UNIVERSITY OF MALAYA

ESSAY GRADING SYSTEM (EGS)

Elaiyakumaran s/o Subramaniam

WEK 010069

Supervisor : Pn. Maizatul Akmar bt. Ismail
Moderator : Pn. Kiran Kaur

Submitted in fulfillment for the Degree of Computer Science (Honours) in the Department of Management of Information System, Faculty of Computer Science and Information Technology, University of Malaya

February, 2004

Department of MIS,
Faculty of Computer Science
& Information Technology,
University of Malaya,
Kuala Lumpur.

ABSTRACT

As a part of my graduation exercise, it is required to develop a system for the Degree of Bachelor in Computer Science and Information Technology, University of Malaya. The system is intended to develop an Automated Essay Grading System (EGS). The EGS hopes to overcome the manual marking that are done for many years. The EGS's aim is to give a fast response and marks to the questions that are answered.

The literature review gives an overview of the marking system and findings of various researches and technology in the effectiveness of the Automated Essay Grading System. Some of the literature review surveys done are examined and their features compared. In this project, sources were collected from various part including the internet, books, and the senior projects.

The Automated Essay Grading System developed by using V-model as a foundation of the systems process flow. The selection of both hardware and software are vital to ensure the success of a system. Criteria such as capability, credibility, cost , speed and size of memory have been emphasized to ensure the hardware-selected meet the requirement of the system to be developed.

System design is very crucial in a system development process as it plays a major role in determining the success of the systems. There are two kinds of user interfaces - the student interface and the marking system (teachers). In conclusion, the system will help teachers to organize their student's exam in an appropriate way. Whereas, the students could learn and practice the History subject efficiently.

ACKNOWLEDGMENTS

There are several parties that the author would like to express the appreciation to them. Firstly, the author would like to express the deepest gratitude to Pn. Maizatul Akmar as the supervisor of the author who had given the advice and supports to the author in order to help the author to overcome the problems occurred during my thesis project. Without Pn. Maizatul's help this project would not be carried on as smoothly as it can.

Secondly, the author would like to express deeply appreciation to my moderators, Pn. Abrizah Abdullah and Pn.Kiran Kaur for their valuable advice and ideas on the project that help the author to improve the weakness of the project. Not forget also the lectures of Faculty of Computer Science and Information Technology who help to build up the foundation of the author in the pass two years.

Thirdly the author would like to convey the sincere thanks to all his friends especially R.Vasanthamaran, S.Kale dasan, G.Raguraman, "Kurau Makkals" and "4th Makkals who had help the author to overcome the problems and give the countless support to the author.

Last but not least. My sincere gratitude to my mother D.Mahaletchimi, sister and my family members for being supportive, motivate and caring throughout my project development.

TABLE OF CONTENTS

CHAPTER 1 PROJECT OVERVIEW

CHAPTER 2 LITERATURE REVIEW

CHAPTER 3 METHODOLOGY

CHAPTER 4 SYSTEM ANALYSIS AND DESIGN

CHAPTER 5 SYSTEM IMPLEMENTATION

CHAPTER 6 SYSTEM TESTING

LIST OF FIGURES

TABLES

CHAPTER 1

PROJECT OVERVIEW

1.1 INTRODUCTION TO PROJECT

This introductory chapter gives a description or purpose of the project and problems to be solved. The significance and rationale of the project will be discussed here. The system functions, limitation and its assumption will also be dwelt into later.

1.2 PROJECT OVERVIEW

As we have move towards the future, time is very important for each and everyone of us. Due to limited teaching resources and the large number of students, instructors would welcome an IT-tool that can be used any time to mark the students' coursework.

Writing is an essential part of the educational process, many teachers find it difficult to incorporate large number of writing essays in their exams due to the effort required to evaluate them. The liability to convey information verbally is an important educational achievement in its own right and on that neither is nor sufficiently well assessed by other kind of tests. In addition the testing is thought to encourage a better conceptual understanding of the material and to reflect a deeper, more useful level of knowledge and application by students.

Thus grading and commenting on written texts is important not only as an assessment method, but also as a feedback device to help students better learn both content and the skills of thinking and writing. Building a system that can mark the essay questions will benefit both the student and teachers.

1.3 PROJECT MOTIVATION

This system can help to mark an essay questions in Malay language specifically history .This will be beneficial to both the students and teachers. As we have move towards the future, usage of the manual system decreasing because computerized system replacing it. So it is a must for us to move along with the new technologies. In education field, the essay marking system could help to reduce time, cost and manpower.

1.4 OBJECTIVE

The broad objective of this thesis is to construct a system which will assist user in knowing how to do the marking process of an essay question given. The objectives of the essay grading system are the following:

- To build a system which will mark students essay as well as suggest the marks that will be given to the marked paper.
- To reduce the time marking the paper manually and user friendly.
- To build a system which, consist of notes and scheme as an extra features for students reference.

1.5 EXPECTATION OF PROJECT

The project is expected to come out with a system which can help to mark student's essays during test or examination which is considered accurate. At the same time it is expected to reduce the marking duration. The teachers find themselves bogged down in their attempt to provide consistent evaluations and high quality feedback to students within as short a timeframe as is reasonable, usually a matter of days rather than weeks. This can cause many problems soon. This system is expected to reduce time, cost and manpower of marking essays. The system is also expected to help teacher and students in education fields.

1.6 SCOPE

The function of this system is to help the users to do the marking based on their requirement. Basically the system will help the user to do all the checking and marking on the answers.

The user of the system is secondary school teachers and students. This project typically only covers one particular subject, which is History (secondary schools) in Malay language, but it can be implemented by other subjects that have similarity with this.

1.7 LIMITATION

This Essay Grading System can only use to mark essays which are in Malay language. Even though the system used to mark history subject but it can be used to mark other subjects also if it is possible. It also covers the history of Malaysia.

1.8 PROJECT SCHEDULE

A project schedules was prepared at the beginning stage of the project to ensure that this project will be completed in time. Basically, this project is divided into 8 stages and the activities for each stage are as follows:

Table 1.0 : Activities in each stage.

1. Preliminary investigation	▪ Identify project objective ▪ Identify system scope
2. Study program tools	▪ ASP.net / PHP ▪ Microsoft Access 2002
3. System Planning	▪ Determine system requirement ▪ Prepare project schedule
4. System Design	▪ Design screen format
5. System Implementation	▪ Code system module
6. System testing	▪ Design test data
7. System maintenance	▪ Make necessary changes to the system
8. Documentation and report	▪ Prepare project report ▪ Prepare user manual

A Gantt Chart is an easy way to schedule tasks. It is essentially is a chart on which bars represent each task on activity. The length of each bar represents the relative length of the task. The Gantt Chart for this project is shown in table 1.1 below. This schedule was re-estimated from time to time during the development life cycle.

Table 1.1: Proposed Schedule of the project

Proposed Schedule (July 2003 – February 2004)								
Activities	**July**	**Aug.**	**Sept.**	**Oct.**	**Nov.**	**Dec.**	**Jan.**	**Feb.**
Preliminary Investigation	■							
Literature Review	■	■						
System Analysis and Design	■	■						
System Development			■	■	■	■	■	
System Testing			■	■	■	■	■	
Implementation and maintenance						■	■	■
Documentation	■	■	■	■	■	■	■	■

LEGEND

Completed task ■

1.9 SUMMARY

This chapter gives an overview of the project that includes the aims, relevance, significance and the scope of the development of the essay grading system. Educational administrators are also concerned with this essay grading system, but in addition must manage the cost of doing this work. Clearly an automated system would be a highly desirable addition to the educational tool-kit, particularly if it can provide less costly and more effective outcome. In conclusion, the essay grading system has been around for years, but the actual implementation is still in its early stages. Besides, it also covers the importance and feasibility of the project as well as the project limitations and expected outcome.

CHAPTER 2

LITERATURE REVIEW

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION TO LITERATURE REVIEW

Literature review is a background study about knowledge and information needed to develop the project. It helps better understanding on the system requirement and methodologies used in project development.

2.2 RESEARCH APPROACH

A lot of information needs to be obtained to develop a system. There are several ways of gathering information:

Table 2.0 Method of research approach

Source	**Method**
Web-sites	Searching by search engine.
Books, Magazines, Newspapers ,etc	Selecting appropriate articles, read and summarize.
Past year thesis	Findings in FSKTM documentation room.
Library	Search related material via OPAC.

Internet is the biggest library that containing many kinds of information. However searching on internet is time consuming. There are two approaches to search information on internet which are directly from website and through search engine. The most popular way of retrieving information is via search engine. It provides us with more related information. Book reading is another way of research for this project. The most important books are reference books and technical books. Newspapers and magazines perhaps are the most traditional and cheap way to perform information retrieval. It contains lots of articles on different categories. While, library has a large collection of books journals, magazines, newspapers, etc. Main library in University of Malaya is visited frequently to search on related topic that might help me in system development. Besides, past year thesis from seniors also can be obtained from documentation room in FSKTM. These samples are useful to provide basic guideline and ideas on how to generate a good report, by evaluate the strength and weaknesses of their work.

2.3 SYLLABUS OF HISTORY SUBJECT

The scope that the grading system covers is form five history subject. According to the research on SPM history reference books and related sources, the syllabus for the history subject for form 5 as following:

- *Bab 1 : Tamadun Awal Di Asia Tenggara*
- *Bab 2 : Warisan Kesultanan Melayu*
- *Bab 3 : Menhadapi Cabaran Barat*
- *Bab 4 : Pembentukan Ekonomoi Moden*
- *Bab 5 : Pembinaan Bangsa dan Negara Yang Berdaulat*
- *Bab 6 : Pembangunan dan Kesejahteraan*
- *Bab 7 : Malaysia dan Hubungan Antarabangsa*

2.4 CONCEPTUAL MODELS FOR AUTOMATED ESSAY GRADING

The first model, Project Essay Grade (PEG), is one of the earliest and longest-lived implementations of automated essay grading. It has been developed by Page and colleagues, and primarily relies on linguistic features of the essay documents.

The second model, E-RATER, is one developed by Burstein et al at the Educational Testing Service (ETS) in the US, which has been implemented to the prototype stage for evaluation. This model uses a hybrid approach of combining linguistic features, derived by using Natural Language Processing (NLP) techniques, with other document structure features.

The third model, the LSA model, makes use of Latent Semantic Analysis (LSA) and the "bag of words" approach, and has been developed and evaluated by Landauer et al at the University of Colorado at Boulder. It ignores document linguistic and structure features.

The fourth model, which uses text categorization techniques, identified in this paper as TCT, has been developed by Larkey at the University of Massachusetts. It uses a combination of modified key words and linguistic features [22].

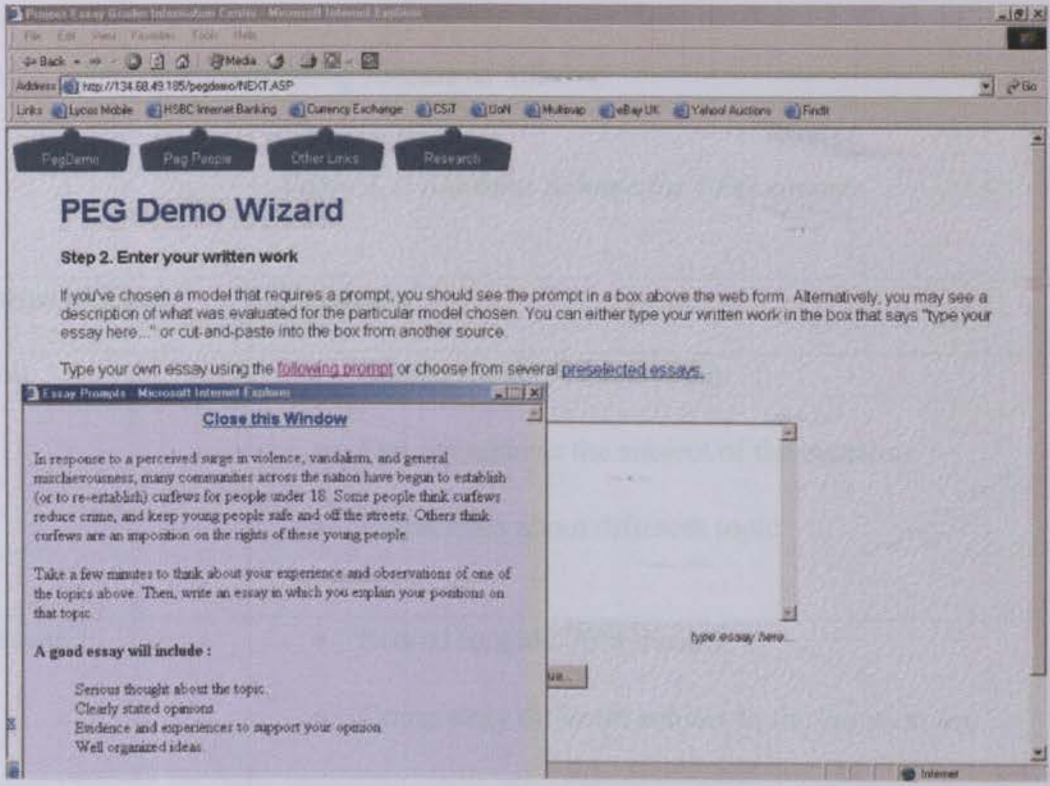
2.4.1 PEG

Ellis Batten Page of Duke University in the USA developed Project Essay Grade (PEG) since the mid 60's. Page uses, what he terms, *proxes*, which are computer approximations or measures of *trins*, in*trins*ic variables of interest within the essay (i.e. what a human grader would look for but the computer can't directly measure), to simulate human rater grading. *Proxes* include, essay length (as the amount of words) to represent the *trin* of fluency; counts of prepositions, relative pronouns and other parts of speech, as an indicator of complexity of sentence structure and variation in word length to indicate diction (because less common words are often longer). *Proxes* are calculated from a set of training essays and are then transformed and used in a standard multiple regressions along with the given human grades for the training essays to calculate the regression coefficients. These regression coefficients are the weighting that best simulates the given human grades when used with the calculated *proxes*. They are then used with *proxes* calculated from the unmarked essays to produce expected grades. Page's latest experiments have achieved results reaching a multiple regression correlation as high as 0.87, which is more reliable than a 6-judge panel, i.e. the computer is predicting the scores that judges will assign to essays better than the judges are predicting each other.

PEG relies purely on a statistical approach that assumes that the quality of the essay, the *trins*, is reflected in the measurable *proxes*. No natural language processing is used and lexical content is not taken into account. PEG also requires training, in the form of assessing a number of previously manually marked essays for *proxes*, in order to calculate the regression coefficients, which enables the marking of new essays. [10]

The following is the PEG system which was created by page [12]:

Figure 2.0: Screen-shot of PEG system.



Back
Media
Address http://134.68.49.185/pegdemo/NEXT.ASP
Go
Links Lycos Mobile HSBC Internet Banking Currency Exchange CSIT UoN Multimap eBay UK Yahoo! Auctions Find!
PegDemo
Peg People
Other Links
Research
Internet

PEG Demo Wizard

Step 2. Enter your written work

If you've chosen a model that requires a prompt, you should see the prompt in a box above the web form. Alternatively, you may see a description of what was evaluated for the particular model chosen. You can either type your written work in the box that says "type your essay here..." or cut-and-paste into the box from another source.

Type your own essay using the following prompt or choose from several preselected essays.

Close this Window

In response to a perceived surge in violence, vandalism, and general mischievousness, many communities across the nation have begun to establish (or to re-establish) curfews for people under 18. Some people think curfews reduce crime, and keep young people safe and off the streets. Others think curfews are an imposition on the rights of these young people.

Take a few minutes to think about your experience and observations of one of the topics above. Then, write an essay in which you explain your positions on that topic.

A good essay will include :

Serious thought about the topic.
Clearly stated opinions
Evidence and experiences to support your opinion
Well organized ideas

type essay here...

The user is presented with a choice of 2 models:

- IUPUI English Placement Exam: This is the standard placement test given to all students entering IUPUI (Indiana University Purdue University Indianapolis). Students are given one hour to write an impromptu exam with the choice of several essay prompts.
- IUPUI Trait Model This is a Beta example of the capabilities of the Trait Modeling system for PEG.

At first, when using the system for marking an essay it did not work, so the Trait model was used. The user is given a choice of typing an essay or using a pre-selected one. For this test the pre-selected one was chosen and the topic can be seen in the pop-up window.

The systems marking scheme as follows:

Table 2.1: Marking Scheme for PEG system.

Results of Essay	**Comments**
Good	• Scored roughly 90% overall. • Did not address the subject of the question. • It describes about different topic.
Average	• Scored roughly 70% overall. • Completely different subject to the question set.
Acceptable	• Scored roughly 50% overall. • This essay contained terrible spelling mistakes and was very short.
Poor	• Scored roughly 50% overall. • Allowed the student to pass.

In conclusion, this system is far perfect. In major trials in the US it achieved good results but these test show it is an easy system to beat. In a university level it would not take long for the students to realize how to come out with a better system and obtain marks by pointless statements.

2.4.2 LATENT SEMANTIC ANALYSIS MODEL

Latent Semantic Analysis (LSA) was developed, in the main, by Thomas K. Landauer of the University of Colorado, Boulder and Peter W. Foltz of New Mexico State University. LSA was not initially developed for use in automated essay grading but has been applied to this. It was first used for indexing documents and information retrieval and, therefore, much work on the essay grading aspect remains unpublished (for example, Landauer, Laham & Foltz (1998), 'Computer-based Grading of the Conceptual Content of Essays').

The technique, which is a method of representing contextual usage of words, proceeds along a number of steps. Firstly the essay is transformed into a matrix representation whereby each row represents a unique word and each column is a 'context', such as a sentence or a paragraph. Each cell then contains the frequency of the word appearing in that context. For example [9]:

The sentences,

"The man likes going on holiday" (Context A)

and, "Spain is a popular holiday destination" (Context B)

Would give the matrix,

Table 2.2 : Matrix Representation

	Context A	**Context B**
Man	1	0
Holiday	1	1
Spain	0	1

Even though not all of the words from the original text would be represented in the matrix (certain *stop words*, such as the, and, if etc. would be removed and morphological differences allowed for) a real essay would obviously create a very large matrix.

The initial matrix is then transformed. Each word occurrence is weighted as an estimate of its importance in the passage, and inversely with the degree to which knowing that a word occurs provides information about which passage it appeared in. This means that if a word appears frequently in one context, but rarely in another, then that word is an important keyword for that particular context. This first transformation of the initial matrix is similar to inverse document frequency (IDF) weighting that is often used in indexing and information retrieval. The weighting gives more importance to index terms that are more specific, i.e. those that occur less, because indexing specificity is inversely proportional to the number of documents an index term occurs in. IDF weighting is also used in the latest work by ETS [9].

Singular Value Decomposition (SVD), a form of factor analysis, is then applied is to the weighted matrix. SVD involves decomposing a rectangular matrix into the product of 3 other orthogonal (i.e. AAT=I, the original matrix multiplied by the transpose is equal to the identity matrix) matrices. This gives you two rectangular/square matrices (depending on the dimensions of the original matrix) and one diagonal matrix consisting of the singular values of the original matrix.

For example, if we have the matrix, $A = \begin{pmatrix} 96 & 172 \\ 228 & 96 \end{pmatrix}$

Its singular value decomposition would be,

$$\begin{pmatrix} 6 & -8 \\ 8 & 6 \end{pmatrix}\begin{pmatrix} 3 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 8 & 6 \\ 6 & -8 \end{pmatrix}$$

Next the dimension of the diagonal matrix is reduced thereby reducing the rank (the number of linearly independent rows and columns) of the original matrix. Finding the optimal dimension for the matrix is important so that correct induction of underlying relations between words and contexts can come through.

A new weighted version of the original matrix is now reconstructed by multiplying these three component matrices together. This dimension reduction increases the dependency of the data on each other, increasing the links between words and contexts, theoretically meaningfully. Cosine correlation (also used in EST's latest work) is used to measure the similarity of the reduced dimensional space constructed from a 'model answer', such as an instructional text taken from a course text or an essay prepared by the class tutor, against a student essay [9].

LSA has produced some impressive results. To use their own words, their approach has produced grades that correlate "approximately as well with experts' assigned scores as such scores correlate with each other, sometimes slightly less well, on average slightly better" (Landauer, 1998). LSA makes no use of word order although they claim (Landauer, 1997b) that it is not the most important factor in collecting the sense of a passage. It also requires large amounts of data in order to be able to construct a suitable matrix representation of word use/occurrence and due to the size of the matrices involved any mathematical calculations will be, computationally, very expensive [21].

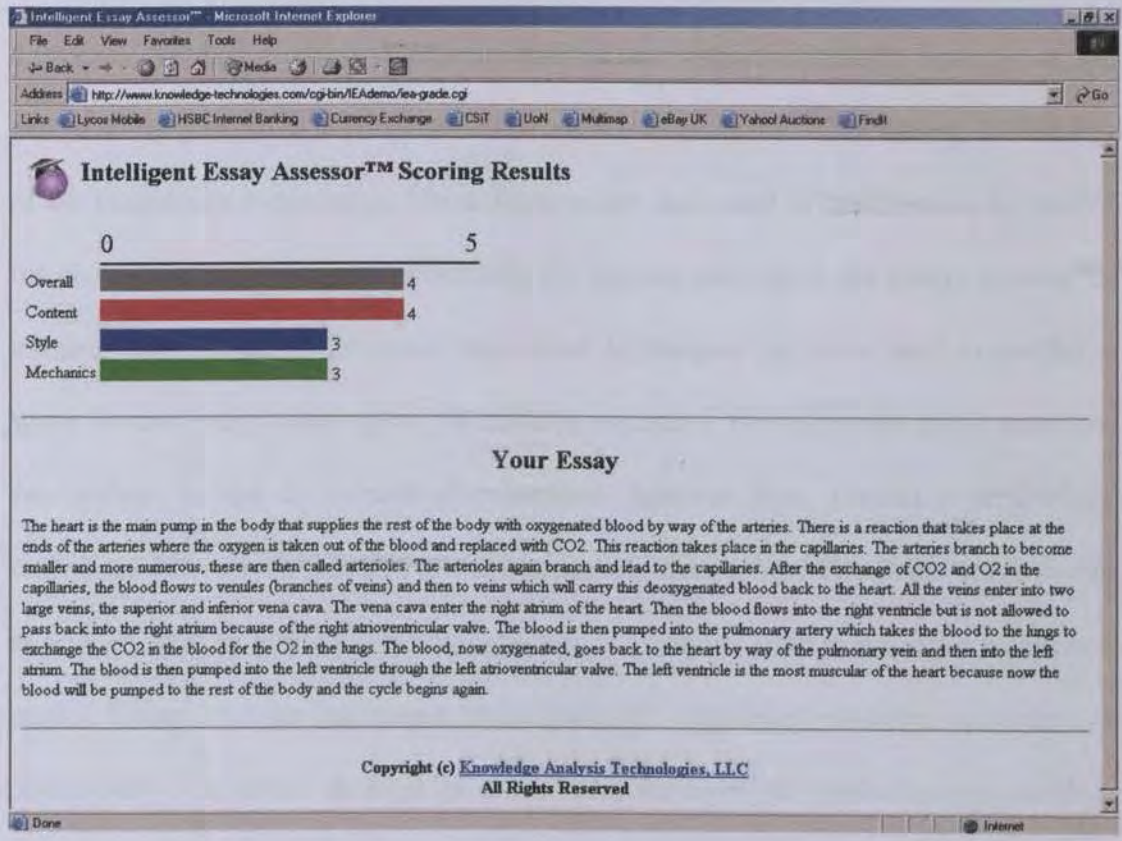
The famous System which is created by using Latent Semantic Analysis is Intelligent Essay Assessor. The Intelligent Essay Assessor (IEA) from Knowledge Analysis Technologies (KAT) is still one of the only commercially available products. In this system the user is given a choice of five essay topics [12]:

- Biology: Function of Heart & Circulatory System
- Psychology 1: Attachment in Children
- Psychology 2: Types of Aphasia
- Psychology 3: Operant Conditioning
- History: The Great Depression

The user is then given a space to write an essay or choose from three example essays. It is not known how the IEA allocates marks but in this situation the essays are graded as they probably would be with a human marker.

Figure 2.1 shows the example essays and the scoring they get [14]:

Figure 2.1 screen-shot of IEA system


Intelligent Essay Assessor™ - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Media
Address http://www.knowledge-technologies.com/cgi-bin/IEAdemo/iea-grade.cgi
Go
Links Lycos Mobile HSBC Internet Banking Currency Exchange CSiT UoN Multimap eBay UK Yahoo! Auctions FindIt
0
5
Done
Internet

Intelligent Essay Assessor™ Scoring Results

Overall	4
Content	4
Style	3
Mechanics	3

Your Essay

The heart is the main pump in the body that supplies the rest of the body with oxygenated blood by way of the arteries. There is a reaction that takes place at the ends of the arteries where the oxygen is taken out of the blood and replaced with CO2. This reaction takes place in the capillaries. The arteries branch to become smaller and more numerous, these are then called arterioles. The arterioles again branch and lead to the capillaries. After the exchange of CO2 and O2 in the capillaries, the blood flows to venules (branches of veins) and then to veins which will carry this deoxygenated blood back to the heart. All the veins enter into two large veins, the superior and inferior vena cava. The vena cava enter the right atrium of the heart. Then the blood flows into the right ventricle but is not allowed to pass back into the right atrium because of the right atrioventricular valve. The blood is then pumped into the pulmonary artery which takes the blood to the lungs to exchange the CO2 in the blood for the O2 in the lungs. The blood, now oxygenated, goes back to the heart by way of the pulmonary vein and then into the left atrium. The blood is then pumped into the left ventricle through the left atrioventricular valve. The left ventricle is the most muscular of the heart because now the blood will be pumped to the rest of the body and the cycle begins again.

Copyright (c) Knowledge Analysis Technologies, LLC
All Rights Reserved

2.4.3 E-RATER

E-rater is the automated scoring system that has been developed to assist in the evaluation of test takers' responses to open-ended essay questions. E-rater uses a combination of statistical and Natural Language Process (NLP) techniques to duplicate the performance of human readers and extract linguistic features of the essays to be graded. It uses several subroutines to extract a wide variety of features of the essays that it evaluates. These features are then used in combination to predict the scores that were assigned previously (by human readers) to the essays in *e-raters* training sample.. Multiple linear regression techniques are then used to predict a score for the essay, based upon the features extracted. For each new essay question, the system is run to extract characteristic features from human scored essay responses. Fifty seven features of the benchmark essays, based upon six score points in an ETS scoring guide for manual grading, are initially used to build the regression model. Using stepwise regression techniques, the significant predictor variables are determined. The values derived for these variables from the student essays are then substituted into the particular regression equation to obtain the predicted score [16].

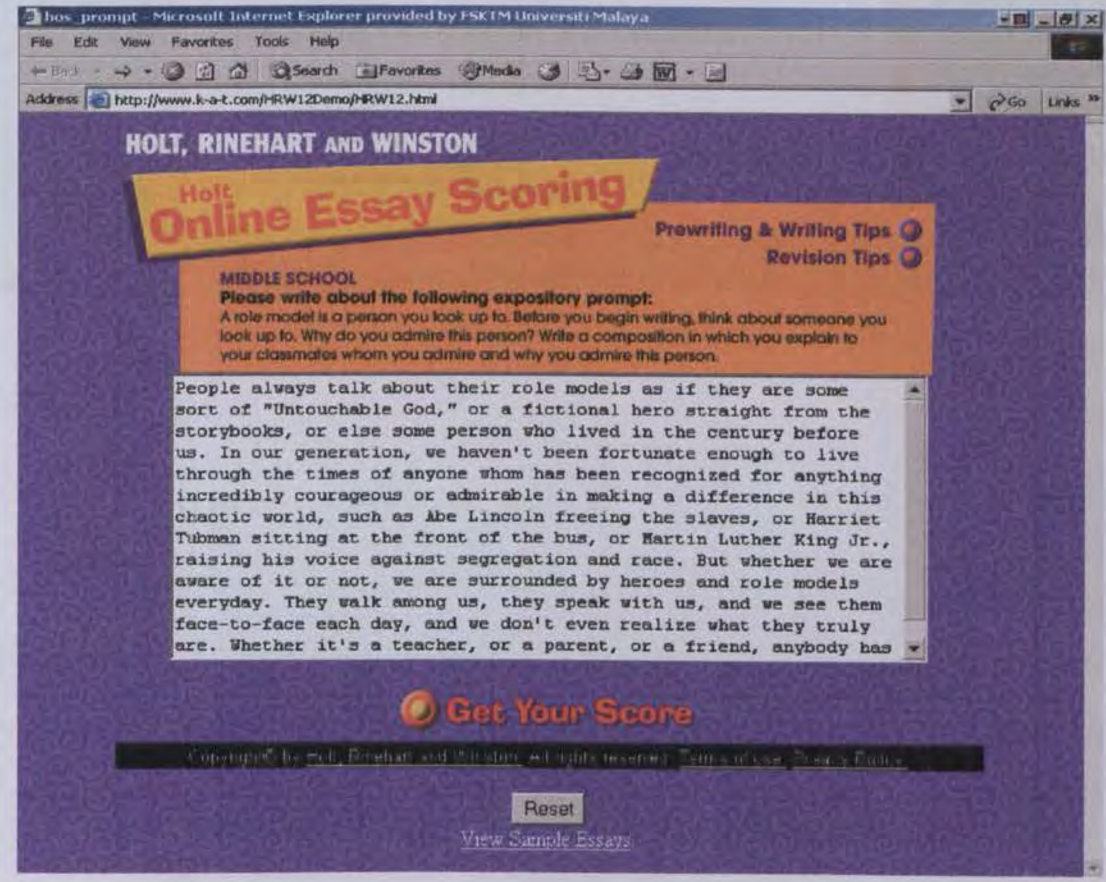
One of the scoring guide criteria is essay syntactic variety. After parsing the essay with an NLP tool, the parse trees are analyzed to determine clause or verb types that the essay writer used. Ratios are then calculated for each syntactic type the essay and per sentence basis. Another scoring guide criteria relates to having well-developed arguments in the essay.

Discourse analysis techniques are used to examine the essay for discourse units by looking for surface cue words and non-lexical cues.

These cues are then used to break the essay up into partitions based upon individual content arguments. The system also compares the topical content of an essay with those of the reference texts by looking at word usage [16].

The following is the online E-rater system which is available in the internet [18].

Figure 2.2: Screen-shot of E-rater System


hos_prompt - Microsoft Internet Explorer provided by FSKTM Universiti Malaya
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.k-a-t.com/HRW12Demo/HRW12.html
Go Links
HOLT, RINEHART AND WINSTON
Holt Online Essay Scoring
Prewriting & Writing Tips
Revision Tips

MIDDLE SCHOOL

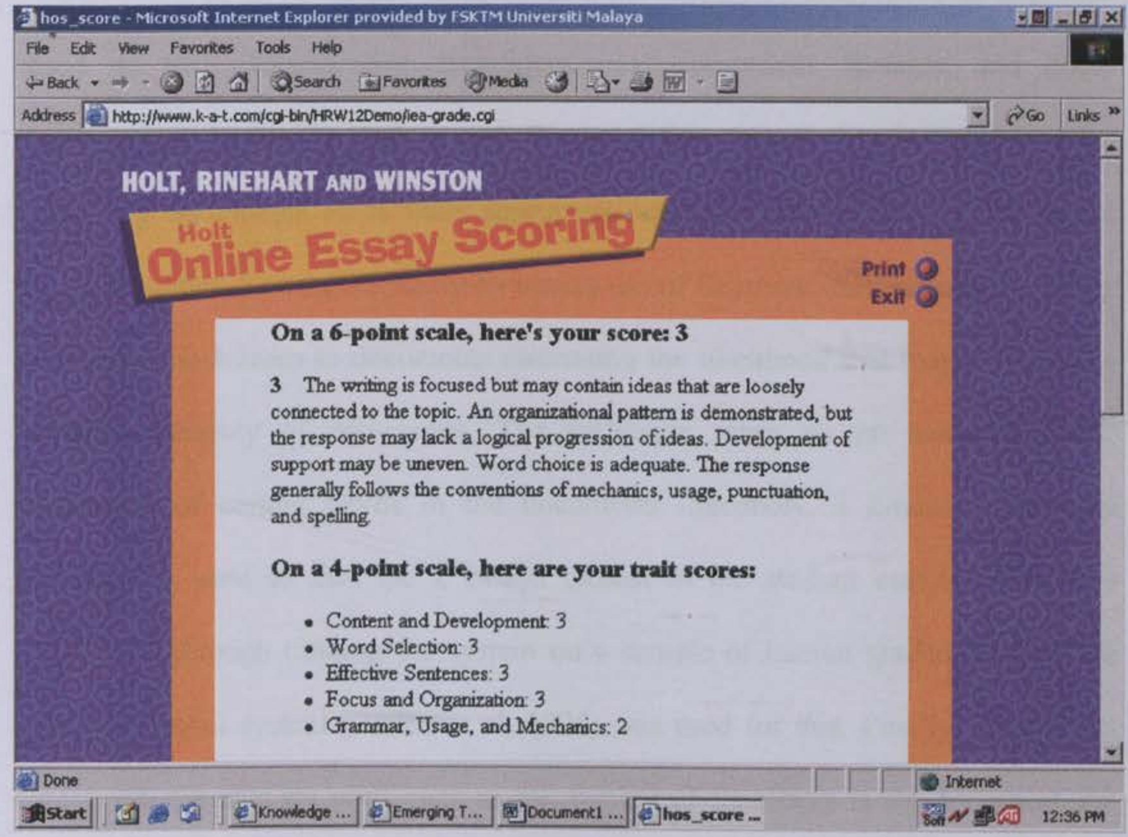
Please write about the following expository prompt:

A role model is a person you look up to. Before you begin writing, think about someone you look up to. Why do you admire this person? Write a composition in which you explain to your classmates whom you admire and why you admire this person.

```
People always talk about their role models as if they are some
sort of "Untouchable God," or a fictional hero straight from the
storybooks, or else some person who lived in the century before
us. In our generation, we haven't been fortunate enough to live
through the times of anyone whom has been recognized for anything
incredibly courageous or admirable in making a difference in this
chaotic world, such as Abe Lincoln freeing the slaves, or Harriet
Tubman sitting at the front of the bus, or Martin Luther King Jr.,
raising his voice against segregation and race. But whether we are
aware of it or not, we are surrounded by heroes and role models
everyday. They walk among us, they speak with us, and we see them
face-to-face each day, and we don't even realize what they truly
are. Whether it's a teacher, or a parent, or a friend, anybody has
```

Get Your Score
Copyright© by Holt, Rinehart and Winston. All rights reserved. Terms of Use. Privacy Policy.
Reset
View Sample Essays

Figure 2.3: Screen-Shot of E-rater system.


hos_score - Microsoft Internet Explorer provided by FSKTM Universiti Malaya
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.k-a-t.com/cgi-bin/HRW12Demo/iea-grade.cgi
Go Links
HOLT, RINEHART AND WINSTON
Holt Online Essay Scoring
Print
Exit
Done
Internet
Start Knowledge ... Emerging T... Document1 ... hos_score ...
12:36 PM

On a 6-point scale, here's your score: 3

3 The writing is focused but may contain ideas that are loosely connected to the topic. An organizational pattern is demonstrated, but the response may lack a logical progression of ideas. Development of support may be uneven. Word choice is adequate. The response generally follows the conventions of mechanics, usage, punctuation, and spelling

On a 4-point scale, here are your trait scores:

- Content and Development: 3
- Word Selection: 3
- Effective Sentences: 3
- Focus and Organization: 3
- Grammar, Usage, and Mechanics: 2

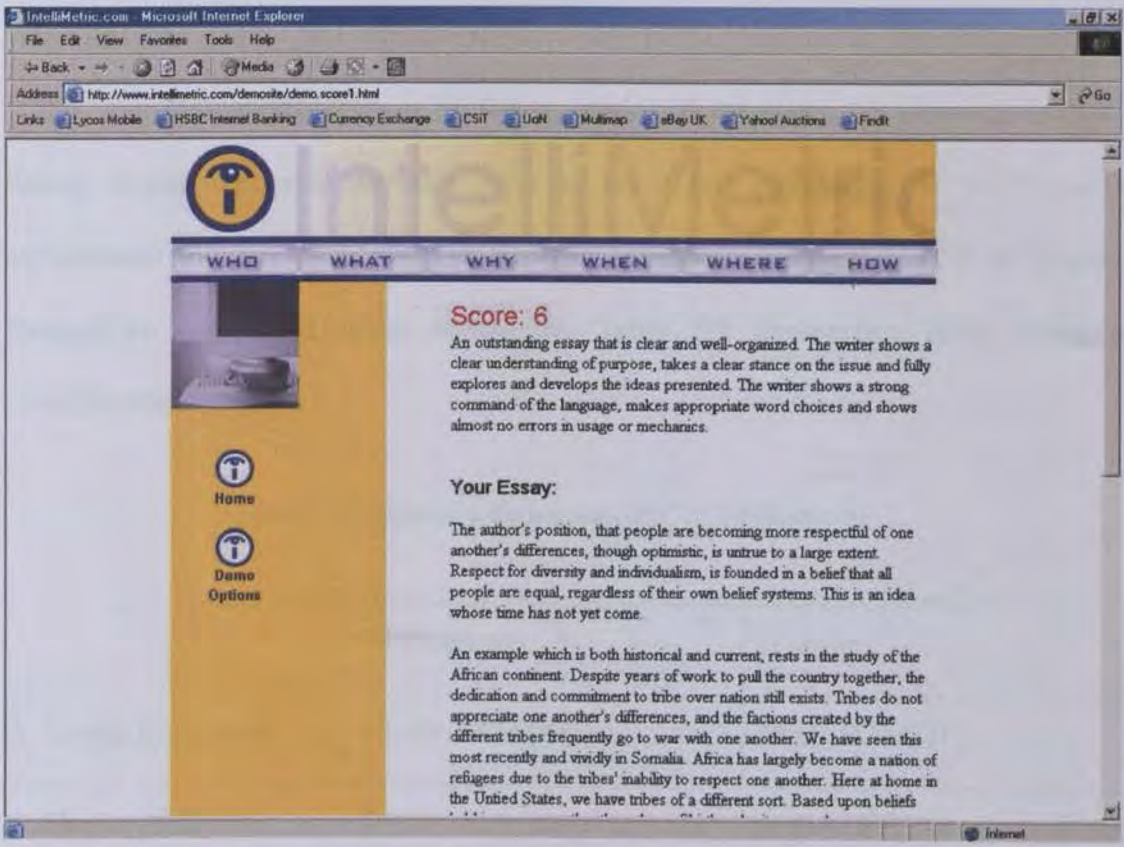
2.4.4 THE TEXT CATEGORIZATION TECHNIQUE (TCT)

In 1998 Larkey implemented an automated essay grading approach based on text categorization techniques, text complexity features, and linear regression methods. The Information Retrieval literature discusses techniques for classifying documents as to their appropriateness of content for given document retrieval queries. The technique firstly makes use of Bayesian independent classifiers to assign probabilities to documents estimating the likelihood that they belong to a specified category of documents. The technique relies on an analysis of the occurrence of certain words in the documents. Secondly, a k-nearest neighbor technique is used to find the k essays closest to the student essay, where k is determined through training the system on a sample of human graded essays. The Inquiry retrieval system (Callan et al, 1995) was used for this. Finally, eleven text complexity features are used, such as the number of characters in the document, the number of different words in the document, the fourth root of the number of words in the document, and the average sentence length.

Larkey conducted a number of regression trials, using different combinations of components. He also used a number of essay sets, including essays on social studies (soc), where content was the primary interest, and essays on general opinion (G1), where style was the main criteria for assessment [17].

The Intellimetric Systems assessments are shown below [15]:

Figure 2.4: Screen-shot of Intellimetric System


IntelliMetric.com - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Media
Address http://www.intellimetric.com/demosite/demo.score1.html
Go
Links Lycos Mobile HSBC Internet Banking Currency Exchange CSiT UoN Multimap eBay UK Yahoo! Auctions FindIt
WHO WHAT WHY WHEN WHERE HOW
Home
Demo Options
Internet

Score: 6

An outstanding essay that is clear and well-organized. The writer shows a clear understanding of purpose, takes a clear stance on the issue and fully explores and develops the ideas presented. The writer shows a strong command of the language, makes appropriate word choices and shows almost no errors in usage or mechanics.

Your Essay:

The author's position, that people are becoming more respectful of one another's differences, though optimistic, is untrue to a large extent. Respect for diversity and individualism, is founded in a belief that all people are equal, regardless of their own belief systems. This is an idea whose time has not yet come.

An example which is both historical and current, rests in the study of the African continent. Despite years of work to pull the country together, the dedication and commitment to tribe over nation still exists. Tribes do not appreciate one another's differences, and the factions created by the different tribes frequently go to war with one another. We have seen this most recently and vividly in Somalia. Africa has largely become a nation of refugees due to the tribes' inability to respect one another. Here at home in the Untied States, we have tribes of a different sort. Based upon beliefs

Intellimetric is a very new piece of software and the company states that Intellimetric provides both holistic and analytical scoring for constructed response essays with accuracy greater than or equal to that of two expert human scorers. More time is needed to evaluate this software correctly and a fully working demonstration would be needed. Time will tell if this is the best marking software but for now e-rater seems to have been established as the industry standard. The assessment of essay based on the ability to express, organize, and support opinions and ideas rather than the position express.

2.4.5 COMPARISON AMONG THE FOUR SYSTEMS

PEG focuses on simple linguistic features, focusing on style, and can be categorized as II (A). E_RATER focuses on linguistic features and document structures, and is thus performing a Master Analysis of style, and falls in the category II (B). The LSA model focuses on the semantics of the essay, but does so using a Rating Simulation, and therefore falls in the I (A) category. The TCT (soc) experiments focused on content in a rating simulation, while the TCT (G1) test focused on style in a rating simulation. Table 2.3 summarizes these models' classifications [22].

Table 2.3: Essay grading models' classifications

	I Content	**II style**
A. Rating Simulation	LSA, TCT (soc)	PEG, TCT (G1)
B. Master Analysis		E_RATER

Table 2.4 shows some of the reported performances, in comparison to human graders, of the various models.

Table 2.4: Comparative performance of models

Model	Measure	Values	Source
PEG	R	0.389-0.743	Page, 1994
E_RATER	%	87-94	Burstein, et al, 1998
LSA	%	85-91	Landauer, 1999
TCT (soc)	R	0.69-0.78	Larkey , 1998
TCT (G1)	R	0.69-0.88	Larkey, 1998

2.5 CLIENT/SERVER COMPUTING

This client/server approach is certainly appealing: a wide range of hardware platforms running the state-of-the-art database servers can now address the performance requirements of most installation [1]. Every hardware and software vendor has something to say about this new technology. Workstation vendors such as Sun and Hewlett-Packard begin to churn out products used as server. Operating system vendors begin to emphasize the client/Server operating system while application-level and database management system vendors, such as Sybase and Oracle push client/server solutions to cater business problems [2].

2.5.1 DEFINITION

A client is defined as a requester of services and a server is defined as the provider of services. A single machine can be both a client and a server depending on the software configuration [4].

2.5.2 CLIENT/SERVER ARCHITECTURE

As a result of the limitations of file sharing architectures, the client/server architecture emerged. With this architecture, the calling module becomes the "client" (that request a service). And the called module becomes the "server" (that which provides the service) [3]. This approach introduced a database server to replace the file server. Using a relational database management system (DBMS), user queries could be answered directly. The client/server architecture reduced network traffic by providing a query response rather than total file transfer. It improves multi-user updating through a GUI front end to a shared database. In client/server architectures, Remote Procedure Calls (RPCs) or standard query language (SQL) statements are typically used to communicate between the client and server [4].

2.5.3 TWO-TIER ARCHITECTURE

Two-tier architecture is where a client talks directly to a server, with no intervening server. It is typically used in small environments (less than 50 users) [5].In this environment, much processing is performed on the client workstation, using the memory space and processing power of the client to provide much of the functionality of the system. Field edits, local lookups, and access to peripheral devices (scanners, printer, and so on) are provided and managed by the client system. Since almost all processing is done at the client, the client portion is sometime refers as fat-client [1].

A common error in client/server development is to prototype an application in a small, two-tier environment and then scale up by simply adding more users to the server. This approach will usually result in an ineffective system, as the server becomes overwhelmed. To properly scale to hundreds or thousands of users, it is usually necessary to move to a three-tier architecture [5].

2.5.4 THREE AND MULTI-TIER ARCHITECTURE

Three-tier architecture introduces a server (or an "agent") between the client and the server. The role of the agent is manifold. It can provide translation services (as in adapting a legacy application on a mainframe to a client/server environment), metering services (as in acting as a transaction monitor to limit the number of simultaneous requests to a given server), or intelligent agent services (as in mapping a request to a number of different servers, collating the results, and returning a single response to the client [5]. The movement from two-tier architecture to three-tier or multi-tier architecture due is to the needs for a scalable and maintainable system and the wide variety of clients within a larger organization.

2.5.5 CHARACTERISTICS OF CLIENT/SERVER ARCHITECTURE

The basic characteristics of client-server architectures are [4]:

1 Combination of a client or front-end portion that interacts with the user, and a server or back-end portion that interacts with the shared resource. The client process contains solution-specific logic and provides the interface between the user and the rest of the application system. The server process acts as a software engine that manages shared resources such as databases, printers, modems, or high powered processors.

2 The front-end task and back-end task have fundamentally different requirements for computing resources such as processors such as processors speeds, memory, disk speeds and capacities, and input/output devices. The environment is typically heterogeneous and multivendor. The hardware platform and operating system of client and server are not usually the same. Client and sever processes communicate through a well-defined set of standard application program interfaces (API's) and RPC's.

3 An important characteristic of client-server systems is scalability. They can be scaled horizontally or vertically. Horizontal scaling means adding or removing client workstations with only a slight performance impact. Vertical scaling means migrating to a larger and faster server machine or multiservers.

2.6 DATABASE

A database is a collection of data files, integrated and organized to minimize redundancy, enhance consistency, ease information retrieval, and enforce data independence [20].

2.6.1 DATABASE MANAGEMENT SYSTEM MODEL

The term database management system (DBMS) usually refers to an electronic system or a computer program designed to work with databases. Microsoft Access and FoxPro are both example of database management system.

Traditional database management systems are based on three data models: hierarchical model, network model and relational model. DBMSs of the 1970s were mainly based on hierarchical and network models. These two data models provide procedural query language interface that forces application programs to navigate through internal database structures to carry out user requests. As a result, any well trained application programmers can write queries against a specific database.

2.6.2 RELATIONAL DATABASE MANAGEMENT SYSTEM

The relational data model offered a fundamentally different approach for storing and accessing data. In this relational approach, all data are stored in tabular data structures (tables) and high level nonprocedural query language interface is provided. Consequently, common end users with little or no programming training are able to query the database directly. Because of its ease of use and elegance in data storage, relational database has dominated the commercial place since mid 1980s. RDBMSes are ideally suited for scalar data such as manes, address fields, and amounts. They are extremely stable and fast. It is possible to execute complex queries. These features make RDBMSes ideally suited for business and financial applications.

2.6.3 OBJECT-RELATIONAL DATABASE MANAGEMENT SYSTEM

Object-relational management system (ORDBMS) is a breed from RDBMs except that it has capabilities to handle complex data types such as images, documents, time-series inputs, or 3-D coordinates, which need special binary encodings to represents data. A time series need a start date and calendar that specifies the intervals between observations, while processing methods for video and audio objects, for example, require rewind capabilities. In essence, complex data types are best represented as objects. They encapsulate structures and methods regarding how the object interacts with other objects. The main challenge for developers are the learning curve required to understand the object-relational framework and being able to develop code within the constraints of API.

2.6.4 OBJECT-ORIENTED DATABASE MANAGEMENT SYSTEM

Object-oriented database management systems (ODBMSes) were one of the hot ideas of the early 1980s. The only problem was that the early ODBMSes were not complete database systems. They lacked backup and recovery functions. Data models were conflicting. Language was proprietary. Because of their structure, it was impossible to do true queries. ODBMSes were not scalable and required huge amounts of memory.

ODBMSes are for niche markets. The object model allows for the complex modeling of objects as they exist rather than trying to squeeze the objects into a 2-D structure. The growing interest in multimedia applications and the Internet has created new markets for ODBMSes. From that perspective, ODBMS technology is ideal for the most popular applications. The huge growth of the internet, video games, multimedia applications, and the development of distributed databases that do not lend themselves to the relational model are bringing renewed attention to ODBMS.

2.6.5 SECURITY REQUIREMENT FOR DATABASES

A database security violation is the unauthorized reading, modifications, or destruction of information stored in the database. The two classifications of security violations are malicious and accidental. Malicious violations are those crimes that exploit loopholes in the system, abusing positions privilege and trust by using data illegitimately. Accidental violations include hardware and software failures, human error, or natural disaster.

CHAPTER 3

METHODOLOGY

CHAPTER 3

METHODOLOGY

3.1 METHODOLOGICAL APPROACH

A methodology is a description of the process required for the development of a system. Methodologies vary in many aspects. They can be very broad, identifying the major steps involved from the conception of the system to the final use of that system, or they can be narrowed .Methodology also vary greatly in their depth. The amount of detail they provide when describing the design process varies considerably.

A process model can be prescriptions for the way software development is done in quality. There are many different process models such waterfall model , v model , prototyping model and etc. there are some reasons for modeling a process:

- Discussion of a process model's sub processes help developers understand what should be and what is in system development.
- When a group writes down a description of the activities, resources and constraints involved in software development.
- Creating and process model helps the development team find inconsistencies, redundancies, and omissions in the process and in its constituents part.
- The model should reflect the goals of development, such as building high-quality software, finding faults early in development ,and meeting required budget and schedule constraints. As the model is built, the development team evaluates candidate activities for their appropriateness in addressing these goals.

For example, the team may include requirement review, so that problems with the requirements can be found and fixed before design begins.

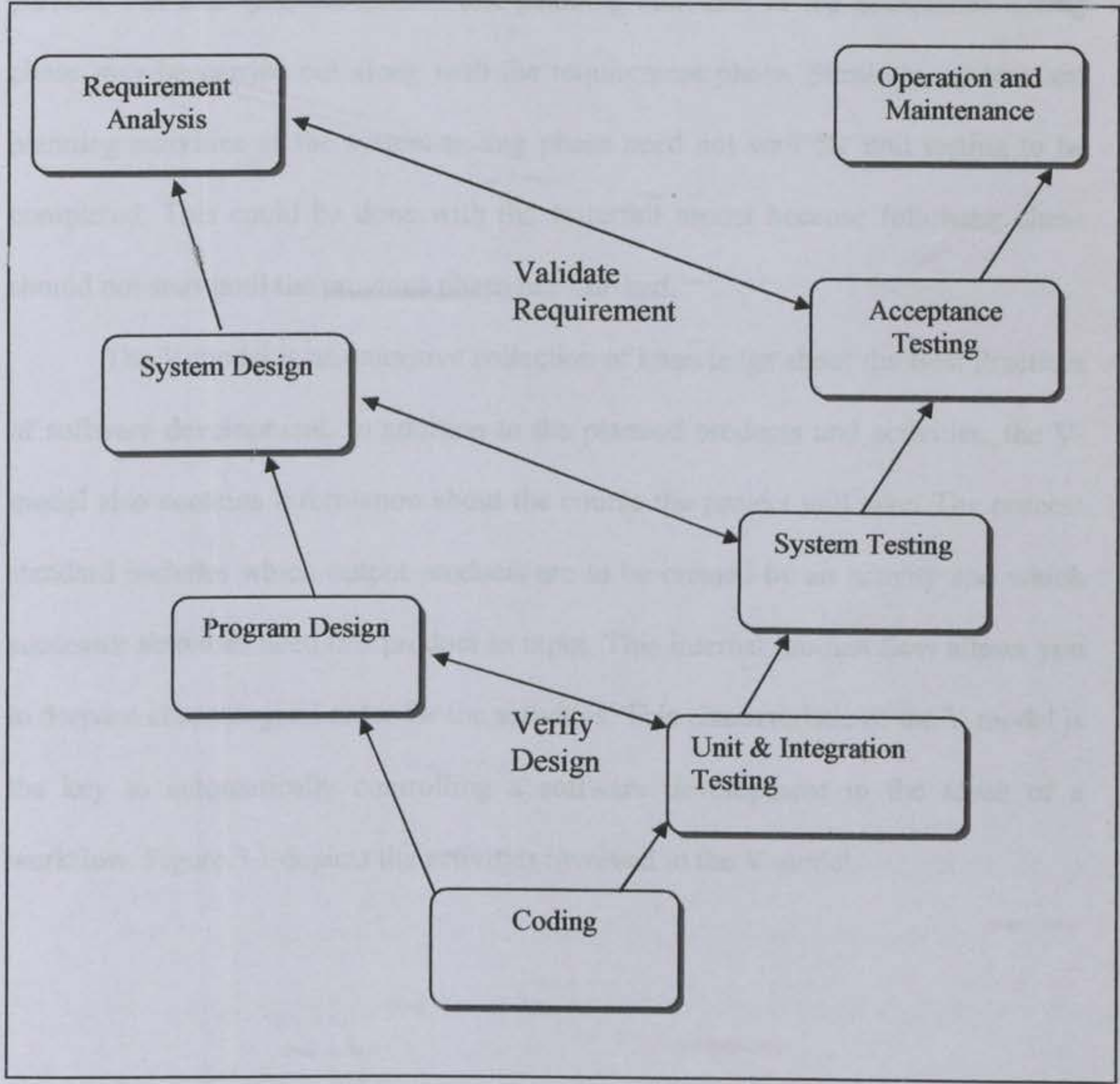
- Every process should be tailored for the special situation in which it will be used. Building a process model helps the development team understand where that tailoring is to occur . every software development process model includes system requirement as input and a delivered product as output.

3.2 PROJECT DEVELOPMENT METHODOLGY

The chosen process model for Essay Grading /system is V- model. The V- model is a variation of the waterfall model that demonstrates how the testing activities are related to analysis and design (German Ministry of Defense 1992). As shown in figure 3.0 ,coding forms the point of the V, with analysis and design on the left, testing and maintenance on the right. Unit and integration testing also be used to verify the program design. That is during unit and integration testing, the coders and test team members should ensure that all aspects of the program design have been implemented correctly in the code. Similarly, system testing should verify the system design, making sure that all system design aspects are correctly implemented. Acceptance testing ,which is conducted by the customer rather than the developer , validates the requirements have been fully implemented before the system is accepted and paid for.

The model's linkage of the left side with the right side of the V implies that if problems are found during verification and validation, then the left side of the V can be re-executed to fix and improve the requirements, design, and code before the testing steps on the right side are reenacted. In other words, the V-model makes more explicit some of the iteration and rework that are hidden in the waterfall depiction. Whereas the focus of the waterfall is often documents and artifacts, the focus of V- model is activity and correctness.

Figure 3.0 : V- Model



3.3 RATIONALE OF METHODOLOGY APPROACH

The suitable and the best model for this project is V-model. It allows software development projects to be automatically controlled in the sense of a workflow – result oriented and with a lot of flexibility. The V-model attempts to give increased importance to testing related activities by dividing the life cycle into 'development' and 'testing' phases. The model relates each development phase to its associated testing phase. Then in this model, work on testing phase is carried out in parallel. For example, acceptance test planning activities of the acceptance-testing phase may be carried out along with the requirement phase. Similarly, system test planning activities of the system-testing phase need not wait for unit testing to be completed. This could be done with the waterfall model because following phase should not start until the previous phase has finished.

The V-model is an extensive collection of knowledge about the Best Practices of software development. In addition to the planned products and activities, the V-model also contains information about the course the project will take. The process standard includes which output products are to be created by an activity and which successor activities need this product as input. This internal product flow allows you to derive a chronological order for the activities. This characteristic of the V-model is the key to automatically controlling a software development in the sense of a workflow. Figure 3.1 depicts the activities involved in the V model.

3.4 FUNCTIONAL REQUIREMENTS

These are the statements of services that the system should provide, how the system should react, to particular inputs and how the system should behave in particular situation. Listed below are the functional requirements involved in the proposed system:-

- User authentication and authorization are very important in order to provide access control against the system. Access controls restrict use of computer system resources to and ensure that users obtain only authentic computer system resources.

- Students who is taking the history subject can access this system to get some notes regarding the subject. They can get some overview on the format that will be used to grade their essays.

- The system can grade the students essay automatically and the marks will given based on the scheme that will be set by the teacher who is conducting the exam.

- The students essays and marks will be saved according to their categories in the server database. Then the system will run a statistics based on their marks. The essays which is saved can be retrieve by the teachers to re-assess the essays, if it is needed.

3.5 NON-FUNCTIONAL REQUIREMENTS

A non-functional requirement or constraint describes a restriction on the system that limits the choice for constructing a solution to the problem. The solutions will narrow the selection of programming language, platform or implementation techniques or tolls. Among the non-functional requirement for the proposed system are:

- Reliability

  A system is considered reliable if it does produce dangerous or costly failures when used in a reasonable manner. A system might be used and the system must be able to handle these situations.

- Accuracy

  Accuracy refers to the precision of the information provided in the homepage it provides various accuracy measures to maintain the accuracy of the information.

- Efficiency

  Efficiency in computer terminology means that enables a system to handle or avoid disaster in the face of unexpected data.

- Robustness

  It refers to the quality that enables a system to handle or avoid disaster in the face of unexpected data.

- Correctness

  Correctness refers to the extent to which a program satisfies its specifications and fulfills the user requirement.

- User friendliness

  The application system is required to have a user –friendly interface for the ease of usage. Graphical User Interface approach should be applied for better visual effect to the user.

- Error handling

  Effective error handling and validation will also help the user to navigate through the system with more confidence. Error message should be displayed to the user to navigate through the system with more confidence. Error message should be display to the user to indicate what is going wrong than for the user to guest what is happening.

- Modularity

  Modularity is a key factor in good program design. The working of the system was broke into modules so that the distinct functions of objects could be isolated from one another. This characteristic makes it easier to perform testing and maintenance.

- Maintainability

  Maintainability can be defined as the ease with which software can be understand, corrected, adapted, and enhanced in the future.

- Expandability

  Expandability measures the degree to which the architecture, data or procedural design can be extended. This system is designed to be expandable in the future.

- Response time

  Web site downloading should be within an acceptable time when retrieving data should be fast enough to make the accessing easy for the web surfer.

3.6 SELECTION OF DEVELOPMENT TOOL

Since the Essay Grading System is web-based system, there are a number of tools that available to develop this system. We must study and analyze the correct tools before implement the tools for developing. The tools also must be support the system modeling during the analysis and design stages of the system process.

In this system there will be a discussion about the two major tools that support this EGS . That is ASP.NET and PHP.

3.6.1 ASP.NET

ASP stands for Application Service Provider, and is sometimes used interchangeably with the term Software as a Service (SaaS).

ASP.NET is the latest Microsoft programming framework that allows for the rapid development of powerful web applications.

In essence, an ASP remotely hosts and manages a software application. It may be helpful to think of an ASP as "renting" a piece of software to customers who access it remotely, most commonly over the Internet or a Wide Area Network (WAN). [23]

The Advantages of ASP

ASP.NET has many advantages, both for the programmers and for the end users.

i. **Powerful database-driven functionality**

ASP.NET allows programmers to develop web application that interface with a database. The advantages of ASP.NET are that it is object-oriented and has many programming tools that allow for faster development and more functionality.

ii. **Faster web applications**

Two aspects of ASP.NET that make it fast is compiled code and caching. In the past, the code was interpreted into "Machine language" when your website visitor viewed your page. Now, with ASP.NET the code is compiled into "machine language" before your visitor ever comes to your site.

Caching is the storage of information that will be reused in a memory location for faster access in the future. ASP.NET allows programmers to set up pages or areas of pages that are commonly reused to be cached for a set period of time to improve the performance of web applications. In addition, ASP.NET allows the caching of data from a database.

iii. **Memory leak and crash protection**

ASP.NET automatically recovers from memory leaks and errors to make sure that your web site is always available to your visitors.

iv. **Multiple language support**

Programmers can actually write their code in more than 25.net languages including VB.Net, C#, and Jscript.Net. This allows programmers to allows programmers to develop your site in the language they know and it means that you can more easily find programmers to support the work on your site.

3.6.2 PHP

PHP stands for PHP: Hypertext Preprocessor. It is a scripting language designed specifically for use on Web pages. PHP is a server side scripting language. A server scripting language has the capability of executing a script on a server and presenting the output as HTML. Server side scripts have the main advantages of interacting directly with server database and performing all types of server manipulations directly.

Client side scripting languages such as JavaScript or Vbscript, act on your local machine by and large and need to pass instruction to the server, rather than directly interacting with it. PHP is a cross platform supported language, which means, PHP can run on various operating systems like Linux, Windows. Both Personal Web Server and Apache (Windows version) can be configured to work with PHP.

Advantages of PHP

i. **Open Source**

PHP is open source, open source is one where, user is given a free license to remodel or recode PHP, according to their wish.

ii. **Multi Platform**

PHP supports various platforms, which mean PHP can be installed on almost all the operating systems, like the window-x and Linux.

iii. **Easy Syntax**

PHP syntax is quite easy to code, all the syntax are similar to the C language syntax, if you are very new to the programming environment then it will be a bit difficult task for you to code the PHP.

3.6.3 Tools Confirmation

By the analysis and survey done on this two main tools, ASP.NET is found to be the best for the development of the EGS.

There are several reasons, why the ASP.NET was chosen:

i. ASP solutions delivered over the Internet are naturally, accessible round-the-clock from any Internet-enabled computer on earth. This means that your content management system is at our fingertips no matter where there are, and no matter what time it is. Content can be added or updated immediately, when it needs to be.

ii. EGS is an online system that helps to assess the students essays and it needs to be updated whenever it is possible. ASP.NET provide a very user-friendly system and easy and fast learning code. So, we can upgrade the system in future without any constraints.

3.7 MICROSOFT ACCESS 2002

Since its introduction in 1992, Microsoft Access has become one of the most versatile applications in the Office suite. This versatility is evidenced by the rich set of tools that even the most experienced database user can take advantage of, while offering the same level of simplicity as the other Office applications for first-time database users[13].

In this new version many functionality has improved to help the developers and experienced users to access and build the powerful databases. The beginners also can easily use the existing applications in Access 2002. It also improves the ability for users to access information from corporate-level, back-end databases such as Microsoft SQL Server. Analysis tools such as PivotTable dynamic views and PivotChart dynamic views has improved and enables users to extend corporate databases applications to the web.

Another important features that Access 2002 developed was the tools that developers needs to build powerful, sophisticated databases solutions that can integrate with enterprise-wide data while ensuring forward and backwards compatibility with new and existing database solutions. They also enhanced tools to build solutions that integrate and leverage internet-standards, such as XML, XSL, and dynamic web pages.

Some more Access provide multinational organizations and multilingual users with a better experience when working the application. This was achieved with enhancements to working with, displaying, and developing international text in databases.

3.8 MINIMUM HARDWARE REQUIREMENTS

i. **Processors**

Intel Pentium II 450Mhz

ii. **RAM**

128 MB

iii. **Hard Disc Space**

100 MB Hard Disk Space

iv. **Display Card**

VGA Card that support 16 bit color at 800x 600 resolution

v. **Operating System**

Windows 2000 Professional/ Windows XP Professional Edition

vi. **CD-ROM Required**

48x CD-ROM Drive

vii. **Monitor**

Standard CRT Monitor

viii. **Internet**

Internet Explorer 5.0 and above

3.9 MINIMUM REQUIREMENTS FOR RUNTIME

i. **Processors**

Pentium 133Mhz

ii. **RAM**

32 MB

iii. **Display Card**

VGA Card that support 16 bit color at 800x 600 resolution

iv. **Monitor**

Standard CRT Monitor

v. **Internet**

Internet Explorer 5.0 and above

CHAPTER 4

SYSTEM ANALYSIS AND DESIGN

CHAPTER 4

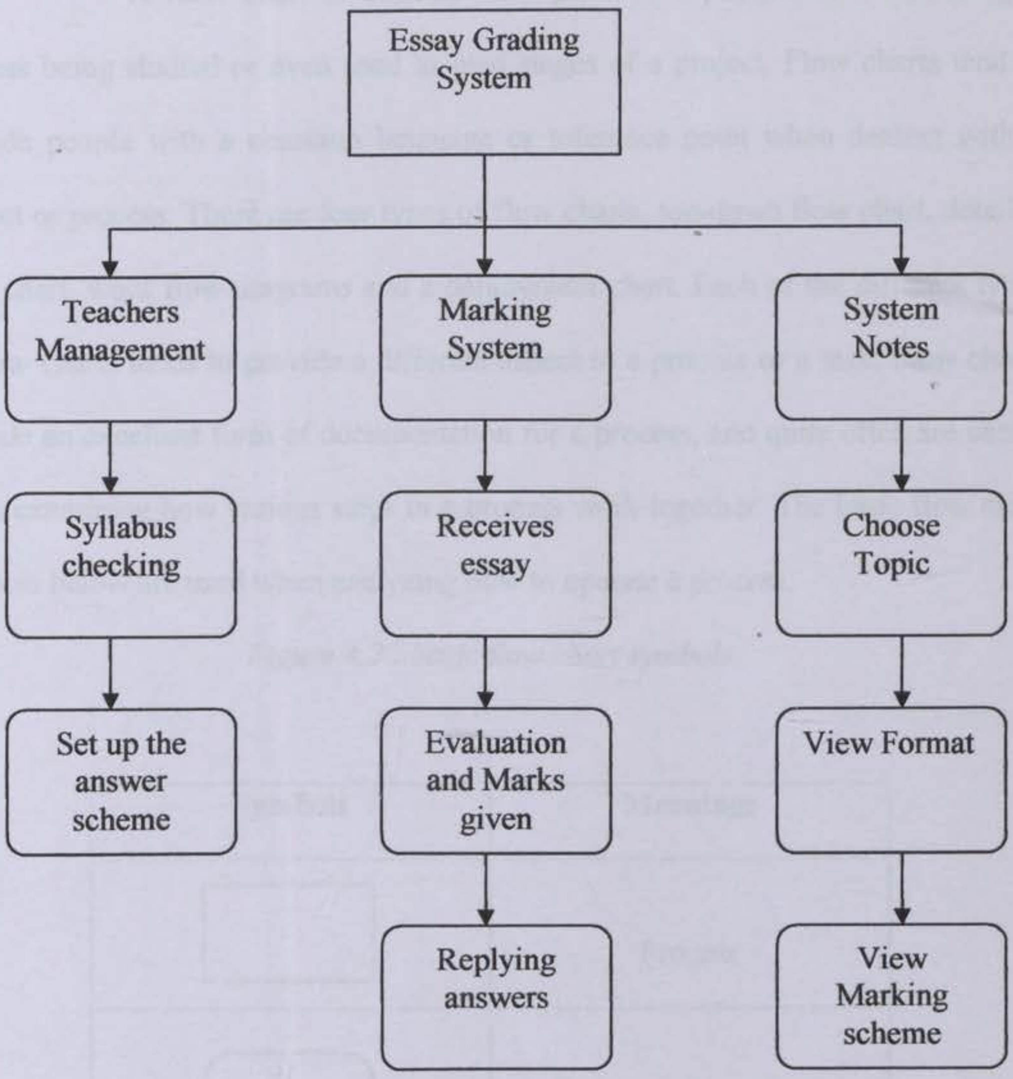
SYSTEM ANALYSIS AND DESIGN

In this chapter, I will cover system design phase in model driven methodology. It is the fifth phase after decision analysis phase. In this phase, all user requirements will be transformed into real world application, which will be developed conceptually or logically. It is a process of transforming problem into solution. Whereas system analysis emphasized the system problem, system design focuses on the technical and implementation concerns of the system.

4.1 SYSTEM HIERARCHY

A system hierarchy is drawn to partition the system into logical subsystems or functions. Moreover, system hierarchy will also be used to determine the appearance frequency of smaller processes. Functional analysis with modeling tools provides important details that will be used often in later stages of analysis. However, it is necessary to note that function approach to issue is not a comprehensive approach. It only shows the functions do not how to do. In a system hierarchy, a function is divided into many smaller functions and each function contains smaller ones. Constructing diagram is a process of division, from higher functions to appropriate smaller functions. The following figure depicts a system hierarchy for the essay grading system.

Figure 4.1 : System Hierarchy



4.2 FLOW CHART OF SYSTEM

A flow chart is defined as a pictorial representation describing a process being studied or even used to plan stages of a project. Flow charts tend to provide people with a common language or reference point when dealing with a project or process. There are four types of flow charts, top-down flow chart, detailed flow chart, work flow diagrams and a deployment chart. Each of the different types of flow charts tends to provide a different aspect to a process or a task. Flow charts provide an excellent form of documentation for a process, and quite often are useful when examining how various steps in a process work together. The basic flow chart symbols below are used when analyzing how to operate a process.

Figure 4.2 : basic flow chart symbols

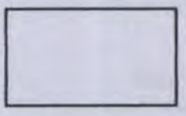
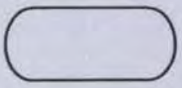
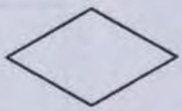
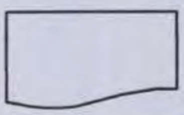
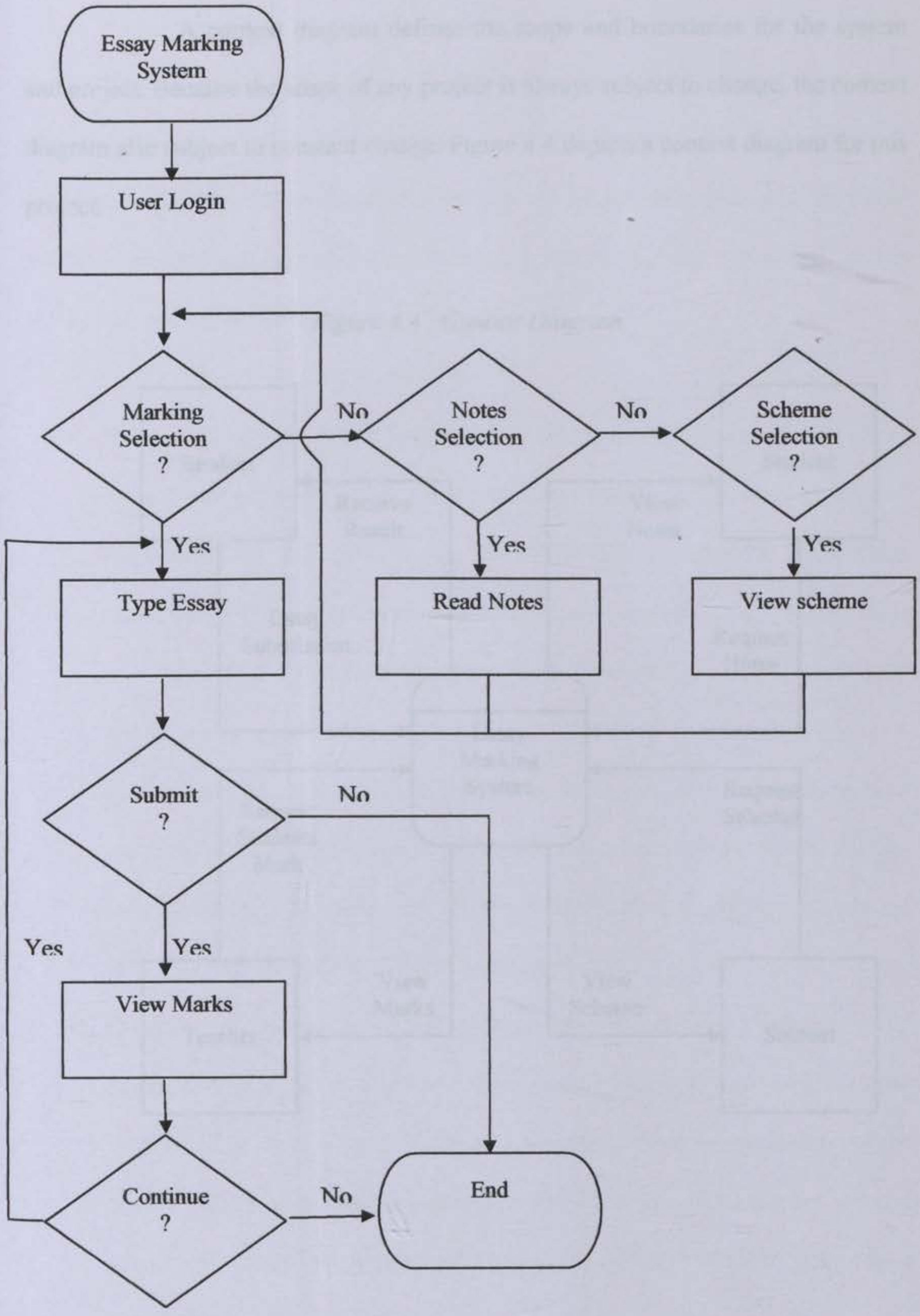
Symbols	Meanings
	Process
	terminator
	Decision
	document

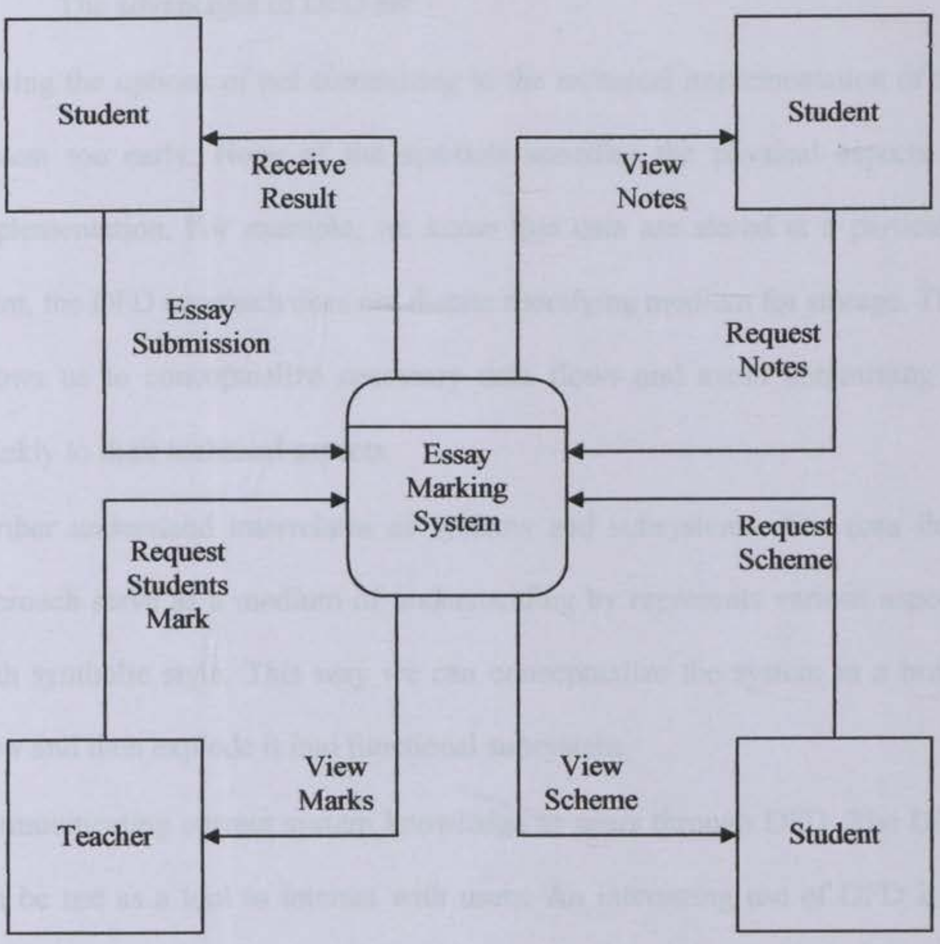
Figure 4.3 : Systems Flow chart



4.3 CONTEXT DIAGRAM

A context diagram defines the scope and boundaries for the system and project. Because the scope of any project is always subject to change, the context diagram also subject to constant change. Figure 4.4 depicts a context diagram for this project.

Figure 4.4 : Context Diagram



4.4 DATA FLOW DIAGRAM

Through a data flow diagram (DFD), we can put together a graphical representation of data processes throughout the application. The DFD approach emphasizes the logic under laying the system. By using combinations of only four symbols, we can create a pictorial depiction of processes that will eventually provide solid system documentation.

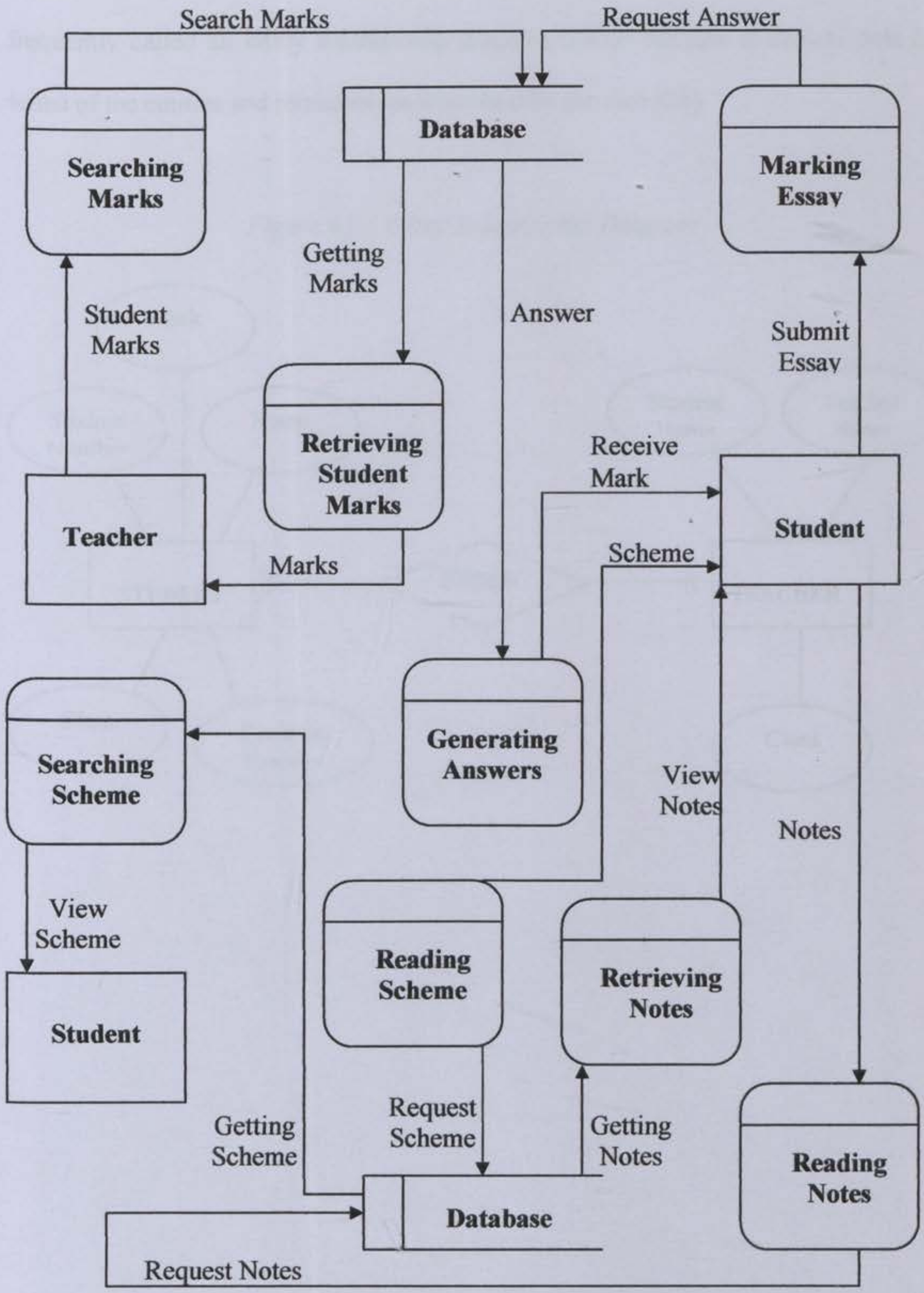
The advantages of DFD are:-

- Having the options of not committing to the technical implementation of the system too early. None of the symbols specifies the physical aspects of implementation. For example, we know that data are stored at a particular point, the DFD approach does not dictate specifying medium for storage. This allows us to conceptualize necessary data flows and avoid committing to quickly to their technical aspects.
- Further understand interrelates of systems and subsystems. The data flow approach serve as a medium of understanding by represents various aspects with symbolic style. This way we can conceptualize the system in a broad view and then explode it into functional subsystem.
- Communicating current system knowledge to users through DFD. The DFD can be use as a tool to interact with users. An interesting use of DFD is to show them to users as incomplete representation of our understanding of the system. Users can then be asked to comment on the accuracy of the conceptualization and changes can be made to reflect the system from user's perspectives.

- Analysis of a proposed system to determine if the necessary data and process have been defined. DFD can allow us to describe each component in a system. Analysis can then be performed to ensure that all necessary output may be obtained from the input data and that processing logic is reflected in the diagram. This will help in detecting and correcting error and design flow in earlier stage of system development phase.

The following figure depicts the Data Flow Diagram of the automated essay grading system.

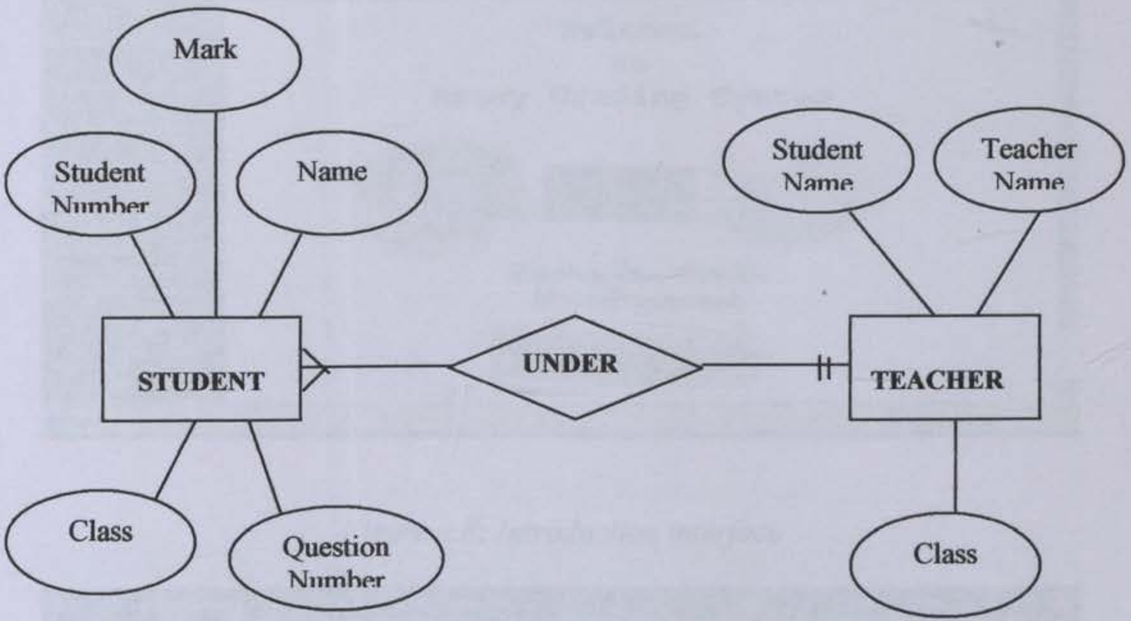
Figure 4.5 : Data Flow Diagram



4.5 DATABASE DESIGN

There are several notations for data modeling. The actual model is frequently called an entity relationship diagram (ERD) because it depicts data in terms of the entities and relationships described by the data [24].

Figure 4.6 : Entity Relationship Diagram



4.6 USER INTERFACE DESIGN

Figure 4.7: Main Menu Interface

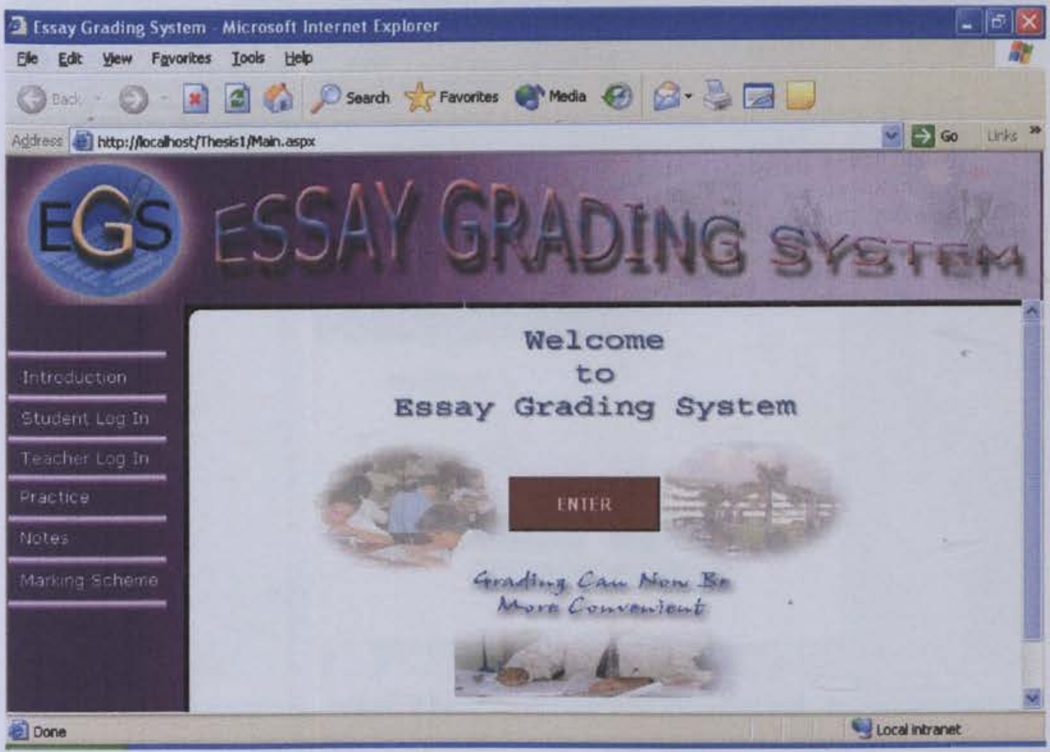
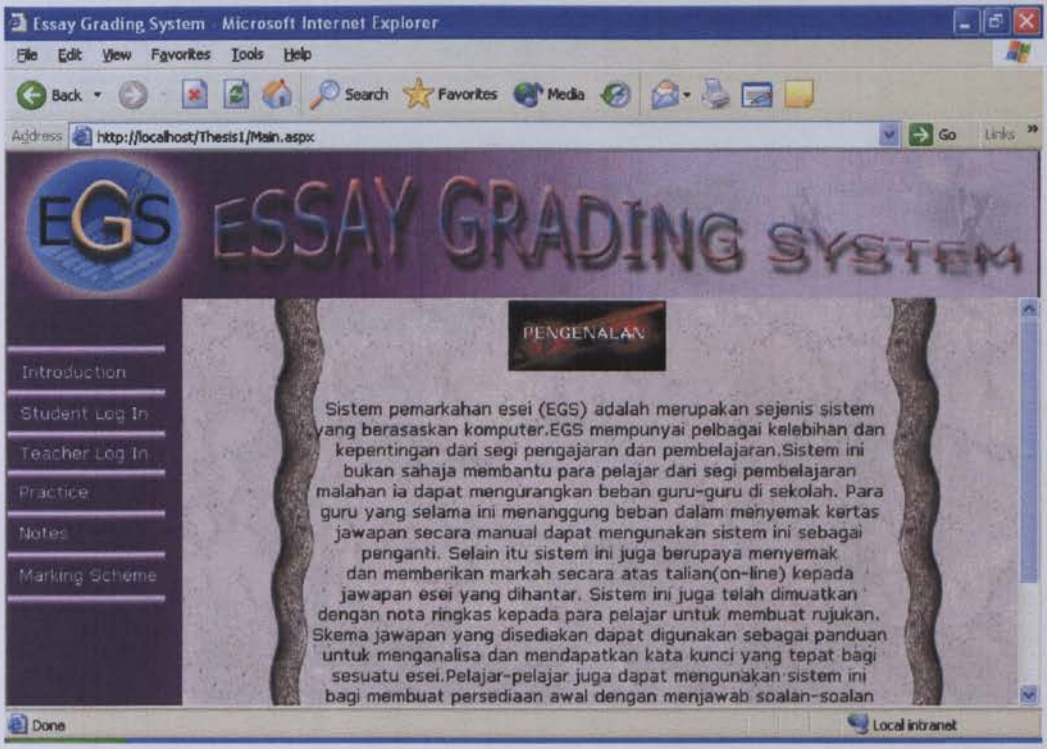

Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Welcome to Essay Grading System
ENTER
Grading Can Now Be More Convenient
Done
Local intranet

Figure 4.8: Introduction interface


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Done
Local intranet

PENGENALAN

Sistem pemarkahan esei (EGS) adalah merupakan sejenis sistem yang berasaskan komputer.EGS mempunyai pelbagai kelebihan dan kepentingan dari segi pengajaran dan pembelajaran.Sistem ini bukan sahaja membantu para pelajar dari segi pembelajaran malahan ia dapat mengurangkan beban guru-guru di sekolah. Para guru yang selama ini menanggung beban dalam menyemak kertas jawapan secara manual dapat mengunakan sistem ini sebagai penganti. Selain itu sistem ini juga berupaya menyemak dan memberikan markah secara atas talian(on-line) kepada jawapan esei yang dihantar. Sistem ini juga telah dimuatkan dengan nota ringkas kepada para pelajar untuk membuat rujukan. Skema jawapan yang disediakan dapat digunakan sebagai panduan untuk menganalisa dan mendapatkan kata kunci yang tepat bagi sesuatu esei.Pelajar-pelajar juga dapat mengunakan sistem ini bagi membuat persediaan awal dengan menjawab soalan-soalan

Figure 4.9: Student login interface

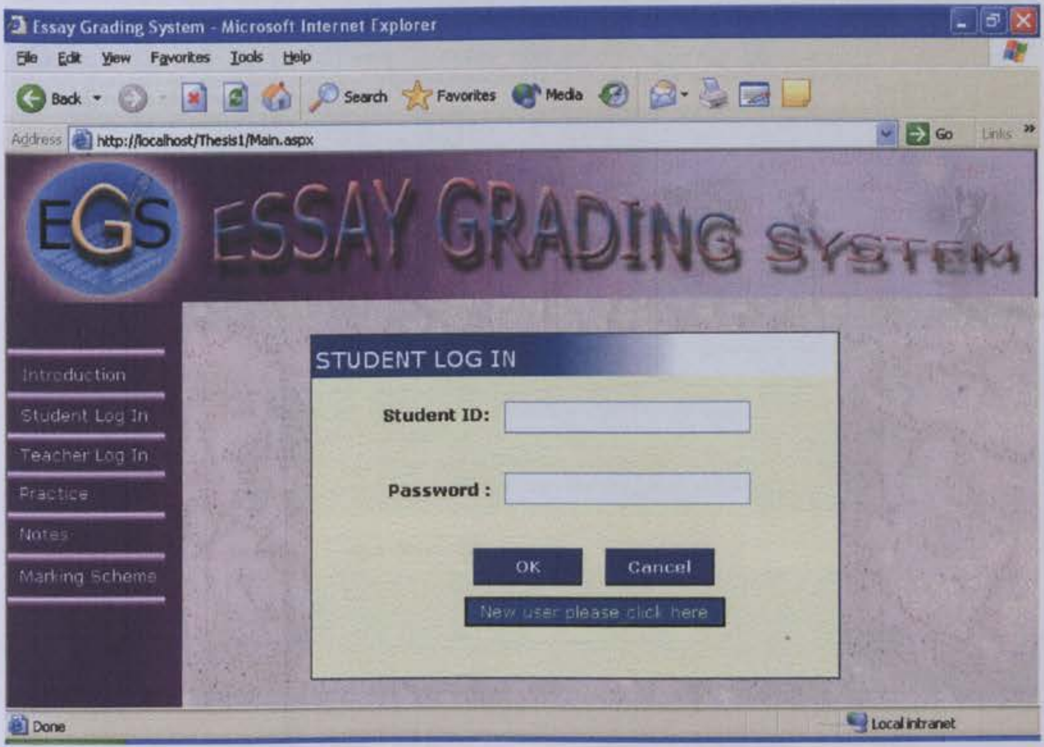

Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
STUDENT LOG IN
Student ID:
Password :
OK
Cancel
New user please click here
Done
Local intranet

Figure 4.10: Teachers login interface

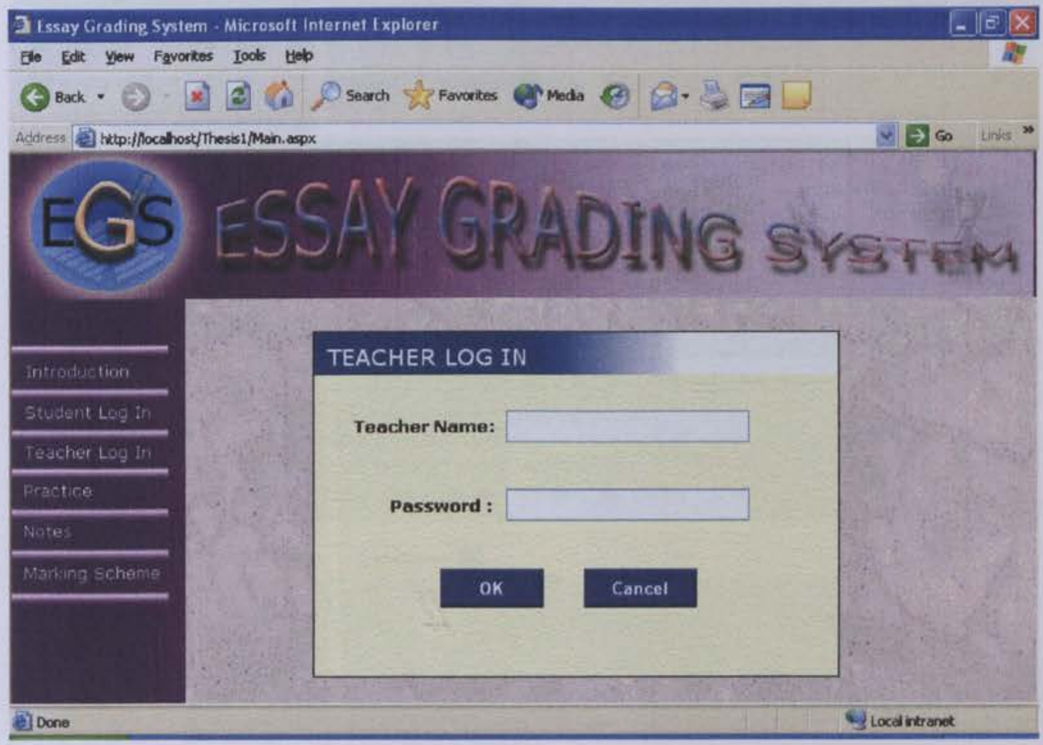
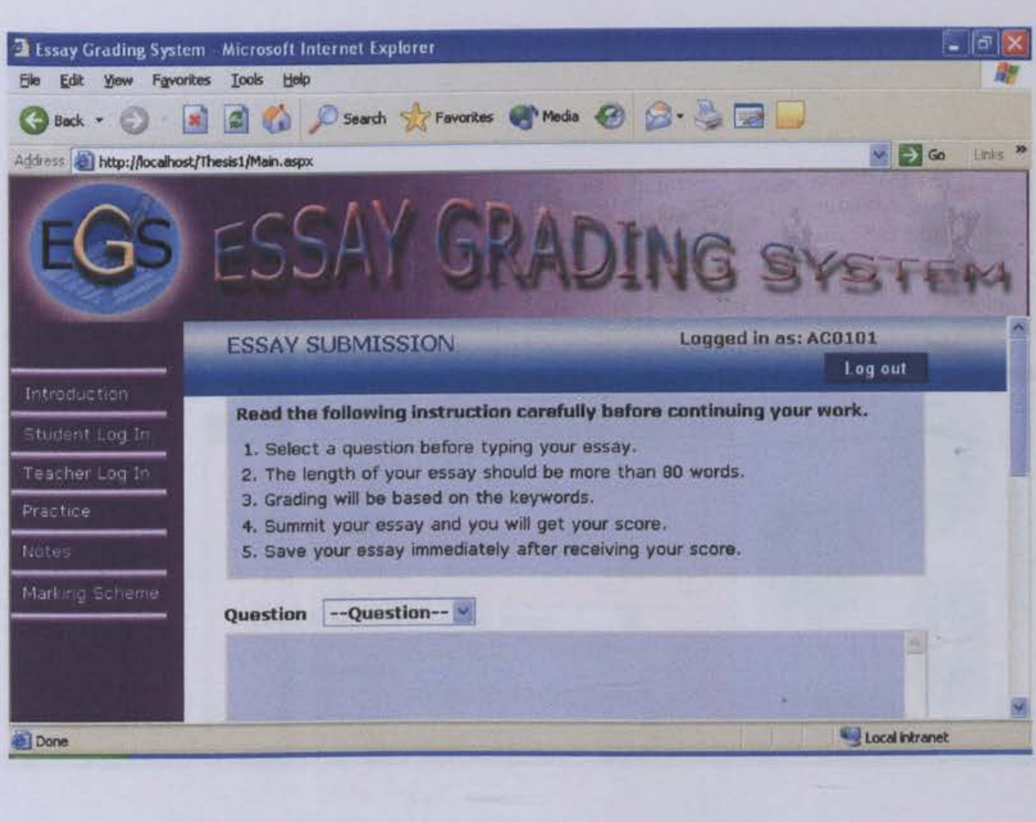


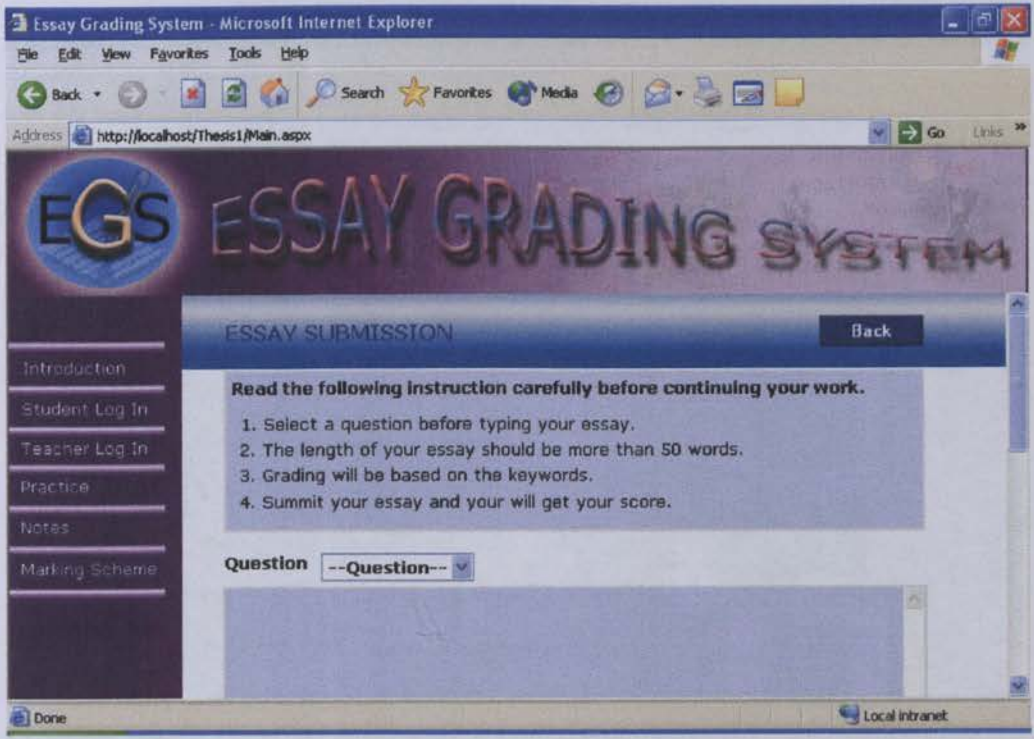
Figure 4.11: Essay Submission interface



Read the following instruction carefully before continuing your work.

1. Select a question before typing your essay.
2. The length of your essay should be more than 80 words.
3. Grading will be based on the keywords.
4. Summit your essay and you will get your score.
5. Save your essay immediately after receiving your score.

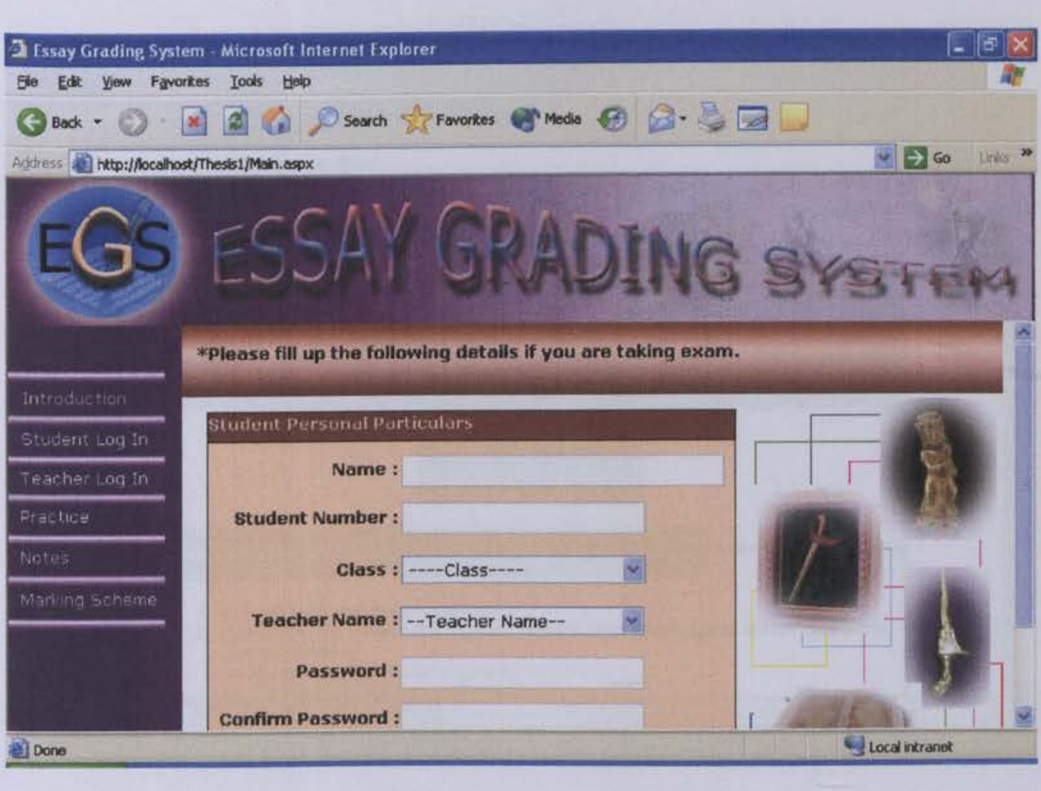
Figure 4.12: Essay Practice interface



Read the following instruction carefully before continuing your work.

1. Select a question before typing your essay.
2. The length of your essay should be more than 50 words.
3. Grading will be based on the keywords.
4. Summit your essay and your will get your score.

Figure 4.13: Students Registration Interface


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Done
Local intranet

*Please fill up the following details if you are taking exam.

Student Personal Particulars

Name :

Student Number :

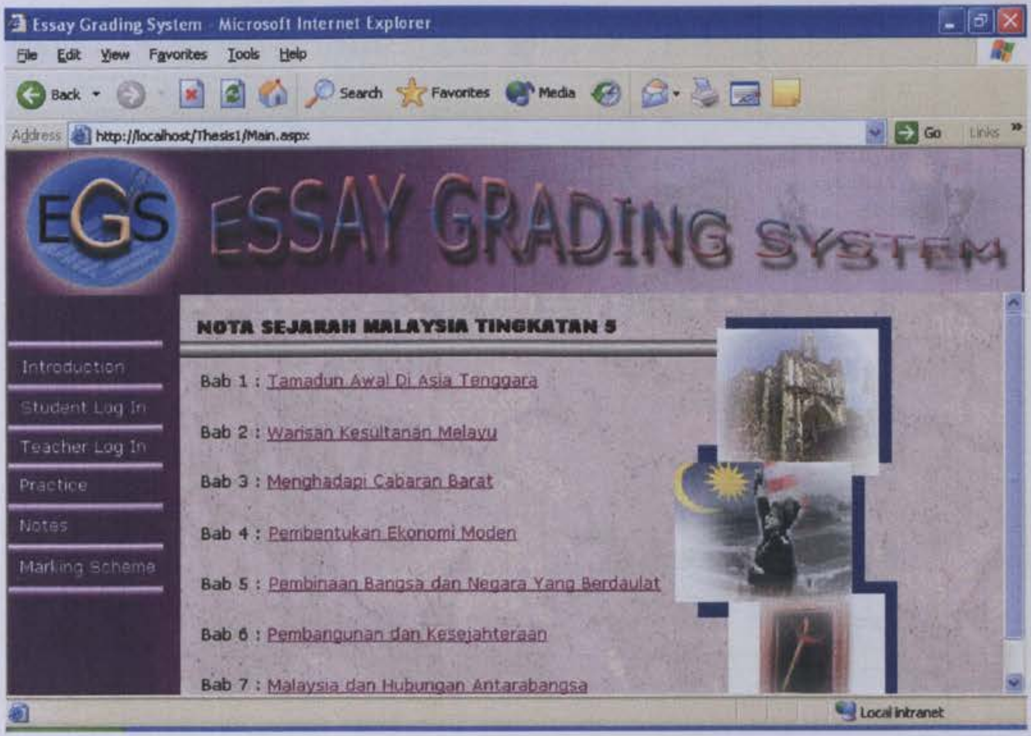
Class : ----Class----

Teacher Name : --Teacher Name--

Password :

Confirm Password :

Figure 4.14: Notes Selection Interface


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Local intranet

NOTA SEJARAH MALAYSIA TINGKATAN 5

Bab 1 : Tamadun Awal Di Asia Tenggara

Bab 2 : Warisan Kesultanan Melayu

Bab 3 : Menghadapi Cabaran Barat

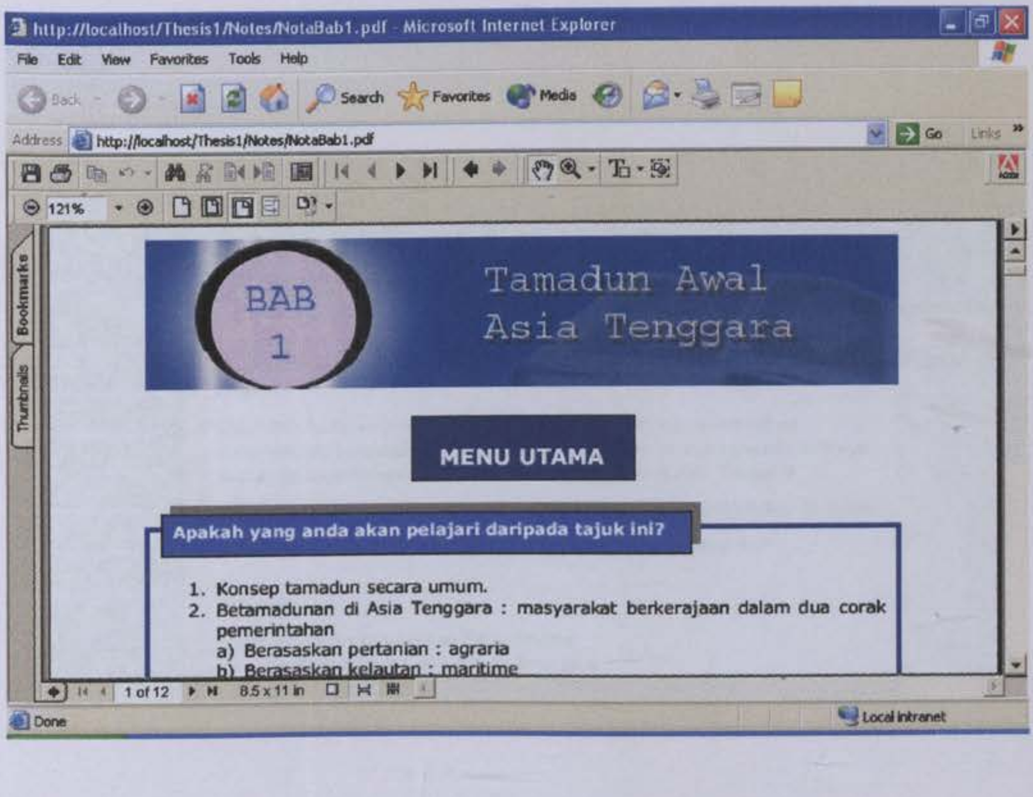
Bab 4 : Pembentukan Ekonomi Moden

Bab 5 : Pembinaan Bangsa dan Negara Yang Berdaulat

Bab 6 : Pembangunan dan Kesejahteraan

Bab 7 : Malaysia dan Hubungan Antarabangsa

Figure 4.15: Notes Interface


http://localhost/Thesis1/Notes/NotaBab1.pdf - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Notes/NotaBab1.pdf
Go Links
121%
Bookmarks
Thumbnails
1 of 12
8.5 x 11 in
Done
Local intranet

BAB 1

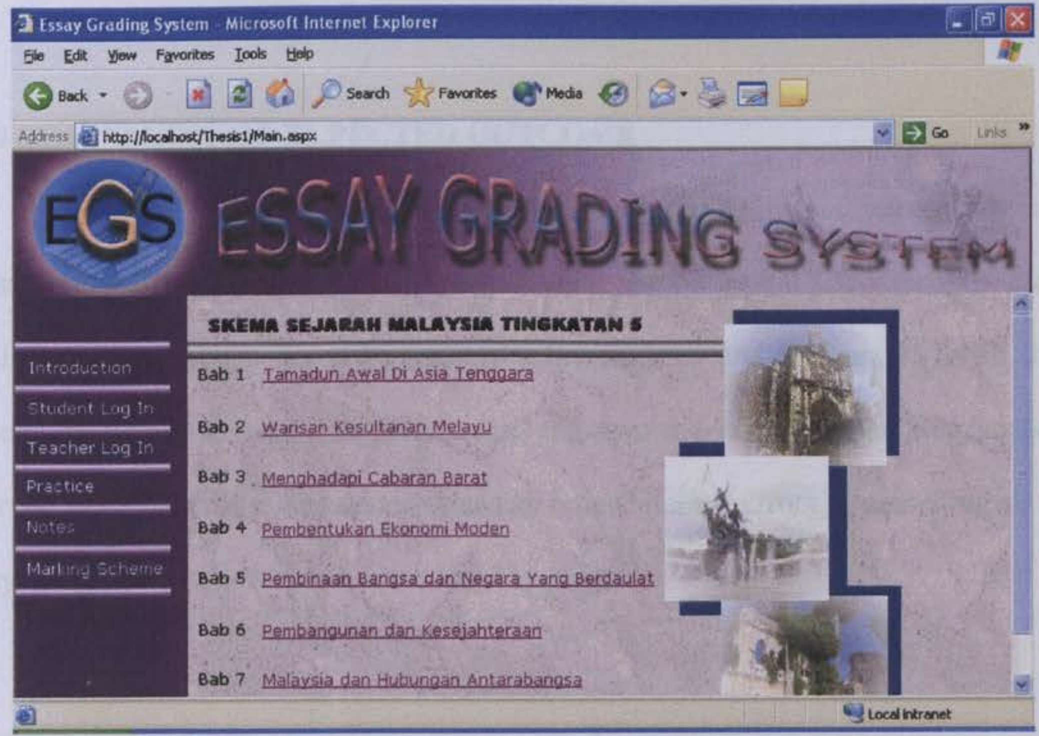
Tamadun Awal Asia Tenggara

MENU UTAMA

Apakah yang anda akan pelajari daripada tajuk ini?

1. Konsep tamadun secara umum.
2. Betamadunan di Asia Tenggara : masyarakat berkerajaan dalam dua corak pemerintahan
   a) Berasaskan pertanian : agraria
   b) Berasaskan kelautan : maritime

Figure 4.16: Scheme Selection Interface


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS
ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Local intranet

SKEMA SEJARAH MALAYSIA TINGKATAN 5

Bab 1 Tamadun Awal Di Asia Tenggara

Bab 2 Warisan Kesultanan Melayu

Bab 3 Menghadapi Cabaran Barat

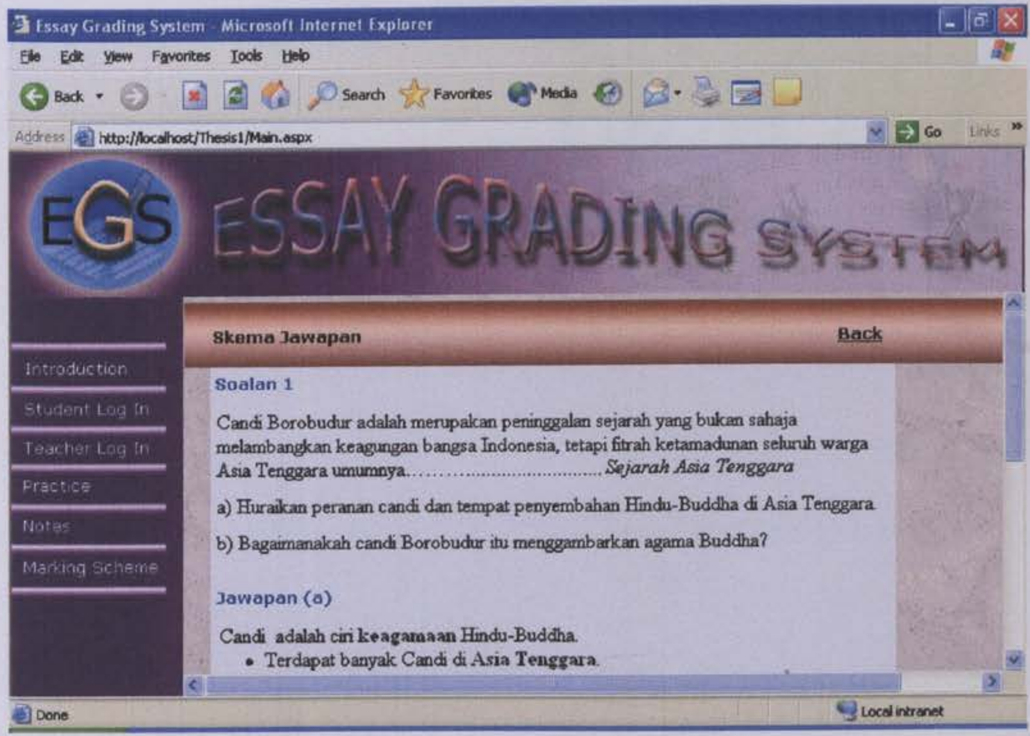
Bab 4 Pembentukan Ekonomi Moden

Bab 5 Pembinaan Bangsa dan Negara Yang Berdaulat

Bab 6 Pembangunan dan Kesejahteraan

Bab 7 Malaysia dan Hubungan Antarabangsa

Figure 4.17: Scheme Interface


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS
ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Done
Local intranet

Skema Jawapan

Back

Soalan 1

Candi Borobudur adalah merupakan peninggalan sejarah yang bukan sahaja melambangkan keagungan bangsa Indonesia, tetapi fitrah ketamadunan seluruh warga Asia Tenggara umumnya.................................*Sejarah Asia Tenggara*

a) Huraikan peranan candi dan tempat penyembahan Hindu-Buddha di Asia Tenggara

b) Bagaimanakah candi Borobudur itu menggambarkan agama Buddha?

Jawapan (a)

Candi adalah ciri keagamaan Hindu-Buddha.

- Terdapat banyak Candi di Asia Tenggara.

4.7 STATEMENT OF EXPECTED OUTCOME

The expected outcome of the automated essay grading system is to come out with a system which can mark students' essays within a few seconds. Since it is an online system it can be access by the students and teachers to fulfill their needs on history subject. The system also can save the students' marks in database for teacher's reference. The system also can reduce human errors by searching all the keywords one by one without leave any key points.

CHAPTER 5

SYSTEM IMPLEMENTATION

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 INTRODUCTION OF SYSTEM IMPLEMENTATION

Usually in prototype model, the requirement analysis, system analysis, system design and implementation phases do not have a clear boundary. Each phase tends to develop one another. After the system design phase on how the system should be functioning, the next process will involve the implementation (development and coding) phase. This phase is an important element for every project. The design phase earlier in the system life cycle is directed towards a final objective, which is to translate the concept of the system into a software representation that is understood by the computer. The coding process involves transforming of the design into a programming language.

The effort spent in this phase will actually determines the success of the system and ease the process of modifications, debugging, testing, verifications, system integration and for future enhancement. The system implementation of Essay Grading System will be divided into three components, which are the development environment, debugging and implementation.

5.2 DEVELOPMENT ENVIROMENT

Development environment plays a major role in determine the speed of developing the system. The combination of software and hardware used to develop the system are as follow:

5.2.1 Hardware used

- Intel Celeron Processor 600 MHz
- Random Access Memory (RAM) 192 MB
- Hard Disk 15 GB
- CD-RW Drive
- CD-ROM Drive
- Floppy Drive 1.44MB
- Other standard desktop PC accessories.

5.2.2 Software used

- Operating System: Windows XP Professional Edition
- Microsoft Visual Studio .NET (ASP.NET)
- Microsoft Access 2002
- Web Server : Internet Information Service (IIS)
- Adobe Photoshop 7.0
- Ulead PhotoImpact 8.0
- Microsoft Paint

5.3 CODING METHODOLOGY

Coding methodology is a process of transformation of design specification into source code that could run on computers. This coding methodology is important in the sense of procedures and standards needs for a project. It prevents human errors by maintaining relations between designing and coding. Bottom-up coding approach was implemented in this project coding. Bottom-up approach is the coding which will start from the small coding and then merge all the coding to be applicable in a bigger program. It helps to prevent a lot of errors and missing functions during coding within modules.

5.4 PROGRAMME DEVELOPMENT AND CODING

Program development is the process of creating the programming needed to satisfy an information system's processing requirement. Developing and coding is the phase which takes the longest time in the development life cycle. Therefore, using the right tool and the right way to develop the system are crucial in determining the success of project. The programming language that is used to develop EGS includes ASP.NET (Visual basic language) and Hypertext Markup Language (HTML). Before starting on the coding process or any other detailed works on the program, a review on the program documentation needs to be done followed by design of the program and finally going into the program coding process.

5.4.1 Program Documentation Review

The first and foremost step to be taken in program development phase is to review the program documentation that was proposed during the earlier phases. The program documentation was proposed in the system design phase of EGS consist of architecture view, concepts and controls, module flow diagram and also the sample layout of the interface.

5.4.2 Program Designing

After reviewing the program documentation, designing the program is the next following process. For this phase, determining how the program can accomplish the features and functions that are described in the program documentation and developing a logical solution to the logic of the program is a step by step solution to the programming problems.

5.4.3 Coding

Coding style is an important attribute of source code. An easy to read source code makes the system easier to maintain and enhance. While doing coding, we need to take some elements into considerations such as internal documentation, standard naming convention and standard graphical user interface. Internal documentation is achieved by using comments while coding, providing clear guide to programmers for future enhancement. Statements of purpose indicating the functions of modules and descriptive comment are embedded into source code to describe the processing functions. A standard naming convention and also a standard usage of graphical user interface components is employed in developing the system marking. Standard naming convention provides programmers with easy identification of variables. While not generate much surprise to them. Usages of these standards perform as a mean towards coding consistency and standardization.

5.5 SYSTEMS CODING

The system coding was done by using code behind to protect the code from users view. Declaration of the buttons, tables, and hyperlinks was put in the HTML designing code. The following codes are the code behind for the systems tasks.

Codes for Student Login:

```
Private Sub btnok_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnok.Click
        lblError.Text = ""
        lblStudentName.Text = ""
        lblCheckPassword.Text = ""
        If txtname.Text = "" Then
            lblStudentName.Text = "Username not entered"
            Exit Sub
        End If

        If txtpassword.Text = "" Then
            lblCheckPassword.Text = "Password not entered"
            Exit Sub
        End If

        Dim boolRedirect As Boolean

        Dim selectSQL As String
        selectSQL = "SELECT * FROM TeachersTable"

        Dim dbConnection As New OleDbConnection(ConnectionString)
        Dim dbCommand As New OleDbCommand(selectSQL, dbConnection)
        Dim dbReader As OleDbDataReader

        Try
            dbConnection.Open()
            dbReader = dbCommand.ExecuteReader
            dbReader.Read()

            Do While dbReader.Read()
                If dbReader("StudentsNumber") = txtname.Text Then
                    If dbReader("Password") = txtpassword.Text Then
                        boolRedirect = True
                        Exit Do
                    Else
                        Exit Do
                    End If
                End If
            Loop

            If boolRedirect = False Then
                lblError.Text = "Invalid username or password"
                txtname.Text = ""

                txtpassword.Text = ""
                dbReader.Close()
                dbConnection.Close()
                Exit Sub
            End If
        selectSQL = ""
        dbReader.Close()

    Catch err As Exception
        lblError.Text = "Error getting data. "
        lblError.Text &= err.Message
    Finally
        If (Not dbConnection Is Nothing) Then
            dbConnection.Close()
            dbConnection.Dispose()
        End If
    End Try
    If boolRedirect = True Then

        Server.Transfer("SubmitF.aspx?ID=" & _
            HttpUtility.UrlEncode(txtname.Text))
    End If
End Sub

Private Sub btncancel_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btncancel.Click
    Response.Redirect("WelcomeForm.aspx")
End Sub

Private Sub btnNew_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnNew.Click
    Response.Redirect("StudentDetail.aspx")
End Sub
```

Codes for Teachers Login:

```
Private Sub btnok_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnok.Click
        If txtname.Text = "" Then
            lblCheckName.Text = "*Administrator name is required!"
            Exit Sub
        End If

        If txtpassword.Text = "" Then
            lblCheckPassword.Text = "*Password is required!"
            Exit Sub
        End If

        Dim DBConn As OleDbConnection
        Dim DBCommand As OleDbDataAdapter
        Dim DSLogin As New DataSet()
        DBConn = New
        OleDbConnection("PROVIDER=Microsoft.Jet.OLEDB.4.0;" _
            & "DATA SOURCE=" _
            & Server.MapPath("/Thesis1/" _
            & "AdminLogindb.mdb;"))
```

```
    DBCommand = New OleDbDataAdapter _
        ("Select AdminName from " _
        & "AdminLoginTable Where " _
        & "AdminName = '" & txtname.Text _
        & "' and Password = '" & txtpassword.Text _
        & "'", DBConn)
    DBCommand.Fill(DSLogin, "AdminLoginTable")
    If DSLogin.Tables("AdminLoginTable"). _
        Rows.Count = 0 Then
        lblError.Text = "The Administrator name and password " _
            & "were not found. Please try again."
    Else
        Session("AdminName") =
        DSLogin.Tables("AdminLoginTable"). _
            Rows(0).Item("AdminName")
        Session("AdminName") = txtname.Text
        Response.Redirect("./TeachersAccess.aspx")
    End If
End Sub

Private Sub btncancel_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btncancel.Click
    Response.Redirect("./WelcomeForm.aspx")
End Sub
```

Codes for Essay Submission :

```
Private Sub btnSubmit_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSubmit.Click
    Dim strContent As String
    strContent = txtAnswer.Text
    lblResults.Text = ""

    Dim x
    Dim i
    Dim currentLetter
    Dim prevLetter
    Dim total

    strContent = Trim(txtAnswer.Text)
    If strContent = "" Then
        lblResults.Text = "There are no words in the text box!"
        Exit Sub

    End If

    x = Len(strContent)
    total = 1

    For i = 2 To x

        currentLetter = Mid(strContent, i, 1)
        prevLetter = Mid(strContent, i - 1, 1)

        If Char.IsWhiteSpace(currentLetter) = True And
           Char.IsWhiteSpace(prevLetter) = False Then
            total = total + 1
        End If
```

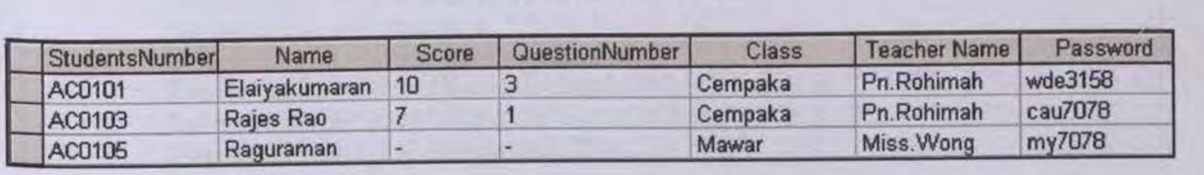
```
 Next

   If total > 80 Then

       If ddlquestion.SelectedItem.Value = 1 Then
  AnswerPath = Server.MapPath("\Thesis1\jawapan\jawapan1.txt")
       ElseIf ddlquestion.SelectedItem.Value = 2 Then
  AnswerPath = Server.MapPath("\Thesis1\jawapan\jawapan2.txt")
        .
        .
        .
        .
       ElseIf ddlquestion.SelectedItem.Value = 14 Then
  AnswerPath = Server.MapPath("\Thesis1\jawapan\jawapan14.txt")
       End If

       Dim contents As String
       Dim contentsPosition, contentsTemp, contentsValue
       Try
           Dim Words
           Dim Counter
           Dim Score
           Dim Content

           Words = Split(txtAnswer.Text)
           Score = 0
           objStreamReader = File.OpenText(AnswerPath)
           Content = objStreamReader.ReadToEnd()
           objStreamReader.Close()

           For Counter = 0 To UBound(Words) - 1
      While InStr(1, Content, Words(Counter), vbTextCompare) > 0

           Content = Replace(Content, Words(Counter), "", 1, 1,
                     vbTextCompare)
                   Score = Score + 1
               End While
           Next
           If Score = 0 Then
               lblResults.Text = "You score is zero"
           End If
           lblDisplayScore.Text = Score
           Content = ""


       Catch err As Exception
           lblResults.Text = "Error reading data. "
           lblResults.Text &= err.Message
       End Try

   Else
       lblResults.Text = "The number of words
                                 not within marking range"
   End If

End Sub .
```

5.6 DATABASE DEVELOPMENT

The EGS uses a database to store users (student) records and administrators (teacher) password. Questions and answers are kept in .txt file, which is considered as database. This is due to the maintenance ease. Questions and answers can be changed easily without any difficulty and the storage space is very big. However, before the system can be developed, the database has to first be completed. This is because algorithms on how the system retrieves data and from which column and record have to be figured out.

a) Students record table

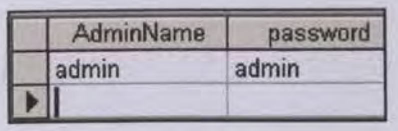
Table 5.0 Students tables



StudentsNumber	Name	Score	QuestionNumber	Class	Teacher Name	Password
AC0101	Elaiyakumaran	10	3	Cempaka	Pn.Rohimah	wde3158
AC0103	Rajes Rao	7	1	Cempaka	Pn.Rohimah	cau7078
AC0105	Raguraman	-	-	Mawar	Miss.Wong	my7078

b) Administrators table

Table 5.1 Administrators tables



AdminName	password
admin	admin

5.6.1 CONNECTING TO THE DATABASE

The database is the main storage of the system where dynamic data is stored on the database can be manipulated. First however the page has to be linked to the database. This is done with ASP.NET coding, the database would first be needed to be linked to the system DSN and the connection and record set has to be defined. It is then ready to be linked to the page. The coding is as follows:-

Adding information to the database :

Students record need to captured and stored for further processing. Students will enter their personal particulars such as student number, name, password and etc. After they have submitted their essays, they will get scores and finally they need to save their question number and the score to the database. Below are the codes for inserting information into a database table:

```
InsertSQL = "INSERT INTO TeachersTable ("
InsertSQL &= "[StudentsNumber], [Name], [Score], "
InsertSQL &= "[QuestionNumber],[Teacher Name], [Class],
              [Password]) "
InsertSQL &= "VALUES ('"

InsertSQL &= txtIdNumber.Text & "', '"
InsertSQL &= txtName.Text & "', '"
InsertSQL &= Score & "', '"
InsertSQL &= QuestionNumber & "', '"
InsertSQL &= ddlTName.SelectedItem.Text & "', '"
InsertSQL &= ddlClass.SelectedItem.Text & "', '"
InsertSQL &= txtPassword.Text & "')"

Dim selectSQL As String
selectSQL = "SELECT [StudentsNumber] FROM TeachersTable"
```

Some of the student's information needs to be updated or removed from the student's database. Deleting and updating student's information done by the teachers. Only teachers are allowed to view, update and delete the student's information. Below are the codes for deleting student's information from the database.

```
        Dim ID As String
        ID = CType(e.Item.FindControl("lblID"), Label).Text

        Dim Delete As String
        Delete = "DELETE FROM TeachersTable "
        Delete &= "WHERE [StudentsNumber] = '" & ID & "'"

        Dim dbConnection As New OleDbConnection(ConnectionString)
        Dim dbCommand As New OleDbCommand(Delete, dbConnection)
```

The following codes are used to update student's information in database.

```
        Dim NewStudentsNumber, NewName As String
        Dim NewScore, NewQuestionNumber As String

        NewName = CType(e.Item.FindControl("txtName"), TextBox).Text
        NewScore = CType(e.Item.FindControl("txtScore"),
TextBox).Text
        NewQuestionNumber =
CType(e.Item.FindControl("txtQuestionNumber"), TextBox).Text

        Dim ID As String
        ID = CType(e.Item.FindControl("lblID"), Label).Text


        Dim Update As String
        Update = "UPDATE TeachersTable SET "
        Update &= "[Name] = '" & NewName & "' , "
        Update &= "[Score] = '" & NewScore & "' , "
        Update &= "[QuestionNumber] = '" & NewQuestionNumber & "' "
        Update &= "WHERE [StudentsNumber] = '" & ID & "'"

        Dim dbConnection As New OleDbConnection(ConnectionString)
        Dim dbCommand As New OleDbCommand(Update, dbConnection)
```

5.7 SUMMARY

System implementation is the phase in which the entire system is coded; this is perhaps the longest, most tedious and difficult, yet the important stage amongst all the other stages. Coding has to be carefully done to ensure that the system operates at full functionality and is reliable. The system was coded in ASP.NET using Visual Basic and HTML language. Database was developed with Microsoft Access 2002.

CHAPTER 6

SYSTEM TESTING

CHAPTER 6

SYSTEM TESTING

6.1 INTRODUCTION TO SYSTEM TESTING

Testing is done to ensure that the system works properly and to introduce techniques that can be used to test programs to discover program faults. Testing provides a method to discover logical error and to test the system reliability. Testing is done throughout system development, not merely at the end of system development. If a system failed after installation will result a waste in cost, time and effort. However successful testing will result in quality software with less errors and work according to specification.

6.2 TESTING OBJECTIVE

The objective of system testing is as follow:

- To identify, isolate and correct as many bugs as possible. Most of the programs have bugs, the most insidious of which appear only with unique combinations of data or events.
- To demonstrate that functionality of the system appears to be working properly and performance requirements appear to be met.
- To do the component testing and integration testing.

6.3 TEST CASE DESIGN

Any system can be tested using one of three types of test case design. They are Ad Hoc testing, white-box testing and black-box testing.

3.6.1 AD-HOC TESTING

Ad-Hoc or ad lib testing means simply examine the functionality units, trying whatever possible in attempt to make the system fail. This type of testing approach is a fast and efficient way of debugging code errors during the early development stage. During this stage of testing, EGS has found many errors that could be solved. The disadvantage of ad-hoc testing is that it usually finds many errors and never be sure what are or what are not to be tested.

6.3.2 WHITE-BOX TESTING

White-box testing is carried out at the early stage of the testing process. It is performed to ensure that the internal operation of a system performs according to specifications and all internal components have been adequately exercised. Using these methods, a system engineer can derive cases that:

1. Guarantee that all independent paths within a model have been exercised at least once.
2. Exercise all logical decisions on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.

4. Exercise internal data structures to assure their validity.

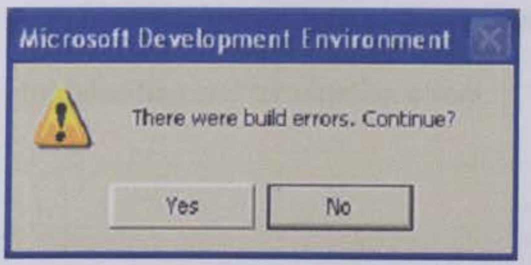
6.3.3 BLACK-BOX TESTING

Functional or black-box testing is an approach to testing where the test is derived from the program or component specification. This testing is conducted to demonstrate that each is fully operational, at the same time searching for errors in each function. This testing approval enables a system engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black-box testing is not an alternative to white-box testing. Rather it is a complementary approach that is likely to uncover a different class or errors than those uncovered by white-box methods.

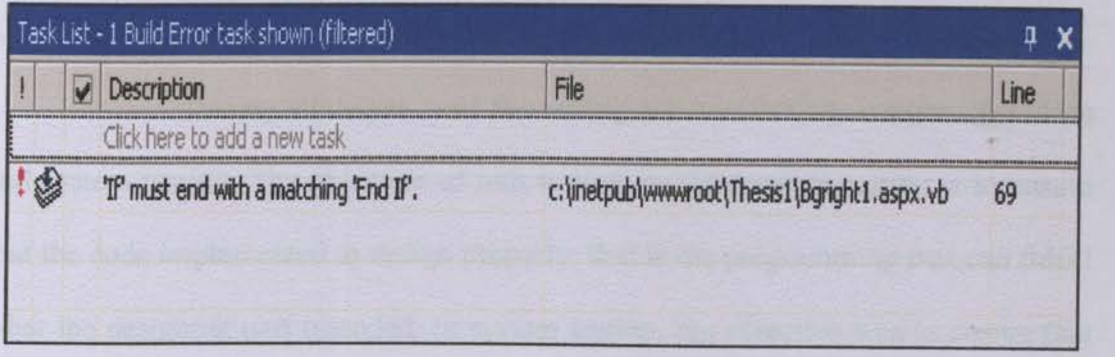
Black-box testing attempts to find errors in the following categories:-

1. Missing and incorrect functions. In the Student login unit, some errors discovered during debugging. Missing of End If statement at the end of function caused an error during debugging. There was a pop-up menu for building error as follows. Then the End If statement was completed to remove the error.

Figure 6.1: Pop-up menu and debugging errors



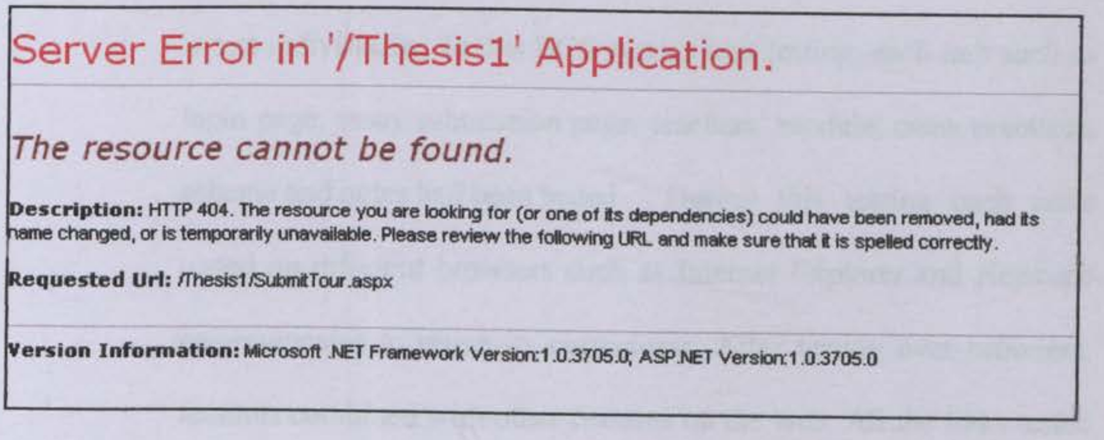
There were build errors. Continue?



!	☑	Description	File	Line
		Click here to add a new task		
!		'If' must end with a matching 'End If'.	c:\inetpub\wwwroot\Thesis1\Bgright1.aspx.vb	69

2. Interface errors and performance errors. One of the links was initialized wrongly. So, there was server error shown by the system as follows:-

Figure 6.2: Server error



Server Error in '/Thesis1' Application.

The resource cannot be found.

Description: HTTP 404. The resource you are looking for (or one of its dependencies) could have been removed, had its name changed, or is temporarily unavailable. Please review the following URL and make sure that it is spelled correctly.

Requested Url: /Thesis1/SubmitTour.aspx

Version Information: Microsoft .NET Framework Version:1.0.3705.0; ASP.NET Version:1.0.3705.0

3. Error in data structures or external database access.
4. Initialization and termination errors.

6.4 TESTING STRATEGIES

Testing strategies used for testing are unit testing, integration testing and system testing. The objective of unit testing and integration testing is to ensure that the code implemented to design properly; that is the programming part can fulfill what the designing part intended. In system testing, the objective was to ensure that the system does what the user wants it to do.

6.4.1 UNIT TESTING

Unit testing or stub testing is the test performed on individual events or modules of a program. In other words, it is the testing of an isolated subset of a program [23]. In the unit testing each component is test individually. In this EGS system unit testing, each unit such as login page, essay submission page, teachers' module, essay practices, scheme and notes had been tested. During this testing each units tested on different browsers such as Internet Explorer and Netscape communicator to prove its correctness. After testing over browsers, features combined with other features on the web. All the links tested to ensure that they navigate to the correct screen when clicked upon. Each error on links and buttons has been removed during this unit testing.

6.4.2 INTEGRATION TESTING

Once individual program components have been tested, they must be integrated to create a partial or complete system. This process involves building the system and testing the resultant system for problems that arise from components interactions [24]. It is not truly system testing because the components are not implemented in the operating environment. There are three basic integration test method, which is All-at-once, Bottom-up testing and Top-down testing. For EGS system bottom-up testing is considered as suitable method because of some important criteria. Bottom-up testing involves individual testing of each module using a driver routine that calls the module and provides it with needed resources. Bottom-up testing often works well in less structured shops. This is because there is less dependency on availability of other resources to accomplish the test. It is a more intuitive approach to testing that also usually finds errors in critical routines earlier than the top-down method. However, in a new system many modules must be integrated to produce system-level behavior, thus interface errors surface late in the process. The following approaches are used in integration testing of EGS. Visual Basic language, HTML language and frames are tested on the browser together. The components are integrated and tested on Internet Explorer 5.0 does not shows any problems. All links were tested to ensure that it is not broken or missing. When each links is clicked, connection is established to the desired target and web site to ensure that the user can access the web site.

6.4.3 SYSTEM TESTING

System testing ensures that application programs written and tested in isolated work properly when they are integrated into the total system [23]. System testing is to ensure that the system is performing well as a completed system. There are several steps in testing a system:

1. **Functional testing**

   Functional test checks that the integrated system performs its functions as specified in the requirements.

2. **Performance testing**

   Performance testing compares the integrated components with the non-functional system requirements as it had been described in chapter 3. That is constraining the way in which the system functions are performed.

3. **Acceptance testing**

   Acceptance test assumes the users that the system that requested is the system that was built for them.

4. **Installation testing**

   A final installation test is run to allow users to exercise system functions and documents additional problems that result from being at the actual site.

6.5 SUMMARY

There are plenty of methods and techniques that could be used to test a developed system. Testing a system is not merely for fault searching but also includes the investigation for system reliability, availability and maintainability. In the testing phase of EGS system, multiple ways are to ensure that the system performs well during and after delivery. However, an error-free or fault-free system is rare, not all faults and errors are able to be discovered in a short period of testing. For a stable and trustworthy system, continues testing and maintenance are required. Due to the limitation of time, the testing of EGS system are carried out to ensure that most of the faults and errors are absent during implementation.

CHAPTER 7

SYSTEM EVALUATION

7.1 INTRODUCTION TO SYSTEM EVALUATION

Systems evaluation is a process that occurs continuously at all phases of the system development, but finally the system should be evaluated using the end users for the system. The evaluation phases were done by form five students from *S.M.K. Gunung Rapat, Ipoh Perak.* Five students were take part in this evaluation phase. The student's information was attached in the Appendix. Evaluation phase was to determine the extent to which the system the expected outcome have been realized and the prescriptive value of the process where extra factors are taken into consideration.

7.2 INTERFACE EVALUATION

The following are the questions that were asked to the students during the evaluation period. Some of the students was answered the question and give a suggestions or ideas that was taken into consideration. Some of their suggestion can not be implemented because there are some specification and rules to be followed. Those questions did helps a lot during the testing period. It also helps to overcome some of the problems that are related to the interfaces.

Table 7.0: Evaluation Question/Answers

Question Asked to students	**Responds by Students**
Does the interface familiar compared to other systems which was used by you before?	➢ The interface is very common and it has all the basic controls that other systems have.
Does the interface behavior surprise you?	➢ No. It never irritating me when I was using the system. ➢ Since I am still new to the computer, the system is new for me but I could handle it by a few practices on it.
Does the interfaces arranged in a proper manner?	➢ Yes. It was arranged in a proper manner, but the page may be able to open in a new page. I may get a better view of its contents.
Does the Interface match the context of the system?	➢ Yes. Its background colors are classic and it supports the context of the subject.
Does the Interface attractive enough to attract students?	➢ It does attract me. It has some historical pictures. It is not mercy and properly arranged in a specific place. Can add some animations to be more attractive.

7.3 NAVIGATION EVALUATION

Navigation evaluation is needed to evaluate whether the systems navigation is working properly. Navigation should be done to a specific page if there are some needs. Any navigation that is not possible might confuse the end users.

Table 7.1: Navigation Evaluation (Question &Answers)

Question Asked to students	**Responds by Students**
Does the navigation works properly between pages?	➢ The navigation between pages is working properly. ➢ Some of the navigation buttons needs to be added.
Does the navigation buttons or hyperlinks confusing in term of surfing?	➢ No. It is clearly written the directions of the navigations. ➢ There is not much navigation buttons used.

7.4 HELP AND PERFORMANCE EVALUATIONS

EGS system has less helps for the students, because it is a simple system and it does not need any complex help features to help the students to operate the Essay Grading System. Some of the brief instructions were added to the essay submission modules, which will help the students to understand the rules that should be followed by them. The students never asked for help features during the evaluation period. It shows that the help features are not necessary for this system, since it could be handled easily. The performance of the system was evaluated by the students. It does have some problems of getting the results, but it is depend on the speed of the computer that was used. The response time might vary while using different computer to access the EGS system.

7.5 RESULTS OF EVALUATION

Results of the evaluation show that some of the systems features could be added or removed form Essay Grading System. It also helps the system to increase its robustness, reliability, stability, maintainability and operation of the system before the final touch to the systems development.

CHAPTER 8

SYSTEM DISCUSSION AND CONCLUSION

8.1 OVERVIEW OF DISCUSSION AND CONCLUSION

Basically, in this chapter we will discuss about the Essay Grading Systems (EGS) strengths and limitations. The systems development from early stages until the systems completion will be discussed here. There are many problems and solutions encountered during the project development. Though, the system was completed successfully by achieving the systems objectives.

8.2 PROBLEMS AND SOLUTIONS

In every project or research, there will always be problems that occur during the development. Throughout this project, many problems have kept unfolding one after another as development work progressed. Although these problems have slowed down the progress of the project, which was planned accordingly, the main objectives are however achieved.

8.2.1 PROBLEM TO DETERMINE THE SCOPE OF THE SYSTEM

Since there is not much experience in developing a system, it was hard to determine to which extent to define the scope of the system, so that the system can be completed within the given time frame. However the risk of ambitious objectives was removed at the beginning level of the system development. At first Latent Semantic Analysis was included in the objectives, but later it was removed because it is time consuming and the LSA is very complicated and a lot of researches need to be done. Since it was not practical to be implemented in this project, so it was removed. There are some groups of expertise or programmers who have been doing researches on the essay marking. They have implemented many techniques to do the programming such as Natural Language Processing and Latent Semantic Analysis and etc. So far, they have reached accuracy of marking around 90s, and it is still not accurate hundred percent. Then a simple matching procedure was adopted to be implemented in the EGS development.

8.2.2 PROBLEMS DURING THE SETUP OF DEVELOPMENT SOFTWARE.

During implementation and development of the system, there were some failures on installing and configuring the Internet Information System (IIS). It causes a lot of problems during the development stages. This results in the shifting of development software from Microsoft Visual Studio .net to WebMatrix development to prevent further problems that may happen during implementation phase and delay the project. Since the WebMatrix software is more complicated to be used, Microsoft Visual Studio .net was tried again and it has given a good result. Without the IIS installed on the computer, it cannot run the system on Internet Explorer. Before starts any installation on Microsoft Visual Studio .net, IIS needs to be installed in Microsoft Windows XP Professional.

8.2.3 TIME CONSTRAINT

Since the ASP.NET is still new and its features were upgraded from the ASP, many of the programmers are undergoing learning process. In this case I need to learn the software well before starts with the project. The learning process takes a lot of time. Due to this constraint, the development time has been tight and hardly enough for any extra or advanced features to be included in the EGS. To solve this problem, some references and sample functions codes from the internet were adopted, modified and used to decrease the time needed for development.

8.2.4 UNEXPECTED PROBLEMS

During the project development, there are many problems to be faced. Problems may occur because of both software and hardware. In this project both problems occurred at different time. The hardware requirement for the Microsoft Visual Studio .net to be installed is 128 MB Random Access Memory (RAM). After the installation, during project development there was some problems faced during the debugging of modules. The debugging process takes plenty of time to run at the Internet Explorer. It takes a lot of memories and Virtual memory was used to overcome this problem. This problem has dragged the process of development. To avoid this kind of problem in the future RAM was upgraded to 198 MB.

At first some coding on the semantic was tried to be implemented in the project. During the debugging process, some critical and serious problems were faced. Then, the software was seriously damaged and it never responds to any other new modules in the project. To overcome this problem, the software needs to be installed for second time.

8.3 SYSTEMS STRENGTH

Although this Essay Grading System does not have powerful features to some extent, but it still has some strength of its own when compared to some existing systems among university students.

8.3.1 INTUITIVE AND CONSISTENT USER INTERFACE

The key strength of the EGS is user friendly. EGS has a user friendly and consistent interface for the ease of users. A set of standard graphical user interface has been implemented and has been set to allow the users to browse in a very short time. The overall user interface is designed using a series of harmonic colors which gives users a smooth and comfortable feeling. Since the systems subject was History, some classic background was chosen.

8.3.2 CROSS BROWSER COMPATIBILITY

EGS can run on Internet explorer and Netscape Navigator or any other server that is compatible with it.

8.3.3 FAST RESPONSE

Each web page is loaded inside a specific frame. These pages are loaded in a reasonable amount of time to ensure a fast view of the page where heavy graphics been avoided.

8.3.4 FLEXIBLE SYSTEM

The EGS can be modified to mark any other subjects very easily without change the systems coding. During the development process, the questions and answers for the subject were stored in .txt files. The contents of the files can be changed and used to mark other subjects as well.

8.3.5 BASIC SECURITY

EGS system provides basic security to authenticate users. Each user level has their own degree of accessibility in EGS. Each user is provided with user login ID and password to be used in the secured section in EGS system. User login ID and user password cannot be changed using the system itself by users. Users have to change or update personal information through their respective teachers.

8.4 LIMITATION

Due to time factor, system developed offers functions and features. The limitations can be described as below:

8.4.1 MULTIPLE LANGUAGES NOT SUPPORTED

The current system is limited to Malay language only in the grading section. Anyway, it is still able to enhance to support more languages to support more languages to adapt to different environment.

8.4.2 TECHNIQUE OF MARKING

The marking techniques used in EGS system may not be suitable for the use of other languages as different languages may have a different form of construction and grammar.

8.4.3 MODIFICATION OF USERS INFORMATION

As a trade off to the security of the system, users are not able to modify on update personal information directly through EGS system. Users need to interact with system administrator (teachers) to modify their personal information. Although such policy may cause great inflexibility to most users, but for security reason, the inflexibility may seem worthy.

8.4.4 BROWSERS

EGS system is developed and tested using the Internet Explorer 5.0, Netscape Navigator and some other web browsers. There are a many web browsers which are not tested using EGS. Therefore, this has make some of the features or functions may not being support, look differently or performed well by using lower version. Other web browsers may cause disorder to the EGS application run on the client's machine.

8.4.5 USER LIMITATION

EGS has its own target users, who are form five students. This limitation was set because the system covers only form five student's syllabuses. Other secondary student's syllabus was not covered here.

8.5 FUTURE ENHANCEMENT

This system can be further enhanced by adding more functions for it to make the site more informational and useful to the user. Some functions such as step-by-step directions of the recipes were not added into the system because of the time constrain. So, more suggestions are given on how this system can be improved in the future.

8.5.1 SUPPORT SEMANTIC

This system developed using sequence matching to search and match the keywords that are grabbed from the student's essay. In the future some semantic coding techniques can be used to mark the student's essay. The semantic coding technique is a very complicated and it needs a lot of researches. This module is time consuming.

8.5.2 STATISTICS

The current system does not have any statistics on the student's marks. In the future some statistics can be implemented to process and to come out with some analysis on the student's marks.

8.5.3 OTHER INFORMATIONAL LINKS

The current system has the basic notes and scheme which is used by the students to refer for their studies. It can be enhanced by add some informational links to other important websites, which can give the students extra information which can increase the student's general knowledge.

8.5.4 MORE ADMINISTRATOR TASK

More powerful admin task such as database backup, HTML protocol can be further added. Statistic calculation of database usage and database cleanup application can also be included to reduce system administrator's work.

8.6 CONCLUSION

As a conclusion of this project, EGS has fulfilled the requirement to deliver the system in time and achieved most of the objectives and requirements as determine during analysis phase. The development of this system is not just to fulfill the requirement for a degree but also as an experience for future use when the student is faced with the task of developing a bigger system.

The EGS system is capable of marking essay questions and gives some scores for students. Marks can be given in a short period of time after the examination. It also helps the teachers to do other tasks than marking student's essays. The interface for the system is very simple and can be pick-up easily by students.

Throughout this project, a lot of valuable knowledge and experience has been learned and gained. Further, skills and techniques in the use of software such as ASP.net, Adobe Photoshop 7.0, Ulead PhotoImpact 8.0, and others have been greatly improved.

The EGS was successfully developed within the time frame given. Most of the functions in the analysis and design phase were incorporated into the system. It is hoped that this system will fulfill the needs of the end user and help them in marking process.

REFERENCE

[1] Chen, K.W., "Internet Student Information System", Thesis Bc., University of Malaya, 1997/1998.

[2] Hulquist,S.(1997)."What is Client-Server Computing?", 18 October 1997, http:// www.cis.ohio-state.edu/hypertext/faq/usenet/client-server-faq/faq-doc-4.html.

[3] Sadoski,D.(2000). "Client/Server Software Architecture". Daniel, Carnegie Mellon University.

http://www.sei.cmu.edu/str/descriptions/clientserver_body.html.

[4] Taylor,S., "What is Three-Tier Architecture?", 18 October 1997, www.cis.ohio- state.edu/hypertext/faq/usenet/client-server-faq/faq-doc-7.html.

[5] Suresh, K.S., " Parallel Database System Technology",Power Technology 1995, The UNIX Technology Conference 1995, 14 April 1995.

[6] TOEFL (1997). http:// www.toefl.org/tstprpmt.html

[7] GMAT (1997). http://www.gmat.org/

[8] Foltz, P.W., Kintsch, W., and Launder, T.(1998) The Measurement of Textual Coherence with Latent Semantic Analysis. Discourse Processes, 25(2&3), 285-307.

[9] "ETS Technology" http://www.etstechnologies.com/products_e.shtml

[10] Bergler, S., (1995), 'From Lexical Semantics to Text Analysis', in P. Saint-Dizie(Ed.), Computational Lexical Semantics, Cambridge University Press.

[11] “Essay Grading System” http://knowledge-technologies.com

[12] “Microsoft Access 2002” http://microsoft.com

[13] “Microsoft Access 2002” http://microsoft.com

[14] “Intelligent Essay Assessor”
http:// www.knowledge-technologies.com/cgi-bin/IEAdemo/iea-grade.cgi

[15] “Intellmetric Automated System” http://www.lboro.ac.uk/

[16] “Web Tools Newsletter, Automated Essay Grading System”
http:// webtools.cityu.edu.hk/news/newslett/automatedessaygrading.htm

[17] Page, E. B., & Petersen, N. S. (1995). The computer moves into essay grading: Updating the ancient test. Phi Delta Kappan, 76, 561-565.

[18] “Holt Online Essay Scoring”
http://www.k-a-t.com/HRW12Demo/HRW12.html

[19] “Advantages of Using ASP.Net” http:// www.brillianceweb.com/

[20] Whitten, L., Jeffrey, Bentley D. Lonnie, Dittman, Kevin .(2002). System Analysis and Design Methods, 5th Ed. New York : Mc Graw Hill.

[21] Burstein, J., K. Kukich, S. Wolff, C. Lu, & M. Chodorow. 1998a. Computer Analysis of Essays. Presented at the NCME Symposium on Automated Scoring, April 1998.

[22] Williams, R. (2001). Automated essay grading: An evaluation of four conceptual models. In A. Herrmann and M. M. Kulski (Eds), *Expanding Horizons in Teaching and Learning*. Proceedings of the 10th Annual Teaching Learning Forum, 7-9 February 2001.

[23] Whitten, L., Jeffrey, Bentley D. Lonnie, Dittman, Kevin .(2002). System Analysis and Design Methods, 5th Ed. New York : Mc Graw Hill.

[24] Sommerville, I., (2001). Software Engineering. 6th Ed. England : Addison Wesley.

APPENDIX A : STUDENT INFORMATION

STUDENT 1

NAME : YAP YEE KEEN

IC NO : 870115 - 05 - 5239 -

FORM : 5

SCHOOL : S. M. K. Seri Ampang, Ipoh, Perak
Darul Ridzuan.

DATE : 30 - 1 - 2004

Students Signature,

(YAP YEE KEEN)

Authors Signature,

(ELAIYAKUMARAN A/L SUBRAMANIAM.)

* The above particulars are taken for the purpose of Essay Grading System evaluation only. It would not be used, or modified for other personal use.

STUDENT 2

NAME : ABDULLAH SHAHIR BIN ABDUL MANAF

IC NO : 870807 - 08 - 5863

FORM : 5

SCHOOL : S.M.K. GUNUNG RAPAT, IPOH,
PERAK DARUL RIDZUAN.

DATE : 30/1/04

Students Signature,

Shahir.

(ABDULLAH SHAHIR BIN ABDUL MANAF

Authors Signature,

(ELAIYAKUMARAN A/L SUBRAMANIAM)

* The above particulars are taken only for the purpose of Essay Grading System evaluation. It won't be used or modified for other personal use.

STUDENT 3

NAME : Amirah binti Yaakob

IC NO : 871013 - 05 - 5894

FORM : 5

SCHOOL : S M K Seri Ampang - Ipoh
Perak

DATE : 31 - 1 - 2004

Students Signature,

(Amirah binti Yaakob)

Authors Signature,

elay

(ELAIYAKUMARAN A/L SUBRAMANIAM)

* The above particulars are taken only for the purpose of Essay Grading System evaluation. It won't be used or modified for other personal use.

APPENDIX B : CODING

1.0 STUDENT LOGIN CODING

```
#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the
         Web Form Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region
    Private ConnectionString As String =
"Provider=Microsoft.Jet.OLEDB.4.0; " & _"Data Source ="
 & Server.MapPath("\Thesis1\Teacherdb.mdb")

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
    'Put user code to initialize the page here
    End Sub

    Private Sub btnok_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnok.Click
        lblError.Text = ""
        lblStudentName.Text = ""
        lblCheckPassword.Text = ""
        If txtname.Text = "" Then
            lblStudentName.Text = "Username not entered"
            Exit Sub
        End If

        If txtpassword.Text = "" Then
            lblCheckPassword.Text =
            "Password not entered"
            Exit Sub
        End If



        Dim boolRedirect As Boolean
        Dim selectSQL As String
        selectSQL = "SELECT * FROM TeachersTable"
        Dim dbConnection As New
    OleDbConnection(ConnectionString)
    Dim dbCommand As New OleDbCommand(selectSQL,
    dbConnection)
    Dim dbReader As OleDbDataReader

    Try
        dbConnection.Open()
        dbReader = dbCommand.ExecuteReader
        dbReader.Read()

        Do While dbReader.Read()
            If dbReader("StudentsNumber") =
               txtname.Text Then
                If dbReader("Password") =
                   txtpassword.Text Then
                    boolRedirect = True
                    Exit Do
                Else
                    Exit Do
                End If
            End If
        Loop

        If boolRedirect = False Then
            lblError.Text = "Invalid username
                                     or password"
            txtname.Text = ""
            txtpassword.Text = ""
            dbReader.Close()
            dbConnection.Close()
            Exit Sub
        End If

        selectSQL = ""
        dbReader.Close()

    Catch err As Exception
        lblError.Text = "Error getting data. "
        lblError.Text &= err.Message
    Finally
        If (Not dbConnection Is Nothing) Then
            dbConnection.Close()
            dbConnection.Dispose()
        End If
    End Try


    If boolRedirect = True Then
            Server.Transfer("SubmitF.aspx?ID=" & _
            HttpUtility.UrlEncode(txtname.Text))
        End If
End Sub
```

```
    Private Sub btncancel_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btncancel.Click
        Response.Redirect("WelcomeForm.aspx")
    End Sub

    Private Sub btnNew_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnNew.Click
        Response.Redirect("StudentDetail.aspx")
    End Sub
End Class
```

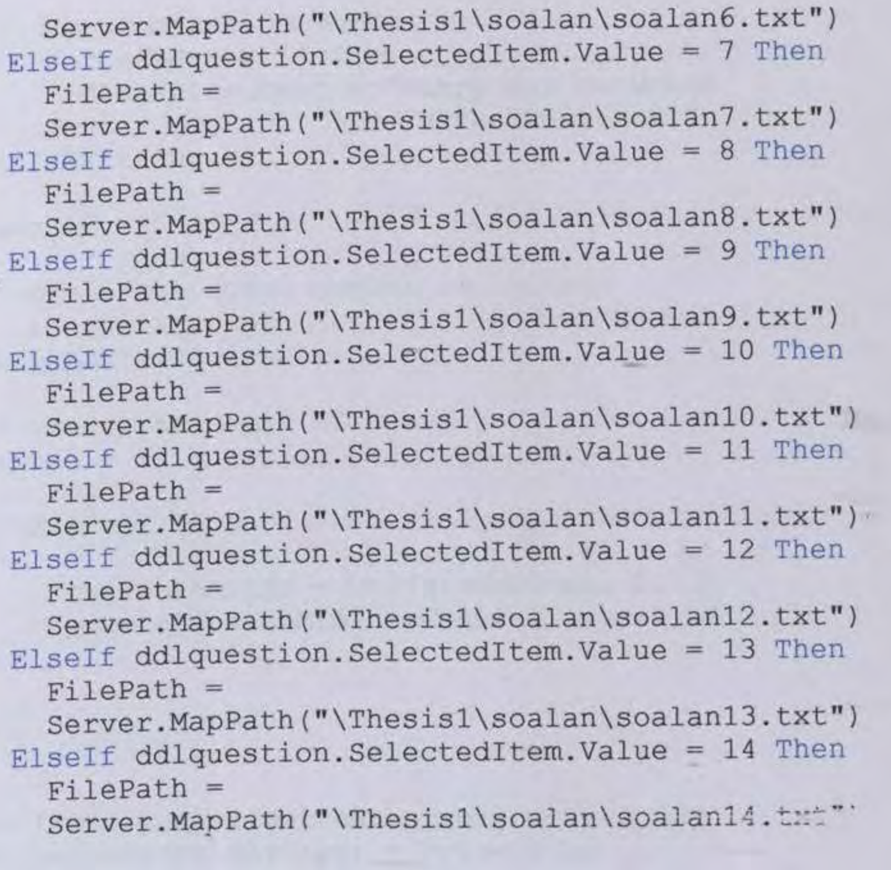
2.0 TEACHERS LOGIN CODING

```
#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form
Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here
    End Sub


Private Sub btnok_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnok.Click
        If txtname.Text = "" Then
            lblCheckName.Text = "*Administrator name
                                 is required!"
            Exit Sub
        End If

        If txtpassword.Text = "" Then
            lblCheckPassword.Text = "*Password is
                                     required!"
            Exit Sub
        End If
        Dim DBConn As OleDbConnection
        Dim DBCommand As OleDbDataAdapter
        Dim DSLogin As New DataSet()
        DBConn = New
OleDbConnection("PROVIDER=Microsoft.Jet.OLEDB.4.0;" _
            & "DATA SOURCE=" _
            & Server.MapPath("/Thesis1/" _
            & "AdminLogindb.mdb;"))
        DBCommand = New OleDbDataAdapter _
            ("Select AdminName from " _
            & "AdminLoginTable Where " _
            & "AdminName = '" & txtname.Text _
            & "' and Password = '" & txtpassword.Text _
            & "'", DBConn)
        DBCommand.Fill(DSLogin, "AdminLoginTable")

If DSLogin.Tables("AdminLoginTable"). _
            Rows.Count = 0 Then
            lblError.Text = "The Administrator
                            name and password " _
                & "were not found. Please try again."
        Else
            Session("AdminName") =
            DSLogin.Tables("AdminLoginTable"). _
                Rows(0).Item("AdminName")
            Session("AdminName") = txtname.Text
            Response.Redirect("./TeachersAccess.aspx")
        End If
    End Sub

    Private Sub btncancel_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btncancel.Click
        Response.Redirect("./WelcomeForm.aspx")
    End Sub
End Class
```

3.0 ESSAY SUBMISSION CODING

```
Imports System.IO
#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init

        InitializeComponent()
    End Sub
#End Region
```

```
    Dim AnswerPath As String
    Dim objFileWriter As StreamWriter
    Dim objStreamReader As StreamReader
    Dim sContents As String = ""
    Dim score

    Private ConnectionString As String =
"Provider=Microsoft.Jet.OLEDB.4.0; " & _
"Data Source =" & Server.MapPath("\Thesis1\Teacherdb.mdb")

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        lblShowUser.Text = Request.QueryString("ID")

    End Sub

    Private Sub ReadFile()
        Try
            objStreamReader = File.OpenText(FilePath)
            Dim contents As String =
            objStreamReader.ReadToEnd()
            TxtQuestion.Text = contents
            objStreamReader.Close()
        Catch err As Exception
            TxtQuestion.Text = "Error reading data. "
            TxtQuestion.Text &= err.Message
        End Try
    End Sub


    Private Sub ddlquestion_SelectedIndexChanged(ByVal sender
As System.Object, ByVal e As System.EventArgs) Handles
ddlquestion.SelectedIndexChanged

         If ddlquestion.SelectedItem.Text <>
           "--Question--" Then
         If ddlquestion.SelectedItem.Value = 1 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan1.txt")
         ElseIf ddlquestion.SelectedItem.Value = 2 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan2.txt")
         ElseIf ddlquestion.SelectedItem.Value = 3 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan3.txt")
         ElseIf ddlquestion.SelectedItem.Value = 4 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan4.txt")
         ElseIf ddlquestion.SelectedItem.Value = 5 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan5.txt")
         ElseIf ddlquestion.SelectedItem.Value = 6 Then
           FilePath =
```



```
          Server.MapPath("\Thesis1\soalan\soalan6.txt")
        ElseIf ddlquestion.SelectedItem.Value = 7 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan7.txt")
        ElseIf ddlquestion.SelectedItem.Value = 8 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan8.txt")
        ElseIf ddlquestion.SelectedItem.Value = 9 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan9.txt")
        ElseIf ddlquestion.SelectedItem.Value = 10 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan10.txt")
        ElseIf ddlquestion.SelectedItem.Value = 11 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan11.txt")
        ElseIf ddlquestion.SelectedItem.Value = 12 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan12.txt")
        ElseIf ddlquestion.SelectedItem.Value = 13 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan13.txt")
        ElseIf ddlquestion.SelectedItem.Value = 14 Then
          FilePath =
          Server.MapPath("\Thesis1\soalan\soalan14.txt"
```

```
          Else
               txtAnswer.Text = ""
               TxtQuestion.Text = ""
               lblDisplayScore.Text = ""
           End If
           ReadFile()
       End If
   End Sub


   Private Sub btnSubmit_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSubmit.Click
       Dim strContent As String
       strContent = txtAnswer.Text
       lblResults.Text = ""

       Dim x
       Dim i
       Dim currentLetter
       Dim prevLetter
       Dim total

       ' Remove leading and trailing spaces of string
         if any.
       strContent = Trim(txtAnswer.Text)
```

```
        ' The text box is empty.
        If strContent = "" Then
            lblResults.Text = "There are no words
                               in the text box!"
            Exit Sub

        End If

        ' Store the total number of letters
          In the string.
        x = Len(strContent)

        ' Start the count at one.
        total = 1

        For i = 2 To x

            currentLetter = Mid(strContent, i, 1)
            prevLetter = Mid(strContent, i - 1, 1)




  If Char.IsWhiteSpace(currentLetter) = True And
Char.IsWhiteSpace(prevLetter) = False Then
                total = total + 1
            End If
        Next

        If total > 80 Then
        If ddlquestion.SelectedItem.Value = 1 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan1.txt")
        ElseIf ddlquestion.SelectedItem.Value = 2 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan2.txt")
        ElseIf ddlquestion.SelectedItem.Value = 3 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan3.txt")
        ElseIf ddlquestion.SelectedItem.Value = 4 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan4.txt")
        ElseIf ddlquestion.SelectedItem.Value = 5 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan5.txt")
        ElseIf ddlquestion.SelectedItem.Value = 6 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan6.txt")
        ElseIf ddlquestion.SelectedItem.Value = 7 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan7.txt")
        ElseIf ddlquestion.SelectedItem.Value = 8 Then
         AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan8.txt")
       ElseIf ddlquestion.SelectedItem.Value = 9 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan9.txt")
       ElseIf ddlquestion.SelectedItem.Value = 10 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan10.txt")
       ElseIf ddlquestion.SelectedItem.Value = 11 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan11.txt")
       ElseIf ddlquestion.SelectedItem.Value = 12 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan12.txt")
       ElseIf ddlquestion.SelectedItem.Value = 13 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan13.txt")
       ElseIf ddlquestion.SelectedItem.Value = 14 Then
        AnswerPath =
        Server.MapPath("\Thesis1\jawapan\jawapan14.txt")
       End If

Dim contents As String
Dim contentsPosition, contentsTemp, contentsValue
           Try
               Dim Words
               Dim Counter
               Dim Score
               Dim Content

               Words = Split(txtAnswer.Text)
               Score = 0
               objStreamReader =
               File.OpenText(AnswerPath)
               Content = objStreamReader.ReadToEnd()
               objStreamReader.Close()

               For Counter = 0 To UBound(Words) - 1
                   While InStr(1, Content,
                   Words(Counter), vbTextCompare) > 0
                   'Replace every occurrance of this
                    word with an empty string so it
                   'won't be found a second time.
               Content = Replace(Content,
               Words(Counter), "", 1, 1, vbTextCompare)
                      Score = Score + 1
                   End While
               Next
               If Score = 0 Then
                   lblResults.Text = "You score is zero"
               End If
               lblDisplayScore.Text = Score
               Content = ""
```

```
            Catch err As Exception
                lblResults.Text = "Error reading data. "
                lblResults.Text &= err.Message
            End Try

        Else
            lblResults.Text = "The number of words not
                               within marking range"
        End If

    End Sub


    Private Sub txtAnswer_TextChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
txtAnswer.TextChanged
        If ddlquestion.SelectedItem.Text =
                 "--Question--" Then
            lblResults.Text = "Please choose a question!"
        Else
            btnSubmit.Enabled = True
        End If
    End Sub

    Private Sub btnLogout_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnLogout.Click
        Response.Redirect("./WelcomeForm.aspx")
    End Sub

    Private Sub btnClear_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnCancel.Click
        txtAnswer.Text = ""
        lblResults.Text = ""
        lblDisplayScore.Text = ""
    End Sub

    Private Sub btnSave_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSave.Click
        Dim updateSQL As String

        updateSQL = "UPDATE TeachersTable SET "
        updateSQL &= "[Score] = '" &
                    lblDisplayScore.Text & "' , "
        updateSQL &= "[QuestionNumber] = '" &
                    ddlquestion.SelectedItem.Text & "' "
        updateSQL &= "WHERE [StudentsNumber] = '" &
                    lblShowUser.Text & "'"

        Dim dbConnection As New
            OleDbConnection(ConnectionString)
        Dim dbCommand As New
            OleDbCommand(updateSQL, dbConnection)
        Dim intUpdated As Integer
        Try
            dbConnection.Open()
            intUpdated = dbCommand.ExecuteNonQuery
            updateSQL = ""
            lblResults.Text = "Score successfully
                              updated"


        Catch err As Exception
            lblResults.Text = "Error updating score. "
            lblResults.Text &= err.Message
        Finally
            If (Not dbConnection Is Nothing) Then
                dbConnection.Close()
                dbConnection.Dispose()
            End If
        End Try

        Dim AnswerFile As String
        AnswerFile =
        Server.MapPath("\Thesis1\jawapanPelajar\")
        AnswerFile &= lblShowUser.Text & "_" &
        ddlquestion.SelectedItem.Text & ".txt"
        objFileWriter = File.CreateText(AnswerFile)
        objFileWriter.Write(txtAnswer.Text)
        objFileWriter.Close()
    End Sub
End Class
```

4.0 ESSAY PRACTICES CODING

```
Imports System.IO
#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form
Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub
#End Region
```

```
    Dim FilePath As String
    Dim AnswerPath As String
    Dim objFileWriter As StreamWriter
    Dim objStreamReader As StreamReader
    Dim sContents As String = ""
    Dim score

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
    End Sub


    Private Sub ReadFile()
        Try
            objStreamReader = File.OpenText(FilePath)
            Dim contents As String =
                objStreamReader.ReadToEnd()
            TxtQuestion.Text = contents
            objStreamReader.Close()
        Catch err As Exception
            TxtQuestion.Text = "Error reading data. "
            TxtQuestion.Text &= err.Message
        End Try

    End Sub

    Private Sub ddlquestion_SelectedIndexChanged(ByVal sender
As System.Object, ByVal e As System.EventArgs) Handles
ddlquestion.SelectedIndexChanged

  If ddlquestion.SelectedItem.Text <>
           "--Question--" Then
         If ddlquestion.SelectedItem.Value = 1 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan1.txt")
         ElseIf ddlquestion.SelectedItem.Value = 2 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan2.txt")
         ElseIf ddlquestion.SelectedItem.Value = 3 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan3.txt")
         ElseIf ddlquestion.SelectedItem.Value = 4 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan4.txt")
         ElseIf ddlquestion.SelectedItem.Value = 5 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan5.txt")
         ElseIf ddlquestion.SelectedItem.Value = 6 Then
           FilePath =
           Server.MapPath("\Thesis1\soalan\soalan6.txt")
```

```
          ElseIf ddlquestion.SelectedItem.Value = 7 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan7.txt")
          ElseIf ddlquestion.SelectedItem.Value = 8 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan8.txt")


          ElseIf ddlquestion.SelectedItem.Value = 9 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan9.txt")
          ElseIf ddlquestion.SelectedItem.Value = 10 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan10.txt")
          ElseIf ddlquestion.SelectedItem.Value = 11 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan11.txt")
          ElseIf ddlquestion.SelectedItem.Value = 12 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan12.txt")
          ElseIf ddlquestion.SelectedItem.Value = 13 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan13.txt")
          ElseIf ddlquestion.SelectedItem.Value = 14 Then
            FilePath =
            Server.MapPath("\Thesis1\soalan\soalan14.txt")
             Else
                 txtAnswer.Text = ""
                 TxtQuestion.Text = ""
                 lblDisplayScore.Text = ""
             End If
             ReadFile()
        End If
    End Sub

    Private Sub btnSubmit_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSubmit.Click
        Dim strContent As String
        strContent = txtAnswer.Text
        lblResults.Text = ""

        Dim x
        Dim i
        Dim currentLetter
        Dim prevLetter
        Dim total

       Dim strContent As String
        strContent = txtAnswer.Text
        lblResults.Text = ""
        Dim x
        Dim i
        Dim currentLetter
        Dim prevLetter
        Dim total

        ' Remove leading and trailing spaces of string
          if any.
        strContent = Trim(txtAnswer.Text)

         ' The text box is empty.
        If strContent = "" Then
            lblResults.Text = "There are no words
                               in the text box!"
            Exit Sub
        End If

        ' Store the total number of letters
          In the string.
        x = Len(strContent)

        ' Start the count at one.
        total = 1

        For i = 2 To x

            currentLetter = Mid(strContent, i, 1)
            prevLetter = Mid(strContent, i - 1, 1)

            If Char.IsWhiteSpace(currentLetter) = True And
Char.IsWhiteSpace(prevLetter) = False Then
                total = total + 1
            End If
        Next
        If total > 10 Then
            'lblResults.Text = Val(total)

        If ddlquestion.SelectedItem.Value = 1 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan1.txt")
        ElseIf ddlquestion.SelectedItem.Value = 2 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan2.txt")
        ElseIf ddlquestion.SelectedItem.Value = 3 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan3.txt")
        ElseIf ddlquestion.SelectedItem.Value = 4 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan4.txt")
        ElseIf ddlquestion.SelectedItem.Value = 5 Then
         AnswerPath =
         Server.MapPath("\Thesis1\jawapan\jawapan5.txt")
```

```
 ElseIf ddlquestion.SelectedItem.Value = 6 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan6.txt")
 ElseIf ddlquestion.SelectedItem.Value = 7 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan7.txt")
 ElseIf ddlquestion.SelectedItem.Value = 8 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan8.txt")
 ElseIf ddlquestion.SelectedItem.Value = 9 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan9.txt")
 ElseIf ddlquestion.SelectedItem.Value = 10 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan10.txt")
 ElseIf ddlquestion.SelectedItem.Value = 11 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan11.txt")
 ElseIf ddlquestion.SelectedItem.Value = 12 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan12.txt")
 ElseIf ddlquestion.SelectedItem.Value = 13 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan13.txt")
 ElseIf ddlquestion.SelectedItem.Value = 14 Then
  AnswerPath =
  Server.MapPath("\Thesis1\jawapan\jawapan14.txt")
 End If


Dim contents As String
Dim contentsPosition, contentsTemp, contentsValue
     Try
         Dim Words
         Dim Counter
         Dim Score
         Dim Content

         Words = Split(txtAnswer.Text)
         Score = 0
         objStreamReader =
         File.OpenText(AnswerPath)
         Content = objStreamReader.ReadToEnd()
         objStreamReader.Close()

         For Counter = 0 To UBound(Words) - 1
             While InStr(1, Content,
             Words(Counter), vbTextCompare) > 0
 'Replace every occurrance of this word
  with an empty string so it
 'won't be found a second time.

  txtAnswer.Text = Replace(txtAnswer.Text, ".", " ")
  txtAnswer.Text = Replace(txtAnswer.Text, ",", " ")
      Words = Split(txtAnswer.Text)
             Score = Score + 1
          End While
        Next
        lblDisplayScore.Text = Score
        Content = ""

      Catch err As Exception
        lblResults.Text = "Error reading data. "
        lblResults.Text &= err.Message
      End Try
    Else
      lblDisplayScore.Text = ""
      lblResults.Text = "The number of words not within
marking range"
    End If
  End Sub

  Private Sub TxtQuestion_TextChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
TxtQuestion.TextChanged
  End Sub

  Private Sub txtAnswer_TextChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
txtAnswer.TextChanged
    If ddlquestion.SelectedItem.Text = "--Question--"
                Then
      lblResults.Text = "Please choose a question!"
    Else
      btnSubmit.Enabled = True
    End If
  End Sub

  Private Sub btnLogout_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnLogout.Click
    Response.Redirect("./WelcomeForm.aspx")
  End Sub


  Private Sub btnClear_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnCancel.Click
    txtAnswer.Text = ""
    lblResults.Text = ""
    lblDisplayScore.Text = ""
  End Sub
End Class
```

5.0 STUDENT REGISTERATION CODING

```
#Region " Web Form Designer Generated Code "

    'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init
        'CODEGEN: This method call is required by the Web Form
Designer
        'Do not modify it using the code editor.
        InitializeComponent()
    End Sub
#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        'Put user code to initialize the page here
    End Sub


    Private ConnectionString As String =
"Provider=Microsoft.Jet.OLEDB.4.0; " & _
"Data Source =" &Server.MapPath("\Thesis1\Teacherdb.mdb")

    Private Sub btnRegister_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
btnRegister.Click
        Dim InsertSQL As String
        Dim ID As String
        Dim QuestionNumber As String
        Dim Score As String



        If txtName.Text = "" Then
            lblError.Text = "Student name required"
            Exit Sub
        ElseIf txtIdNumber.Text = "" Then
            lblError.Text = "Student number required"
            Exit Sub
        ElseIf ddlClass.SelectedItem.Text =
            "----Class----" Then
            lblError.Text = "Class name not selected"
            Exit Sub
        ElseIf ddlTName.SelectedItem.Text =
            "--Teacher Name--" Then
            lblError.Text = "Teacher name not selected"
            Exit Sub
```

```
ElseIf txtPassword.Text = "" Then
    lblError.Text = "Password required"
    Exit Sub
ElseIf txtCPassword.Text = "" Then
    lblError.Text = "Password required"
    Exit Sub
ElseIf txtPassword.Text <> txtCPassword.Text Then
    lblError.Text = "Password does not match"
    Exit Sub
End If

QuestionNumber = "-"
Score = "-"

InsertSQL = "INSERT INTO TeachersTable ("
InsertSQL &= "[StudentsNumber], [Name],
            [Score], "
InsertSQL &= "[QuestionNumber],[Teacher Name],
            [Class], [Password]) "
InsertSQL &= "VALUES ('"
InsertSQL &= txtIdNumber.Text & "', '"
InsertSQL &= txtName.Text & "', '"
InsertSQL &= Score & "', '"
InsertSQL &= QuestionNumber & "', '"
InsertSQL &= ddlTName.SelectedItem.Text & "', '"
InsertSQL &= ddlClass.SelectedItem.Text & "', '"
InsertSQL &= txtPassword.Text & "')"

Dim selectSQL As String
selectSQL = "SELECT [StudentsNumber] FROM
            TeachersTable"
Dim dbConnection As New
    OleDbConnection(ConnectionString)
Dim dbCommand As New OleDbCommand(selectSQL,
    dbConnection)
Dim dbReader As OleDbDataReader

Try
    dbConnection.Open()
    dbReader = dbCommand.ExecuteReader
    dbReader.Read()

    Do While dbReader.Read()
        If dbReader("StudentsNumber") =
           txtIdNumber.Text Then
            lblError.Text = "Student number
                             already exists"
            dbReader.Close()
            dbConnection.Close()
            Exit Sub
        End If
    Loop
```

```
            selectSQL = ""
            dbReader.Close()

        Catch err As Exception
            lblError.Text = "Error getting data. "
            lblError.Text &= err.Message
        Finally
            If (Not dbConnection Is Nothing) Then
                dbConnection.Close()
            End If
        End Try

Dim cmd As New OleDbCommand(InsertSQL, dbConnection)
Dim Added As Integer
        Try
            dbConnection.Open()
            Added = cmd.ExecuteNonQuery

        Catch err As Exception
            lblError.Text = "Error inserting record."
            lblError.Text &= err.Message

        Finally
            If (Not dbConnection Is Nothing) Then
                dbConnection.Close()
                dbConnection.Dispose()
            End If
        End Try

        If Added > 0 Then
           Server.Transfer("SubmitF.aspx?ID=" & _
           HttpUtility.UrlEncode(txtIdNumber.Text))
        End If
    End Sub

    Private Sub btnCancel_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnCancel.Click
        Response.Redirect("WelcomeForm.aspx")
    End Sub
End Class
```

6.0 ADMINISTRATOR CODING (TEACHER)

```
Imports System.Data
Imports System.Data.OleDb
Imports System.IO
Imports System.Data.OleDb.OleDbCommand
Imports System.Data.OleDb.OleDbConnection
Imports System.Web.UI.WebControls
```

```
    Public Function GetDataSet() As DataSet
        Dim SelectSQL As String
        SelectSQL = "SELECT * FROM TeachersTable
                    ORDER BY [StudentsNumber]"
        Dim dbConnection As New
            OleDbConnection(ConnectionString)
        Dim dbCommand As New OleDbCommand(SelectSQL,
            dbConnection)
        Dim adapter As New OleDbDataAdapter(dbCommand)
        Dim dsPubs As New DataSet()
        adapter.Fill(dsPubs, "TeachersTable")
        dbConnection.Close()
        Return dsPubs
    End Function

    Public Sub BindGrid(ByVal ds As DataSet)
        dg1.DataSource = ds.Tables("TeachersTable")
        Me.DataBind()
    End Sub

    Dim objStreamReader As StreamReader

Private AnswerPath As String =
Server.MapPath("\Thesis1\jawapanPelajar\")
    Private ConnectionString As String =
"Provider=Microsoft.Jet.OLEDB.4.0; " & _
                                        "Data Source =" &
Server.MapPath("\Thesis1\Teacherdb.mdb")
#Region " Web Form Designer Generated Code "


 'This call is required by the Web Form Designer.
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()

    End Sub

    Private Sub Page_Init(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Init
                'Do not modify it using the code editor.
        InitializeComponent()
    End Sub

#End Region

    Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        If Me.IsPostBack = False Then
            Dim ds As DataSet = GetDataSet()
            BindGrid(ds)
            CreateFileList()
        End If
    End Sub
    Private Sub dg1_EditCommand(ByVal source As Object, ByVal
e As DataGridCommandEventArgs) Handles dg1.EditCommand
        dg1.EditItemIndex = e.Item.ItemIndex
        Dim ds As DataSet = GetDataSet()
        BindGrid(ds)
    End Sub

    Private Sub dg1_CancelCommand(ByVal source As Object,
ByVal e As System.Web.UI.WebControls.DataGridCommandEventArgs)
Handles dg1.CancelCommand
        dg1.EditItemIndex = -1
        Dim ds As DataSet = GetDataSet()
        BindGrid(ds)
    End Sub


    Private Sub dg1_UpdateCommand(ByVal source As Object,
ByVal e As DataGridCommandEventArgs) Handles dg1.UpdateCommand
        Dim NewStudentsNumber, NewName As String
        Dim NewScore, NewQuestionNumber As String


        NewName = CType(e.Item.FindControl("txtName"),
                  TextBox).Text
        NewScore = CType(e.Item.FindControl("txtScore"),
                  TextBox).Text
        NewQuestionNumber =
        CType(e.Item.FindControl("txtQuestionNumber"),
        TextBox).Text

        'Retrieve the ID to look up the record.
        Dim ID As String
        ID = CType(e.Item.FindControl("lblID"),
             Label).Text

        'Define the Update Statement.
        Dim Update As String
        Update = "UPDATE TeachersTable SET "
        Update &= "[Name] = '" & NewName & "' , "
        Update &= "[Score] = '" & NewScore & "' , "
        Update &= "[QuestionNumber] = '" &
                   NewQuestionNumber & "' "
        Update &= "WHERE [StudentsNumber] = '" & ID & "'"

        'create object.
        Dim dbConnection As New
            OleDbConnection(ConnectionString)
        Dim dbCommand As New
            OleDbCommand(Update,dbConnection)
        'Open database and execute
        Try
            dbConnection.Open()
            Dim NumberUpdated As Integer
            NumberUpdated = dbCommand.ExecuteNonQuery()
            lblStatus.Text = NumberUpdated.ToString()
            lblStatus.Text &= " record updated"

        Catch err As Exception
            lblStatus.Text = "Error updating data."
            lblStatus.Text &= err.Message

              Finally
            If Not (dbConnection Is Nothing) Then
                dbConnection.Close()
            End If
        End Try


        'cancel edit mode
        dg1.EditItemIndex = -1

        'Rebind the grid.
        Dim ds As DataSet = GetDataSet()
        BindGrid(ds)
    End Sub

    Private Sub CreateFileList()
        Dim FileList As Array =
        Directory.GetFiles(AnswerPath)
        lstFiles.DataSource = FileList
        lstFiles.DataBind()
    End Sub

    Private Sub lstFiles_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
lstFiles.SelectedIndexChanged
        Dim filename As String =
        lstFiles.SelectedItem.Text
        AnswerPath = filename
        Try
            objStreamReader = File.OpenText(AnswerPath)
            Dim contents As String =
            objStreamReader.ReadToEnd()
            txtEssay.Text = contents
            objStreamReader.Close()
        Catch err As Exception
            txtEssay.Text = "Error reading data. "
            txtEssay.Text &= err.Message
        End Try
    End Sub
    Private Sub dg1_DeleteCommand(ByVal source As Object,
ByVal e As System.Web.UI.WebControls.DataGridCommandEventArgs)
Handles dg1.DeleteCommand
        Dim ID As String
        ID = CType(e.Item.FindControl("lblID"), Label).Text

        'Define the Update Statement.
        Dim Delete As String
        Delete = "DELETE FROM TeachersTable "
        Delete &= "WHERE [StudentsNumber] = '" & ID & "'"

        'create object.
        Dim dbConnection As New
            OleDbConnection(ConnectionString)
        Dim dbCommand As New
            OleDbCommand(Delete, dbConnection)

        'Open database and execute
        Try
            dbConnection.Open()
            Dim NumberUpdated As Integer
            NumberUpdated = dbCommand.ExecuteNonQuery()
            lblStatus.Text = NumberUpdated.ToString()
            lblStatus.Text &= " record deleted"

        Catch err As Exception
            lblStatus.Text = "Error deleting data."
            lblStatus.Text &= err.Message

        Finally
            If Not (dbConnection Is Nothing) Then
                dbConnection.Close()
            End If

        End Try

        'cancel edit mode
        dg1.EditItemIndex = -1

        'Rebind the grid.
        Dim ds As DataSet = GetDataSet()
        BindGrid(ds)
    End Sub

    Private Sub btnlogout_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnlogout.Click
        Response.Redirect("WelcomeForm.aspx")
    End Sub

    Private Sub dg1_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
dg1.SelectedIndexChanged

    End Sub
End Class
```

APPENDIX C :USER MANUAL

TABLE OF CONTENTS

USER MANUAL

LIST OF FIGURES

1. OVERVIEW OF THE SYSTEM

Essay Grading System (EGS) is a brief example of automated marking system for History subject. It covers form five students in secondary schools. EGS is a client/server application, which is designed to help school teachers grade their student essays via online system. It also limits the usage of paper. This system is also easy to be used by students and user friendly with simple user interface.

1.1 ABOUT THE MANUAL

This manual is divided into five main headings, which are:

- A brief overview
- Getting started
- EGS basic
- Student module
- Teacher module

1.2 CONVENTIONS

A consistent typographic is used to guide the users to understand the manual better.

Conventions	Descriptions
[button]	Indicates the button name
[hyperlink]	Indicates the hyperlink name

2. HARDWARE AND SOFTWARE REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

Listed below is the hardware requirements needed to run the system.

2.1.1 SERVER

- Minimum Intel Pentium II 450 MHz (or above are recommended as the power of server is essential to provide optimum system performance.
- Hard disk space 100 MB or above.
- Network adapter.

2.1.2 CLIENT

- Minimum Intel Pentium II 450 MHz (or above are recommended)
- Minimum 64 MB RAM (or above are recommended)
- Graphic Adapter
- Keyboard and mouse as input devices
- Network card
- Minimum 16 bit color monitor
- 800 x 600 pixels screen monitor (for optimum screen layout)

2.2 SOFTWARE REQUIREMENTS

The software requirement needed to run the system is:

2.2.1 SERVER

- Windows XP
- Internet Information Server 5.0 and above
- Microsoft Visual Studio .NET

3. GETTING STARTED

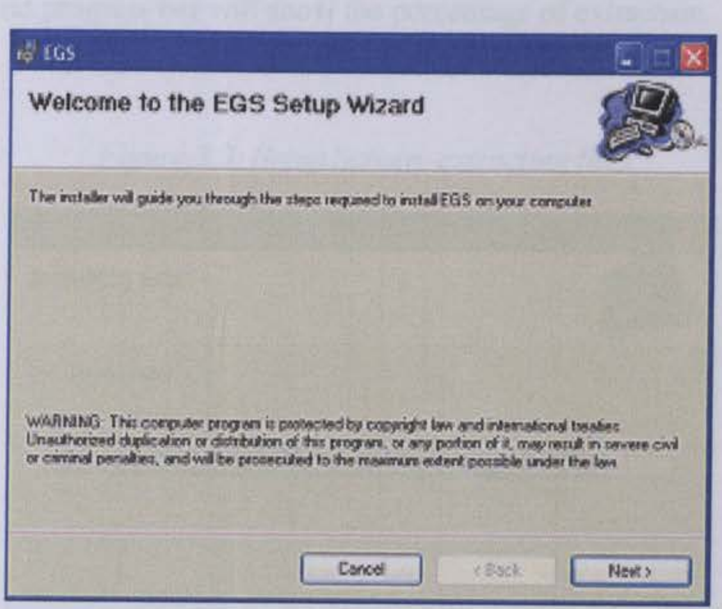
As EGS is a client/server application, it needs an established network connection to put the system into function. Before users are able to run the application, they must meet all those hardware and software requirements and need to go through a setup process.

3.1 EGS INSTALLATION

User must follow the steps below in order to install the system.

1. Execute the setup file with the name and file extension, setup.exe.
2. A setup welcome screen will appear as shown in Figure 3.1. Select [NEXT].

Figure 3.1: Setup Screen



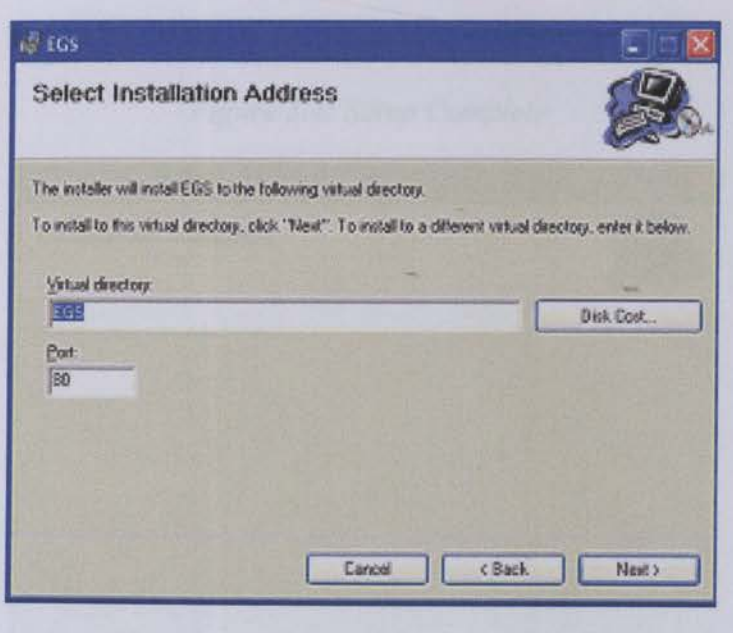
Welcome to the EGS Setup Wizard

The installer will guide you through the steps required to install EGS on your computer

WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.

3. Next you will see Figure 3.2. Change the default folder name to the original folder name and Select [NEXT]. Then Select [NEXT] to the Installation Confirnation.

Figure 3.2: Select Installation Folder



EGS

Select Installation Address

The installer will install EGS to the following virtual directory.

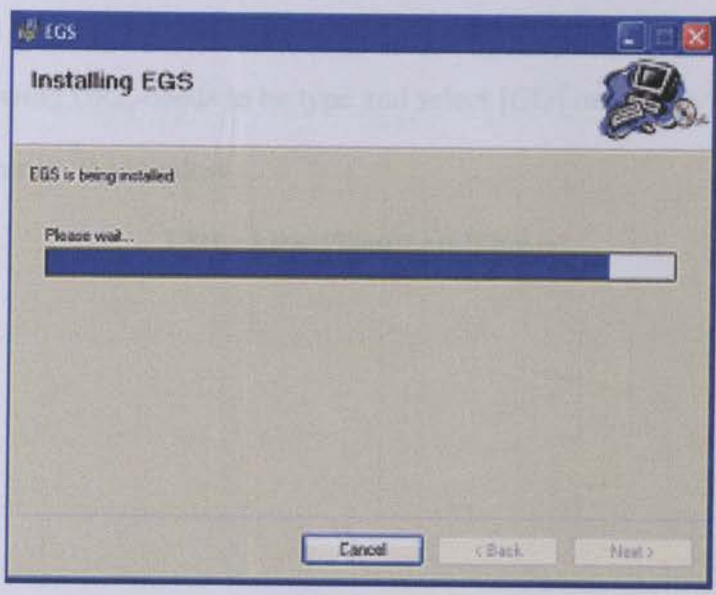
To install to this virtual directory, click "Next". To install to a different virtual directory, enter it below.

Virtual directory: EGS

Port: 80

4. A message as in Figure 3.3 will appear indicating that the files are currently extracted and progress bar will show the percentage of extraction.

Figure 3.3: Setup Screen–extracting files



EGS

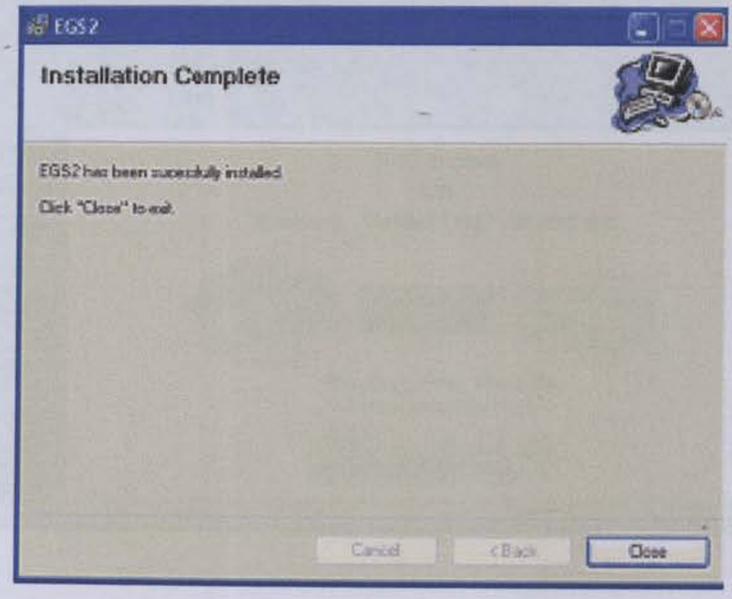
Installing EGS

EGS is being installed.

Please wait...

5. Lastly, it will inform you that setup was completed successfully as shown in figure 3.4. Select [CLOSE] on that window.

Figure 3.4: Setup Complete



Installation Complete

EGS2 has been successfully installed.

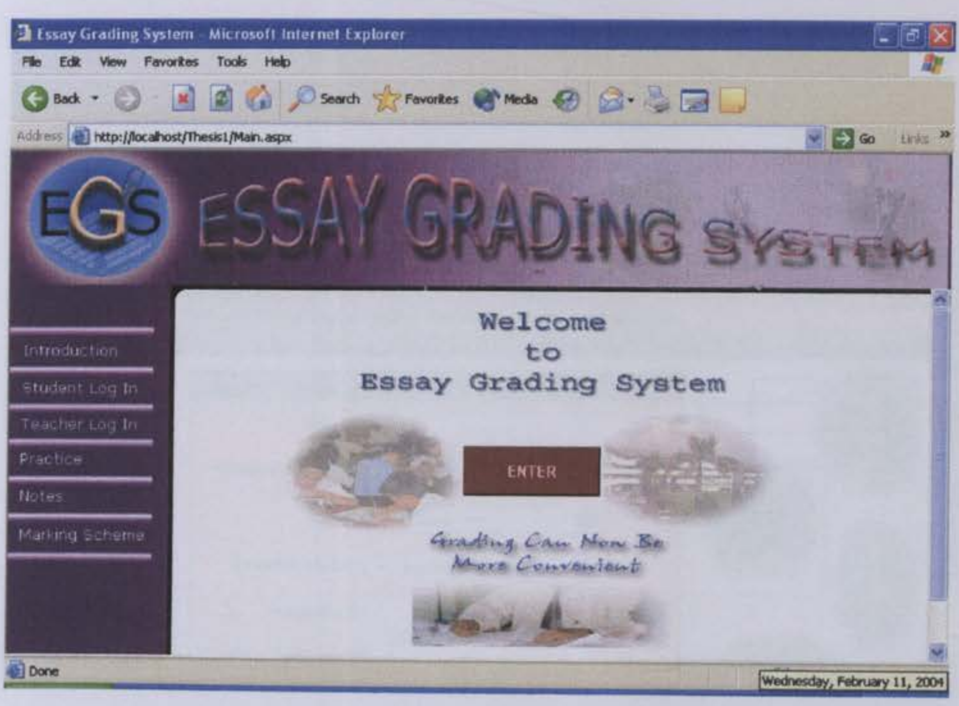
Click "Close" to exit.

3.2 STARTING EGS

To run the system users need to double click on the Internet Explorer icon. Then the following URL needs to be type and select [GO] on it. The following main page will appear on the window.

URL: http://localhost/Thesis1

Figure 3.5: EGS Main page



Welcome to Essay Grading System

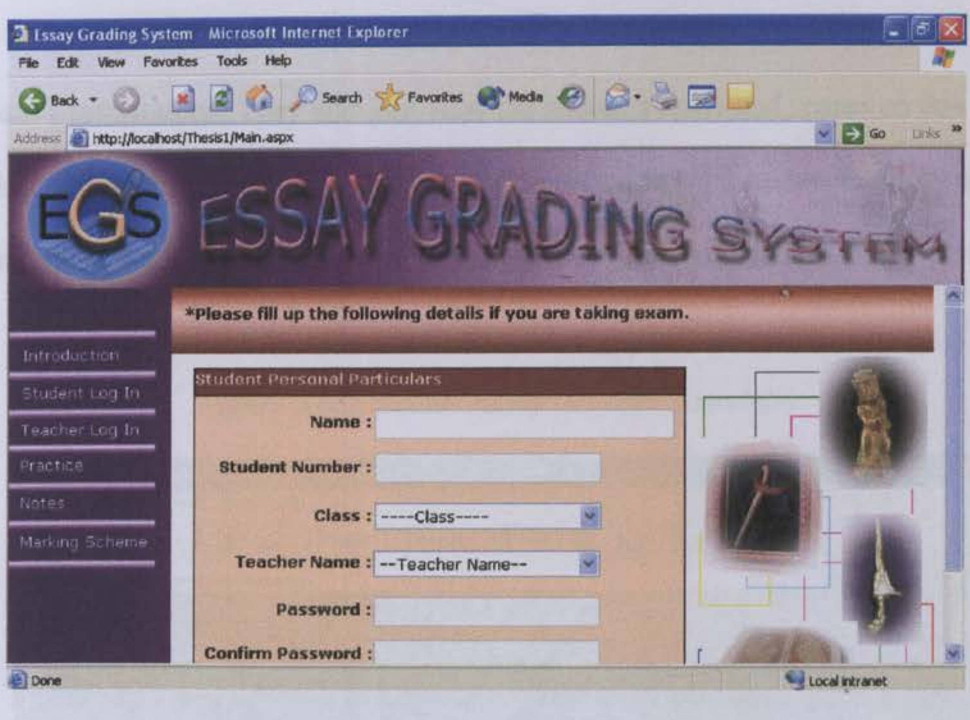
Grading Can Now Be More Convenient

Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
ENTER
Done
Wednesday, February 11, 2004

3.3 Logon

Login form will appear if you clicked on 'student log on' navigation or 'Teacher log on' navigation. The Students and Administrator (Teachers) must enter an authorized username and valid password. Username and password validation will be done on server. If you are a new user you need to fill up the form before you get the username and password. Figure 3.6 shows the registration form.

Figure 3.5: New Users Registration Form


Essay Grading System Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS
ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Done
Local intranet

*Please fill up the following details if you are taking exam.

Student Personal Particulars

Name :	
Student Number :	
Class :	----Class----
Teacher Name :	--Teacher Name--
Password :	
Confirm Password :	

3.4 Logoff

The student and the teachers need to log off after completing their task or to leave the program. This is for security purpose so that no other person uses their task, especially unauthorized person.

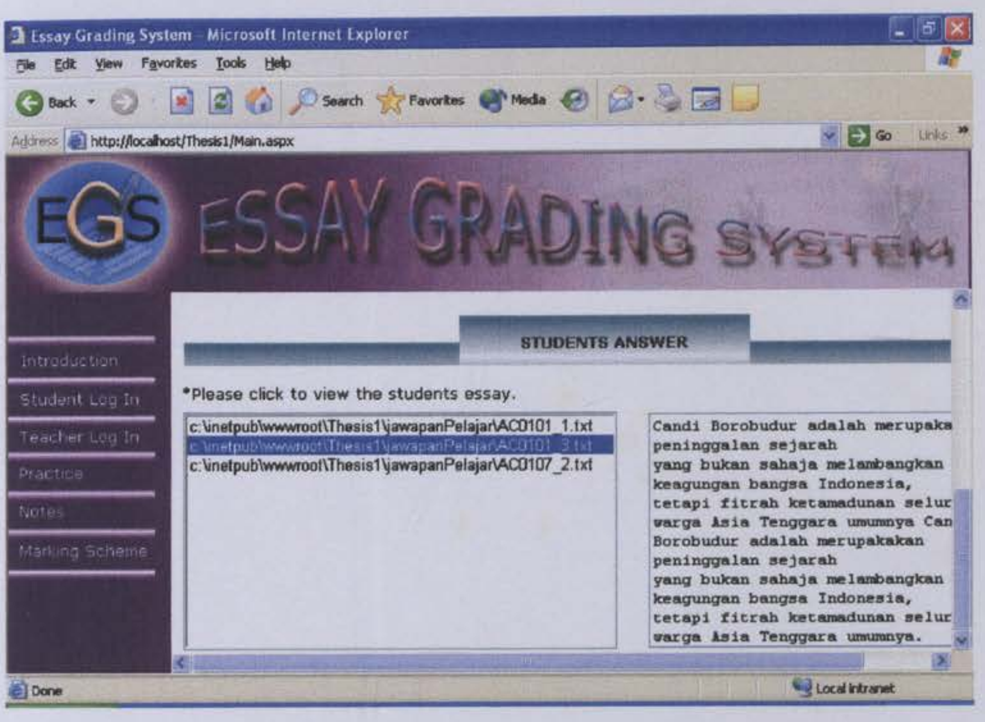
3.5 Students Module

Students can enter the "essay submission page" after logon. Then they need to follow a brief instruction on top of the page to do their task. After complete their essay they need to submit the essay by click on [SUBMIT] and the score will be displayed. Then the student need to click on [SAVE] to save their score and question number on the administrator database.

3.6 Teacher Module

Teachers can access this module to update, delete and view the student's information. They also can view the students essay and scores on this module. To view the student's essay the teacher need to click on the directory path. Figure 3.6 shows the students essay which was saved in the database.

Figure 3.6: Students Essay (teacher's module)


Essay Grading System - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/Thesis1/Main.aspx
Go Links
EGS ESSAY GRADING SYSTEM
Introduction
Student Log In
Teacher Log In
Practice
Notes
Marking Scheme
Done
Local intranet

STUDENTS ANSWER

*Please click to view the students essay.

```
c:\inetpub\wwwroot\Thesis1\jawapanPelajar\AC0101_1.txt
c:\inetpub\wwwroot\Thesis1\jawapanPelajar\AC0101_3.txt
c:\inetpub\wwwroot\Thesis1\jawapanPelajar\AC0107_2.txt
```

```
Candi Borobudur adalah merupaka
peninggalan sejarah
yang bukan sahaja melambangkan
keagungan bangsa Indonesia,
tetapi fitrah ketamadunan selur
warga Asia Tenggara umumnya Can
Borobudur adalah merupakakan
peninggalan sejarah
yang bukan sahaja melambangkan
keagungan bangsa Indonesia,
tetapi fitrah ketamadunan selur
warga Asia Tenggara umumnya.
```