
   ON a.ansWord = b.ansWord WHERE a.ID=@i)
   BEGIN
      --Do calculation for Scheme Table
      SET @tmpScmVSM =(SELECT frequency FROM ' + @sTableName + ' WHERE ID=@i)
      SET @tmpScmNN =(SELECT frequency FROM ' + @sTableName + ' WHERE ID=@i)
      UPDATE ' + @sTableName + '
      SET weightVSM = SQUARE(@tmpScmVSM),
            weightNN = @tmpScmNN,
            att1 = 1
      WHERE ID = @i
      --Do calculation for Answer Table
       SET @tmpAnsVSM =(SELECT frequency FROM ' + @aTableName + ' WHERE ID=@i)
       SET @tmpAnsNN =(SELECT frequency FROM ' + @aTableName + ' WHERE ID=@i)
      UPDATE ' + @aTableName + '
      SET weightVSM = SQUARE(@tmpAnsVSM),
            weightNN = @tmpAnsNN,
            att1 = 1
      WHERE ID = @i
      --Do calculation for Matching Terms
      DECLARE @TEMPORARY1 as FLOAT
      SET @TEMPORARY1 = ISNULL(@tmpScmVSM,0) * ISNULL(@tmpAnsVSM,0)
      SET @SumOfProductVSM = @SumOfProductVSM + @TEMPORARY1
   END
SET @i = @i + 1
END
SET @i = 1

--Start matching in synonym list
WHILE @i <= ( SELECT MAX(a.ID) FROM ' + @aTableName + ' a INNER JOIN ' + @sTableName + '
b ON a.ansWord = b.ansWord )
BEGIN
   IF EXISTS(SELECT ID FROM ' + @aTableName + ' WHERE ID=@i AND att1=0)
   BEGIN
      -- Try to find match in synonym list
      SET @processWord = (SELECT ansWord FROM ' + @aTableName + ' WHERE ID=@i)
      IF EXISTS(SELECT a.ID FROM ' + @aTableName + ' a INNER JOIN List_Of_Synonym b
         ON    a.ansWord = b.synonym WHERE a.ID=@i)
        BEGIN
           SET @synonymWord =(SELECT originalWord FROM List_Of_Synonym
              WHERE synonym=@processWord)
              --now try to match the synonym with the document
              IF EXISTS(SELECT ansWord FROM ' + @sTableName + '
              WHERE ansWord=@synonymWord)
              BEGIN
                 SET @weight = (SELECT weight FROM List_Of_Synonym
                 WHERE synonym=@processWord    AND originalWord = @synonymWord )
                 SET @tmpAnsVSM =(SELECT frequency FROM ' + @aTableName + '
                 WHERE ID=@i)* @weight
                 SET @tmpAnsNN =(SELECT frequency FROM ' + @aTableName + '
                 WHERE ID=@i)* @weight
                 UPDATE ' + @aTableName + '
            SET weightVSM = SQUARE(@tmpAnsVSM),
                weightNN = @tmpAnsNN,
                att1 = 1
            WHERE ID = @i

--Do calculation for Matched Terms
            DECLARE @TEMPORARY2 as FLOAT
            SET @TEMPORARY2 = ISNULL(@tmpScmVSM,0) * ISNULL(@tmpAnsVSM,0)
            SET @SumOfProductVSM = @SumOfProductVSM + @TEMPORARY2
          END
        END
    END
SET @i = @i + 1
END

-- Compute result for VSM
DECLARE @tmp1 float
DECLARE @tmp2 float
DECLARE @resultVSM float
SET @tmp1 = (SELECT SQRT(SUM(weightVSM)) FROM ' + @aTableName + ' )
SET @tmp2 = (SELECT SQRT(SUM(weightVSM)) FROM ' + @sTableName + ' )
SET @resultVSM = @SumOfProductVSM/(@tmp1 * @tmp2)

-- Compute result for NN
DECLARE @tmp3 float
DECLARE @tmp4 float
DECLARE @tmp5 float
DECLARE @resultNN float
SET @tmp3 = (SELECT isnull( SUM(weightNN), 0 ) FROM ' + @sTableName + '
WHERE NOT(att1 = 1))
SET @tmp4 = (SELECT SUM(weightNN) FROM ' + @aTableName + ')
SET @tmp5 = (SELECT SUM(weightNN) FROM ' + @sTableName + ')
SET @resultNN = @tmp3/@tmp5

--Deciding Grade using VSM
DECLARE @Result varchar(10)
IF @ resultVSM > 0.9
BEGIN
SET @Result = "A"
END
ELSE
IF @ resultVSM > 0.8
BEGIN
  SET @Result = "B"
END
ELSE
IF @ resultVSM > 0.7
BEGIN
  SET @Result = "C"
END
ELSE
IF @ resultVSM > 0.5
BEGIN
  SET @Result = "D"
END
ELSE
```

```
  SET @Result = "F"
--save result into output table
SELECT ISNULL(@tmp6,0) AS ResultNN, ISNULL(@tmp7,0) AS ResultVSM, ISNULL(@Result,"F")
as Result
'

EXEC (@SQL)
SET NOCOUNT OFF
GO
```

After obtaining the results from the essay grading stored procedure, another SQL stored procedure is called to write the result into the table that stores the student's result.

```
CREATE  proc usp_GRADING_Result
@StudID varchar(30),
@TestID varchar(30),
@QuestionID int,
@ResultNN float,
@ResultVSM float,
@Result varchar(30)
As

DECLARE @StudName varchar(30)
SET @StudName = (SELECT studName FROM Mst_Student WHERE studID = @StudID)

DECLARE @teacherID varchar(30)
SET @teacherID = dbo.func_Get_TeacherID_By_StudID(@StudID)

INSERT INTO
Mst_Student_Result(studID,studName,teacherID,testID,questionID,resultNN,resultVSM,result)
VALUES(@StudID
            ,@StudName
            ,@TeacherID
            ,@TestID
            ,@QuestionID
            ,@ResultNN
            ,@ResultVSM
            ,@Result)
SELECT 'OK' as flag
GO
```

6.4 Chapter Summary

In this system implementation phase, all the design phases that have been discussed are directed toward a goal to translate the representation of system design into a piece of working system that can be understood and execute by the computer.

Chapter 7 System Testing

7.0 System Testing

Testing is a crucial process for uncovering bug and to test the system reliability. The main objective of testing is to uncover different types of errors that exist while executing the system. System testing is a crucial phase to assure software quality and it represents the ultimate review of specification, design and coding. However, testing can only show that software defects are present.

When developing a system, testing usually involves several stages. An example of testing process is shown as below:

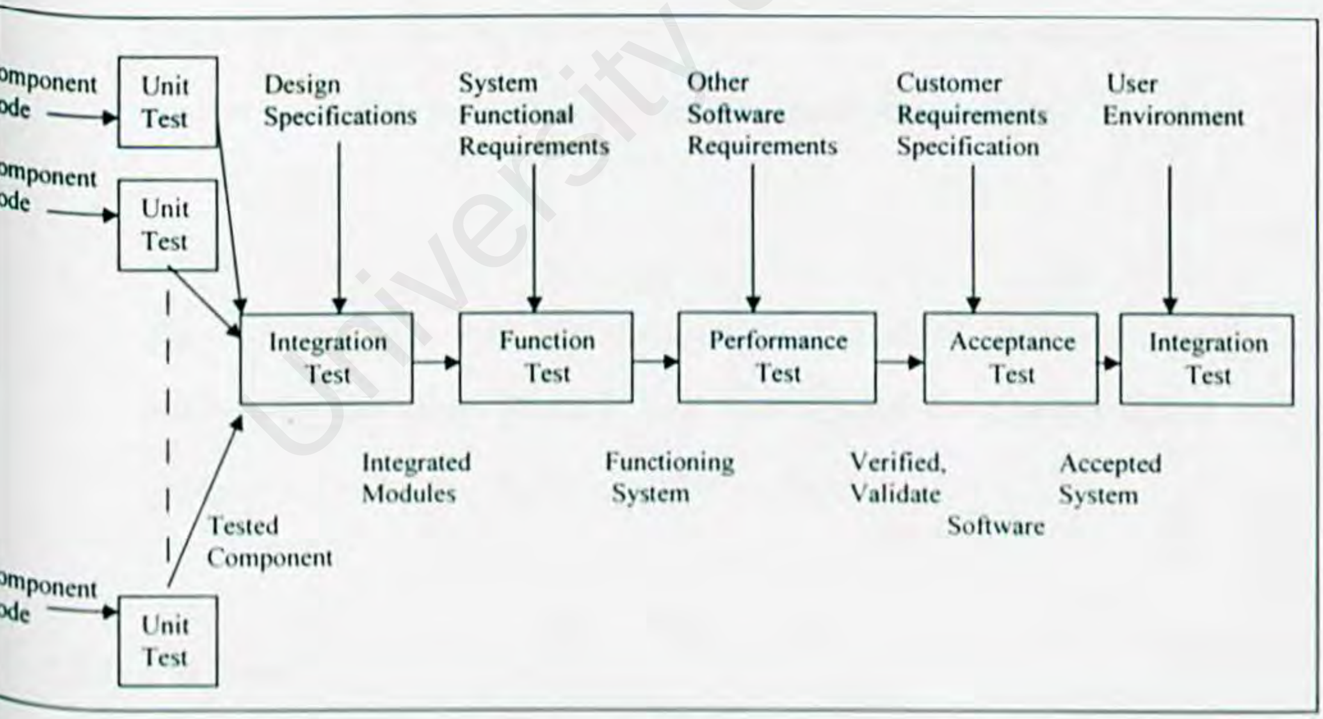


Figure 7-1: Testing process

Generally, there were 3 stages involve altogether and were listed down as below:

a. Unit Testing

This is the first stage of testing where each program component is tested on its own and is isolated from the other components in the system. It verifies that the component functions work properly with the types of input and output expected from studying the component's design. After each component has been tested, the interaction between these components must be tested again to ensure that the components can be integrated.

b. Integration Testing

This stage ensures that the interfaces among the components are defined and handled properly. It is the process of verifying that the system modules work together as described in the system and program design specifications

c. System Testing

This is the last stage which is performed to find out errors, which result from unanticipated interactions of system components or units. It is to ensure that the whole system works according to users' specifications.

7.1 Type of Faults

Types of faults that can be found in automated essay grading system are algorithmic faults and computation faults.

7.1.1 Algorithmic faults

Algorithmic faults use to happen when a component's algorithm or logic fails to produce the expected output for a given input. These kinds of faults happen due to wrong proceeding steps. Example of algorithmic faults in automated essay grading system is: Using the wrong weight to calculate the result.

7.1.2 Computation/Precision faults

Computation or precision faults use to occur when a particular formula is being implemented wrongly or the component does not complete a result to a required accuracy. This type of faults is being tested in component such as the grading system.

7.2 Testing Techniques Used

7.2.1 Ad Hoc Testing

Ad hoc testing is an attempt to break the program or make it fail with trying whatever comes to mind. Normally, many errors will be found during the testing.

7.2.2 White Box Testing

White Box testing is the type of testing that deals directly with the structure of the code within a module or a code segment. There are basically six types of code coverage in white box testing. Most of the testing is discussed in the unit testing

- **Segment Coverage**

Each and every segment of the node between control structures is supposed to execute at least once.

- **Branch Node Coverage**

  Each and every branch of every possible direction is taken at least once.

- **Compound Condition Coverage**

  When multiple conditional appear in the code, every possible combination is tested based on a truth table.

- **Basis Path Testing**

  Each independent path through the code is usually taken as predetermined order. When dependencies appears in the code, each path where dependency appears exists must be tested.

- **Data Flow Testing**

  This approach is to uncover anomalies such as variables, which are used but not properly initialized.

- **Loop Testing**

  This type of testing is related to testing single loop, concatenated loops (sequence of loop) and nested loops (one or more loops within loops).

White box testing is mainly done to check the system. For example: whether the timer function in the exam mode works as it should or not. All the functions in the system are checked using the stated tests.

7.2.3 Black Box Testing

This type of testing involves testing functions of a module without knowing the logic structure of the code. It focuses on the most important aspects of a module in the term of how well the module meets its specification. The purpose of this testing is to make sure that the output of the code will be what should be expected. Following is the unit testing of the grading system by using Black Box Testing.

To test the essay grading algorithm, the exact answer is feed into the system to see whether it achieves a perfect match or not. When a perfect match is achieved, the Nearest Neighbour Algorithm will give a result of 0 and the Vector Space Model will give a result of 1. After that, the content in the exact answer will be reduced by deleting one paragraph. The new reduced text will be fed into the system to see whether the result changes as it should. This process will be repeated until an empty document is given to the system. When an empty document is given, the result should be 1 for the Nearest Neighbour Algorithm and 0 for the Vector Space Model. During the test, all the results are recorded and these results are included in the following table. NN Result is the result of the Nearest Neighbour Algorithm and VSM Result is the result of the Vector Space Model Algorithm. A column with the name of "1 – NN Result" is added so that the comparison of both results is easier.

Sample	NN Result	1 - NN Result	VSM Result
1	0	1	1
2	0.0675	0.9325	0.969270027
3	0.165	0.835	0.920230156
4	0.215	0.785	0.880954646
5	0.3	0.7	0.826384196
6	0.39	0.61	0.772245892
7	0.4575	0.5425	0.740768183
8	0.505	0.495	0.720178453
9	0.5975	0.4025	0.664779955
10	0.6975	0.3025	0.564842712
11	0.825	0.175	0.470734436
12	0.9675	0.0325	0.156446555
13	1	0	0

Table 7-1: Testing Result

From the table above, it is obvious that the changes of the result are the same as the algorithm should works. Then the data in the table above are plot into a graph, in Figure 7-2. This graph shows that the Nearest Neighbour Algorithm will give a line that is increasing from 0 to 1 which represents that the document is very close at the beginning and further away in the end. The Vector Space Models result starts with a 1 and ends with 0 which also represents the document is further from the answer scheme.

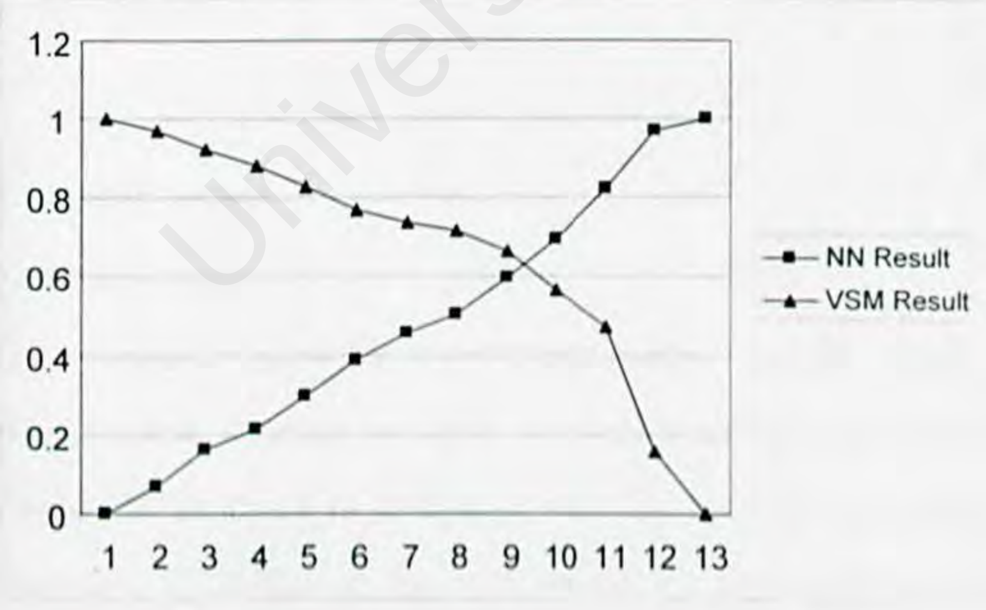


Figure 7-2: Graph of testing result 1

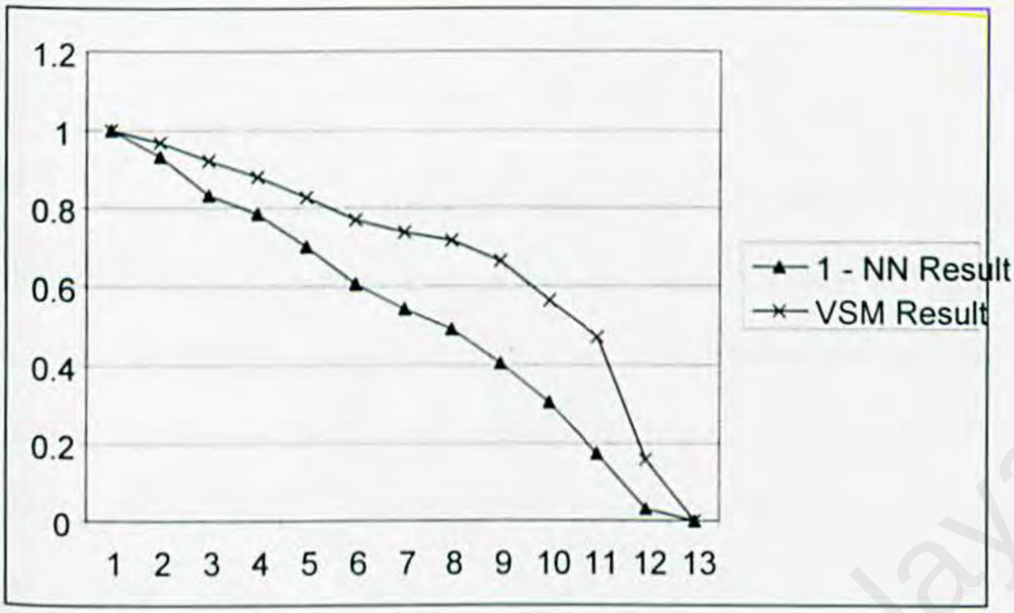


Figure 7-3: Graph of testing result 2

When the graph of "1-Nearest Neighbour result" and Vector Space Model result is drawn, a graph in Figure 7-3 could be seen. From the graph above, it could be conclude that both of the algorithms are well implemented because it gives the desired outcome when using the black box testing.

7.3 Type of Testing

7.3.1 Unit Testing

The unit testing in automated essay grading system is done by testing every function when it is built. All stored procedures are tested in the SQL query analyzer when it is written. The result will be checked by verifying the result in respective tables. All functions in Visual Basic are also tested when it is written. Buttons drop down list, and data grids are tested by running the respective web form.

7.3.2 Integration Testing

The purpose of the integration testing is to know whether the entire software is able to work as one program. It will also verify that each module will be able to function together. Integration testing concentrates on module interaction and the detection of interface errors. The design specification is referred for the purpose of verification and helps to test the software according to the dependencies present in particular module that being tested.

In the automated essay grading system, the most important integration test is to integrate the indexing module and the essay grading module. When doing the testing on the integration of these two modules, the inputs and outputs of each function is checked extensively to make sure that the right type of data type goes in and out of each involved functions. Care had to be taken in this testing phase as the indexing module and grading module are both complex modules and any minor mistakes might cause unanticipated errors. Another important integration test is to test the integration of the system with the user interface. Care had to be taken to make sure all the data types are well defined.

7.3.3 System Test

Functions Testing

First, all the functions in the system are tested again after the integration. This is to check whether there is any error after the integration. Each function in each web form will be tested.

Performance Testing

The purpose of performance testing is to test whether the system could perform in a acceptable response time or not. This test targets the run-time performance of the system. The tests done to verify the performance are Stress Testing, Security Testing and Human Factor Testing. The main purpose of the stress testing is to determine whether the system can handle, as it should, large and varies workload at one time. It subject system to high loads over a short period of time. This test is done by using multiple computers to log into the system at the same time, and run some functions of the system. Then the security testing is done to verify the protection mechanism such as making sure that the user uses a password with a combination of characters and numbers. And also try to log into a teachers account using a students login ID. Then the human Factor Testing is run by some users to get the feedback regarding the appearance and the interaction of the system. All aspect that may be related to ease of use, such as display screen, will be examined.

7.4 Debugging Strategies

Debugging is actually of finding and fixing the errors. There are several debugging strategies that applied in developing the automated essay grading system such as:

- Built-in Error Detection

  Error will be discovered if a program is not performing well. ASP.NET has built-in error detection where an error message together with the lines number where the error occurred will be debugged. With this features, the debugging work becomes much easier and faster.

- Reviewing the Algorithm Used

  Reviewing algorithm and computations for the correctness and efficiency will help to discover logic error or database error. Usage of different algorithms will sometime increase the efficiency of the program.

- Display the Passing Value On Screen

  By displaying the passing value on screen, it helps to ensure that the correct value has been passed to the next program for processing.

- Check Success Status

  The success status is checked to determine whether to continue the process or exit from the program and display error message whenever there is failure in the previous process.

- Using Query Analyzer Provided by SQL Server 2000

  Query analyzer will helps to test the SQL statement and information about the error will be provided. Query Analyzer is also used to correct the SQL statement when wrong information is being retrieved.

7.5 Chapter Summary

Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Unit, integration and system testing has been carried out for automated essay grading system. At the end of the testing phase, the system should be able to perform the tasks required and free of some errors.

Chapter 8

8.0 System Evaluation

Before delivering a system to the end user, it is crucial that a system should be evaluated. This is the ultimate phase of developing a system. System evaluation is implemented by more than simply comparing the information obtained with the information which is expected. It was related to user environment, attitudes, information priorities and several other concerns that are to be considered carefully before effectiveness can be concluded. At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on a variety of sources and information. In this system evaluation, the main focus is on the evaluation of the essay grading algorithm.

8.1 Evaluation of the Essay Grading Algorithms.

To evaluate the efficiency of the Vector Space Model and Nearest Neighbour Algorithm, a fixed answer scheme is used to grade other collected samples. In the system, the indexing process and the matching process is the same for both algorithms. Then, the thirteen testing samples which are also modification from the answer scheme, another ten samples of related answers and ten samples of unrelated answers are collected and feed into the system. All these answers will be graded or compared against the fixed answer scheme mentioned above.

First, ten unrelated answer samples are fed into the system. The answer samples used here are answer scheme for other questions in other chapters. This means that although their lengths are almost the same, the content is totally different with the exceptions of common words. The reason that unrelated answers are used first is to see the effectiveness of the system to identify unrelated materials. The collected results are recorded in table 8-1.

A column "1 – NN Result" is added to the table so that the result of the Nearest Neighbour algorithm could be compared easily with the result of the Vector Space Model. It is subtracted by 1 because the ranges of results from both algorithms are between 0 and 1, but the result of Nearest Neighbour is opposite with the result of the Vector Space Model. By subtracting with 1, we will inverse the result of the Nearest Neighbour Algorithm and still give it a range of 0 and 1.

Sample	NN Result	1-NN Result	VSM Result
1	0.8725	0.1275	0.091163146
2	0.8825	0.1175	0.103637577
3	0.8525	0.1475	0.121676186
4	0.865	0.135	0.104989066
5	0.86	0.14	0.175763336
6	0.915	0.085	0.0571163
7	0.86	0.14	0.109586092
8	0.895	0.105	0.094628446
9	0.86	0.14	0.129503867
10	0.8575	0.1425	0.088017157

Table 8-1 Result of Unrelated Samples

To have a better view of the results in the table, a graph of the Nearest Neighbour Algorithms result and the Vector Space Models result are plotted. As shown in the graph in Figure 8-1 below, two lines that are almost straight could be seen. These two lines that seem quite straight actually show that both algorithms will distinguish unrelated documents and give a result in the range of those two lines. Both algorithms still give a certain value to unrelated documents because there are still common words in the documents that are not eliminated.

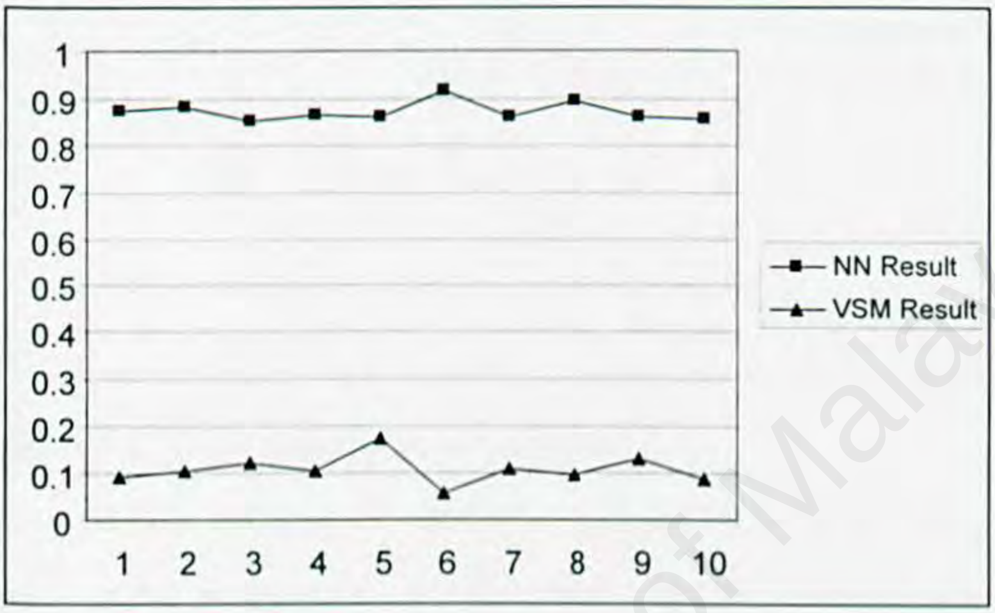


Figure 8-1: Graph of unrelated samples result 1.

When the graph of "1 – NN Result" and the Vector Space Models result are plotted, the graph in Figure 8-2 below could be observed. The results show two lines with a similar pattern. When observed carefully, the graph also shows that the Vector Space Model is more sensitive to the data it evaluates. This could be seen through the changes of the result from Vector Space Model is bigger than the changes of result from the Nearest Neighbour Algorithm. The result from Nearest Neighbour Algorithm formed a line which is straighter when compared to the other one. Derived from the observation above, it could be said that when evaluating a set of documents in the same category, the Vector Space Model could give a better judgment by using a larger range to describe the document it evaluates.

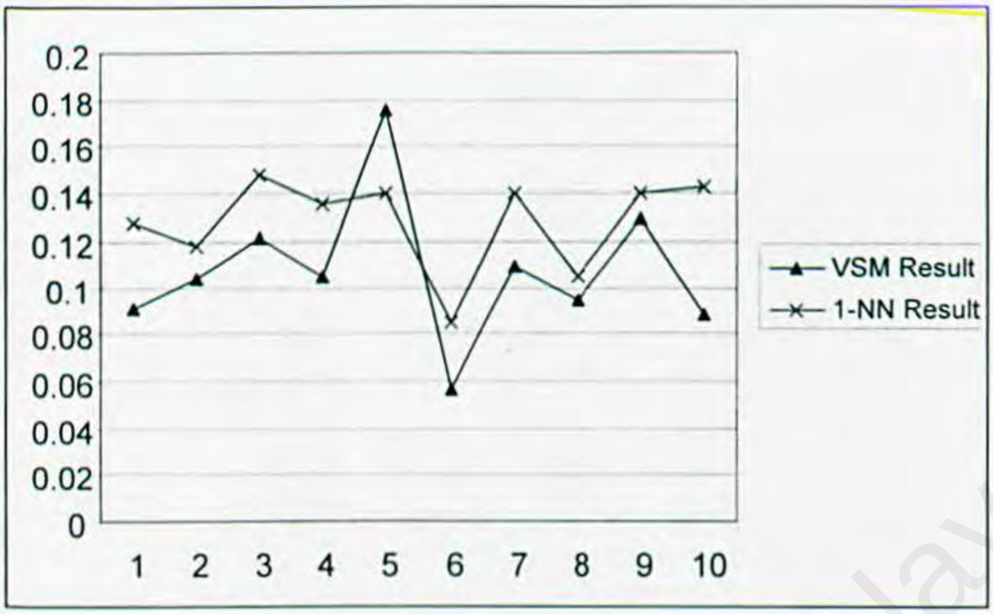


Figure 8-2: Graph of unrelated samples result 2.

Besides that, from the result of giving unrelated documents into the system, we could see that the highest result is lower than 0.18. This is the level in the system that shows that a documents that are not related to the answer scheme. For the rest of this report, without further enhancing the elimination of the common words by adding more common words in the list, it would be assumed that when an unrelated document is given to the system, the estimated output would be in the range of 0 to 0.2. In other words, if a document gets any result lower than 0.2, it would be estimated to be unrelated to the answer scheme. In the case of essay grading, then any answers that give a result with 0.2 or lower should get an 'F' or considered fail.

Then, the result from the testing phase will be used here. Table 8-2 is the result from the testing phase. The first sample is actually the answer scheme itself. The following samples are actually modifications of the answer scheme itself. The content of the answer scheme was reduced by the length of one paragraph and feed into the system. The process repeats until the whole essay is empty.

Sample	NN Result	1 - NN Result	VSM Result
1	0	1	1
2	0.0675	0.9325	0.969270027
3	0.165	0.835	0.920230156
4	0.215	0.785	0.880954646
5	0.3	0.7	0.826384196
6	0.39	0.61	0.772245892
7	0.4575	0.5425	0.740768183
8	0.505	0.495	0.720178453
9	0.5975	0.4025	0.664779955
10	0.6975	0.3025	0.564842712
11	0.825	0.175	0.470734436
12	0.9675	0.0325	0.156446555
13	1	0	0

Table 8-2 Results of Testing Samples

A graph in Figure 8-3 was plotted according to the result above, using the "result of Vector Space Model" and the "result of Nearest Neighbour minus by 1". This graph shows two descending lines. The line that represents the "Nearest Neighbour Results minus by 1" descends linearly. The "Vector Space Model Results" descends in a curve manner which tends to reach 0 a little slower than the other line. The curve is also similar to the beginning of the COS functions curve. This is due to the function used to calculate the Vector Space Models result is the cosine of two documents. From this graph, we could see clearly that the "Nearest Neighbour result minus by 1" will give a result that reach zero very fast. And the Vector Space Model approach will reach the result of zero a little slower.

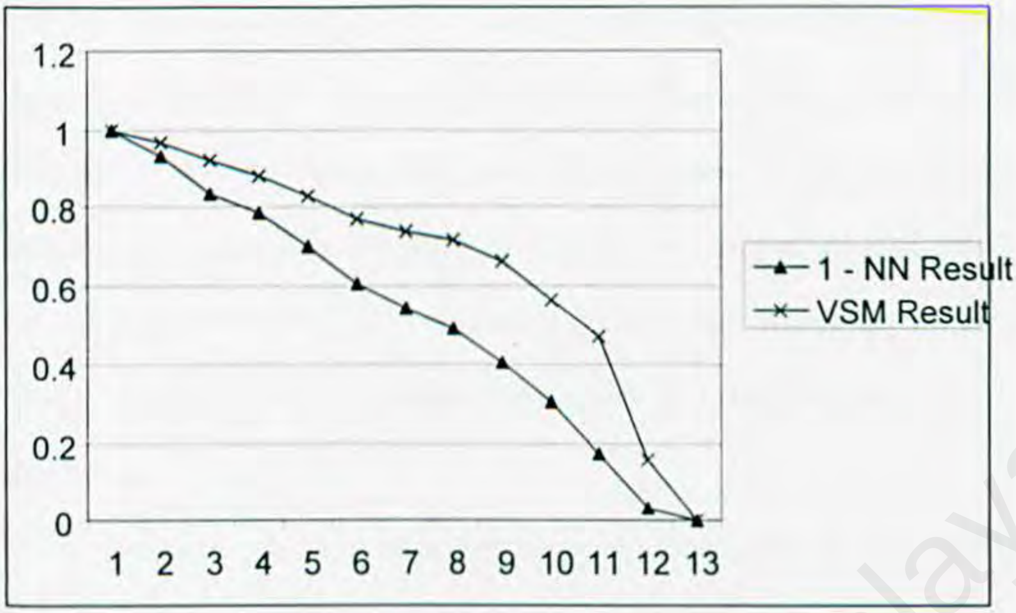


Figure 8-3: Graph of testing samples result.

From here, we could see that the Vector Space Model will have a larger range when evaluating two documents. This will give us more precision when compared to the Nearest Neighbour approach. For example, from document 1 to document 9, the Nearest Neighbour approach had used the value from 1 to 0.4. In the Vector Space Model approach, it had only been decreasing from 1 to 0.66. This shows that the Vector Space Model approach is better because in a same range, it could categorize the evaluated essay into more detailed category or into more categories. It could be said that when evaluating essays from different categories, the Vector Space Model uses a smaller range partitions which enable the algorithm to have more categories.

Besides that, by using the earlier conclusion when using unrelated documents, any documents that have a result of 0.2 or lower could be considered as unrelated documents. In the result above, there are 3 samples that falls under the category of unrelated document when judged by the Nearest Neighbour Algorithm. There are only two samples that falls under this category when judged by the Vector Space Model. In other words, 3 samples will get an 'F' when using the Nearest Neighbour Algorithm and only 2 samples will get an 'F' when using the Vector Space Model. In the case here, while all the samples are actually modification of the original answer scheme, the Vector Space Models judgment seems more reasonable. Besides that, the ability of the Vector Space Model to distinguish an unrelated document had been shown by using unrelated documents.

Sample 11 is the sample getting an 'F' when judge by the Nearest Neighbour Algorithm. Even if this sample should get an 'F', when using the Vector Space Model, the failing range could be modified to be higher. By using the Vector Space Model, we could also define a certain range as a range for considerations, which is the border line between a pass or fail where the human essay grader could give the judgment whenever a result falls in this range.

After that the ten samples of related answers are given into the system and the results are shown in the Table 8-3 below.

Document	NN Result	1 - NN Result	VSM Result
1	0.795	0.205	0.175174795
2	0.81	0.19	0.191381137
3	0.78	0.22	0.201178826
4	0.8525	0.1475	0.205393419
5	0.695	0.305	0.2694577
6	0.66	0.34	0.304935305
7	0.68	0.32	0.425312399
8	0.6	0.4	0.513754581
9	0.535	0.465	0.559671978
10	0.4225	0.5775	0.76064229

Table 8-3 Results of related samples

Graphs could be plot from the results; this will give a clearer view of the results gained. Figure 8-4 is the graph of Nearest Neighbour Algorithms result and Vector Space Models result. The table below has the same columns as the table above to ease the comparison between the results of two different algorithms.

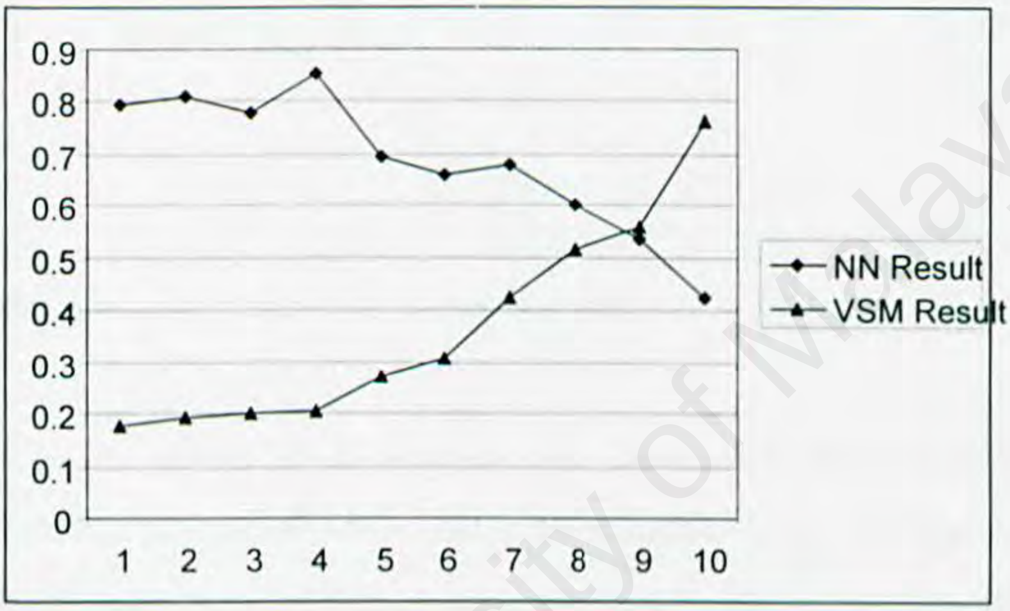


Figure 8-4: Graph of related samples result.

The graph of Nearest Neighbour Algorithms result and Vector Space Models result shows that the chosen samples have different similarity with the answer scheme. To have a clearer view of the effect of both algorithms on the sample answers, the graph of Nearest Neighbour Algorithms result subtracted by 1 and the Vector Space Models result should be plot.

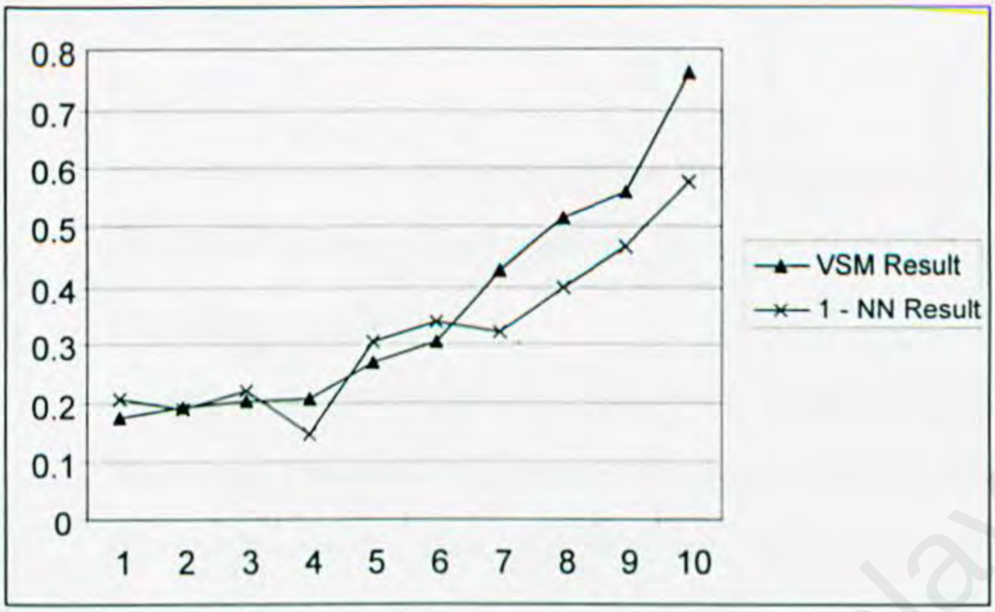


Figure 8-5: Graph of related documents result.

By viewing the graph above, it is clearer that both algorithms judge the documents in a similar pattern. This could be derived by the two lines that have the similar shape. We could also see that in this graph, sometimes, the Vector Space Model gives a higher value and sometimes it gives a smaller value. But when observed carefully, we could see that the graph stills obeys the pattern similar to the one in Figure 8-3. This proves the conclusion derived earlier, which is Vector Space Model, is a better algorithm because it could categorize different document into a much detailed category or into more categories. This is because with more categories, we could define the outcome grade with a more precise manner.

8.2 System Strengths

- Security

  The system checks and enforces a rule that the user must use a password with the combination of characters and numbers.

- Test result report generator

  The students can get their result in a short time. The teacher can view the result of the students. Teachers can also save a soft copy of the result list in the format of Microsoft Excel with a single click. When saved in Excel form, the teacher could modify, print, generate a graph or do anything that the Microsoft Excel could do.

- Online Test

  Test is conducted online in a fixed time. Before and after the fixed time, the question can't be access by the students.

8.3 System Constraints

The system constraints are described as below:

- Could not give a correct grade

  Even though the system could identify the similarity of two documents, it could not assign a correct grade to the answer. The grade assigning process is hard coded into the system and this is inflexibility of the system.

- Could not accept scanned text

  The system does not accept text in the form of scanned image. Thus, handwritten text must be type into a computer text file before it could be evaluated.

8.4 Future Enhancement

Due to the limitation of this system, there are a few suggestions that may be useful to future enhancement of the automated essay grading system. The suggestions are as below:

- Implement a function that could accept scanned image.

  This would help to save the time to type in the question or essay. The teacher could upload handwritten questions and the teacher could scan the students essay into image and upload it for evaluation.

- Implement an intelligent grade assigning function

  An intelligent grade assigning function is needed so that the grade assigned to the evaluated essay would be more accurate.

8.5 Knowledge and Experience Gained

Besides knowledge on technical aspects such as Windows XP Server, ASP.Net, VB.Net and SQL Server, there are also other valuable experiences gained from working on this project such as:

- Being exposed to the real system development environment especially dealing with users
- Learn how to manage a project as in time and resource
- Concept on how to integrate and fully utilize various technologies into developing system
- Experience on how to set up and configure various technologies to be able to serve as a live system.
- Learn to work independently and in team
- Cultivated skills in writing documentations and reports

- Boost self-confidence, self-esteem and good communication skill

8.6 Reviews on Goals

There should be certain expectation and objective achieved at the final stage of the project.

8.6.1 Expectation Achieved

The system had fulfilled the expectation stated at the early stage of the project. All the basic foundation of the system was being designed and implemented. Moreover, the end product met the criteria such as user friendliness, reliability, manageability, expandability and so on.

8.6.2 Objective Achieved

The system created had fulfilled all the requirements stated in the early chapter, therefore, the objectives to establish the application had been achieved.

8.7 Chapter Summary

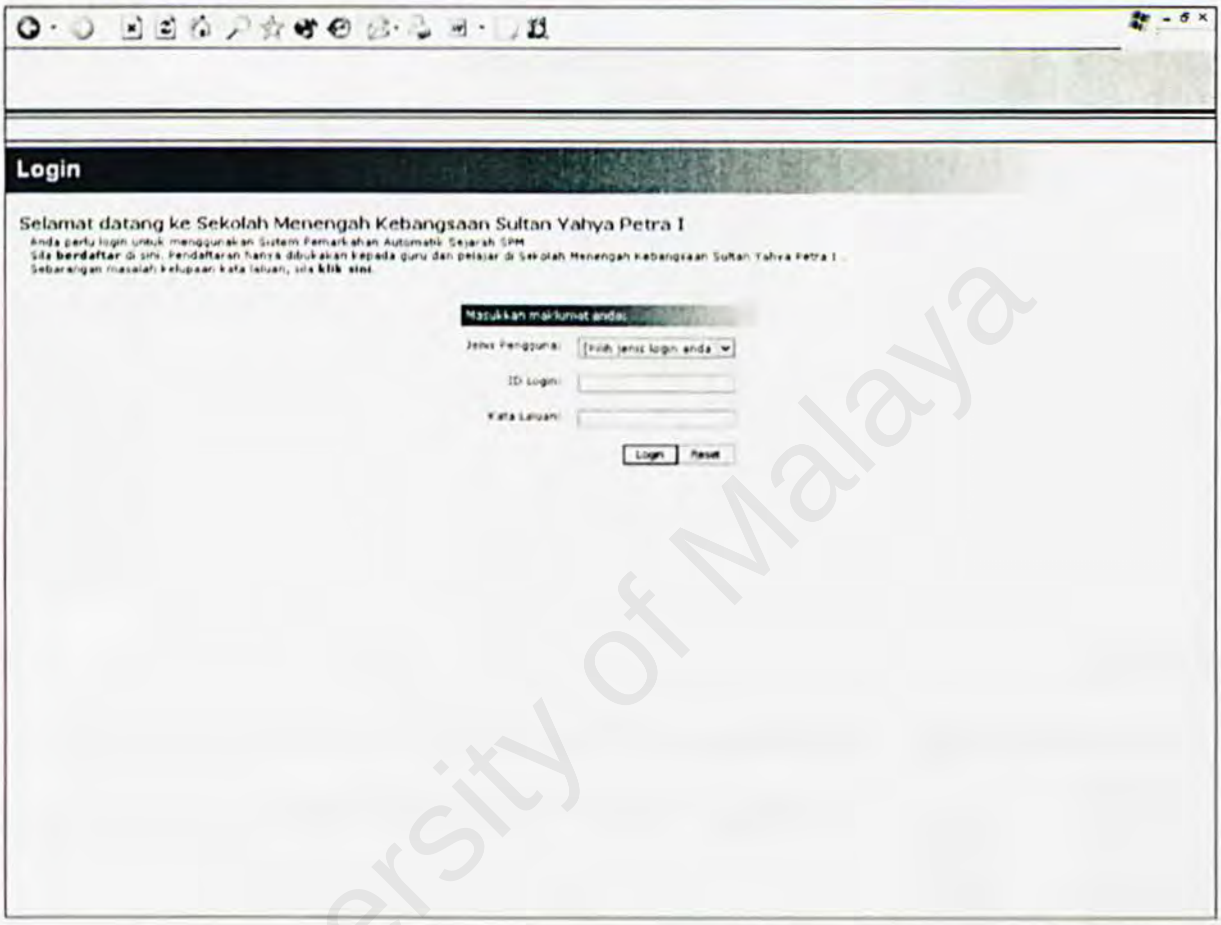
As a conclusion, this project was succeeded in achieving the objectives of developing a web based automated essay grading system. Although this automated essay grading system cannot grade essays exactly like a human essay graders, it successfully imitates the grading ability by using algorithms that compares the similarity of two documents. After the development of this system, the effectiveness of the Vector Space Model and the Nearest Neighbour Algorithm had also been compared and come to a conclusion that using the Vector Space Model had a better advantage than the Nearest Neighbour Algorithm.

Besides that, throughout the development of this project, a lot of precious knowledge on web based programming was gained. This included the configuration and management of Windows XP Server and IIS, programming knowledge in ASP.Net and VB.Net as well as the techniques and concepts in implementing database like using Stored Procedures. This project has been a very useful experience which exposes the idea of research work to the developer

APPENDIX A

USER MANUAL

Pengguna :: Login



Login

Selamat datang ke Sekolah Menengah Kebangsaan Sultan Yahya Petra I

Anda perlu login untuk menggunakan Sistem Pemarkahan Automatik Sejarah SPM

Sila **berdaftar** di sini. Pendaftaran hanya dibukakan kepada guru dan pelajar di Sekolah Menengah Kebangsaan Sultan Yahya Petra I .

Sebarangan masalah kelupaan kata laluan, sila **klik sini**

Masukkan maklumat anda:

Jenis Pengguna: Pilih jenis login anda

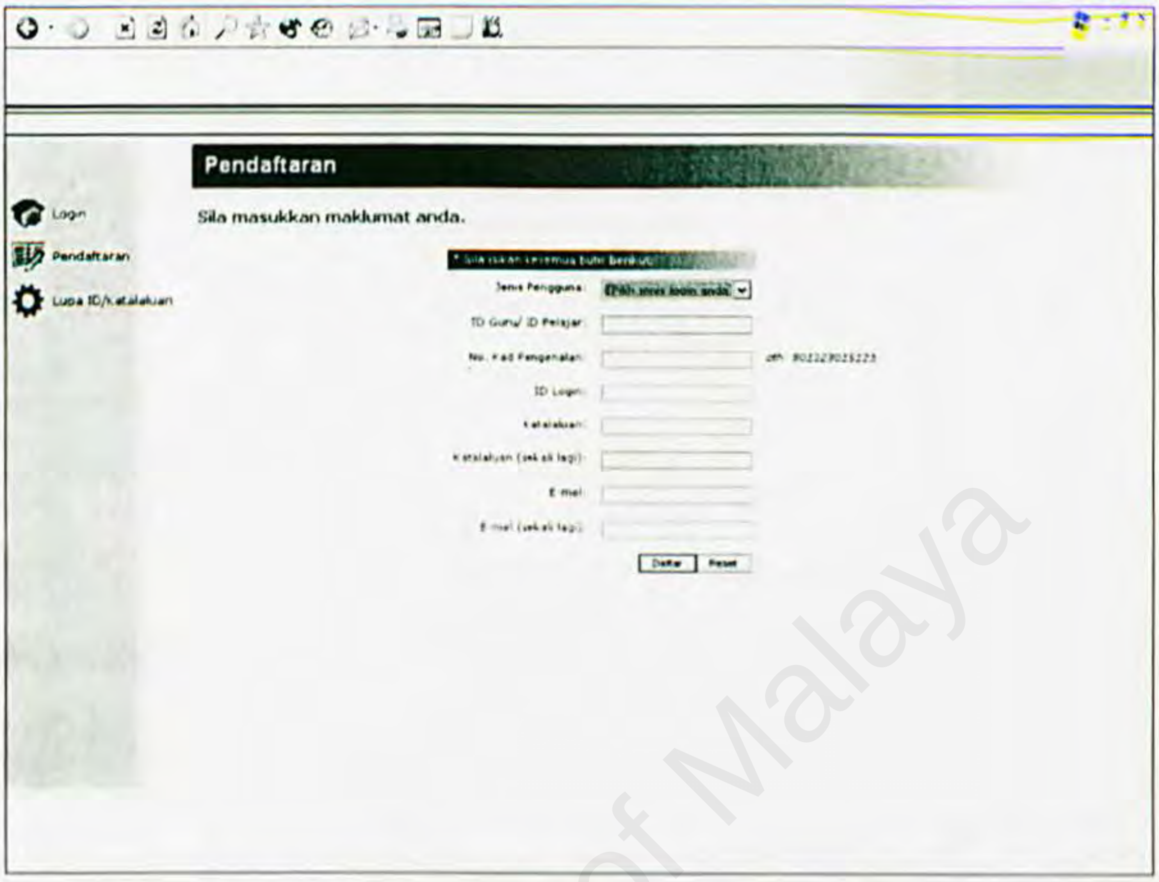
ID Login:

Kata Laluan:

Pada mukasurat Login, anda dikehendaki memilih jenis login iaitu Guru atau Pelajar. Selepas itu, pasangan ID Login dan katalaluan yang betul hendak dimasukkan pada petak yang telah dilabel.

Pengguna akan dijemput ke laman utama jika pasangan ID Login dan katalaluan yang betul telah dimasukkan.

Pengguna :: Login :: Pendaftaran



Pendaftaran

Sila masukkan maklumat anda.

* Sila isikan kesemua butir berikut

Jenis Pengguna: Pilih jenis login anda

ID Guru/ ID Pelajar:

No. Kad Pengenalan: cth: 901123025223

ID Login:

Katalaluan:

Katalaluan (sekali lagi):

E mel:

E mel (sekali lagi):

Jika pengguna adalah Guru atau Pelajar dan belum memiliki ID Login, mereka boleh klik pautan berdaftar dan mendaftarkan ID Login yang baru.

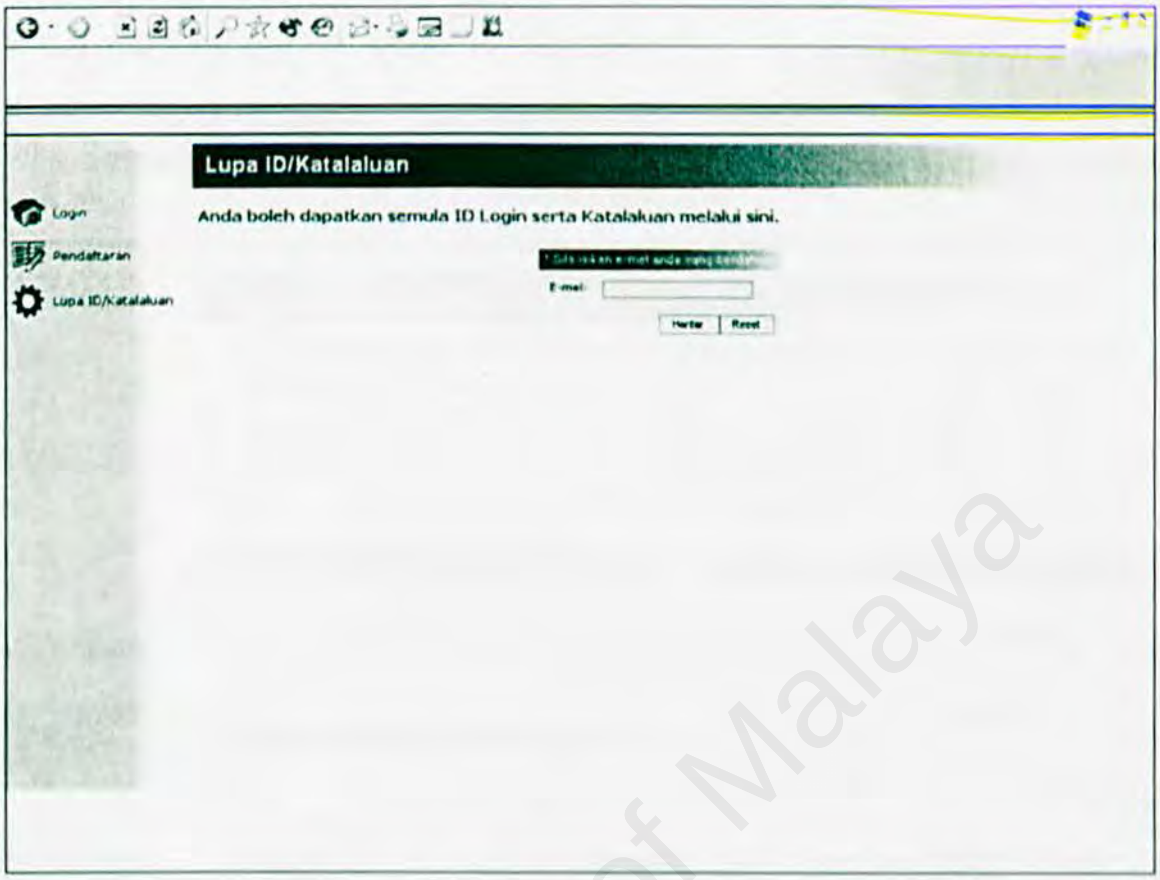
Dalam Skrin pendaftaran, semua butir-butir dikehendaki diisikan dengan betul. Setelah butir-butir diisikan, pengguna dikehendaki menekan butang 'Daftar' untuk berdaftar.

Butang 'Reset' adalah untuk mengkosongkan semua petak supaya butiran yang baru dapat diisikan.

Jika pengguna hendak balik ke skrin Login, pengguna dikehendaki menekan butang Login di sebelah kiri.

Hanya ID Guru atau Pelajar yang wujud dalam sistem dapat mendaftar. Jika ID anda tidak wujud, sila jumpa system administrator untuk memasukkan ID ke dalam sistem.

Pengguna :: Login :: Lupa ID Login/Katalaluan



Lupa ID/Katalaluan

Anda boleh dapatkan semula ID Login serta Katalaluan melalui sini.

Jika pengguna lupa ID Login atau katalaluan, pengguna boleh klik pautan klik sini.Selepas itu, pengguna dikehendaki memasukkan emel yang telah mereka daftarkan dalam sistem untuk mendapatkan ID Login dan katalaluan mereka. ID Login dan katalaluan akan dihantar ke emel tersebut.

Emel yang salah tidak dipedulikan.

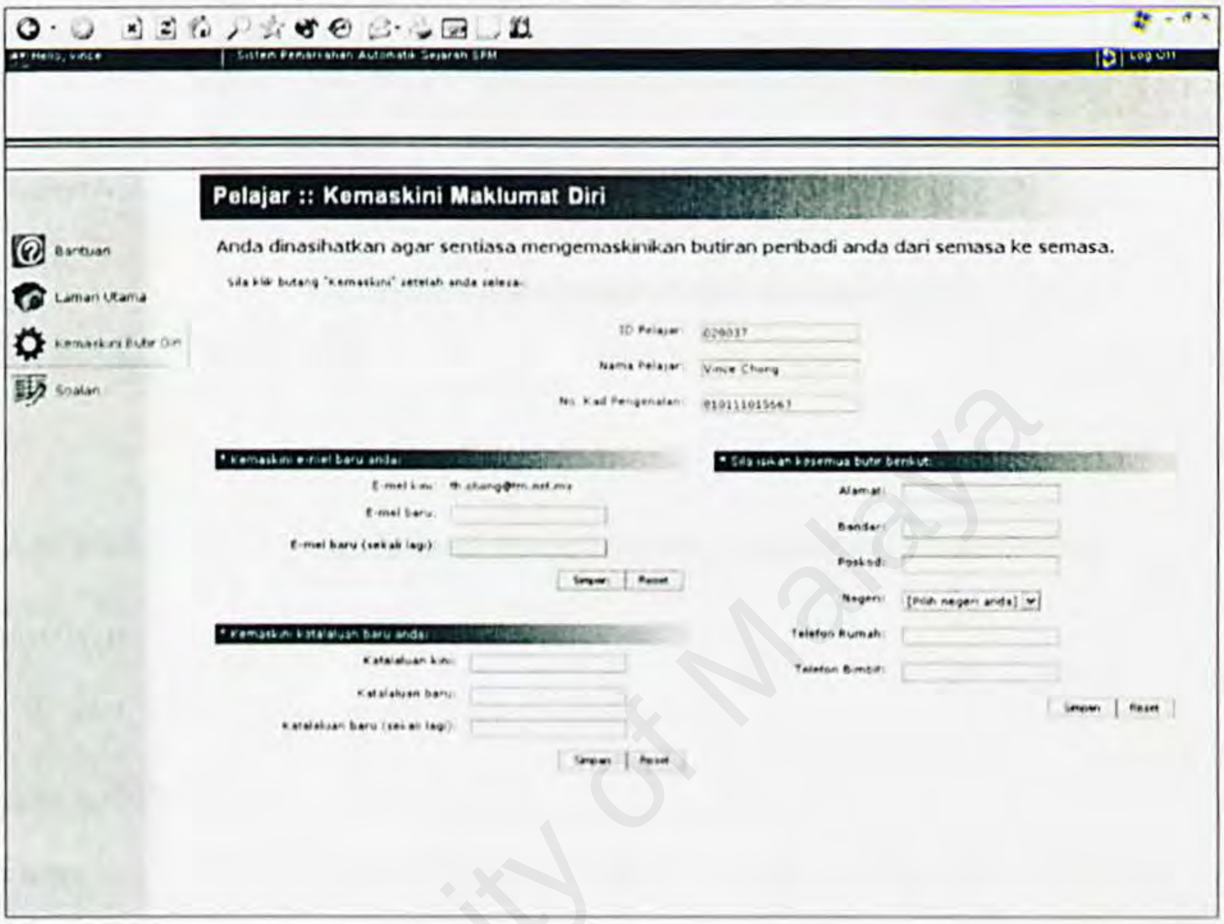
Jika pengguna hendak balik ke skrin Login, pengguna dikehendaki menekan butang Login di sebelah kiri.

Pengguna :: Logoff

Pengguna diingati supaya menekan logoff di sebelah kanan atas setiap kali selepas selesai menggunakan sistem ini.

Pengguna :: Pelajar

Pengguna :: Pelajar :: Kemaskini Butir Diri



Pelajar :: Kemaskini Maklumat Diri

Anda dinasihatkan agar sentiasa mengemaskinikan butiran peribadi anda dari semasa ke semasa.

Sila klik butang "Kemaskini" setelah anda selesai.

ID Pelajar: 020037

Nama Pelajar: Vince Chong

No. Kad Pengenalan: 810111015567

* Kemaskini e-mel baru anda:

E-mel kini: th.chang@tm.net.my

E-mel baru:

E-mel baru (sekali lagi):

* Kemaskini katalaluan baru anda:

Katalaluan kini:

Katalaluan baru:

Katalaluan baru (sekali lagi):

* Sila isikan kesemua butir berikut:

Alamat:

Bandar:

Poskod:

Negeri: [Pilih negeri anda]

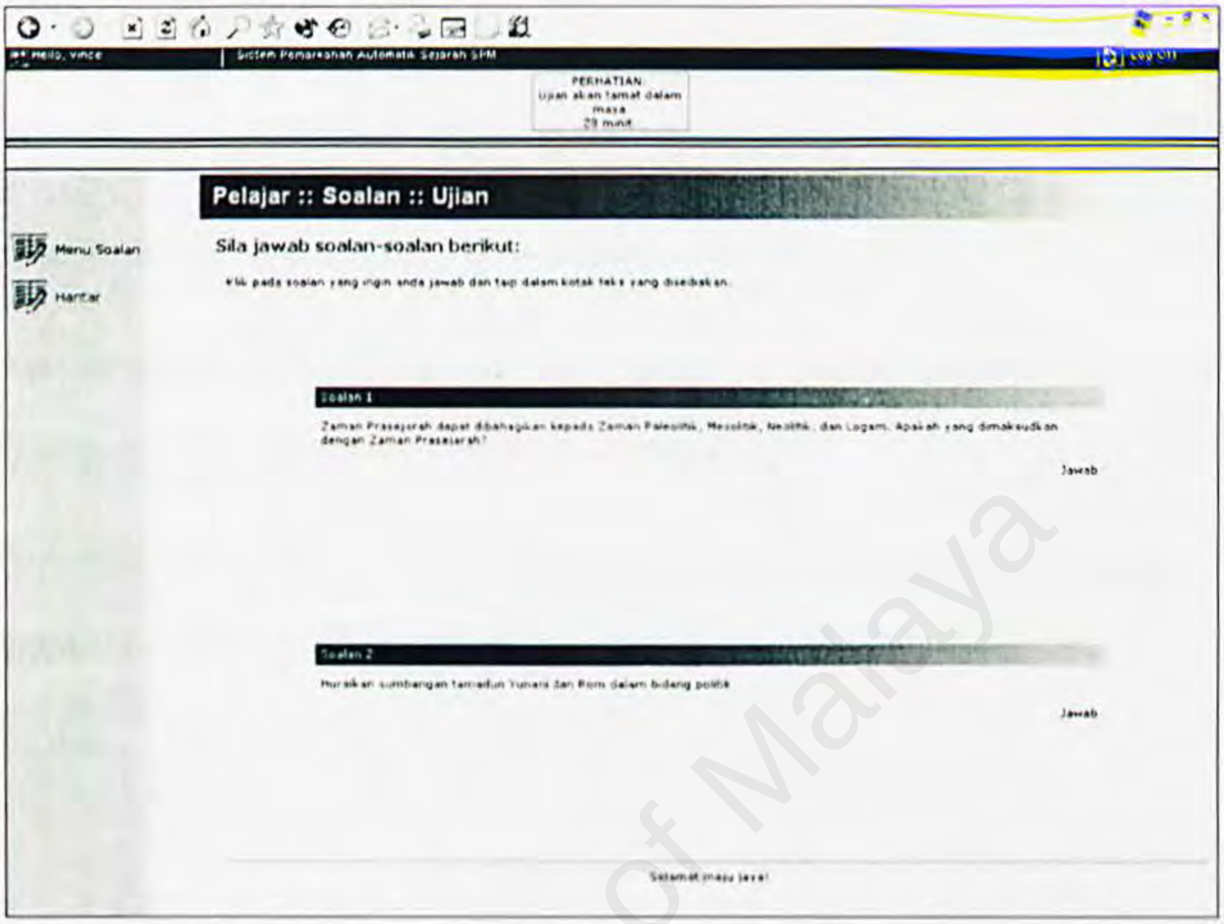
Telefon Rumah:

Telefon Bimbit:

Pelajar boleh mengkemaskinikan butir-butir diri dengan fungsi ini. Butir-butir seperti Alamat, Bandar, Poskod, Negeri, Telefon Rumah, Telefon Bimbit, E-mel, dan katalaluan boleh diubah atau dikemaskinikan oleh Pengguna Pelajar

ID Pelajar, Nama Pelajar dan No. Kad Pengenalan tidak dapat ditukar. Hanya pengguna SysAdmin mempunyai hak untuk menukar butiran-butiran tersebut.

Pengguna :: Pelajar :: Soalan



Sictem Pemarkahan Automatik Sejarah SPM

PERHATIAN
Ujian akan tamat dalam masa
29 minit

Pelajar :: Soalan :: Ujian

Sila jawab soalan-soalan berikut:

Klik pada soalan yang ingin anda jawab dan taip dalam kotak teks yang disediakan.

Soalan 1

Zaman Prasejarah dapat dibahagikan kepada Zaman Paleolitik, Mesolitik, Neolitik, dan Logam. Apakah yang dimaksudkan dengan Zaman Prasejarah?

Soalan 2

Huraikan sumbangan tamadun Yunani dan Rom dalam bidang politik

Selamat maju jaya!

Fungsi ini adalah untuk membenarkan Pelajar membuat latihan atau menduduki sesuatu ujian. Selepas memilih fungsi ini, pengguna dikehendaki memilih sama ada pengguna hendak memasuki Latihan atau Ujian.

Selepas itu, arahan untuk ujian atau latihan akan diberikan kepada pelajar.

Pelajar dikehendaki menekan butang 'Mula'selepas arahan dibaca.

Bergantung kepada ujian atau latihan yang disediakan oleh guru, pelajar boleh mimilih soalan yang hendak dijawab dengan menekan pautan Jawab

Selepas masuk ke dalam skrin menjawab soalan, butang 'Simpan' berfungsi untuk menyimpan jawapan pelajar. Pelajar boleh menekan butang 'Simpan' bila-bila masa. Selepas jawapan disimpan, pelajar boleh teruskan dengan soalan yang lain. Pelajar juga boleh mengembali ke jawapan yang telah disimpan untuk penyemakan dan membuat perubahan.

AMARAN : Tekan butang 'Simpan' tidak akan hantar jawapan untuk disemak.

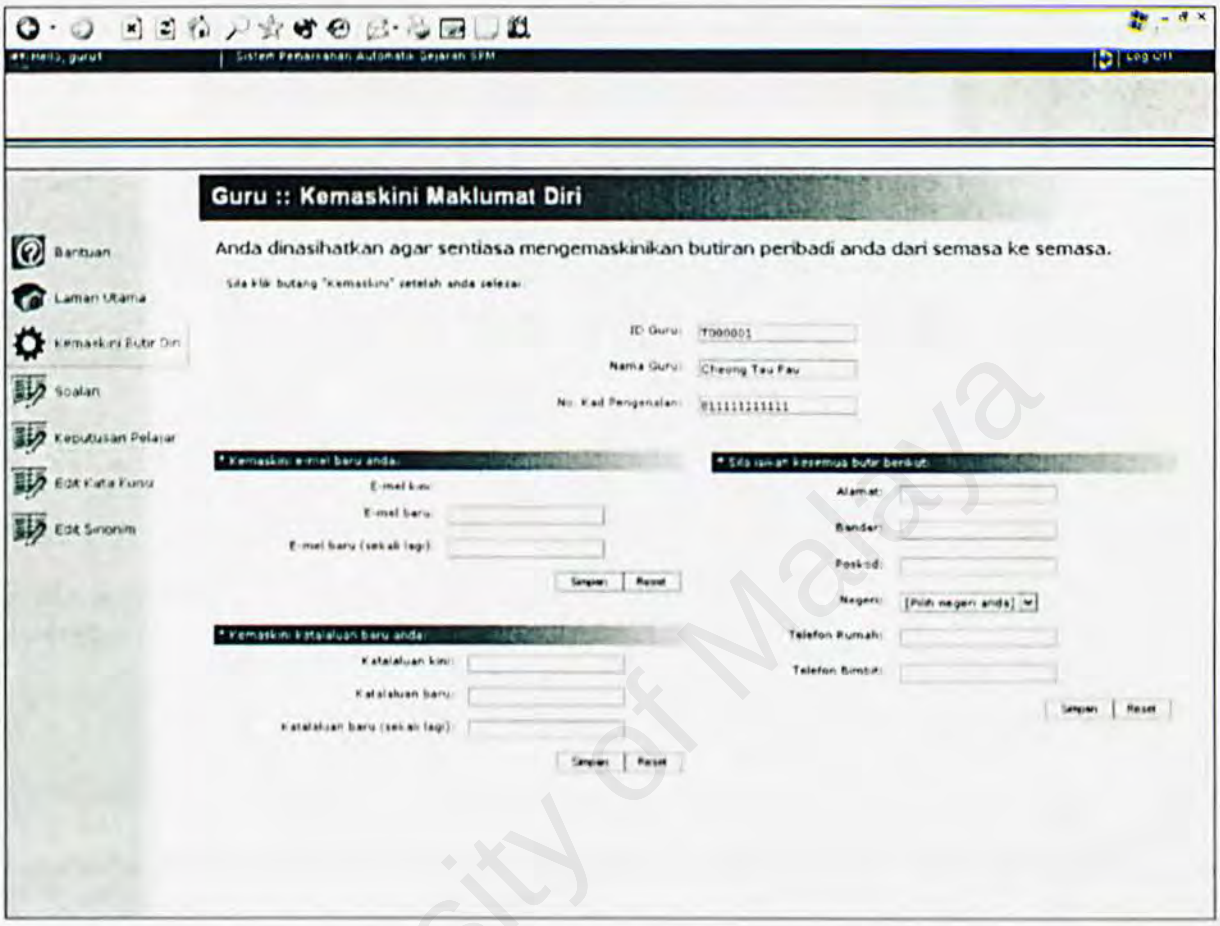
AMARAN : Jika butang 'Simpan' tidak ditekan, jawapan akan hilang.

Selepas pelajar menyiapkan semua soalan, pelajar dikehendaki menekan butang 'Hantar' untuk menghantarkan jawapan untuk disemak.

Masa akan dikira dan masa yang tinggal akan dipaparkan di atas. Apabila masa tamat, pelajar akan dibawa ke skrin untuk hantar jawapan.

Pengguna :: Guru

Pengguna :: Guru :: Kemaskini Butir Diri



Guru :: Kemaskini Maklumat Diri

Anda dinasihatkan agar sentiasa mengemaskinikan butiran peribadi anda dari semasa ke semasa.

Sila klik butang "Kemaskini" setelah anda selesai

ID Guru: T000001
Nama Guru: Cheong Tau Fau
No. Kad Pengenalan: 811111111111

* Kemaskini e-mel baru anda:
E-mel kini:
E-mel baru:
E-mel baru (sekali lagi):

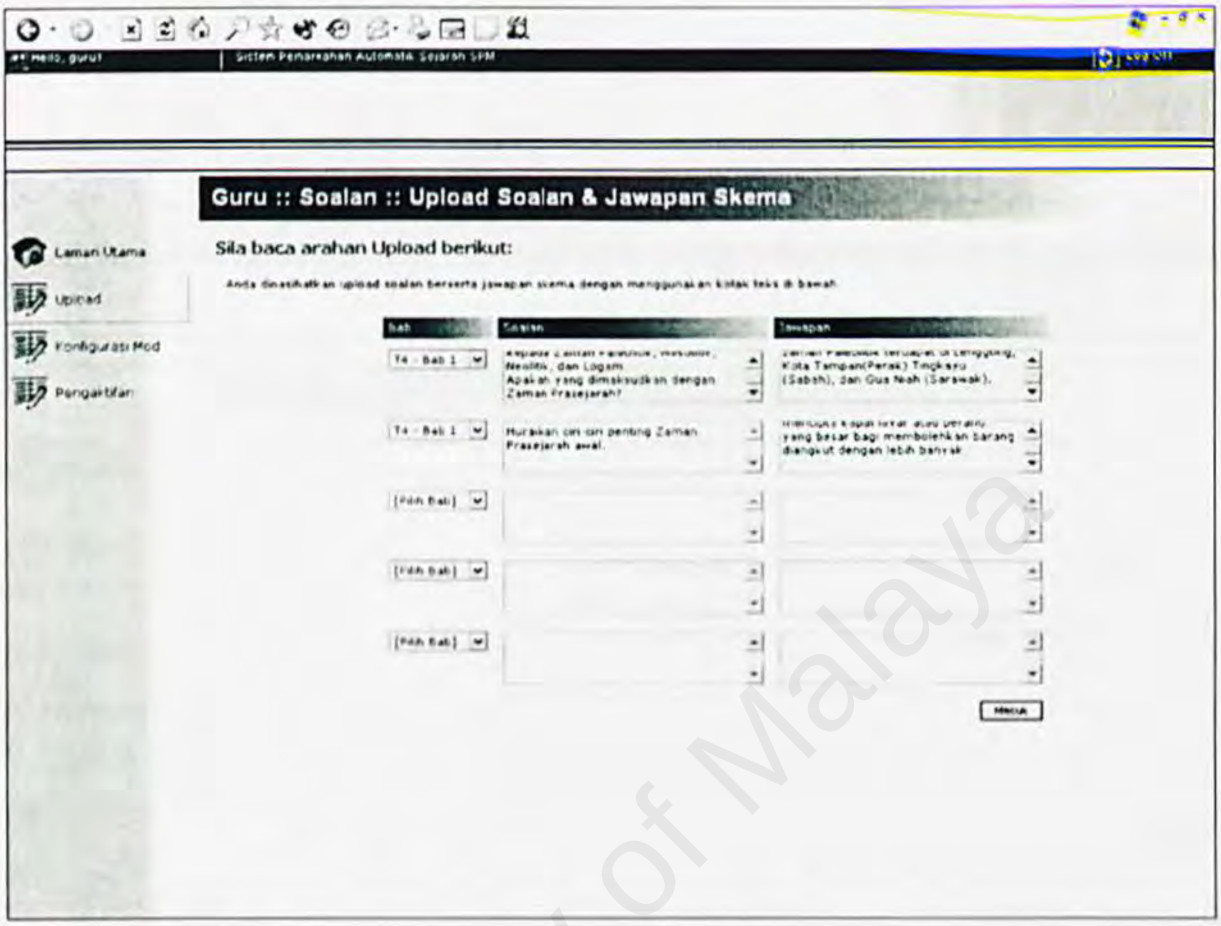
* Kemaskini katalaluan baru anda:
Katalaluan kini:
Katalaluan baru:
Katalaluan baru (sekali lagi):

* Sila isikan kesemua butir berikut:
Alamat:
Bandar:
Poskod:
Negeri: [Pilih negeri anda]
Telefon Rumah:
Telefon Bimbit:

Guru boleh mengkemaskinikan butir-butir diri dengan fungsi ini. Butir-butir seperti Alamat, Bandar, Poskod, Negeri, Telefon Rumah, Telefon Bimbit, E-mel, dan katalaluan boleh diubah atau dikemaskinikan oleh Pengguna Guru

ID Pelajar, Nama Pelajar dan No. Kad Pengenalan tidak dapat ditukar. Hanya pengguna SysAdmin mempunyai hak untuk menukar butiran-butiran tersebut.

Pengguna :: Guru :: Soalan :: Upload



Hello, guru1 | Sistem Pemarkahan Automatik Sejarah SPM

Log Off

Guru :: Soalan :: Upload Soalan & Jawapan Skema

Sila baca arahan Upload berikut:

Anda dinasihatkan upload soalan berserta jawapan skema dengan menggunakan kotak teks di bawah.

Bab	Soalan	Jawapan
T4 - Bab 1	Neolitik, dan Logam Apakah yang dimaksudkan dengan Zaman Prasejarah?	Kota Tampan(Perak) Tingkayu (Sabah), dan Gua Niah (Sarawak).
T4 - Bab 1	Huraikan ciri-ciri penting Zaman Prasejarah awal.	yang besar bagi membolehkan barang diangkut dengan lebih banyak
[Pilih Bab]		
[Pilih Bab]		
[Pilih Bab]		

Masuk

Fungsi ini adalah untuk memasukkan soalan baru ke dalam sistem. Guru dikehendaki memasukkan pasangan soalan dengan jawapannya.

Guru harus memilih bab soalan yang hendak dimasukkan.

Guru boleh menaip soalan dan jawapan di ruang yang disediakan.

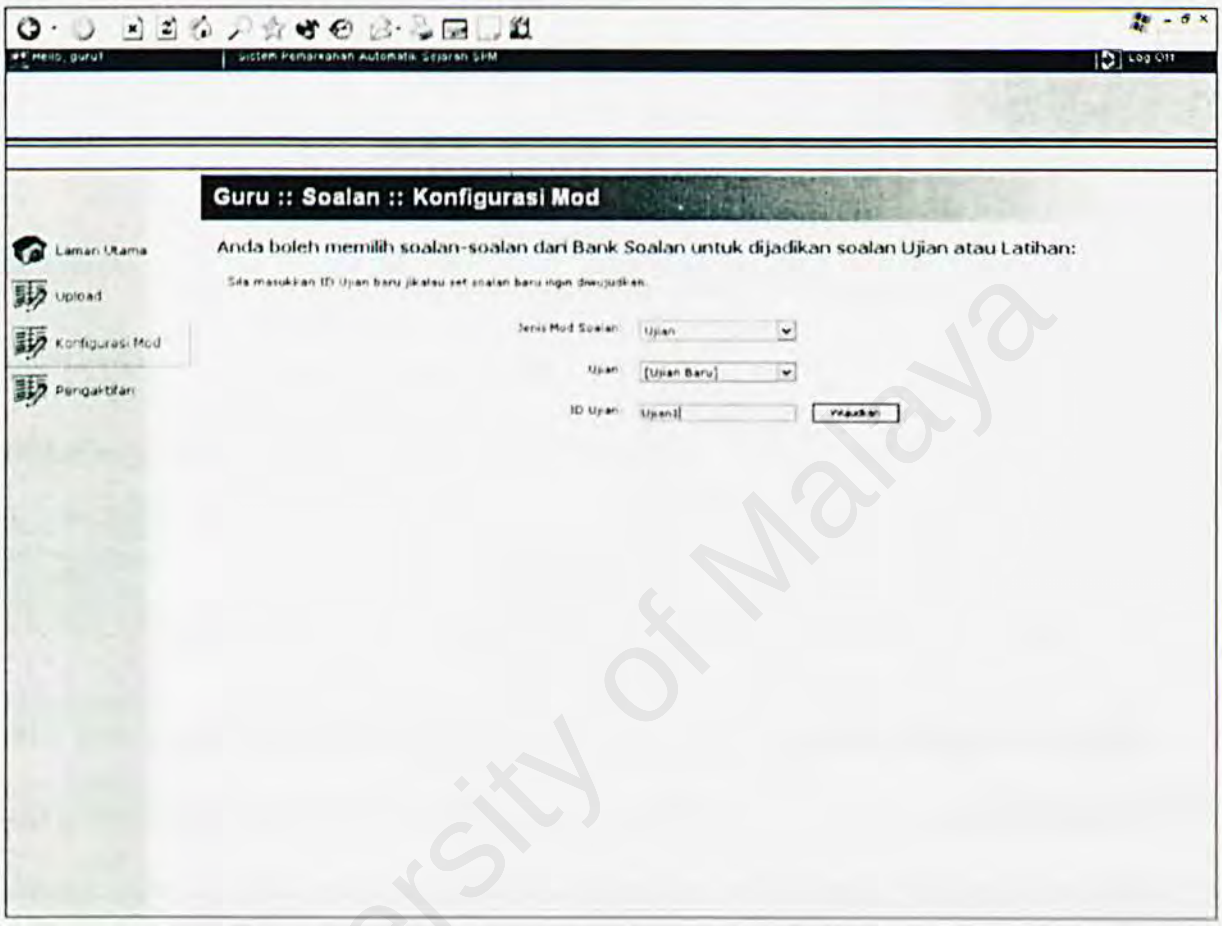
Soalan tidak akan disimpan jika ruangan jawapan adalah kosong.

Halaman ini membekalkan lima tempat ruangan, iaitu satu hingga lima soalan dan jawapannya boleh disimpan.

Selepas guru bersedia untuk memasukkan soalan dan jawapan ke dalam sistem, guru dikehendaki menekan butang "Masuk"

Sistem akan memberi amaran dan proses memasukkan soalan akan gagal jika soalan yang cuba dimasukkan itu telah wujud dalam sistem.

Pengguna :: Guru :: Soalan :: Konfigurasi Mod



Guru :: Soalan :: Konfigurasi Mod

Anda boleh memilih soalan-soalan dari Bank Soalan untuk dijadikan soalan Ujian atau Latihan:

Sila masukkan ID Ujian baru jikalau set soalan baru ingin diwujudkan.

Jenis Mod Soalan: Ujian

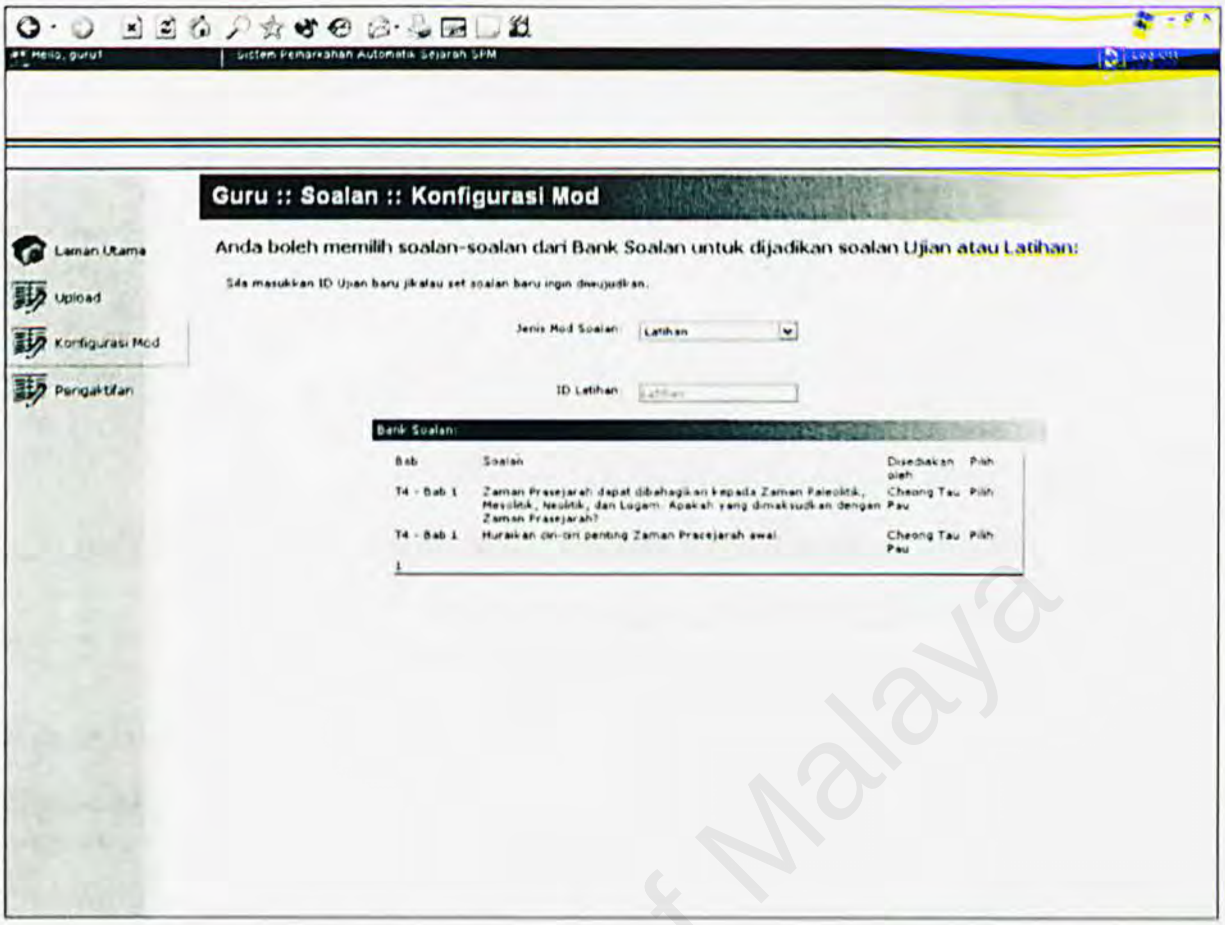
Ujian: [Ujian Baru]

ID Ujian: Ujian1

Fungsi ini adalah untuk memilih soalan untuk ujian dan latihan

Pilihan mod adalah untuk memilih sama ada menyediakan soalan untuk latihan atau menyediakan soalan ujian.

Jika guru ingin menyediakan latihan, pilih pilihan latihan. Selepas itu, guru boleh memilih sebanyak lima soalan daripada bank soalan yang telah dimasukkan ke dalam sistem.



Guru :: Soalan :: Konfigurasi Mod

Anda boleh memilih soalan-soalan dari Bank Soalan untuk dijadikan soalan Ujian atau Latihan:

Sila masukkan ID Ujian baru jikalau set soalan baru ingin diwujudkan.

Jenis Mod Soalan: Latihan

ID Latihan: Latihan

Bank Soalan:

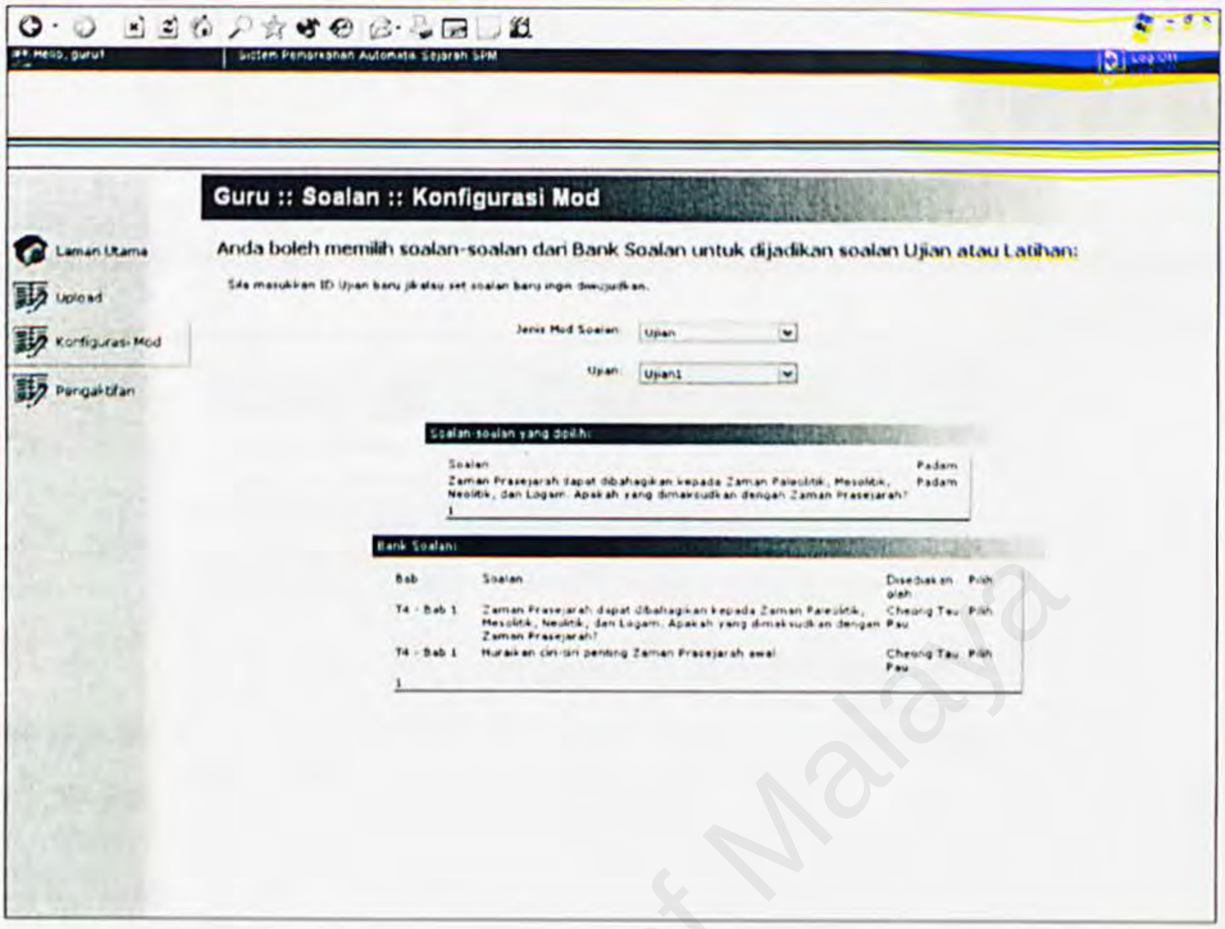
Bab	Soalan	Disediakan oleh	Pilih
T4 - Bab 1	Zaman Prasejarah dapat dibahagikan kepada Zaman Paleolitik, Mesolitik, Neolitik, dan Logam. Apakah yang dimaksudkan dengan Zaman Prasejarah?	Cheong Tau Pau	Pilih
T4 - Bab 1	Huraikan ciri-ciri penting Zaman Prasejarah awal	Cheong Tau Pau	Pilih

1

Jika guru ingin menyediakan ujian, pilihan ujian harus dipilih. Untuk menyediakan suatu ujian baru, guru boleh menaip nama ujian baru di dalam petak yang disediakan (Nama ujian ini tidak boleh menganduni aksara tempat kosong). Selepas nama ujian ditaip, butang "Wujudkan" harus ditekan. Selepas itu, nama ujian baru ini akan wujud dalam senarai dan dapat dipilih.

Untuk ujian yang telah wujud dalam senarai, guru boleh memilih ujian yang hendak disediakan. Selepas itu, guru boleh memilih sebanyak lima soalan daripada bank soalan yang telah dimasukkan ke dalam sistem.

Setiap soalan yang dipilih akan disimpan dengan otomatik.



Guru :: Soalan :: Konfigurasi Mod

Anda boleh memilih soalan-soalan dari Bank Soalan untuk dijadikan soalan Ujian atau Latihan:

Sila masukkan ID Ujian baru jikalau set soalan baru ingin diwujudkan.

Jenis Mod Soalan: Ujian

Ujian: Ujian1

Soalan-soalan yang dipilih:

Soalan	Padam
Zaman Prasejarah dapat dibahagikan kepada Zaman Paleolitik, Mesolitik, Neolitik, dan Logam. Apakah yang dimaksudkan dengan Zaman Prasejarah?	Padam

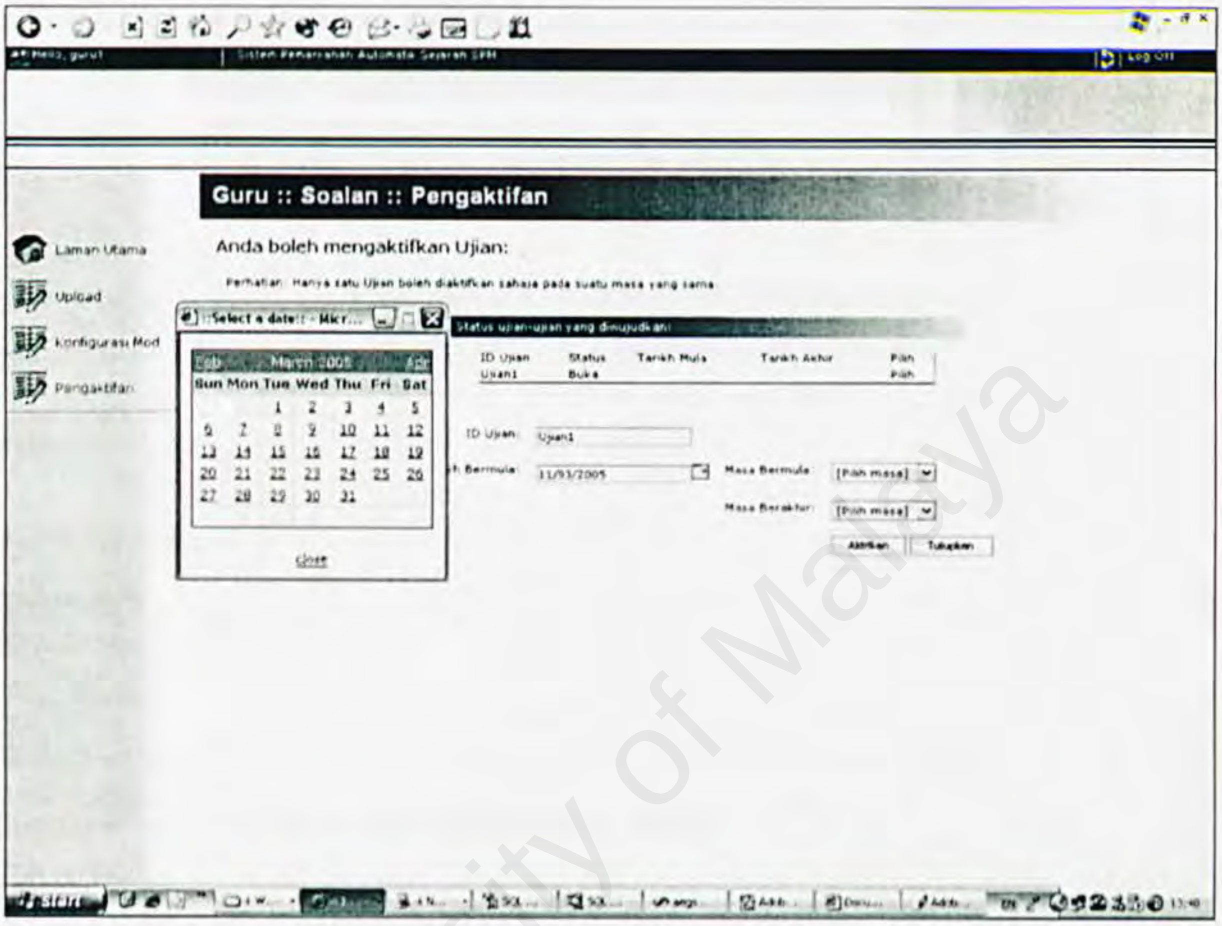
1

Bank Soalan:

Bab	Soalan	Disediakan oleh	Pilih
T4 - Bab 1	Zaman Prasejarah dapat dibahagikan kepada Zaman Paleolitik, Mesolitik, Neolitik, dan Logam. Apakah yang dimaksudkan dengan Zaman Prasejarah?	Cheong Tau Pau	Pilih
T4 - Bab 1	Huraikan ciri-ciri penting Zaman Prasejarah awal	Cheong Tau Pau	Pilih

1

Pengguna :: Guru :: Soalan :: Pengaktifan



Guru :: Soalan :: Pengaktifan

Anda boleh mengaktifkan Ujian:

Perhatian: Hanya satu Ujian boleh diaktifkan sahaja pada suatu masa yang sama

Status ujian-ujian yang diwujudkan:

ID Ujian	Status	Tarikh Mula	Tarikh Akhir	Pilih
Ujian1	Buka			Pilih

ID Ujian: Ujian1

...h Bermula: 11/03/2005

Masa Bermula: [Pilih masa]

Masa Berakhir: [Pilih masa]

Feb			March 2005			Apr
Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

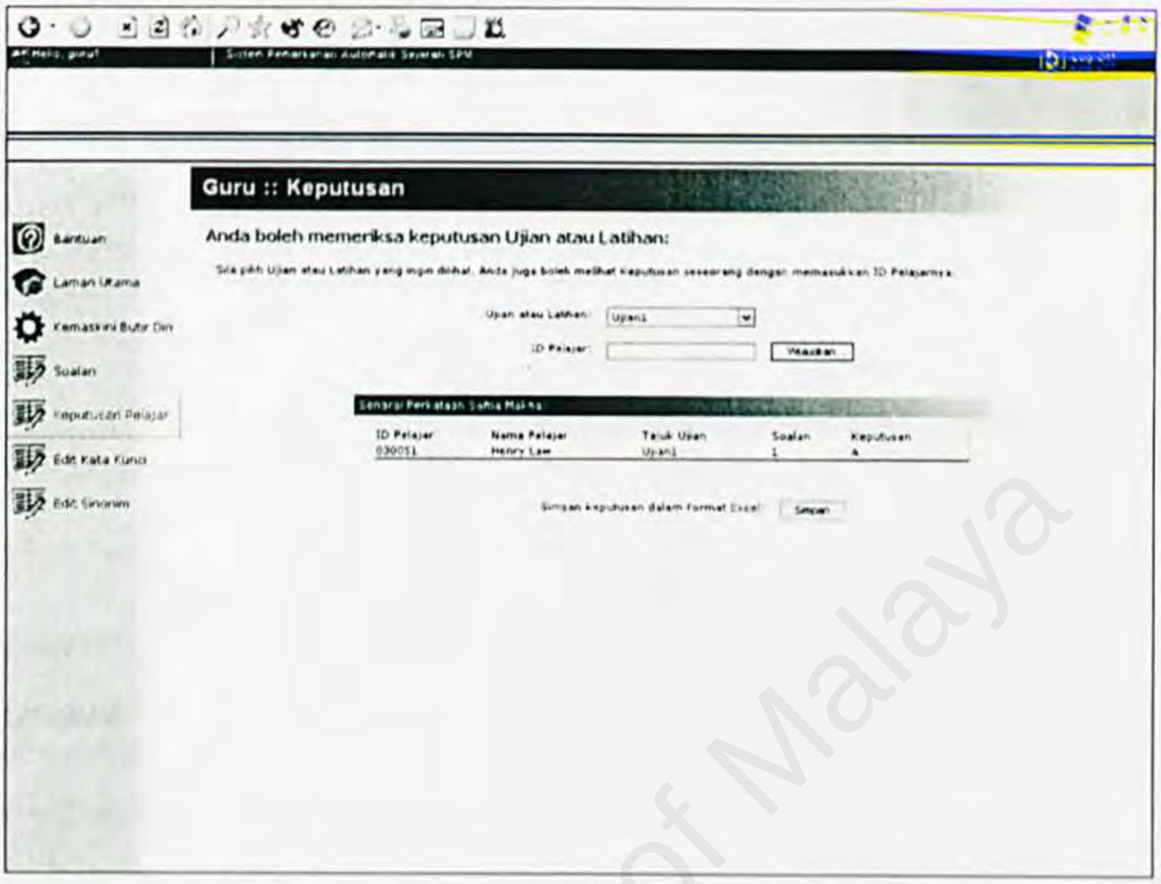
Fungsi ini adalah untuk mengaktifkan ujian yang telah disediakan

Semua ujian yang telah disediakan akan ditunjukkan di dalam jadual. Untuk mengaktifkan sesuatu ujian, guru dikehendaki klik pada perkataan pilih pada baris ujian yang hendak diaktifkan.

Selepas itu, guru boleh klik pada kelender yang disediakan untuk memilih tarikh ujian. Seterusnya, masa mula dan masa tamat peperiksaan juga boleh dipilih dan ditentukan.

Selepas langkah-langkah tersebut, guru dikehendaki klik butang "Aktifkan" untuk mengaktifkan ujian. Butang "Tutupkan" akan menamatkan ujian itu.

Pengguna :: Guru :: Keputusan Pelajar



Hello, guru!

Sistem Pemarkahan Automatik Sejarah SPM

Guru :: Keputusan

Anda boleh memeriksa keputusan Ujian atau Latihan:

Sila pilih Ujian atau Latihan yang ingin dilihat. Anda juga boleh melihat Keputusan seseorang dengan memasukkan ID Pelajarnya.

Ujian atau Latihan: Ujian1

ID Pelajar:

Senarai Perkataan Sama Makna

ID Pelajar	Nama Pelajar	Tajuk Ujian	Soalan	Keputusan
030051	Henry Law	Ujian1	1	A

Simpan keputusan dalam format Excel:

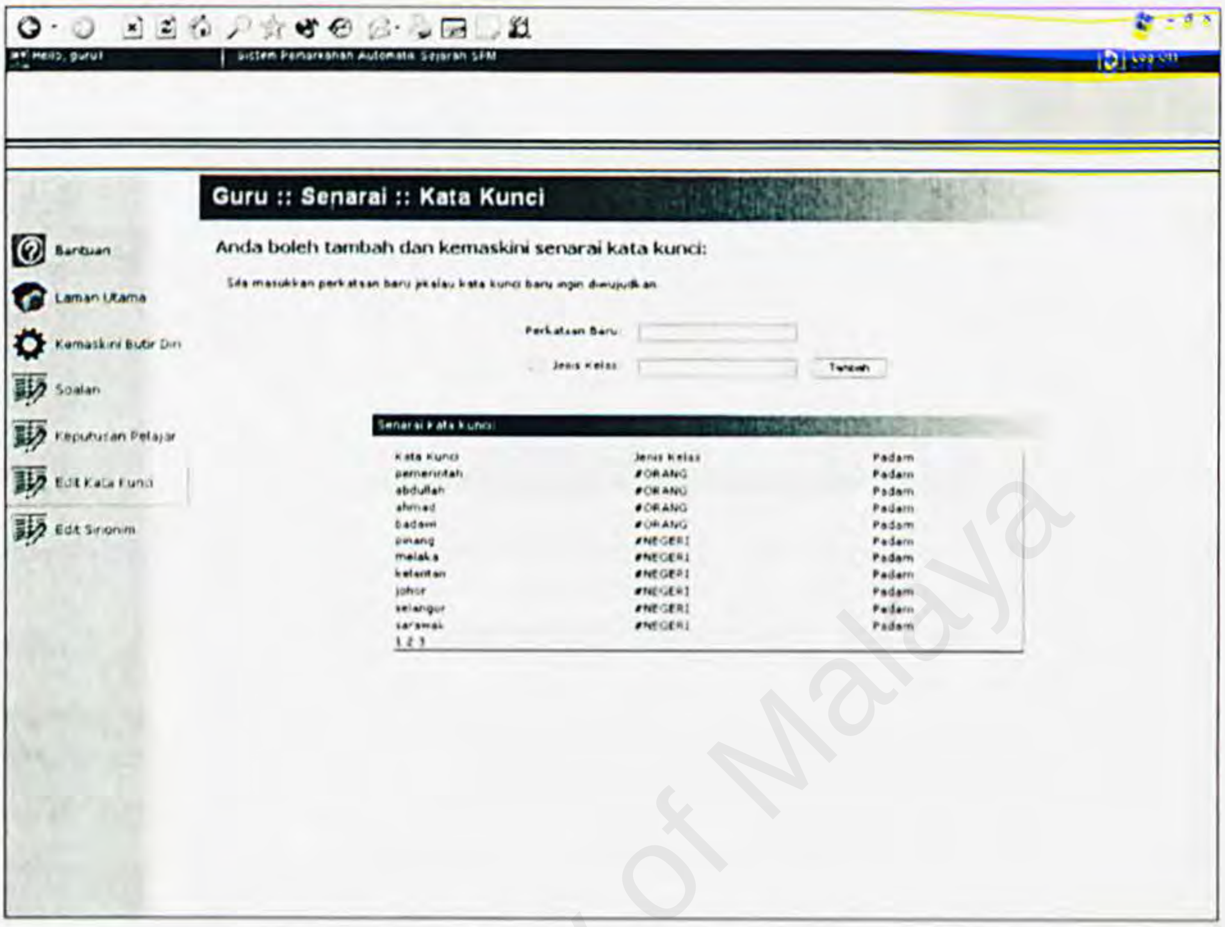
Fungsi ini adalah untuk menyemak keputusan pelajar

Apabila guru memasuki halaman ini, semua keputusan pelajar yang diajar akan ditunjukkan.

Guru boleh memilih untuk melihat keputusan pelajar bagi ujian yang tertentu sahaja dengan memilih ujian pada pilihan ujian.

Guru juga boleh melihat semua keputusan seseorang pelajar dengan menaip ID pelajar tersebut dalam petak yang disediakan dan menekan butang "Wujudkan"

Pengguna :: Guru :: Edit Kata Kunci



Guru :: Senarai :: Kata Kunci

Anda boleh tambah dan kemaskini senarai kata kunci:

Sila masukkan perkataan baru jikalau kata kunci baru ingin diwujudkan

Perkataan Baru:

Jenis Kelas:

Senarai Kata Kunci

Kata Kunci	Jenis Kelas	Padam
pemerintah	#ORANG	Padam
abdullah	#ORANG	Padam
ahmad	#ORANG	Padam
badawi	#ORANG	Padam
pinang	#NEGERI	Padam
melaka	#NEGERI	Padam
kelantan	#NEGERI	Padam
johor	#NEGERI	Padam
selangor	#NEGERI	Padam
sarawak	#NEGERI	Padam

1 2 3

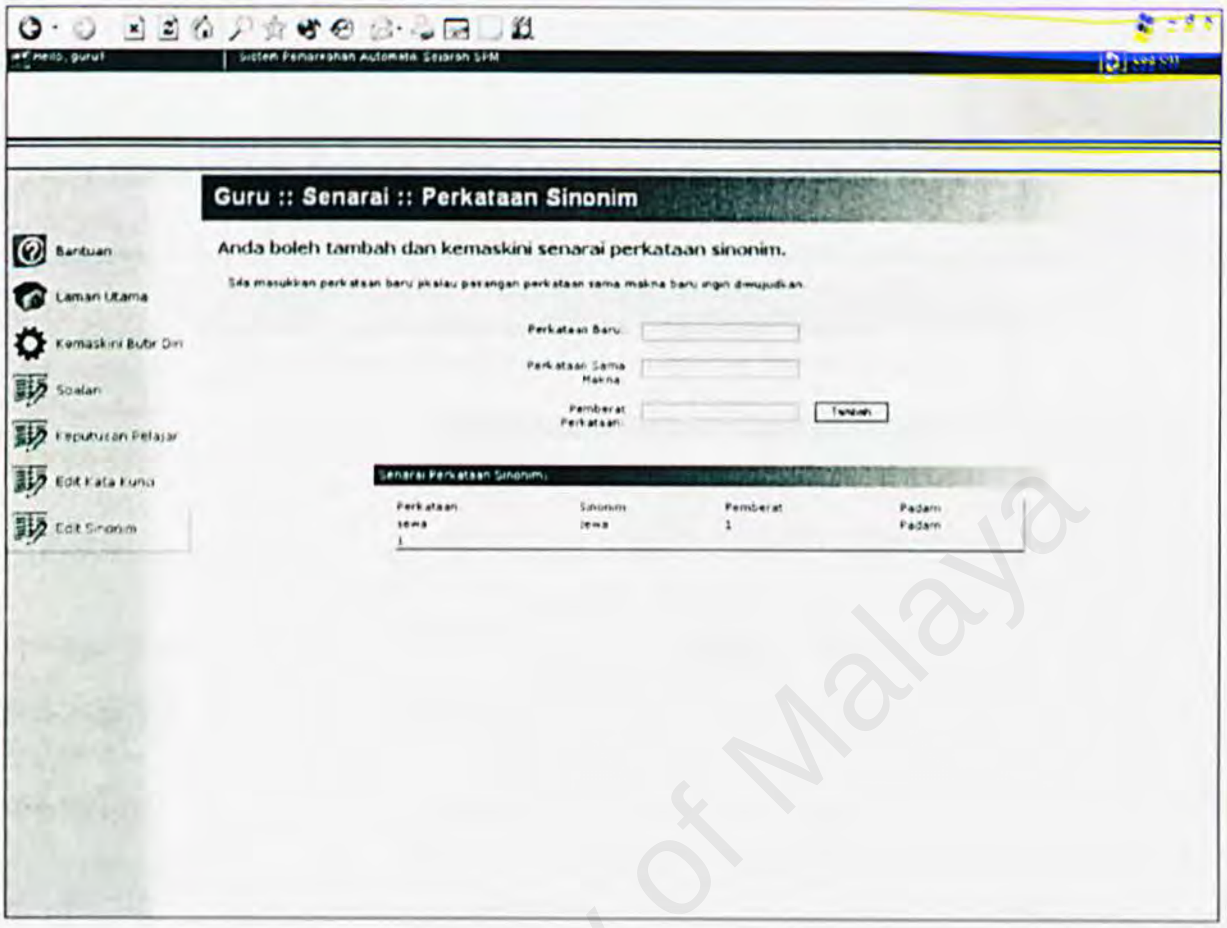
Fungsi ini adalah untuk menambah kata kunci

Guru dikehendaki menaip kata kunci yang ingin ditambah di petak yang disediakan. Selepas kata kunci ditaip, butang 'tambah' hendaklah ditekan untuk menambah kata kunci yang telah ditaip.

Kata kunci yang telah ditambah akan ditunjukkan dalam jadual. Jika guru ingin memadam sesuatu kata kunci, guru boleh klik perkataan padam pada baris kata kunci yang hendak dipadam.

Jenis kelas adalah pilihan. Jika perkataan tidak dimasukkan dalam ruang ini, kata kunci baru akan ditambah dengan bahagian jenis kelas sebagai "#UNKNOWN"

Pengguna :: Guru :: Edit Sinonim



Guru :: Senarai :: Perkataan Sinonim

Anda boleh tambah dan kemaskini senarai perkataan sinonim.

Sila masukkan perkataan baru jikalau pasangan perkataan sama makna baru ingin diwujudkan.

Perkataan Baru:

Perkataan Sama Makna:

Pemberat Perkataan:

Senarai Perkataan Sinonim:

Perkataan	Sinonim	Pemberat	Padam
sewa	jewa	1	Padam
1			

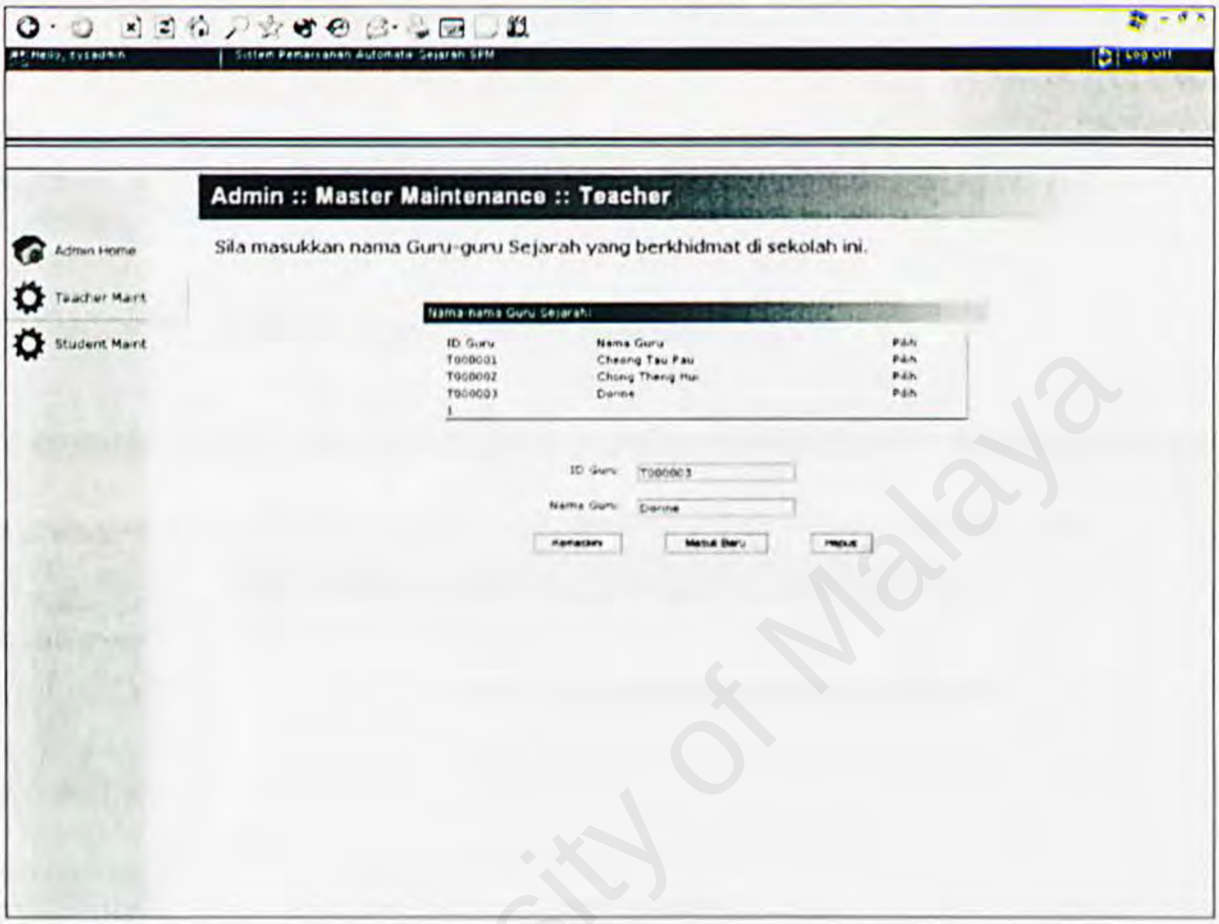
Fungsi ini adalah untuk menambah perkataan sinonim

Guru dikehendaki menaip pasangan kata asal dan sinonim yang ingin ditambah di petak yang disediakan. Selepas pasangan kata asal dan sinonim ditaip, butang 'tambah' hendaklah ditekan untuk menambah pasangan kata asal dan sinonim yang telah ditaip.

Sinonim yang telah ditambah akan ditunjukkan dalam jadual. Jika guru ingin memadam sesuatu sinonim, guru boleh klik perkataan padam pada baris kata kunci yang hendak dipadam.

Pengguna :: System Admin

Pengguna :: System Admin :: Teacher Maintenance



Admin :: Master Maintenance :: Teacher

Sila masukkan nama Guru-guru Sejarah yang berkhidmat di sekolah ini.

Nama-nama Guru Sejarah:

ID Guru	Nama Guru	Pilih
T000001	Cheong Tau Fau	Pilih
T000002	Chong Theng Hui	Pilih
T000003	Denne	Pilih
1		

ID Guru: T000003

Nama Guru: Denne

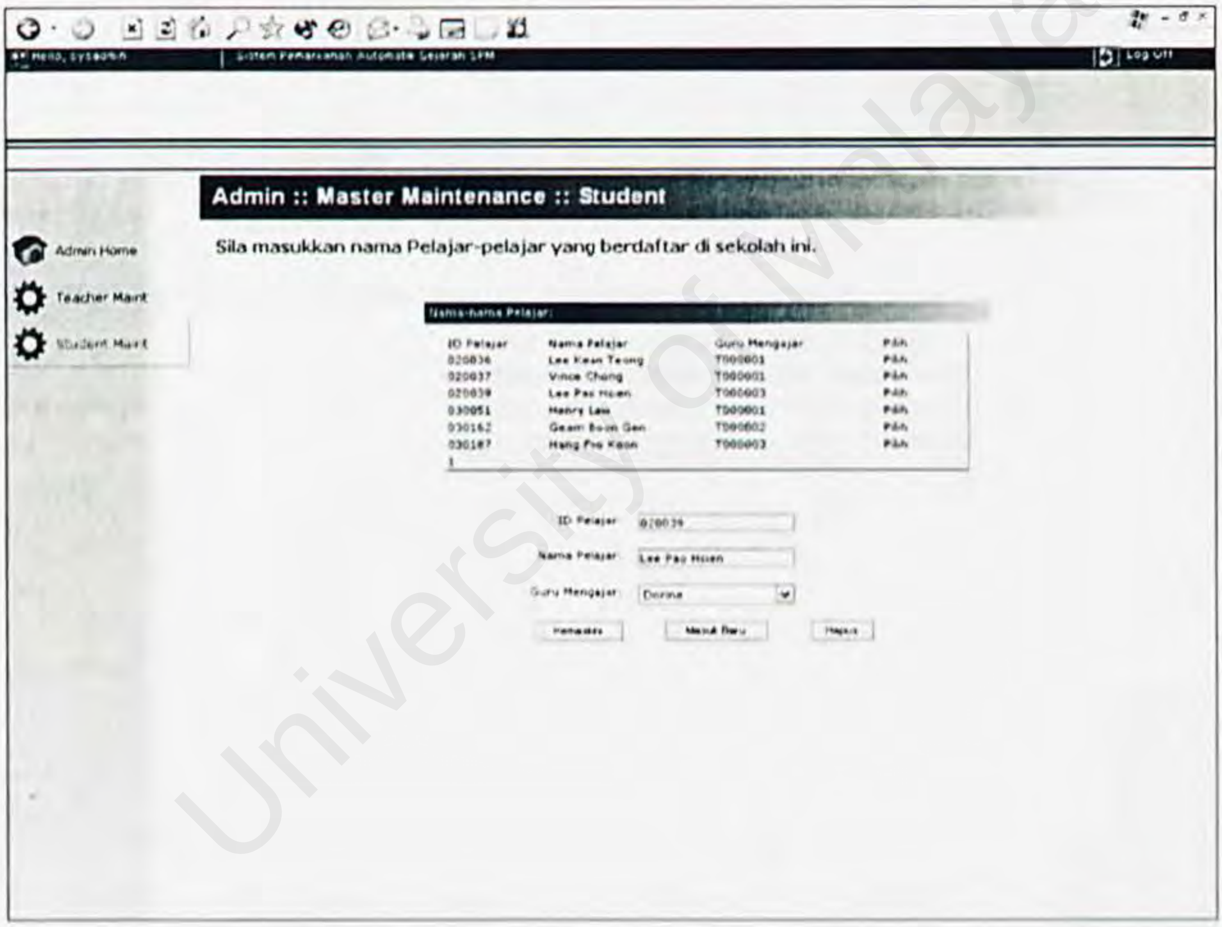
Fungsi ini adalah untuk menambah ID dan Nama Pengguna supaya pengguna itu dapat mendaftar dan menggunakan sistem ini. Pengguna yang dimasukkan ke dalam jadual ini boleh mendaftar sebagai pengguna guru sahaja.

Jadual akan memaparkan semua data yang telah dimasukkan ke sistem. Untuk mengkemaskini data yang telah wujud, pengguna boleh klik perkataan pilih pada baris data yang ingin dikemaskinikan. Selepas itu, data yang telah wujud akan dipaparkan di dalam petak yang disediakan. Selepas mengubah data itu, pengguna harus menekan butang "Kemaskini" untuk menyimpan data yang telah diubah. Jika butang "Masuk Baru" ditekan, baris yang baru akan ditambah dan data yang wujud itu tidak akan diubah. Ubahan tidak dapat dibuat terhadap ID. Jika ID diubah, data yang baru hendaklah disimpan.

Untuk memasukkan data baru, pengguna boleh mengisikan petak Nama dan ID seterusnya menekan butang "Masuk Baru".

Butang "Padam" adalah untuk memadam sesuatu data daripada sistem.

Pengguna :: System Admin :: Student Maintenance



Admin :: Master Maintenance :: Student

Sila masukkan nama Pelajar-pelajar yang berdaftar di sekolah ini.

Nama-nama Pelajar:

ID Pelajar	Nama Pelajar	Guru Mengajar	Pilih
020036	Lee Kean Teong	T000001	Pilih
020037	Vince Chong	T000001	Pilih
020039	Lee Pai Hoen	T000003	Pilih
030051	Henry Lew	T000001	Pilih
030162	Geam Boon Gen	T000002	Pilih
030187	Hang Foo Keon	T000003	Pilih
1			

ID Pelajar: 020039
Nama Pelajar: Lee Pai Hoen
Guru Mengajar: Dorina

Fungsi ini adalah untuk menambah ID dan Nama Pengguna supaya pengguna itu dapat mendaftar dan menggunakan sistem ini. Pengguna yang dimasukkan ke dalam jadual ini boleh mendaftar sebagai pengguna pelajar sahaja.

Fungsi ini adalah untuk menambah ID dan Nama Pengguna supaya pengguna itu dapat mendaftar dan menggunakan sistem ini.

Jadual akan memaparkan semua data yang telah dimasukkan ke sistem. Untuk mengkemaskini data yang telah wujud, pengguna boleh klik perkataan pilih pada baris data yang ingin dikemaskinikan. Selepas itu, data yang telah wujud akan dipaparkan di dalam petak yang disediakan. Selepas mengubah data itu, pengguna harus menekan butang "Kemaskini" untuk menyimpan data yang telah diubah. Jika butang "Masuk Baru" ditekan, baris yang baru akan ditambah dan data yang wujud itu tidak akan diubah. Ubahan tidak dapat dibuat terhadap ID. Jika ID diubah, data yang baru hendaklah disimpan.

Untuk memasukkan data baru, pengguna boleh mengisikan petak Nama, ID dan memilih guru yang mengajar seterusnya menekan butang "Masuk Baru".

Butang "Padam" adalah untuk memadam sesuatu data daripada sistem.

APPENDIX B

Answer Scheme used to test the system.

Zaman Prasejarah merupakan zaman silam di mana manusia pada masa itu tidak mempunyai sistem tulisan atau sebarang catatan untuk merekodkan segala kegiatan kehidupan mereka.

Zaman Prasejarah adalah berbeza antara satu kawasan dengan kawasan yang lain. Walaupun Zaman Prasejarah tidak meninggalkan sebarang rekod bertulis atau catatan, tetapi kewujudan manusia pada zaman tersebut dapat diketahui melalui kajian arkeologi dengan menggunakan kaedah cari gali ke atas artifak, fosil, feature, dan ekofak yang ditemui.

Ciri penting untuk mengenali Zaman Prasejarah ialah:
Manusia hidup secara berpindah-randah dari satu tempat ke tempat lain untuk mencari kawasan baru bagi memulakan penghidupan. Kegiatan ekonomi mereka ialah memungut hasil hutan, berburu binatang, dan menangkap ikan.

Tempat tinggal mereka di tepi sungai, di dalam gua, dan di tepi tasik secara berkumpulan. Alatan batu yang dicipta kasar buatannya dan lebih mementingkan fungsi daripada nilai estetika.

Pakaian mereka dibuat daripada kulit binatang, daun kayu, dan ranting kayu. Alatan batu yang dibuat digunakan untuk memotong, mengikis, dan sebagai senjata. Satu alatan batu yang penting ialah kapak tangan yang digunakan sebagai pisau, kapak, dan geraji. Di Malaysia, petempatan awal manusia zaman Paleolitik terdapat di Lenggong, Kota Tampan(Perak) Tingkayu(Sabah), dan Gua Niah (Sarawak).

Kepentingan zaman Neolitik kepada peradaban manusia dapat dilihat dengan jelas dalam aspek berikut:
Asas kepada petempatan kehidupan kekal dan tetap dan tidak lagi berpindah-randah. Perkembangan aktiviti pertanian yang lebih giat dan mula menternak binatang.

Petempatan kekal berkembang sehingga mewujudkan bandar atau kota besar seperti Catal Huyuk (Turki), Jarmo (Iraq), dan Jericho (Palestine). Ia merangsangkepada kehidupan yang lebih selesa dan sejahtera. Perkembangan aktiviti pertanian merangsang pemikiran manusia untuk mencipta alat pertanian seperti pisau, sabit dan bajak bagi meringankan aktiviti pekerjaan mereka, di samping mencipta tempat menyimpan makanan, bijirin, air dan sebagainya.

Aktiviti pertanian dan penternakan dijalankan secara besar-besaran membolehkan berlakunya lebihan yang menjadi pendorong kepada kegiatan tukar barang (sistem barter) dan menjadi asas kepada hubungan perdagangan dengan kawasan yang lebih jauh. Kewujudan hubungan perdagangan merangsang pemikiran manusia untuk mencipta kapal layar atau perahu yang besar bagi membolehkan barang diangkut dengan lebih banyak.

Aktiviti perdagangan memerlukan satu kaedah yang baik dalam urusan jual beli. Oleh itu, manusia mula mencipta sistem mata wang yang paling awal, terdiri daripada siput sebelum wujudnya mata wang perak dan emas.

Ekoran perkembangan penduduk, ini menimbulkan keperluan bagi mengawal masyarakat yang semakin kompleks. Keadaan ini mewujudkan satu sistem undang-undang atau peraturan bagi mengawal masyarakat agar kestabilan masyarakat dapat dikawal.

Undang-undang dan peraturan tersebut merupakan satu kawalan sosial yang perlu dipatuhi. Justeru, muncul ketua yang memastikan undang-undang tersebut dipatuhi. keadaan ini mewujudkan golongan yang memerintah dan golongan yang diperintah.

Bagi menampung pertambahan penduduk yang semakin pesat, ekonomi perlu dibangunkan dengan meningkatkan hasil pengeluaran makanan. Ini melahirkan pula pengkhususan pekerjaan dan mewujudkan organisasi sosial atau susun lapis dalam masyarakat. Setiap individu dalam susun lapis masyarakat mempunyai kedudukan tertentu dan memainkan peranan masing-masing. Misalnya menjadi raja, pembesar, artisan, petani, pendeta, tentera, dan hamba.

References

[1] N. Idris. (2003). *Automated Grading Essay for Malay Text using information retrieval and nearest neighbor algorithm.*

[2] X. Li. (2000). *Syntactic Features in Question Answering.*

[3] Darus, S., S. H. Stapa. (2001). *Lecturer's Expectations of a Computer-Based Essay Marking Systems.* The English Teacher. Journal of the Malaysian English Language Teachers' Association (MELTA). Vol. XXX. June. Pp. 47-56.

[4] S. Y. Tai, C. S. Ong, N. A. Abdullah. (1990). *On Designing an Automated Malaysian Stemmer for the Malay Language.* Proceedings of the 5th International Workshop Information Retrieval with Asian Languages.

[5] J. Palmer, R. Williams, H. Dreher. (2002). *Automated Essay Grading System Applied to a First Year University Subject – How Can We do it Better ?* Proceedings of the Informing Science and IT Education (InSITE) Conference, Cork, Ireland, 1221-1229.

[6] R. Williams. (2001). *Automated essay grading: An evaluation of four conceptual models.* Proceedings of the 10th Annual Teaching Learning Forum, 7-9 February 2001. Perth: Curtin University of Technology.

[7] T. Mitchell, N. Aldridge, P. Broomhead. *Computerised Marking of Short-Answer Free-Text Responses.*

[8] M.-F. Moens, J. Dumortier. (2000). *Automatic categorization of magazine articles.*

[9] S. valenti, F. neri and A. Cucchiarelli. (2003). *An Overview of current research on automated essay grading.* Journal of Information Technology Education. Volume 2, 2003.

[10] M. J. Strait, J. A. Haynes. *Application of Latent Semantic Analysis to Lessons Learned Systems.*

[11] B. Rehder, M.E. Schreiner, M. B. W. Wolfe, D. Laham, T. K. Landauer, W. Kintsch *Latent Semantic Analysis and Knowledge Assessment.*

[12] D. Laham. *Latent Semantic Analysis Approaches to Categorization.*

[13] T. K. Landauer. (2003). *Applications of Latent Semantic Analysis.*

[14] J. Quesada, W. Kintsch, E. Gomez. (2001). *A Computational Theory of Complex Problem Solving Using the Vector Space Model.*

[15] B. Lemaire, G. Denhiere (2003). *Cognitive models based on Latent Semantic Analysis.* ICCM'2003.

[16] S. R. Hota. *Content Analysis of Text using Nearest Neighbor.*

[17] D. Haley, P. Thomas, B. Nuseibeh, J. Taylor, P. Lefrere. (2004). *E-Assessment using Latent Semantic Analysis.*

[18] L. Streeter, J. Psotka, D. Laham, D. MacCuish. (2002). *The Credible Grading Machine: Automated Essay Scoring in the DOD.*

[19] N. Evangelopoulos. (2003). *Intelligent Exam Grading: Text-Mining Free Text Answers.* Master Thesis. University of Kansas.

[20] D. Bramanis. *The Evolution of Automated Essay Grading.*

[21] T. Hoffman. (1999). *Probabilistic Latent Semantic Analysis.* Uncertainity in Artificial Intelligence, UAI'99, Stockholm.

[22]P. W. Hastings. (1999). *How Latent is Latent Semantic Analysis.*

[23] T. Doan. *Investigation on how to improve latent semantic analysis performance.*

[24]C. P.Rose, A. Roque, D. Bhembe, K. VanLehn. *A hybrid Approach to Content Analysis for Automatic Essay Grading.*

[25] S. C. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R. Harshman. (1990). *Indexing by Latent Semantic Analysis.* Journal of the American Society for Information Science, 41(6), 391-407.

[26] T. K. Landauer. (1998). *An Introduction to Latent Semantic Analysis.* Discourse Processes, 25.

[27]Marti A Hearst The *debate on automated essay grading.*

[28] T. K. Landauer, D. Laham. *Learning Human-Like Knowledge by Singular Value Decomposition: A Progress Report.*

[29]C. Gonzalez *Learning in Dynamic Decision Making: The Recognition Process.*

[30]J. J. Little *Computerized Evaluation of Essays.*

[31] W. Kintsch, A. R. Bowles. *Metaphor Comprehension: What makes a metaphor difficult to understand?*

[32]P. W.Foltz, S. Gilliam, S. A. Kendall. *Supporting content-based feedback in online writing evaluation with LSA.*

[33] M. D. Shermis, H. R. Mzumara, J. Olson Susanmarie Harrington *On-line Grading of Student Essays : PEG goes on the World Wide Web.*

[34] T. R. Payne *Dimensionality Reduction and Reprsentation for Nearest Neighbour Learning.*

[35] S. T. Dumais, Bellcore *Using Latent Semantic Indexing for Information Retrieval, Information Filtering and Other Things.*

[36] E. Hong(Sam)Han, G. Karypis, V. Kumar *Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification.*

[37]R. Baeza, B. Ribeiro. (1999). *Modern Information Retrieval*, Pearson Education Limited.

[38] Page, E.B.(1966). *The Imminence of Grading Essays by Computer*, Phi Delta Kappan, January,238-243

[39] Page E.B.(1994). *Computer Grading of Student Prose, Using Modern Concepts and Software*, Journal of Experimental Education, 62,127-142.

[40] N. Idris and S. M. F. D. Syed Mustapha. (2001). *Automated Essay Grading: An Application for Historical Malay Text*. International Conference on Artificial Intelligence and Applications (IC-AI 2001).

[41] S. R. Schach. (2004). *Object-Oriented and Classical Software Engineering*. 6th Edition. McGraw Hill.

[42] G. Salton, C. Buckley. (1988) *Term-Weighting Approaches in Automatic Text Retrieval*. Information Processing and Management, 24(5), 513-23.