e-Sign: AN ELECTRONIC
SIGN LANGUAGE DICTIONARY
(LETTERS A-M)

LIM PEK SAN
WEK030105
Supervisor: Assoc. Prof. Dr. Ow Siew Hock
Moderator: Assoc. Prof. Raja Noor Ainon Zabariah
Raja Zainal Abidin

FACULTY OF COMPUTER SCIENCE AND
INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
SESSION 2005/2006

ABSTRACT

e-Sign Dictionary is an electronic sign language dictionary that focused on Malaysian Sign Language. It can be a good learning tool for the deaf community as well as the public in general.

e-Sign Dictionary is a trilingual system with instruction in English, Malay and Chinese. Through this system, users will be able to learn not only the sign language of a particular word but also the meaning of the word in English, Malay and Chinese. The synonym and antonym of the word as well as examples of its usage in a sentence are also provided. This system provides three kinds of searching methods that are, search by category, search by alphabetical order and recent search. Most of the words in this system are based on a book named "Belajar Bahasa Isyarat Dalam Sepuluh Jam" (Mr. Tan Yap,1998).

The most attractive feature of this system is the animation. Users can view the sign language of a word from different angles. The animation can be rotated 90° to the left or to the right for viewing. In addition, this system also provides "zoom-in" and "zoom-out" features which allow the users to adjust the size of the animation according to their preferences. They can also choose their favourite animated model (i.e. man, woman, boy or girl) to demonstrate the sign language.

This system was developed using Visual Basic.Net as the programming language, Microsoft SQL Server for managing databases and 3D Poser 6.0 to create the 3D animations. Besides, it has a function to help the users to learn the pronunciation of the words they are searching.

It is hoped that e-Sign Dictionary can bring benefits not only to the deaf community but also to the public at large. In this regard, the primary objective of this system is to foster greater closeness and understanding between the deaf community and the public.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Project Objectives

We cannot deny that there is a communication barrier between the deaf community and the public at large. The e-Sign Dictionary is specifically designed to assist those who are interested in learning the sign language. This system can help to close the gap between the deaf people and the public by giving them a chance to communicate to each other. The main objectives of this project are:

- ❖ To develop an Electronic Sign Language Dictionary that is purely based on Malaysian Sign Language.

- ❖ To provide a learning resource for the deaf community and the public to learn Malaysian Sign Language.

- ❖ To provide a systematic and an attractive method of learning sign language.

- ❖ To improve the communication between the public and the deaf community.

- ❖ To enable the deaf community and the public to learn Malaysian Sign Language at flexible time and location.

- ❖ To bring the deaf community as well as the public closer to the information technology.

The e-Sign Dictionary will be developed according to the objectives mentioned above to ensure that the community will benefit from it.

1.2   Project Team

e-Sign Dictionary development team consists of the following members.

Table 1.1: e-Sign Dictionary Team Members

Name	Role	Responsibility
Associate Professor Dr. Ow Siew Hock	Project Group Supervisor	- Give advices and train all project group members. - Maintain the progress of the e-Sign Dictionary development team.
Tee Say Hong (WEK030223)	Project Group Leader	- Manage the project team so that the team can work co-operatively. - Distribute task to every group member. - Report to project supervisor about the progress of the development. - In charge for the following literature reviews: i. Programming language: Java programming language ii. Database System: Microsoft Access iii. 3D tools: 3D Poser 6.0 iv. Sound tools: Goldwave

		v. Research Methodology: Rapid Application Development • Responsible for documenting a minimum total of 40 alphabet A's English words and the associated meaning. • Responsible for documenting a minimum total of 40 alphabet B's English words and the associated meaning. • Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented English words from alphabet A to B. • Construct the animation characters for all of the English words from alphabet A to B. • In charge in the following modules. i. Password encryption. ii. Admin
Leong Wei Sian (WEK030098)	Project Group Member	• In charge for the following literature reviews: i. Programming language: Visual Basic .net ii. Database System: Paradox

Chin Foh Har
(WEK100060)
	Project
	Group
	Leader
		iii. 3D tools:
		
		    Animoid 3D Movie Maker
		
		iv. Sound tools:
		
		    3D mp3 Sound Recorder
		
		v. Research Methodology:
		
		    Spiral model
		
		• Responsible for documenting a minimum total of 40 alphabet E's English words and the associated meaning.
		
		• Responsible for documenting a minimum total of 40 alphabet F's English words and the associated meaning.
		
		• Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented English words of alphabet E and F.
		
		• Construct the animation characters for all of the English words of alphabet E and F.
		
		• In charge in the following modules.
		
		  i. Admin
		
		  ii. Text-to-Speech

| Lim Pek San (WEK030105) | Project Group Member | • In charge for the following literature reviews:

  i. Programming language:
     See sharp (c#) programming language

  ii. Database System:
     dBase

  iii. 3D tools:
     3D Canvas™

  iv. Sound tools:
     Sound Forge

  v. Research Methodology:
     Waterfall model

• Responsible for documenting a minimum total of 25 alphabet K's English words and the associated meaning.

• Responsible for documenting a minimum total of 25 alphabet L's English words and the associated meaning.

• Responsible for documenting a minimum total of 30 alphabet M's English words and the associated meaning.

• Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented |

		English words of alphabet K, L and M.
		
		 • Construct the animation characters for all of the English words of alphabet K, L and M.
		
		 • In charge in the following modules.
		
		   i. Main Search
		
		   ii. Animation control

Low Wei Yee
 	
 (WEK030120)
	Project
	
	 Group
	
	 Member
		• In charge for the following literature reviews:
		
		   i. Programming language:
		
		      Visual Basic 6.0
		
		   ii. Database System:
		
		      Microsoft SQL Server 2000
		
		   iii. 3D tools:
		
		      Maya Complete
		
		   iv. Sound tools:
		
		      Audacity 1.2.3
		
		   v. Research Methodology:
		
		      Iterative and incremental model
		
		 • Responsible for documenting a minimum total of 40 alphabet C's English words and the associated meaning.
		
		 • Responsible for documenting a minimum total of 40 alphabet D's English words and the associated

		meaning.
		• Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented English words of alphabet C and D.
		• Construct the animation characters for all of the English words of alphabet C and D.
		• In charge in the following module. i. Help and information function

Lu Peik Luan
(WEK030121)
	Project
	Group
	Member
		• In charge for the following literature reviews:
		  i. Programming language:
		     C++ programming language
		  ii. Database System:
		     MySQL
		  iii. 3D tools:
		     3D Sci-Fi Movie Maker
		  iv. Sound tools:
		     Text-to-speech
		  v. Research Methodology:
		     V model
		• Responsible for documenting a minimum total of 40 alphabet G's English words and the associated meaning.

		• Responsible for documenting a minimum total of 40 alphabet H's English words and the associated meaning.
		• Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented English words of alphabet G and H.
		• Construct the animation characters for all of the English words of alphabet G and H.
		• In charge in the following module. i. Report

Soo Hsaio Pei
(WEK030201)
	Project Group Member	• In charge for the following literature reviews:
		
		   i. Programming language:
		       Perl programming language
		
		   ii. Database System:
		       Oracle
		
		   iii. 3D tools:
		       3Ds Max 6.0
		
		   iv. Sound tools:
		       Live Sound Recorder
		
		   v. Research Methodology:
		       Prototyping
		
		• Responsible for documenting a minimum total of

40 alphabet I's English words and the associated meaning.

- Responsible for documenting a minimum total of 40 alphabet J's English words and the associated meaning.

- Responsible for documenting the associated Malay words and meaning, as well as the Chinese words and meaning for all of the documented English words of alphabet I and J.

- Construct the animation characters for all of the English words of alphabet I and J.

- In charge in the following modules.

   i. Search by category and recent search.

   ii. User interface

1.3 Project Scope

The e-Sign Dictionary is developed according to some scopes. It is important to follow these scopes, so that, we can ensure the system will be delivered according to its main purpose, which is to develop an electronic sign language dictionary that is purely based on Malaysian Sign Language. The scopes for the e-Sign Dictionary system are as follow:

i.   Deaf Community and Public

The target users for the e-Sign Dictionary system are the deaf community and the public.

❖ Deaf community

The e-Sign Dictionary can be a useful learning resource for the deaf community. From this system, they will be able to learn the sign for each word. This can help to build an effective communication between the deaf community and the public, as they can use the sign language that they have learned to communicate with one another.

❖ Public

The main targeted user is the public. It is hoped that e-Sign Dictionary can be a good reference for the public to learn sign language. They can learn it at anytime and anywhere that is convenient to them.

ii.   Malaysian Sign Language

All the contents in the e-Sign Dictionary system are based on Malaysian Sign Language and Malaysian culture.

10

iii.   Words

The words that have been covered in e-Sign Dictionary are the words that are started with the alphabet 'A' to the alphabet 'M'. There are about 180 words saved in the database. Anywhere, I am just responsible for the words that are started with the alphabet 'K' to the alphabet 'M', which are about 30 words.

iv.   Trilingual System

The e-Sign Dictionary system will be built in with three types of languages which are Malay, English and Chinese. The meaning for each word will be given not only in English but also Malay and Chinese.

v.   3D graphical animation

The sign for each word in the e-Sign Dictionary is developed by using 3D Poser 6.0. User can rotate the signs into certain degrees, so that, they will have a better understanding on how to sign for a particular word. In other words, user will be able to see the sign for each word in different angles according to their preferences.

vi.   Antonym and Synonym

Besides providing the meaning and the sign of a word, the e-Sign Dictionary also provides the antonym and synonym of the word to the user. This is

important as it can increase the understanding of the user on the word that they are searching.

vii.   Basic meaning of the word

In the e-Sign Dictionary system, the meaning of a word is given in a basic form that is easily understandable by the user.

1.4 Project Schedule

The e-Sign Dictionary system is due to be completed by March 2006. The sooner the project is completed, the sooner the community will benefit from it. This is important as we would like to improve the communication between the public and the deaf community, which is one of the project's objectives. The completed set of e-Sign Dictionary system will have all the functions embedded inside of it from searching function to the pronunciation function.

The project schedule is important as it can give a clear timeline on the project. So, we will be able to deliver our project on time. The schedule is divided into two parts. The first part concentrates mainly on the documentation, whereas the second part is concentrates on the development of the system.

➤ Part 1: This part contains the Project Initiation, Requirement Study, System Analysis and the System Design stage. The first two stages are to be completed in the end of July 2005, while the rest are expected to be completed by the mid of August 2005.

> Part 2: This part contains the Prototyping, Implementation, Testing and Installation phase. Prototyping phase is scheduled to be completed in the end of September 2005, whereas Implementation phase will be ended in the end of January 2006. Testing and Installation will be done after the Implementation. Both tasks are expected to be finished by the end of February. Basically, Part 2 is based on the information collected in Part 1.

The project schedule for the e-Sign Dictionary system is shown in the figure below:

	Task Name	Duration	Start	Finish
1	Project Initiation	7 days	Mon 7/11/05	Sun 7/17/05
2	Requirement Study	14 days	Mon 7/18/05	Sun 7/31/05
3	System Analysis	14 days	Mon 7/25/05	Sun 8/7/05
4	System Design	14 days	Mon 8/8/05	Sun 8/21/05
5	Prototyping	35 days	Mon 8/22/05	Sun 9/25/05
6	Implementation	128 days	Mon 9/26/05	Tue 1/31/06
7	System Testing	14 days	Wed 2/1/06	Tue 2/14/06
8	System Installation	7 days	Wed 2/15/06	Tue 2/21/06
9	Documentation	226 days	Mon 7/11/05	Tue 2/21/06

Figure 1.1: Project Schedule for the e-Sign Dictionary System

1.5 Summary of Contents

This report contains the information about the whole activities that involve in the development of this project, which is from the introduction of the system to the evaluation of the e-Sign Dictionary system. The summary of each chapter are as follow:

Chapter 1: Introduction

This chapter explains mainly about the e-Sign Dictionary system, which includes the introduction, project objective, project team, project scopes and project schedule of the system.

Chapter 2: Project Proposal

Introduction about the deaf community and their related statistic are presented in this chapter. Besides that, all the research methods and the system requirements which includes functional and non-functional requirements will also been discussed in this chapter.

Chapter 3: System design

All the e-Sign Dictionary system's design is presented in this chapter. It ranges from data and process design to input and output design and finally the module and the interface design.

Chapter 4: System Development

The process that involves in the development of this system will be discussed in this chapter. Besides that, all the development tools and technologies that are used in the e-Sign Dictionary system will also been discussed under this chapter.

Chapter 5: System Testing

Testing is one of the important steps that cannot be abandoned in the development of a system. So, all the testing techniques that are used in our system are being discussed in this chapter. Not only that, the testing tools and all the test cases will also been describe in this chapter.

Chapter 6: Discussions and Conclusion

Not to deny that, there are a lot of problems that we have encountered during the development of this system. So, in this final chapter, we are going to describe about the problems that we had encountered as well as the solution that we have found. Besides, the evaluation for the system will also been discussed. All the strengths and the weaknesses of this system will be stated under this chapter. Anywhere, the most important part in this chapter is the conclusion. What we have gained from the development of this system will be described under the conclusion part.

CHAPTER 2 PROJECT PROPOSAL

2.1 Literature Review

Literature review is an important process that must be included in the development of any new project. This process involves several activities such as studying all the technology, tools and methodology that can be used to develop a system. From that, we will be able to choose the most suitable technology, tools and methodology to develop our system.

2.1.1 Deafness

2.1.1.1 Definition of Deafness

Deafness is the result of damage to any part of the ear and the degree of hearing loss depends on the severity of that damage. The implications of hearing loss vary from person to person and are related to the individual's circumstances.

2.1.1.2 Causes of Deafness

Not to deny that, there are actually many causes of hearing loss. Some of them are:

❖ a disease such as mumps or meningitis

❖ certain drugs, such as aspirin in high doses, or antibiotics called amino glycosides.

❖ frequent exposure to loud noise

❖ a serious head injury

- ❖ medical conditions such as Meniere's disease
- ❖ aging (progressive deterioration of hearing in older people, which is a natural part of aging process)

2.1.1.3 Types of Deafness

Hearing losses can either be conductive, perceptive, neural or any combination of it. But, the two main types of deafness are:

i. Conductive deafness

Conductive deafness refers to the situation where sound has difficulty passing through the outer or middle ear. Conductive deafness can be treated medically. Causes of conductive deafness including ear wax, Otitis media, foreign body in ear, ear inflammation, glue ear, Tonsillitis, Adenoiditis, Perforated eardrum and eardrum scarring.

ii. Sensorineural deafness

Sensorineural deafness refers to the situation where the cause of deafness is in the cochlea or hearing nerve. In this case, the sound waves traverse the outer and middle ear normally, but the inner ear is unable to perceive in because of disease in the cochlea of the nerves pathways. There are many causes of sensorineural deafness. Some of them are congenital rubella syndrome, Congenital Syphilis, Jaundice, birth injury, mumps and Meniere's disease.

2.1.1.4 Classification of Deafness

dBHL is used to measure the degree of deafness and hearing loss. dB is the abbreviation for decibels, whereas HL means hearing level. Hearing losses are divided into some categories, such as:

i.   NORMAL: 0 to 19 decibels

       No difficulty following speech.

ii.   MILD DEAFNESS: 20 to 40 decibels

       Have some difficulty following speech, mainly in noisy situations

iii.   MODERATE DEAFNESS: 41 to 70 decibels

       Have difficulty following speech without a hearing aid.

iv.   SEVERE DEAFNESS: 71 to 95 decibels

       Rely a lot on lipreading, even with a hearing aid.

v.   PROFOUNDLY DEAF

       May communicate by using sign language or lipreading.

2.1.1.5 Supporting Devices and Services for Deafness

Nowadays, there are many types of equipment and support devices created to facilitate the deaf or hard of hearing people in their daily life. Some of them are as below:

i.   Hearing aid

A hearing aid is an electronic device, with special and extremely small batteries and consists of three components which are microphone, amplifier and receiver. Several types of hearing aids have been designed to fulfill the needs of the deaf people. For example:

❖ BTE (Behind the Ear) which is worn behind the ear and connected to a plastic ear mold that fit inside the outer ear.

❖ ITE (Inside the Ear) which fit completely in the outer ear and are used for mild to severe hearing loss.

❖ ITC (In the Canal) which customized to fit the size and shape of the ear canal and is used for mild to moderately severe hearing loss.

❖ CIC (Completely in the Canal) which are largely concealed in the ear canal and are used for mild to moderately severe hearing loss.

ii. Textphones

Textphone can be used by people who are severely or profoundly deaf. Textphone have a display screen and a keyboard for the user to type their message and at the same time read what is being typed in reply.

iii. Videophones

Videophone is quite popular among the deaf because it can be used to communicate in sign language.

iv. Loop and infrared system

A loop system helps people who use a hearing aid or loop listener to hear sounds more clearly by reducing background noise. In the other hand, infrared system is an alternative to loop system. Both systems are often can be found in places like theatres, cinemas, banks, shopping centre and train stations.

v. Cochlear implant

Cochlear implant is a small, complex electronic device that can help restore totally or profoundly hearing loss. Cochlear implant electrically stimulates auditory nerve fibers in the cochlea when the organ of Corti has degenerated or fails to develop. The implant is surgically placed under the skin behind the ear. An implant has four basic parts:

➢ A microphone, which picks up sound from the environment,

➢ A speech processor, which selects and arranges sounds picked up by the microphone,

➢ A transmitter and receiver/stimulator, which receive signals from the speech processor and convert them into electric impulses,

> An electrode, which collects the impulses from the stimulator and send them to the brain.

vi. Signaling Device

Signaling devices are those devices that used to get deaf people's attention such as light signaling devices.

NEGERI State	KETURUNAN Race						
	Melayu	Cina	India	BI	BBI	L	Jumlah
Johor	1,413	679	257			81	2,436
Kedah	903	194	66			10	1,173
Kelantan	1,074	36	1			2	1,113
Melaka	423	233			1	3	723
Negeri Sembilan	462	201			1		764
Pahang	526		31			6	674
Perak	916	577	175	2	1	31	1,727
Perlis		18	2			2	273
Pulau Pinang		657	133			4	1,388
Selangor	1,055	566	346			15	1,958
Terengganu	1,123	13	3				1,137
WP Kuala Lumpur	765	683	250			6	1,681
WP Labuan	17	13		1	3		34
Sabah		11			1,341		1,352
Sarawak	150	672			137		1,359
JUMLAH / Total	9,861	4,582	1,416	6	1,484	145	17,692

Sumber : JKMM / Social Welfare Department Malaysia
* Symbol: BI – Muslim Bumis, BBI – Non-Muslim Bumis, L – Others

2.1.2 Deaf Community in Malaysia

According to the Social Welfare Department of Malaysia, there were 17,692 registered
hearing impairment peoples in Malaysia in the end of 2000. [e-pek@k, 2005]
The distribution of deaf population in Malaysia according to state and race is shown in
Table 2.1.

Table 2.1 Registered Deaf Populations in Malaysia by the End of 2000

NEGERI State	KETURUNAN Races						
	Melayu	Cina	India	BI	BBI	L	Jumlah
Johor	1,419	679	257	-	-	81	2,436
Kedah	939	154	68	-	-	10	1,171
Kelantan	1,074	36	1	-	-	2	1,113
Melaka	423	233	63	-	1	5	725
Negeri Sembilan	462	201	97	-	1	3	764
Pahang	526	111	31	-	-	6	674
Perak	958	579	175	2	2	11	1,727
Perlis	255	13	3	-	-	2	273
Pulau Pinang	534	627	123	-	-	4	1,288
Selangor	1,037	560	346	-	-	15	1,958
Terengganu	1122	13	2	-	-	-	1,137
WP Kuala Lumpur	745	680	250	-	-	6	1,681
WP Labuan	17	13	-	1	3	-	34
Sabah	-	11	-	-	1,341	-	1,352
Sarawak	350	672	-	-	337	-	1,359
JUMLAH / Total	9,861	4,582	1,416	4	1,684	145	17,692
Sumber : JKMM / Social Welfare Department Malaysia							

* Symbol: BI - Muslim Bumis, BBI - Non-Muslim Bumis, L - Others

2.1.3 Education for the Deaf People in Malaysia

We cannot deny that education is very important for everyone including the deaf community. There are two types of education for the deaf people in Malaysia, which are:

- ➤ Special education school for the deaf.

- ➤ Integration program for the deaf.

2.1.3.1 Special Education School for the Deaf

Special education schools for the deaf people are also divided into two, which are special education primary school and special education secondary school. The following table shows the number of special primary and secondary school for the deaf people in Malaysia in the year 2002:

Table 2.2: Special School for the Deaf People in Malaysia

Special Education School for the Deaf	
Special Education Primary School	23
Special Education Secondary School	2
Total	25

2.1.3.2 Integration Program for the Deaf

The following table shows the number of integration program for the deaf people in Malaysia in the year 2002:

Table 2.3: Integration Program for the Deaf People in Malaysia

Integration Program for the Deaf		
State	Primary	Secondary
Perlis	0	1
Kedah	0	1
Pulau Pinang	1	0
Perak	6	3
Selangor	10	10
Wilayah Persekutuan	1	4
Negeri Sembilan	1	1
Melaka	2	1
Johor	10	8
Pahang	3	2
Terengganu	0	2
Kelantan	4	1
Sarawak	2	4
Sabah	1	1
Labuan	0	0
Total	41	39

2.1.4 Malaysian Sign Language

Every country will have their own sign language. Malaysian Sign Language (MSL) is the most widely used method of signed communication in Malaysia. It is also known as "Bahasa Isyarat Malaysia (IBM)". MSL developed naturally, as languages do. It uses both manual and non-manual components: hand shapes and movements, facial expression and shoulder movement. MSL is widely used in many parts of Malaysia. It has many dialects, differing from state to state. American Sign Language (ASL) has a strong influence on MSL, but both are different enough to be considered separate languages. Besides, there are some others sign languages in used in Malaysia, such as:

❖ Penang Sign Language (PSL)

It was developed in Malaysia by deaf children, outside the classroom, when oralism was predominant. It is now mainly used by older people, although many younger people can understand it.

❖ Selangor Sign language (SSL or KLSL)

It is also known as Kuala Lumpur Sign Language. It was originally based on American Sign Language (ASL) but has diverged significantly and now it can be considered as a language on its own. It is used mainly in the state of Selangor, rather than Kuala Lumpur. Because of that, the deaf themselves call it Selangor Sign Language. Like Penang Sign Language, it is mainly used by older people, although many younger people can understand it.

❖ Kod Tangan Bahasa Malaysia or Manually Coded Malay (KTBM)

It is the only form of sign language recognized by the government in Malaysia as the language of communication for the deaf. It is also referred to as Bahasa

25

Malaysia Kod Tangan, which is the form found in the Ethnologue. KTBM is adapted from American Sign Language, with the addition of some local signs, and grammatical signs representing affixation of nouns and verbs that used in Malay. It is used in deaf school for the purpose of teaching the Malay language.

2.1.5 Review on Existing Electronic Sign Language Dictionary

Nowadays, new technologies are implied particularly in the development of the equipment and systems for the deaf or hard of hearing people. There are some existing sign language dictionary systems that can be found in the market. Some of the systems are given as a free source while the others will require the customer to purchase it. Below are the examples of the sign language dictionary system that are available in the market.

Name of product: ASL PRO.com Dictionaries

Company: ASL Pro.com

Cost: Free

Features:

- It is a free online dictionary that consist over 6000 ASL signs

- Only provide English version.

- Different people or model to present the sign.

- Video dictionary.

- Speed controlling

Weakness:

- Cannot view the sign from different angles.

- No description of meaning about the words being signs.

- The speed for the sign is fast. Although it provides speed controller to the user, but the movement for each sign is not easily to follow by the user.

- No searching by categories features.

Name of product: American Sign Language Dictionary Gold Edition

Producer: M2K

Cost: $19.95

Features:

- Provide over 2600 signs.

- Include finger spelling.

- Include learning games.

- Searching capabilities in five languages

- Instant access to a sign can be gained by typing a keyword or phrase.

Name of product: British Sign Language Dictionary

Company: Stories in the Air

Cost: Free

Features:

- Online British Sign Language Dictionary (BSL)

- It is a free web application.

- Include facial expression.

- Provide speed controller on the animation of the sign.

Weakness:

- Cannot view the sign from different angles.

- No description of meaning about the words being signs.

- Must install the QuickTime software to watch the sign.

- No sound effects.

Name of product: M2K American Sign language Dictionary Gold

Producer: M2K

Cost: $12.50

Features:

- American Sign Language Dictionary combines text, video examples and animated illustrations into a reference tool.

- Practice more than 2600 signs with video demonstrations and illustrations.

- Built on what we have seen through audio explanations and on-screen descriptions.

- Learn the important skill of finger spelling.

Name of product: The American Sign Language Dictionary

Producer: Multimedia 2000

Cost: $9.24

Features:

- Include over 2600 signs.

- Include finger spelling and learning games.

- Provide searching capabilities in five languages.

- Online connection to download new signs each month.

2.1.6 Review on Technology

2.1.6.1 Programming Language

C#

The C# programming language was developed at Microsoft by a team led by Anders Heljsberg and Scott Wiltamuth. It was designed specifically for the .NET platform as a language that would enable programmers to migrate easily to .NET. This migration is made easy due to the fact that C# has roots in C, C++ and Java, adapting the best features of each and adding new features of its own. Basically, the purpose of C# is twofold:

- It gives access to many of the facilities previously available only in C++, while retaining some of the simplicity to learn of VB.

- It has been designed specifically with the .NET framework in mind, and hence is very well structured for writing code that will be compiled for .NET.

C# is an event-driven, fully object-oriented, visual programming language in which programs are created using an Integrated Development Environment (IDE). With the IDE, a programmer can create, run, test and debug C# program conveniently, thereby reducing the time it takes to produce a working program to a function of the time it would have taken without using the IDE. C# can be viewed at three different perspectives.

- As a .NET programming language that lets us conveniently and easily use the new .NET framework.

- As an object-oriented programming language that let us program in a natural object-oriented way.

- As an intermediate level programming language that combines the best features of the existing pre-.NET language of VB and C++.

2.1.6.2   Database System

dBase

dBase is a popular database management system produced by Ashton Tate Corporation. The original version, called Vulcan, was created by Wayne Ratliff. In 1981, Ashton-Tate bought Vulcan and marketed it as dBase II. Subsequent versions with additional features are known as dBase III, dBase III+ and dBase IV, all of which were owned and developed by Borland Corporation (now Inprise), but were sold to dBase Inc. in March 1999.

The dBase format for storing data has become a de facto standard and is supported by nearly all database management and spreadsheet systems. Even systems that do not use the dBase format internally are able to import and export data in dBase format. dBase's underlying file format, the dbf file, is widely used in many other applications needing a simple format to store structural data. So, it can say that a major legacy of dBase is its .dbf file format, which has been adopted in a number of other applications. Besides,

dBase's database system was one of the first to provide a "header" section for describing the structure of the data in the file. This meant that the program no longer required advance knowledge of the data structure, but rather could ask the data file how it was structured.

As with any software package, dBase has a number of limitations when applied to scientific databases and systems. The number of fields available is limited so that a number of files must be set up to handle all the data elements. Having fixed field lengths is justified for administrative systems, but not for scientific data. For example, a good deal of disk space is wasted when the chemical name filed lengths must be set using the longest chemical name available. Entering data using the dBase numeric field is necessary for searching, but creates a problem in printouts. If a parameter field is empty, the dBase numeric default is 0 (zero). Thus, it is possible for someone easily to misinterpret a blank as a data value. Conversion of the data to character information for printout is a solution, but clearly involves additional time and storage.

2.1.6.3 3D Tool

3D Canvas

3D Canvas is a real-time, 3D-modelling and animation tool that incorporates a drag-and-drop approach. Complex models can be constructed from simple 3D primitives or created using 3D Canvas's object building tools. By using 3D Canvas, creating an animated scene is as simple as positioning the objects within our scene for each point in time in our animation. 3D Canvas will ensure the resulting animation runs smoothly.

We can paint 3D objects and animate their shapes and positions in real time, or record to an AVI video file for distributing our creation. The 3D Canvas interface is extremely intuitive, with objects and palettes conveniently accessible from clearly visible tabs and panels. This freeware version of 3D Canvas is an excellent program with a multitude of varied features and plenty of things to play with. In version 6.5.1, six new import formats are available. There are MD2, Biovision Motion Capture (bvh), Neutral Object Format (nof), Raw Triangles (raw), Wavefront (obj) and WorldToolKit (nff).

2.1.6.4   Sound Tool

Sound Forge

Sound Forge is an award-winning digital audio editor that includes a powerful set of audio processes, tools and effects for recording and manipulating audio. This industry standard application is the professional's choice for audio editing, audio recording, effects processing, streaming content creation and more. Sound Forge 7.0 software continues to build on its decade-long legacy by adding features that make it more powerful than ever to enhance our workflow and productivity. These include DirectX plug-in effects automation; automated time-based recording and audio-threshold record triggering; VU/PPM meters for RMS playback and record monitoring; enhanced spectrum-analysis tools; white-, pink- and brown- noise generator.

Audio-editing applications have a reputation for clutter. The recently rebranded Sony Sound Forge 7.0 avoids that trap with a sleek layout that houses professional 32-bit audio and endless features. For audio-editing applications, it's par for the course to cram as many features as possible into a tiny toolbar at the top of the screen, assigning a vague

icon to each function. Anywhere, Sound Forge doesn't fall into that trap. This industry-standard audio editor sports a clean layout, a massive feature list and up-to-32-bit professional-quality audio.

Sound forge 7.0 software includes many new features and enhancements, such as automated time-based recording; audio threshold record triggering; VU meters for recording and playback; enhanced spectrum Analysis Tool; DirectX Plug-in effects automation; White, Pink and Brown noise generators; Media Explorer Previewing; Project File Creation; support for 24fps native DV video files and more. The application includes a set of customizable processes, effects and tools for manipulating audio, creating steaming media and also offers support for a range of file import and export option for most industry formats.

2.1.7 Research Methodologies

Generally, the Software Development Life Cycle (SDLC) covers the whole life of software. It covers not only feasibility study, analysis specification, design and development but also the operation, maintenance and enhancement aspects which take place after the software has been accepted by the end users. So, in another words, a software lifecycle depicts the significant phases of activities of a software project from conception until the product is retired.

The selection of an appropriate Software Development Life Cycle is important for the project. It brings us into the project with a clear idea of the general form that the development is going to take.

Waterfall Model

The waterfall model was derived from engineering model to put some order in the development of large software products. It consists of different stages which are processed in a linear fashion. [Adrian Als and Charles Greenidge, 2003]

The phases in the waterfall model are as follow:

❖ Requirement analysis phase

In the requirement analysis phase, all the problems are specified along with the desired project's objectives (goals). Besides, the constrained are identified as well.

❖ Specification phase

In the specification phase, the project specification is produced from the detailed definition of the requirement analysis. This document will clearly define the software function.

❖ Design phase

In the design phase, the problem definition specified in the earlier phase is translated into design which will solve the problem. The design phase is the bridge between what the user wants and the code that will be created to satisfy the requirements.

❖ Implementation and testing phase

In the implementation and testing phase, the designs are translated into the software domain. Detailed documentation from the design phase can significantly reduce the coding effort. On the other hand, testing at this stage

is focuses on making sure that the errors are identified and the software meets its required specification.

❖ Integration and system testing phase

In the integration phase, all the program units are integrated and tested to ensure that the complete system meets the software requirements. After this stage, the software is delivered to the customer.

❖ Maintenance phase

In this phase, the software is updated to:

> Meet the changing customer needs.

> Adapted to accommodate changes in the external environment.

> Enhancing the efficiency of the software.
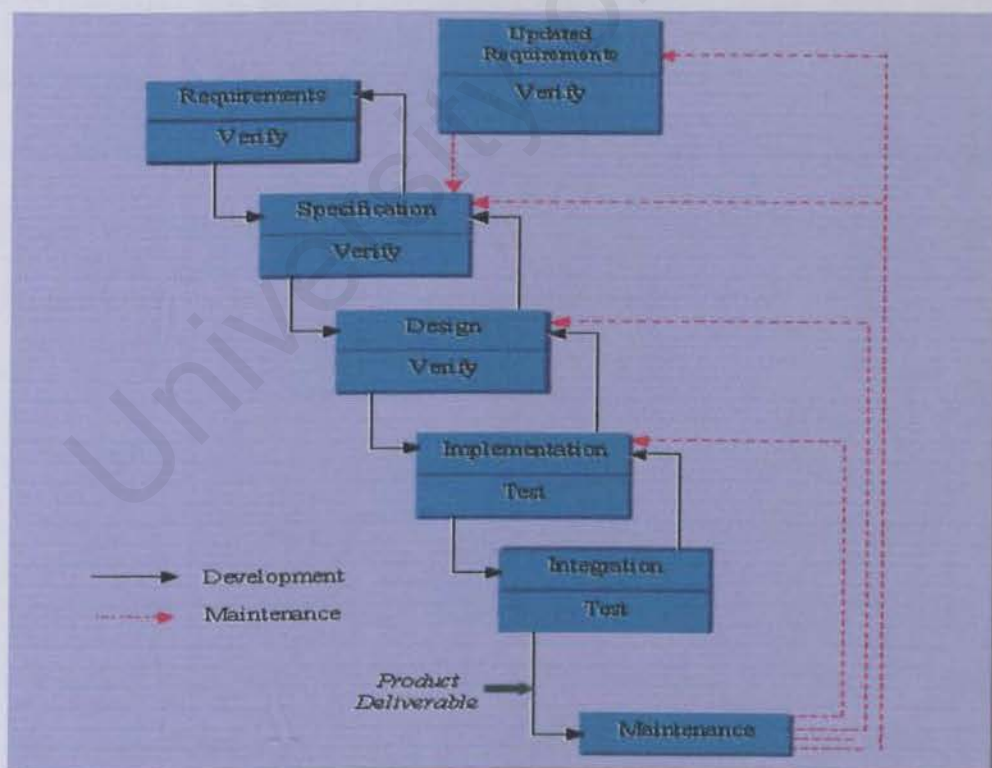
Figure 2.1 Waterfall Model

Advantages of waterfall model:

- ✓ Testing is inherent to every phase of the waterfall model.
- ✓ It is documentation driven, that is, documentation is produced at every stage.
- ✓ It is an enforced disciplined approach.

Disadvantages of waterfall model:

- ✓ Real projects rarely follow the sequential flow that the model proposes.
- ✓ The model assumes that requirements are clearly specified at the beginning of the project. The model has no mechanism to handle changes to the requirements that are identified.
- ✓ Customer only sees a working version of the product after it has been coded.

2.2 Research Methods

2.2.1 Requirements Elicitation Technique

Requirements elicitation involves understanding the application domain, the specific problem to be solved and the specific facilities needed by system stakeholders. [Gerald Kotonya and Ian Sommerville, 2003]

The techniques that have been used in gathering the requirements for this system are as below.

2.2.1.1 Survey

One of the requirement elicitation techniques is to conduct survey through questionnaire. Questionnaire is a special purpose document that allows the analyst to collect information and opinions from the respondents. The document can be mass produced and distributed to the respondents, who will then complete the questionnaire on their own. The purpose of the questionnaire is to survey how much or how deep did the people know about the sign language and the sign language dictionary. This will give us some feedback and ideas on the development of our system.

2.2.1.2 Interview

Interviewing is a technique that can be used to elicit information from those people that are knowledgeable in certain relevant field. It is through the use of interviews that a developer can identify those areas of the project that make up the system. By using interviews, detailed information can be obtained from the stakeholders about such things as workflows and factors that affect the operation of the system. My interviewees are Mr.

Tan Yap and the teachers in the deaf school. The reason I chose them is because they are the most experience and suitable people that can give me some information about the deaf community and the sign languages. From the interviews, I hope that I can get some valuable feedbacks and suggestions about the features that can be incorporated into the e-Sign Dictionary system.

2.2.1.3 Group Discussion

During the requirement elicitation phase, discussion has been carried out with my group members to discuss about the details of the system. Any uncertainty is discussed during the discussion. From the interview, ideas have been generated to point out every aspect that can be improved and ways to improve them in order to get a better result. So, I can say that it is important to have a group discussion as it allows brainstorming among our group members.

2.2.1.4 Consultation with Supervisor

During the requirement gathering process, we have consulted our Supervisor, Prof. Dr. Ow Siew Hock for her valuable suggestions and opinions on our e-Sign Dictionary system. Prof. Dr. Ow Siew Hock has given us some ideas and suggestions about the features that should be incorporated into the system. Besides, she had showed us some existing systems which were done by our seniors. From that, she pointed out some of the strength and weaknesses of the particular system, which I think it is really important and helpful in the development of our system.

2.2.1.5 Direct Observation

Not to deny that, direct observation is also an important technique to elicit information. We have made some observation on the existing sign language dictionaries that can be found in the internet. From that, we will be able to come out with the good features for the e-Sign Dictionary. Not only that, we will also be able to point out some of the weaknesses of the existing system and try to avoid it in the development of our system.

Table 2.4: Comparisons of Existing Electronic Sign Language Dictionary Software

Features	ASL PRO.com Dictionary	American Sign Language Dictionary Gold Edition	British Sign Language Dictionary	M2K American Sign Language Dictionary Gold	The American Sign Language Dictionary
Sign Language	American Sign Language	American Sign Language	British Sign Language	American Sign Language	American Sign Language
Instruction Language	English	English	English	English	English
Media	Web-based	CD-ROM	Web-based	CD-ROM	CD-ROM
Alphabets	Yes	Yes	Yes	Yes	Yes
Phrases	Yes	Yes	Yes	Yes	Yes
Description of meaning	No	Yes	No	Yes	Yes

2.2.2 Comparison with Existing Systems

Not to deny that, there are many electronic sign language dictionaries available in the market. These dictionaries come with a lot of new features and functions that bring benefit not only to the deaf community but also the public. Five existing electronic sign language dictionaries have been studied. Below is the comparison table for these sign language dictionaries.

Table 2.4: Comparison of Existing Electronic Sign Language Dictionary Software

Features	ASL PRO.com Dictionary	American Sign Language Dictionary Gold Edition	British Sign language Dictionary	M2K American Sign Language Dictionary Gold	The American Sign Language Dictionary
Sign Language	American Sign Language	American Sign Language	British Sign Language	American Sign Language	American Sign Language
Instruction Language	English	English	English	English	English
Media	Web-based	CD-ROM	Web-based	CD-ROM	CD-ROM
Alphabets	Yes	Yes	Yes	Yes	Yes
Numbers	Yes	Yes	Yes	Yes	Yes
Description of meaning	No	Yes	No	Yes	Yes

40

Spelling	Yes	Yes	Yes	Yes	Yes
Audio	No	No	No	Yes	No
Video	Yes	Yes	Yes	Yes	Yes
3D Graphics	No	No	No	Yes	No
Graphic rotation	No	No –	No	Yes	No
Platform	Windows	Windows	Windows	-	Windows
Search by Category	No	No	Yes	No	No

2.2.3 Comparison of Programming Languages

Six programming languages have been compared and the result is showed in the table below.

Table 2.5: Comparison of Programming Languages

Features/Tools	Perl	Visual Basic	Visual Basic.NET	Java	C#	C++
Full object – orientation	No	No	Yes	Yes	Yes	Yes
Case Sensitive	Yes	No	No	Yes	Yes	Yes
Build web-based and window-based	Yes	Yes	Yes	Yes	Yes	Yes

application						
Full implementation inheritance and polymorphism	Yes	No	Yes	Yes	Yes	Yes
Flexible, simple data access	Yes	Yes	Yes	Yes	Yes	Yes
XML web services	No	No	Yes	Yes	No	No
Difficulty to learn	Easy	Easy	Easy	Moderate	Easy	Easy
Easiest, and productive	Yes	Yes	Yes	Yes	Yes	Yes
Availability of tutorial material	Vast	Vast	Vast	Yes	Vast	Vast

- ✓ Perl – Please refer to Soo Hsaio Pei's report under section 2.1.6.1

- ✓ Visual Basic - Please refer to Low Wei Yee's report under section 2.1.6.1

- ✓ Visual Basic.NET – Please refer to Leong Wei Sian's report under section 2.1.6.1

- ✓ Java – Please refer to Tee Say Hong's report under section 2.1.6.1

- ✓ C++ - Please refer to Lu Peik Luan's report under section 2.1.6.1

42

2.2.3.1 Why Visual Basic.NET?

Visual Basic.NET has been chosen as the authoring tool for our e-Sign Dictionary system. This is because VB.NET provides a complete set of system development tool for web application as well as standalone system. There are many Rapid Application Development (RAD) tools that can be found in VB.NET, such as drag-and-drop form design, popular "code behind" forms model and automatic control anchoring.

Not to deny, VB.NET has a user-friendly interface that is easy-to-use and easy-to-learn. So, it will not be a burden for us in mastering this language. Besides, we can obtain most of the important function that suits our system from the library class of VB.NET. It provides methods that enable the interaction between graphics, audio, video and animation. This is important as our system includes a lot of animation to display the sign for each word.

Bear in mind, VB.NET is inexpensive and affordable by small and moderate developers. It makes the coding process becomes easier and at the same time enhancing the performance and the security of the system.

2.2.4 Comparison of 3D Tools

Six 3D tools have been compared and the result is showed in the table below.

Table 2.6: Comparison of 3D Tools

Features/Tools	3D Poser 6.0	Maya Complete	3D Sci-Fi Movie Maker 1.09	Animoids 3D Movie Maker 2.2.07 DL	3D Studio Max 8	3D Canvas
Animation	Yes	Yes	Yes	Yes	Yes	Yes
Modeling	Yes	Yes	Yes	Yes	Yes	Yes
Visual effects	Yes	Yes	Yes	Yes	Yes	Yes
Rendering	Yes	Yes	No	Yes	Yes	Yes
Texture mapping	Yes	No	No	No	Yes	Yes
Brush – based technology	No	Yes	No	Yes	No	No
Lighting	Yes	No	No	Yes		No
Intuitive user interface	Yes	Yes	Yes	Yes	Yes	Yes
Games support	No	No	Yes	Yes	Yes	Yes
Materials and mapping	Yes	No	No	No	Yes	No
Cameras	Yes	No	Yes	Yes	Yes	Yes

Modeler workflow	Yes	No	No	No	No	No
Viewport interaction	No	No	No	No	Yes	No
MAXScript®	No	No	No	No	Yes	No
Layout workflow	No	Yes	No	No	No	No
SDK-LScript	No	No	No	No	Yes	No
Fast 2D interface	No	No	No	No	No	Yes
Tree lab	No	Yes	No	No	No	No

- ✓ 3D Poser 6.0 – Please refer to Tee Say Hong's report under section 2.1.6.3

- ✓ Maya Complete – Please refer to Low Wei Yee's report under section 2.1.6.3

- ✓ 3D Sci-Fi Movie Maker 1.09 – Please refer to Lu Peik Luan's report under section 2.1.6.3

- ✓ Animoids 3D Movie Maker 2.2.07 DL – Please refer to Leong Wei Sian's report under section 2.1.6.3

- ✓ 3D Studio Max 8 – Please refer to Soo Hsaio Pei's report under section 2.1.6.3

2.2.4.1 Why 3D Poser 6.0?

3D Poser 6.0 was chosen as the 3D graphic and animation development tool for our e-Sign Dictionary system. This is because 3D Poser 6.0 provides ready-to-use human model that include male, female and children. This feature or function is important for us as it can help to save our time on developing the human model. Besides, this tool is easy to master and affordable. So, it is the most suitable tool for us as we do not have any knowledge on creating 3D graphics.

Not only that, 3D Poser 6.0 also gives a detail and direct realistic 3D animation to the user. It able to manipulate individual body parts and facial expressions into any pose with hundreds of morph targets. It also gives a detail poses by using multiple light sources to cast highlights and shadows. This is very useful for our system because we need to show the sign for each word to the users accurately.

In general, we can say that 3D Poser 6.0 is able to direct animation of figures, body parts, lights, cameras, props, morph targets and materials to achieve intricate 3D animations. So, it is suitable to be used in developing our system.

2.2.5 Comparison of Sound Tools

Six sound tools have been compared and the result is showed in the table below.

Table 2.7: Comparison of Sound Tools

Features/ Tools	Sound Forge 8	Gold Wave	Audacity 1.2.3	3D MP3 Sound Recorder	Text To Speech	Wave Flow 5.1
DirectMusic Support	Yes	Yes	Yes	Yes	No	Yes

DirectShow Support	Yes	No	Yes	No	No	No
DirectSound Support	Yes	Yes	Yes	Yes	Yes	Yes
Unlimited simultaneous sounds	Yes	Yes	Yes	Yes	No	Yes
Ability to make unique sounds	No	Yes	Yes	Yes	Yes	Yes
Hardware or software sound mixing	No	No	Yes	No	Yes	No
Sound Editing	Yes	Yes	Yes	No	No	Yes
Movie playing	Yes	No	Yes	No	No	No
3D Sound Support	No	No	Yes	No	No	No
Effects (Reverb, Echo)	Yes	Yes	Yes	No	Yes	Yes

Format supported	MP3, WAV, AIF/SND, AU/SNG, AVI, DIG/SD, IVC, MOV, MPG, OGG, PCA, QT, RAW, RM, SWF, VOX, W64, WMA, WMV	MP3, WAV, WMA, AU, IFF, VOC, SND, MAT, AIFF	WAV, AIFF, MP3	MP3, WAV	WAV, AIFF, AU	MP3, WAV

✓ Goldwave – Please refer to Tee Say Hong's report under section 2.1.6.4

✓ Audacity 1.2.3 – Please refer to Low Wei Yee's report under section 2.1.6.4

✓ 3D MP3 Sound Recorder – Please refer to Leong Wei Sian's report under section 2.1.6.4

✓ Wave Flow 5.1 – Please refer to Soo Hsaio Pei's report under section 2.1.6.4

✓ Text to Speech – Please refer to Lu Peik Luan's report under section 2.1.6.4

2.2.5.1 Why Text-to-Speech?

In our e-Sign Dictionary system, we have a function that can teach the users on the actual pronunciation of a word. So, we have chosen to use Text-to-Speech module to build this function. This software has been chosen because both of the English and Malay

Text-to-Speech are freeware. The Malay speech synthesizer and the English speech synthesizer (which provided by Visual Basic Microsoft Agents) have high interoperability and easy to be incorporated into our e-Sign Dictionary system.

Moreover, by using speech synthesizer, we can reduce the size of the whole system. This is important because in future, we may add in more sign into the system. Besides, it can help to reduce the times taken to record audio files for each and every word.

2.2.6 Comparison of Database Systems

Six database systems have been compared and the result is showed in the table below.

Table 2.8: Comparison of Database System

Features/Tools	Oracle	Microsoft Access	MSSQL Server	MySQL	DBase	Paradox
Database access by .Net application using ADO.NET	Yes	No	Yes	No	No	No
Processing using SQL Statement	No	Yes	Yes	Yes	No	Yes
Query Analyzer for database query	Yes	Yes	Yes	No	Yes	No
Support for XML	No	No	Yes	No	No	No

Format						
Able to use HTTP to send queries to the database	No	No	Yes	Yes	No	Yes
Backup Wizard	Yes	Yes	No	Yes	No	Yes
High Reliability	Yes	Yes	Yes	Yes	Yes	Yes
Easy Installation and Use	Yes	Yes	Yes	Yes	Yes	Yes
Supports from other software	Yes	Yes	Yes	Yes	No	No
Application use to run small business	Yes	Yes	Yes	Yes	Yes	Yes
Personal applications	Yes	Yes	Yes	Yes	Yes	Yes
Corporation-wide applications	Yes	Yes	No	Yes	No	Yes
Front-end for enterprise-wide client or server applications	Yes	Yes	Yes	No	No	Yes
Departmental	Yes	Yes	Yes	No	No	Yes

applications						

✓ Oracle – Please refer to Soo Hsaio Pei's report under section 2.1.6.2

✓ Microsoft Access – Please refer to Tee Say Hong's report under section 2.1.6.2

✓ SQL Server – Please refer to Low Wei Yee's report under section 2.1.6.2

✓ MySQL – Please refer to Lu Peik Luan's report under section 2.1.6.2

✓ Paradox – Please refer to Leong Wei Sian's report under section 2.1.6.2

2.2.6.1 Why Microsoft SQL Server?

Microsoft SQL Server has been chosen to manage the database for our e-Sign Dictionary system. This is because it has a Query Analyzer for databases query and it is processed using SQL statement. So, it will not be a big problem for us to write a query for the database, as most of us already have some knowledge on writing SQL statement.

Not only that, Microsoft SQL Server provides backup wizard that scheduled database backups. This is important as we need to have a backup for our database to prevent any unpredictable situation that brings harm to our data. Another point that makes SQL Server important is, it can be accessed by .NET application using ADO.NET. It is fully integrated with Microsoft Office, BizTalk Server and Microsoft Commerce Server.

2.2.7 Comparison of Research Methodologies

Six software development lifecycle have been compared and the result is showed in the table below

Table 2.9: Comparison of Software Development Life Cycle

Features/Tools	Waterfall Model	Proto-typing Model	Iterative and Incre-mental Model	Spiral Model	V-Model	Rapid Applica-tion Develop-ment (RAD)
Allow user feedbacks	Yes	Yes	Yes	Yes	Yes	Yes
Complexity of project	Low	Medium	High	High	Low	Medium
Product technology	Existing	New	New	New	Existing	New
Availability of resources	All	Some	Some	Some	All	Some
Risk management perspective	No	Yes	No	Yes	No	Yes
Iteration and allow rework	No	Yes	Yes	Yes	Yes	Yes
Fast development	Yes	Yes	Yes	No	No	Yes

Schedule constraint	Medium	Low	Medium	Medium	Medium	Medium
Problem domain knowledge	High	Fair	Poor	Poor	High	High
Understanding of user requirements	Specific	Vague	Vague	Vague	Specific	Specific

- ✓ Prototyping – Please refer to Soo Hsaio Pei's report under section 2.1.7

- ✓ V-Model – Please refer to Lu Peik Luan's report under section 2.1.7

- ✓ Iterative and Incremental Model – Please refer to Low Wei Yee's report under section 2.1.7

- ✓ Spiral Model – Please refer to Leong Wei Sian's report under section 2.1.7

- ✓ Rapid Application Development – Please refer to Tee Say Hong's report under section 2.1.7

2.2.7.1 Why Waterfall Model and Prototyping Model?

Waterfall model and prototyping model have been chosen in developing our e-Sign Dictionary system. Both models have been chosen because each of it has its own advantages that benefited our system.

Waterfall model provides sequencing of activities and it addresses elements of quality management through verification and validation. It also provides configuring management by base lining products at the end of the stage. Besides, it is documentation-driven, which mean, documentation is produced at every stage. This is important because we need to have a complete documentation for our system.

As for prototyping model, it can help to improve the communication between the developers and the end users. Definitely, this will bring benefit to us as we will be able to understand the needs of our system stakeholders. Not only that, in prototyping model, the proposed system is expected to have considerable user interface.
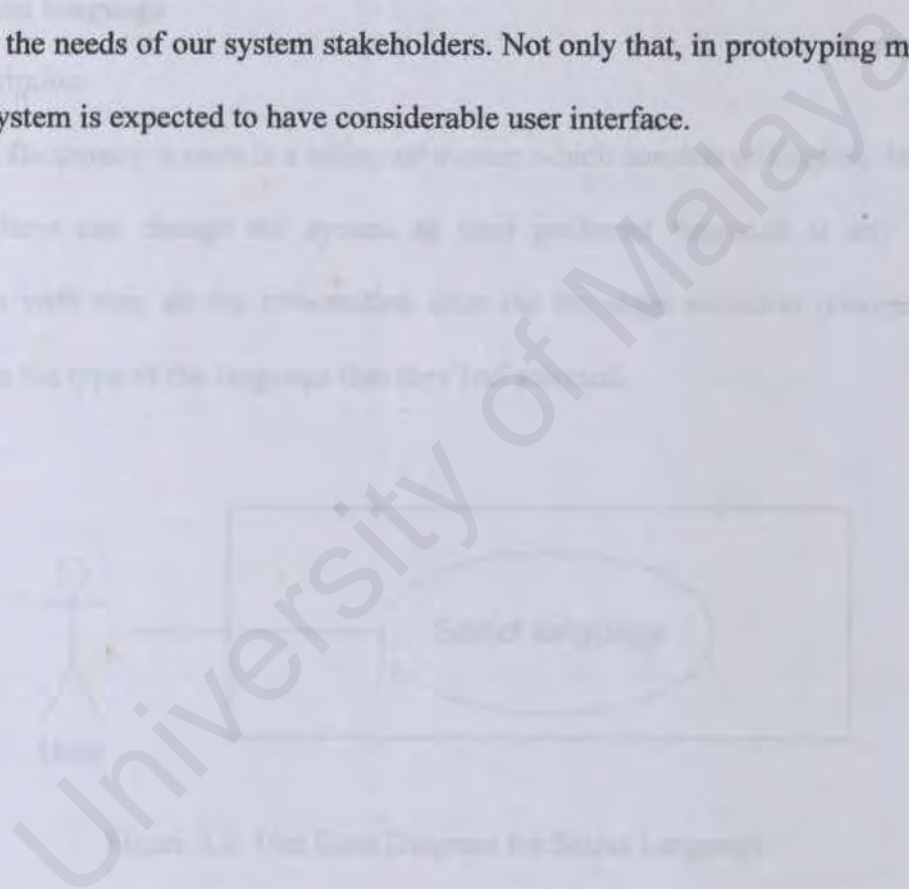
2.3 Functional and Non-Functional Requirements

2.3.1 Functional Requirements

Functional requirements are a statement of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. The functional requirements for the e-Sign Dictionary system are as follow:

2.3.1.1 Select language

Brief Description:

The e-Sign Dictionary system is a trilingual system which consists of English, Malay and Chinese. Users can change the system to their preferred language at any time. In conjunction with this, all the information after the language selection process will be displayed in the type of the language that they had selected.
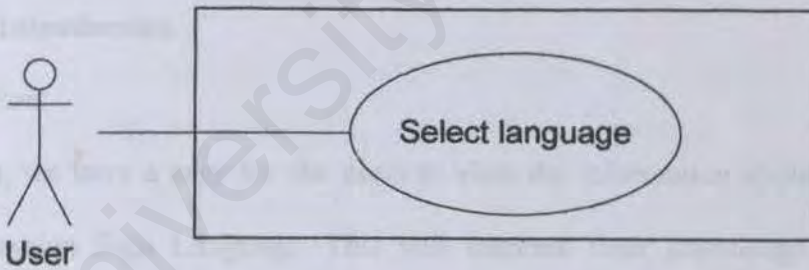
Figure 2.2: Use Case Diagram for Select Language

2.3.1.2 Select animation model

Brief description:

At the startup of the system, users are required to select the animation model. In our e-Sign Dictionary system, we have provided four types of human models for the users to choose. There are man model, woman model, little boy model and little girl model. So, users are required to choose one of the human models. All the sign of a word will be showed using the selected model.
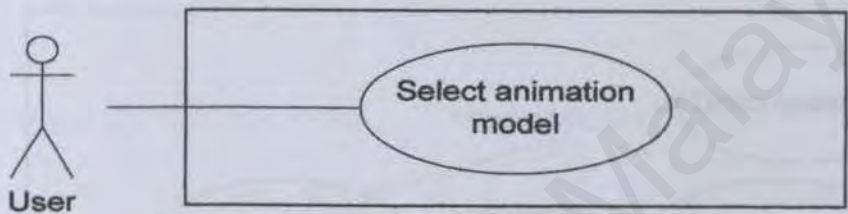

Figure 2.3: Use Case Diagram for Select Animation Model

2.3.1.3 View Introduction

Brief description:

In our system, we have a page for the users to view the information about the deafness and the Malaysian Sign Language. This will increase their knowledge on the deaf community as well as the Malaysian Sign language.
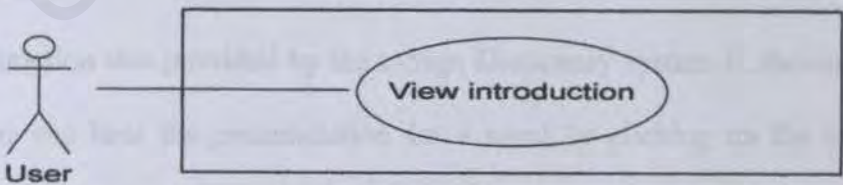
Figure 2.4: Use Case Diagram for View Introduction

2.3.1.4 Search word

Brief description:

The main function of this system is to allow user to search the sign language and the meaning for a particular word. User can choose to search according to the category, last search or direct search. Direct search provides two capabilities; either users type in the full spelling of a word or a portion of it. The system will return the meaning to the users in few seconds.
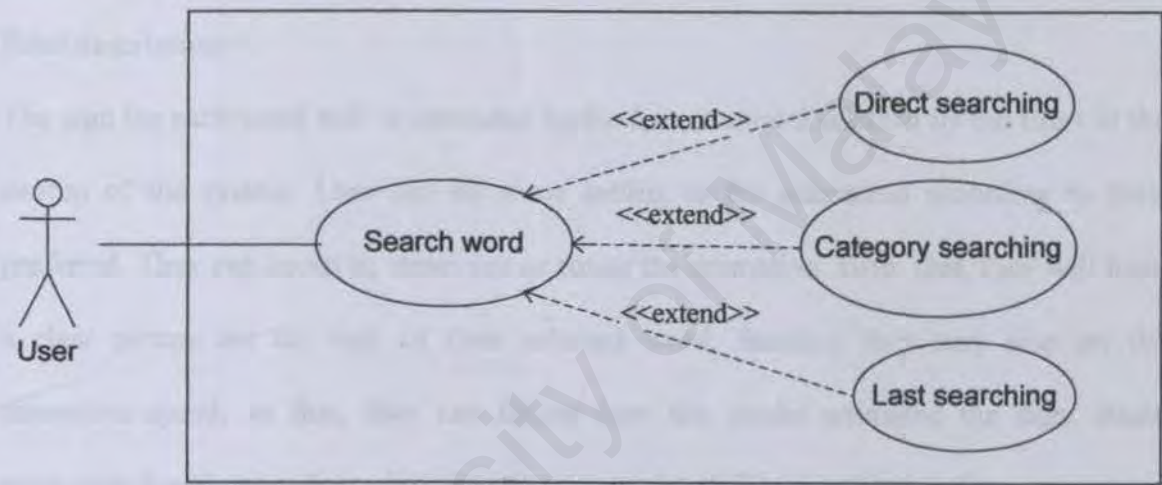
Figure 2.5: Use Case Diagram for Search Word

2.3.1.5 Play sound speech

Brief description:

One of the function that provided by the e-Sign Dictionary system is showing the sound speech. Users can hear the pronunciation for a word by clicking on the specific icon. From that, they will be able to learn the actual pronunciation of a word. Users can set the volume of the sound speech or playback the sound according to their preferred.
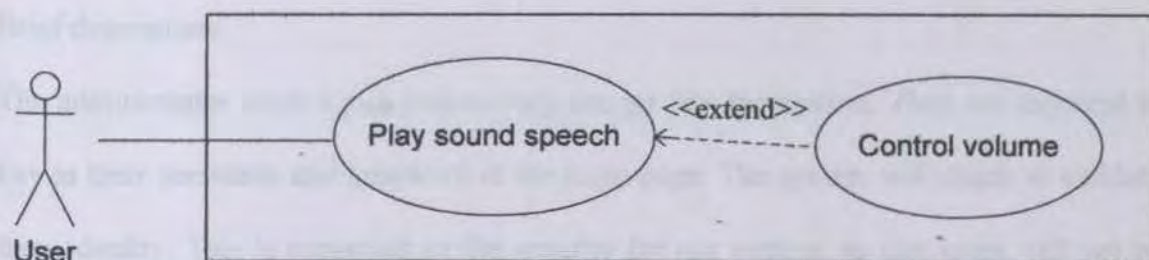
Figure 2.6: Use Case Diagram for Play Sound Speech

2.3.1.6 Play animation

Brief description:

The sign for each word will be animated by the human model selected by the users at the startup of the system. User can do some setting to the animation according to their preferred. They can zoom in, zoom out or rotate the animation. From that, they will have a clear picture for the sign of their selected word. Besides, they may also set the animation speed, so that, they can follow how the model animated the sign. Basic animation functions such as play, stop and pause also will be provided in the system.
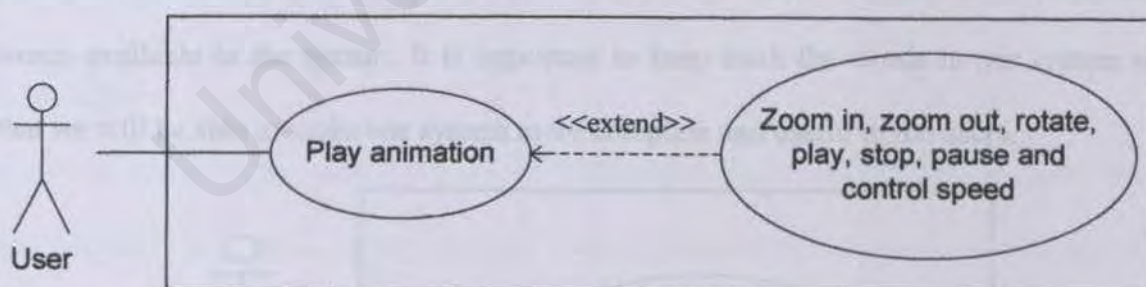
Figure 2.7: Use Case Diagram for Play Animation

2.3.1.7 Log-in

Brief description:

The administrator must log-in before they can go into the system. They are required to key in their username and password at the login page. The system will check to validate their identity. This is important as the security for our system, so that users will not be able to make changes to the contents of our system.
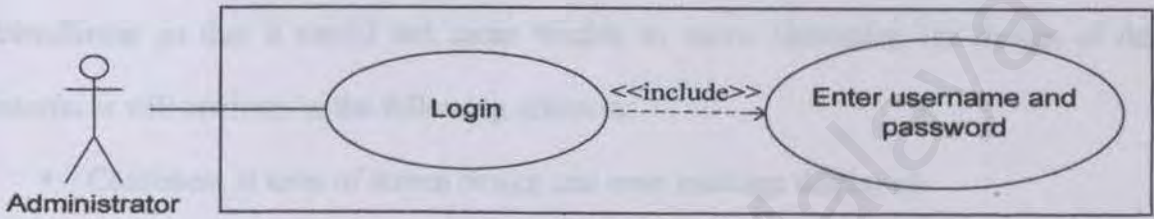
Figure 2.8: Use Case Diagram for Log-in

2.3.1.8 Add, delete and update content

Brief description:

The administrator will be able to manipulate the data in our system. In future, they may add in some new words to the system. Besides, they can delete or update some particular words available in the system. It is important to keep track the words in our system so that we will be able to make our system more complete and useful to the users.
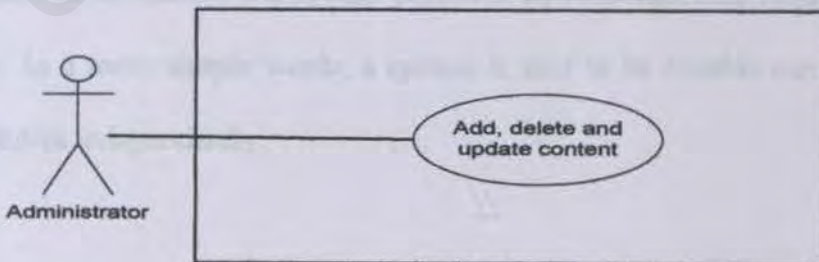
Figure 2.9: Use Case Diagram for Add, Delete and Update Content

2.3.2 Non-Functional Requirements

Non-functional requirements are the constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process and standards.

User-friendliness

User interface design creates an effective communication medium between a man and a computer. Therefore, it is very important to make sure that the interfaces fulfill user-friendliness so that it would not cause trouble to users. Generally, the design of the interfaces will conform to the following criterion:

- Consistent in term of screen design and error message displayed.

- Simplicity refers on keeping the screen properly in a manner that focuses the user's attention.

- High degree of understandability and avoid memorization of events and command.

- Attractiveness which infers that user should enjoy using or attracted to use the system due to their appealing design.

Reliability

A system is said to be reliable if a system performs its functions with required precision and accuracy. In a more simple words, a system is said to be reliable can be trusted to perform the duties independently.

Flexibility

The system should have the capabilities to change following the introduction of new technologies. New environment and resources should be easy to be implemented into the system with this non-functional requirement.

Efficiency

The system is said to be efficient when it can be called or accessed in an unlimited number of time to produce an expected results at a creditable pace or speed.

Functionality

The functionalities stressed here are the searching and retrieving capability, which is important in our system. This is because our system needs to deal with the data from the database. So, any errors occurred during the retrieval of data will cause the failure in our system.

CHAPTER 3: SYSTEM DESIGN

3.1 Architectural Design

Architectural designs are used to represent all the entities and their relationships in a particular system. The architectural design for the e-Sign Dictionary system can be represented using the class diagram below.
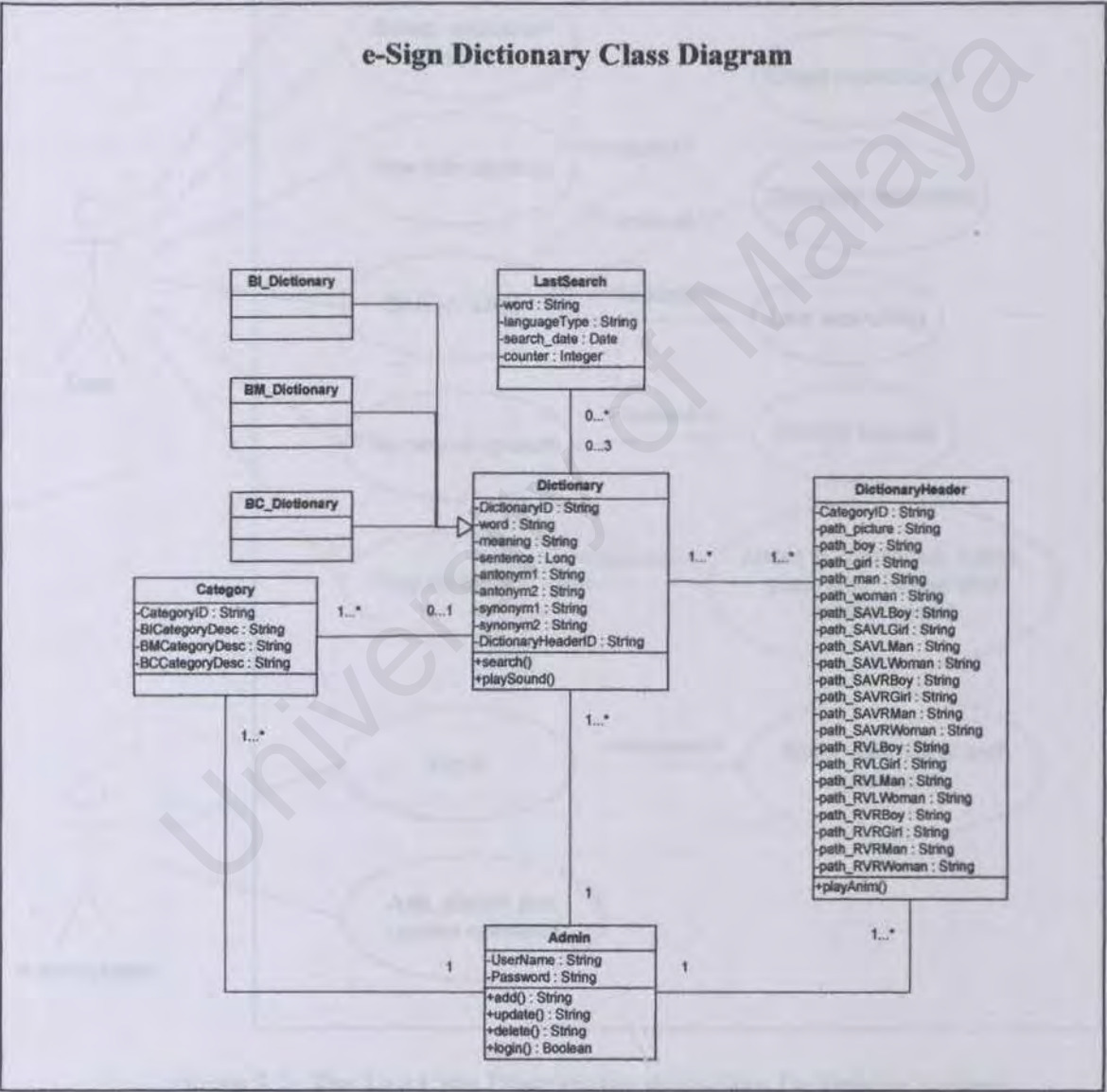
Figure 3.1: The Class Diagram for the e-Sign Dictionary System

3.2 Functional Design

The functions that will be included in the e-Sign Dictionary system are represented by the use cases below.
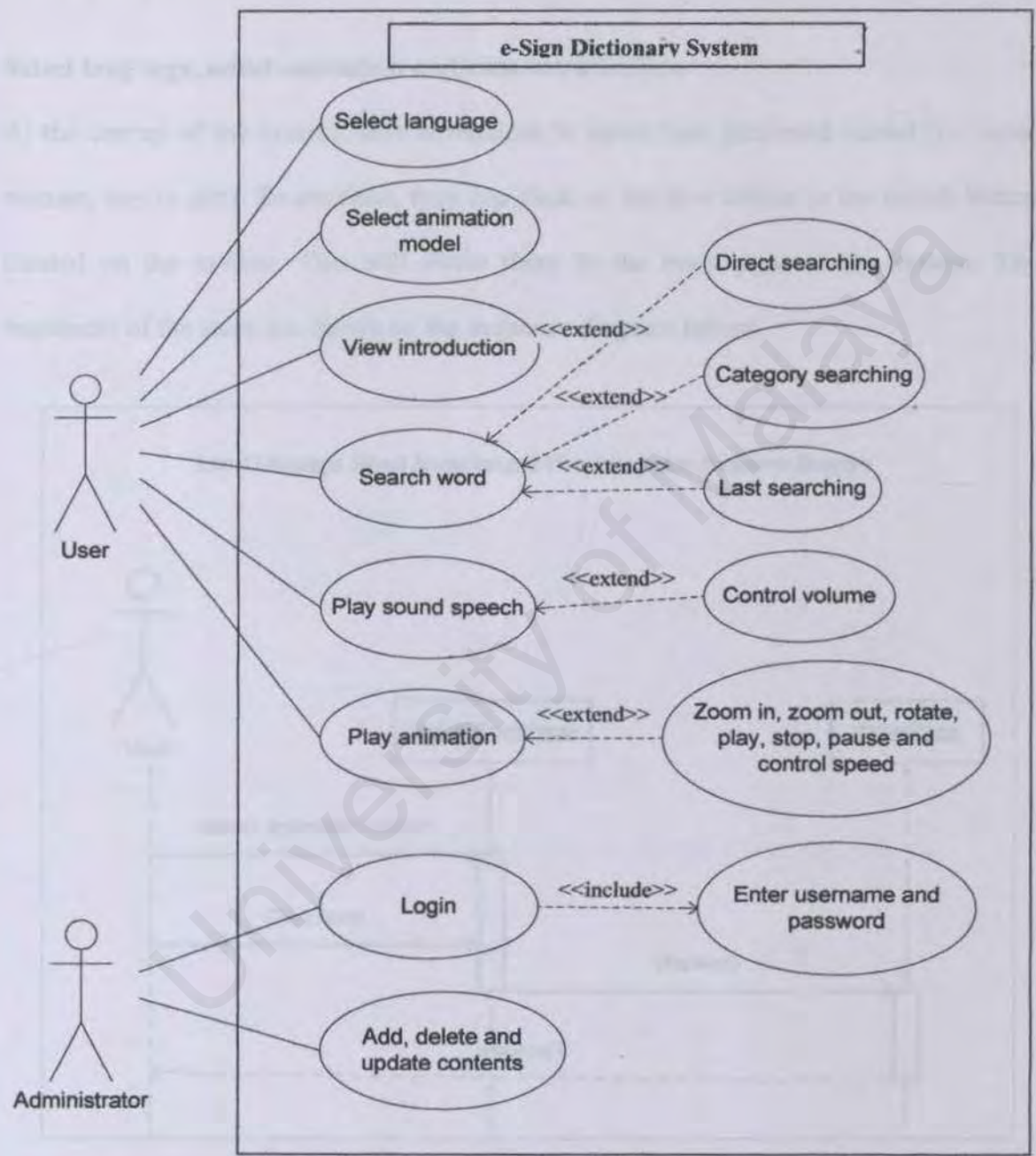
Figure 3.2: The Use Case Diagram for the e-Sign Dictionary System.

The detail workflow for each use case can be represented using the sequence diagram. A sequence diagram is an interaction diagram that details how operations are carried out, what messages are sent and when.

Select language, select animation and view introduction

At the startup of the system, user is required to select their preferred model (i.e. man, woman, boy or girl). To continue, they can click on the next button or the search button located on the toolbar. This will direct them to the main page of the system. The sequences of the steps are shown on the sequence diagram below:
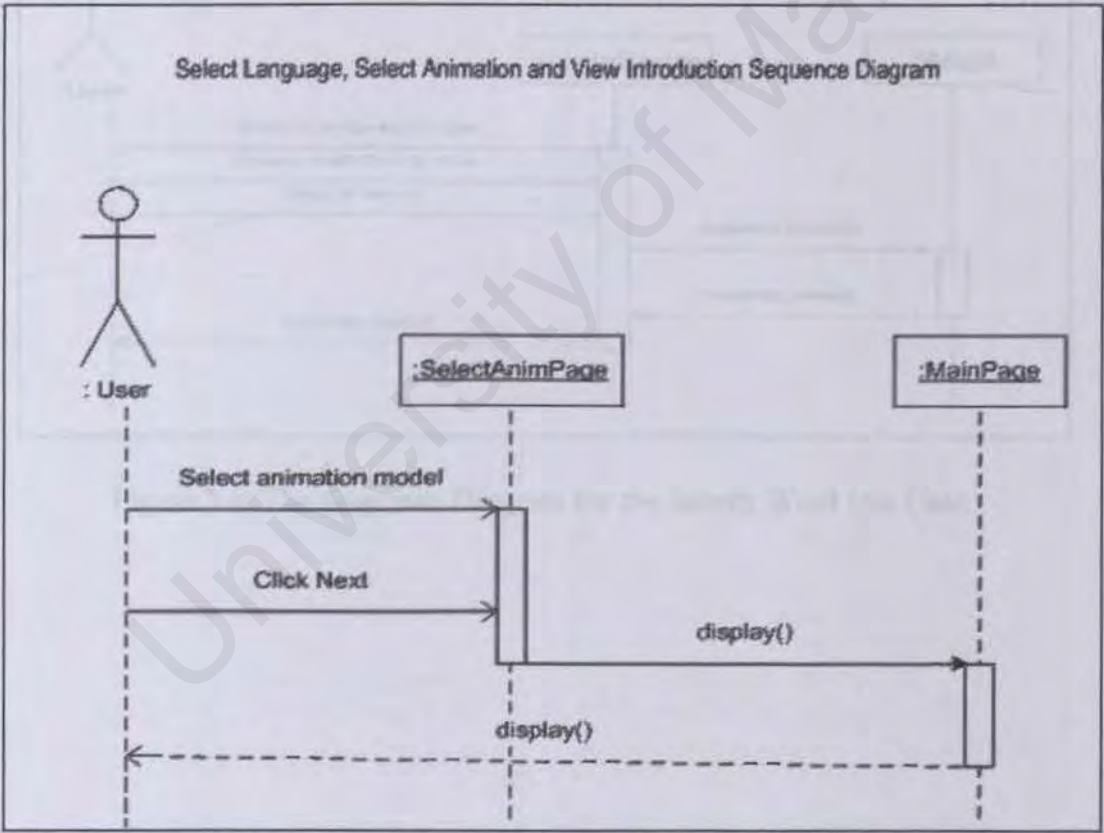
Figure 3.3: Sequence Diagram for Select Animation and Enter Main Page

Search word

In order to search a word, the users must select their preferred language type (i.e. English, Malay or Chinese). Besides, they have to select the searching method that they want. They may search according to alphabetical order, category or last search. After choosing the searching method, they are required to key in the searched word. The system will return the results by displaying them on the listbox. The sequences for this process are shown below.
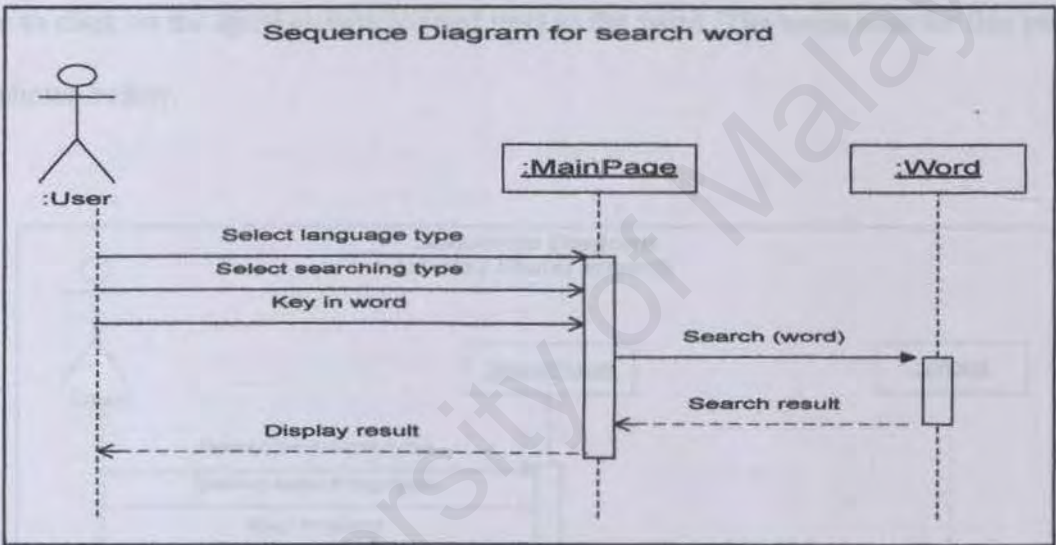
Figure 3.4: The Sequence Diagram for the Search Word Use Case.

Play sound speech

In order to play the sound speech of a word, the users must select their preferred language type (i.e. English, Malay or Chinese). Besides, they have to select the searching method that they want. They may search according to alphabetical order, category or last search. After choosing the searching method, they are required to key in the searched word. The system will return the results by displaying them on the listbox. Next, the user will have to click on the word that they would like to search. All the details of the word will be displayed at the bottom of the listbox. To hear the pronunciation of the word, users will have to click on the speaker icon located next to the word. The sequences for this process are shown below.
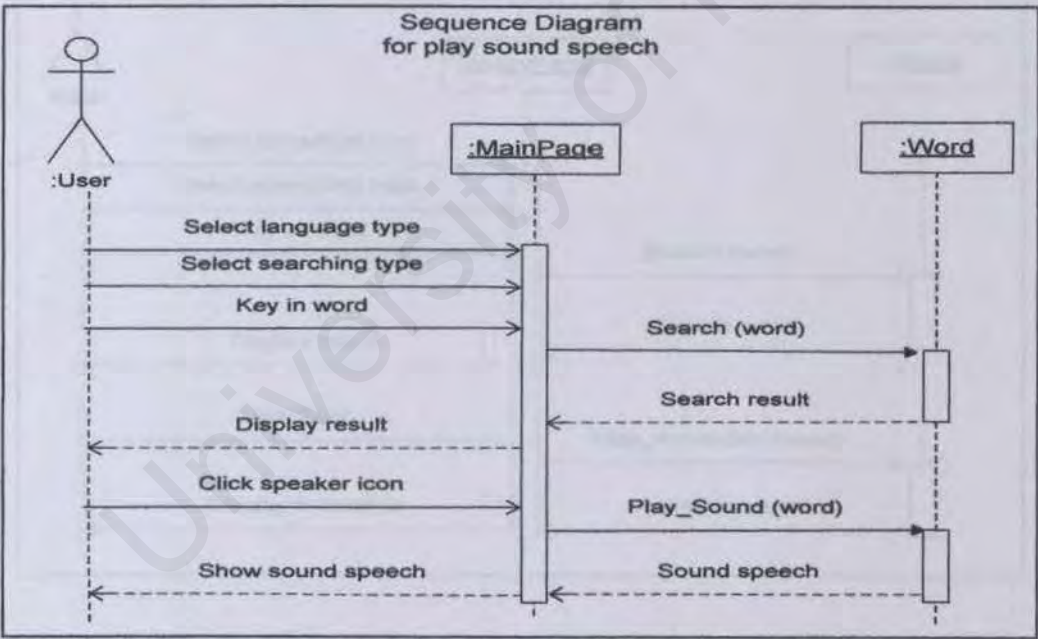
Figure 3.5: The Sequence Diagram for the Play Sound Speech Use Case.

Play animation

In order to play the animation of a word, the users must select their preferred language type (i.e. English, Malay or Chinese). Besides, they have to select the searching method that they want. They may search according to alphabetical order, category or last search. After choosing the searching method, they are required to key in the searched word. The system will return the results by displaying them on the listbox. Next, the user will have to click on the word that they would like to search. All the details of the word will be displayed at the bottom of the listbox. To play the animation of the word, users will have to click on the play button. The sequences for this process are shown below.
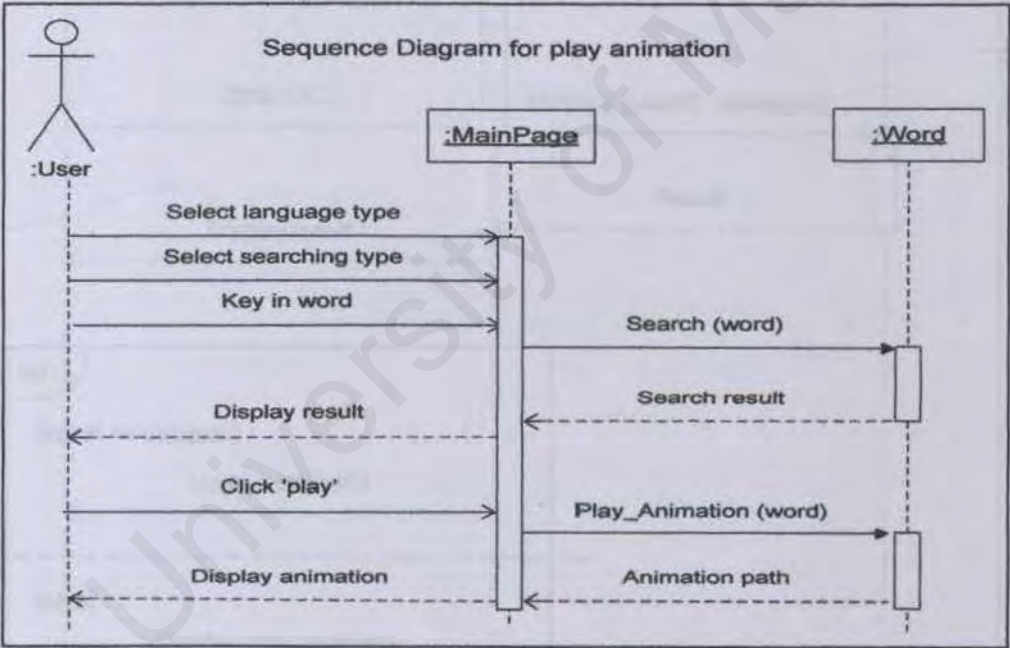
Figure 3.6: The Sequence Diagram for the Play Animation Use Case.

Administrator login

To login to the admin page, administrator needs to enter his/her user ID and password.
The system will validate the user ID and password. If the user ID and password are valid,
the administrator will be directed to the admin page. The sequences for this process are
shown below.
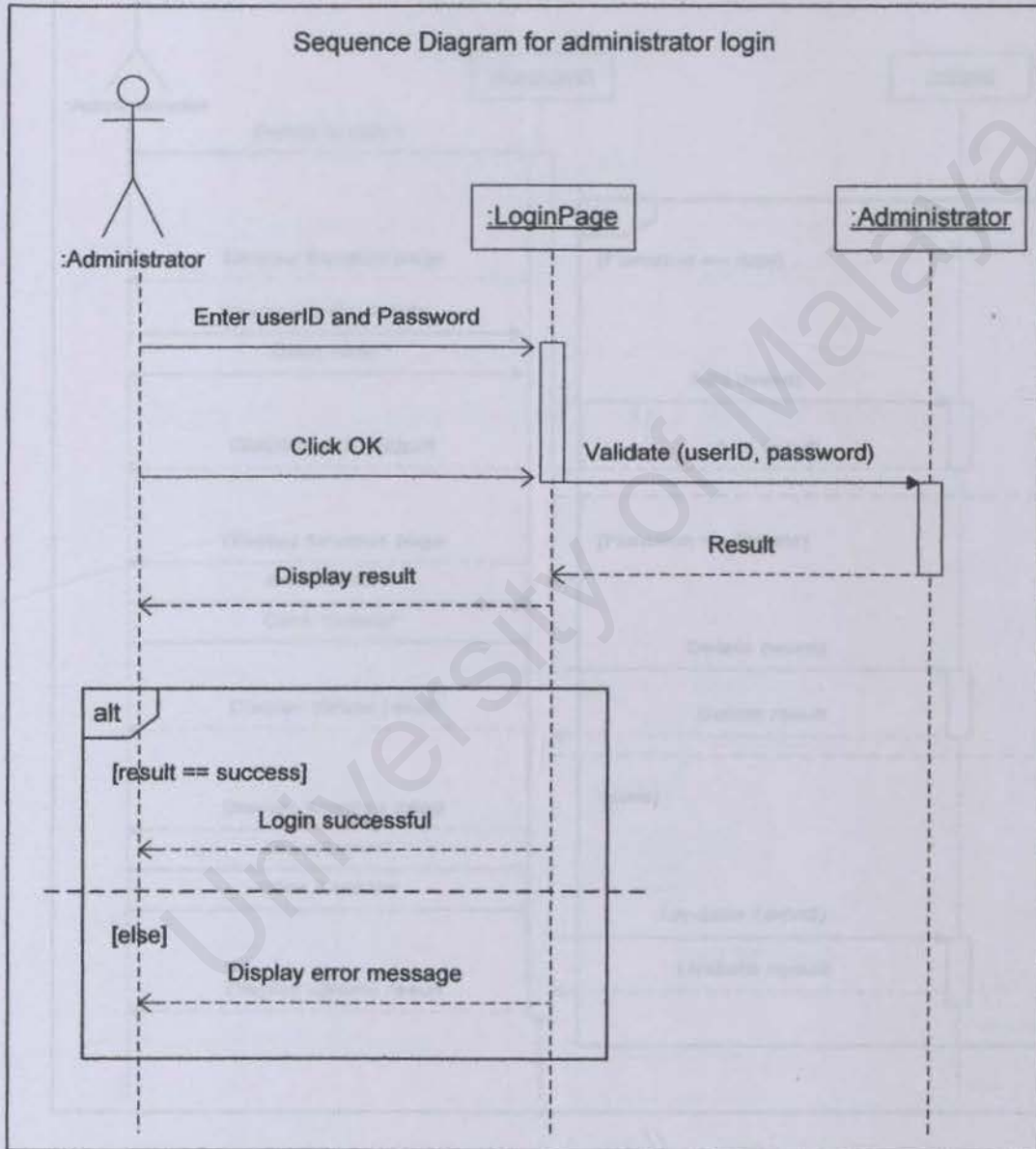
Figure 3.7: The Sequence Diagram for the Login Use Case

Add, update and delete content

In the admin page, users can add, update or delete content. Users need to choose the action that they would like to perform by clicking on the button located on the left corner of the page. The details activities that can be performed by the users are shown below.
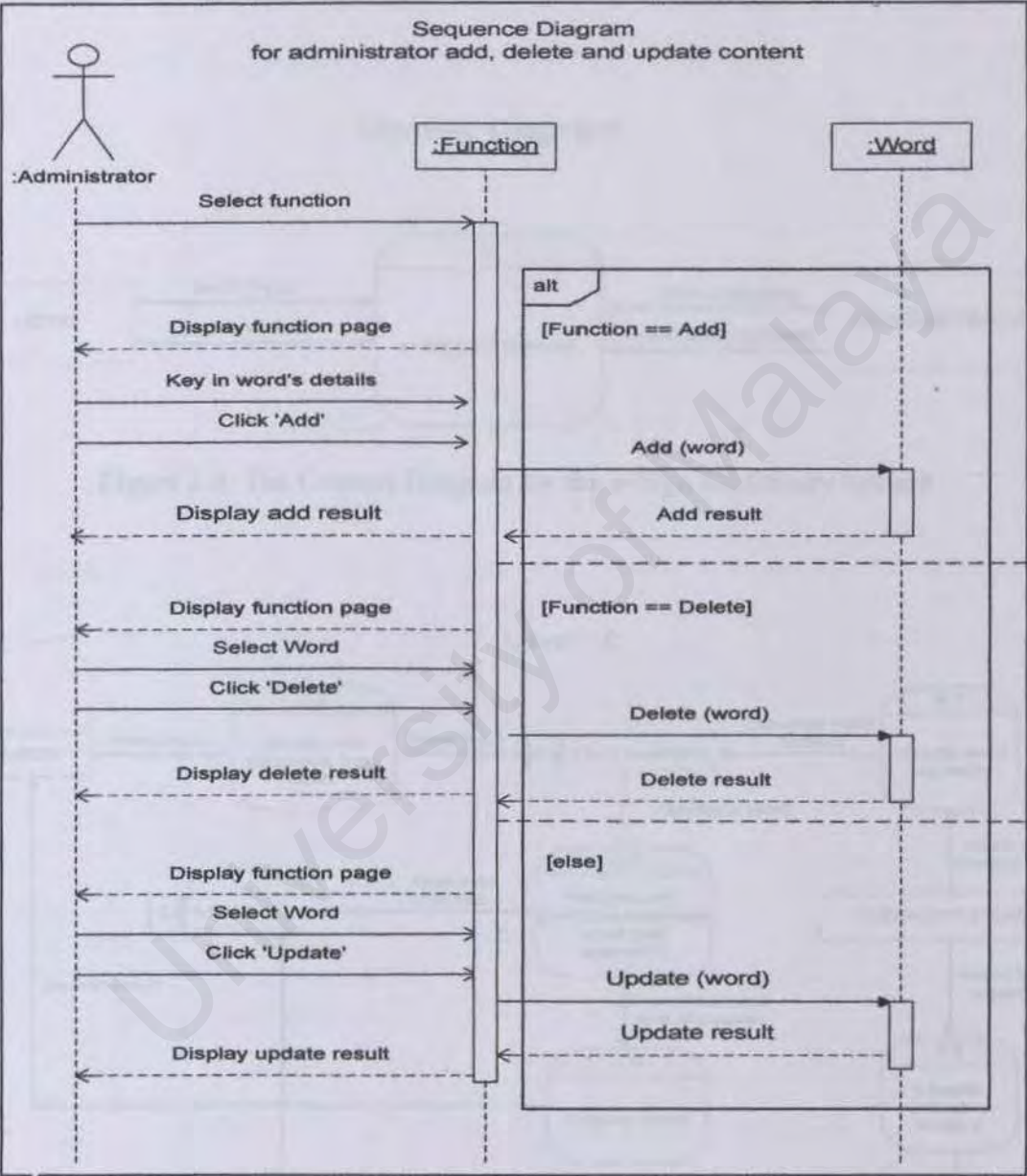
Figure 3.8: The Sequence Diagram for the Add, Delete and Update Content Use Case.

3.3 Data Flow Diagrams

Data flow diagram is a picture of the movement of data between external entities and the processes and data stores within a system. In our system, the two main sources are the user and the administrator. The data flow diagrams for our system are as follow:

Context Diagram
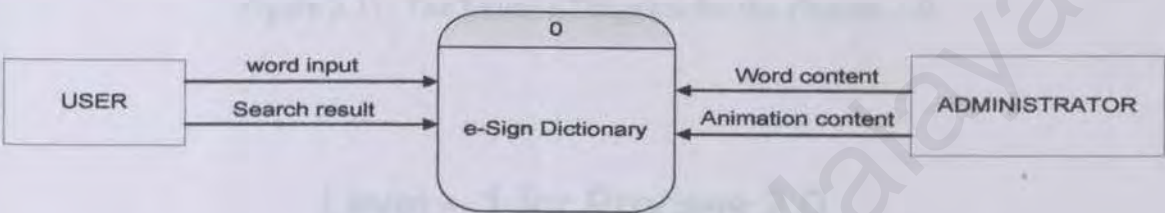
Figure 3.9: The Context Diagram for the e-Sign Dictionary System

Level - 0
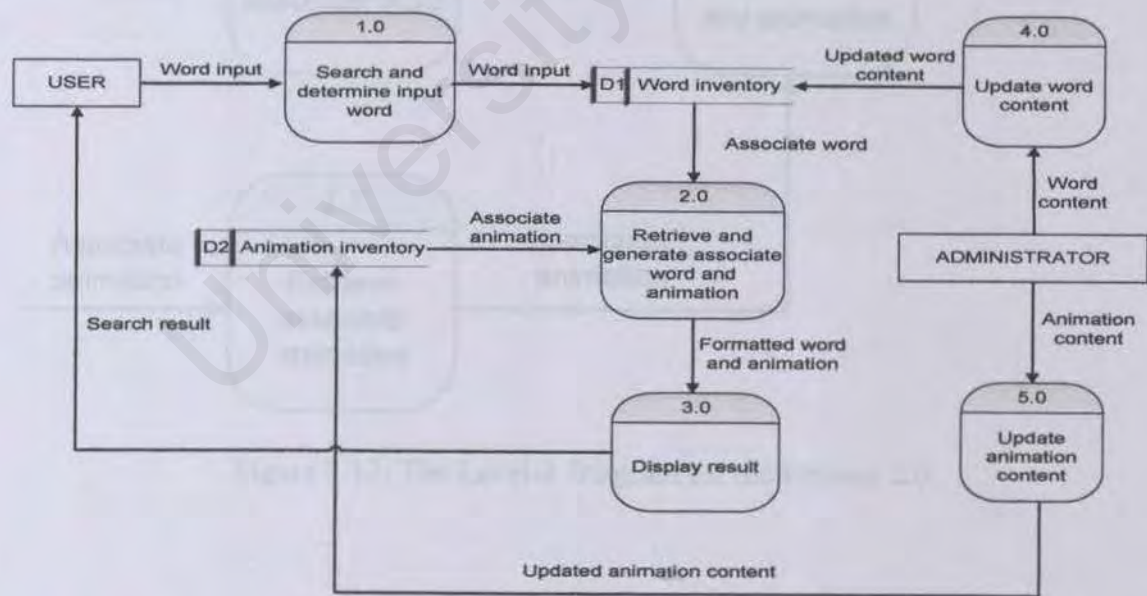
Figure 3.10: The Level-0 Diagram for the e-Sign Dictionary System

Level – 1 for Process 1.0
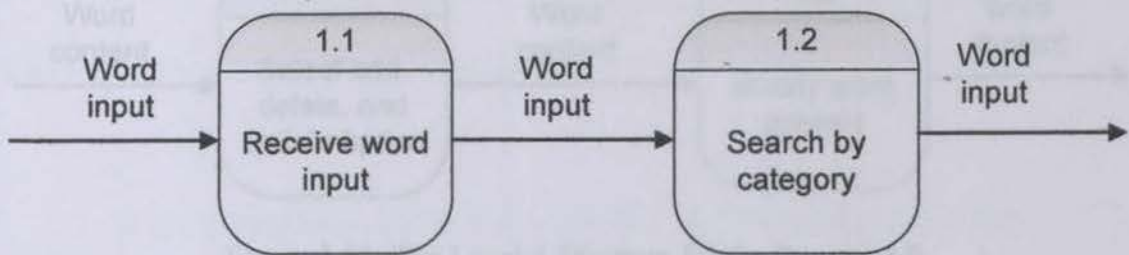
Figure 3.11: The Level-1 Diagram for the Process 1.0

Level – 1 for Process 2.0
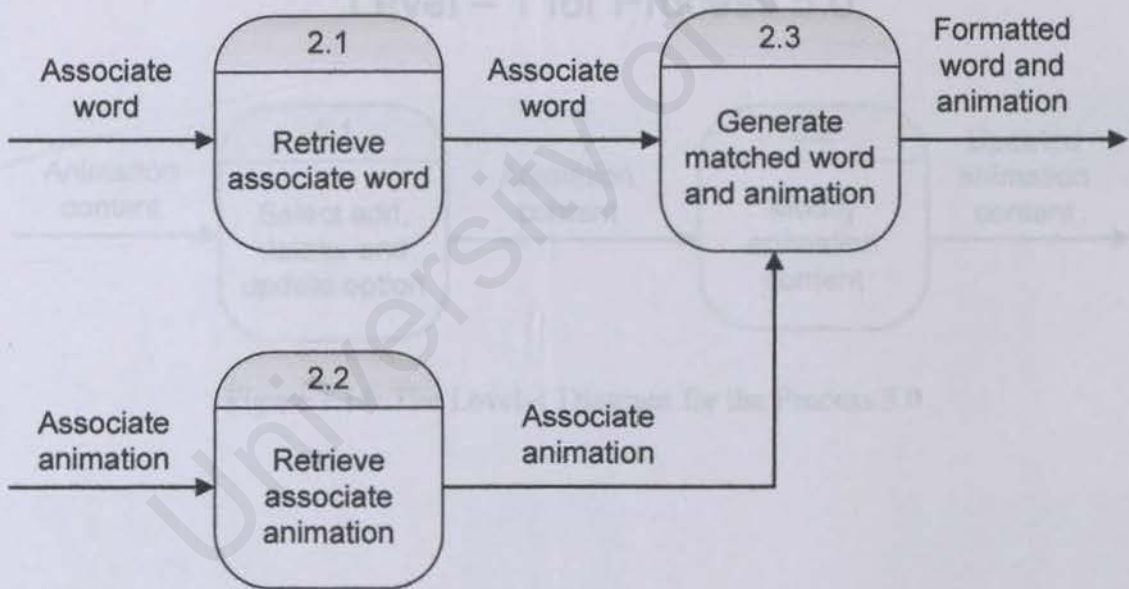
Figure 3.12: The Level-1 Diagram for the Process 2.0

71

Level – 1 for Process 4.0
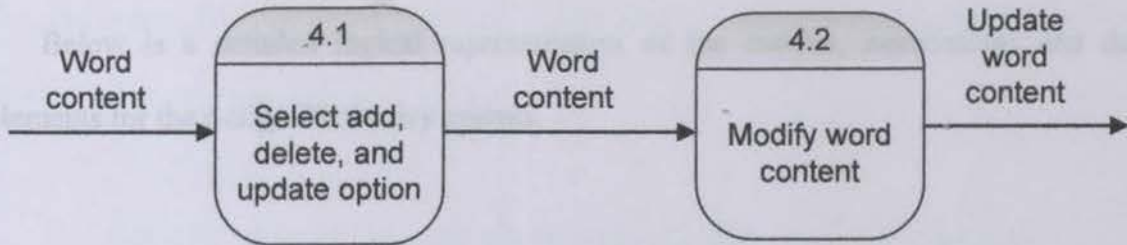
Figure 3.13: The Level-1 Diagram for the Process 4.0

Level – 1 for Process 5.0
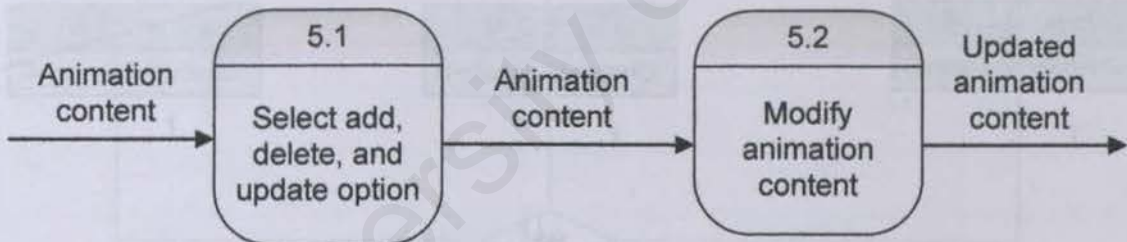
Figure 3.14: The Level-1 Diagram for the Process 5.0

3.4 Database Design

3.4.1 Entity-Relationship Diagram

Below is a detailed logical representation of the entities, associations and data elements for the e-Sign Dictionary system.

Sign Language Dictionary Entity-Relationship Diagram
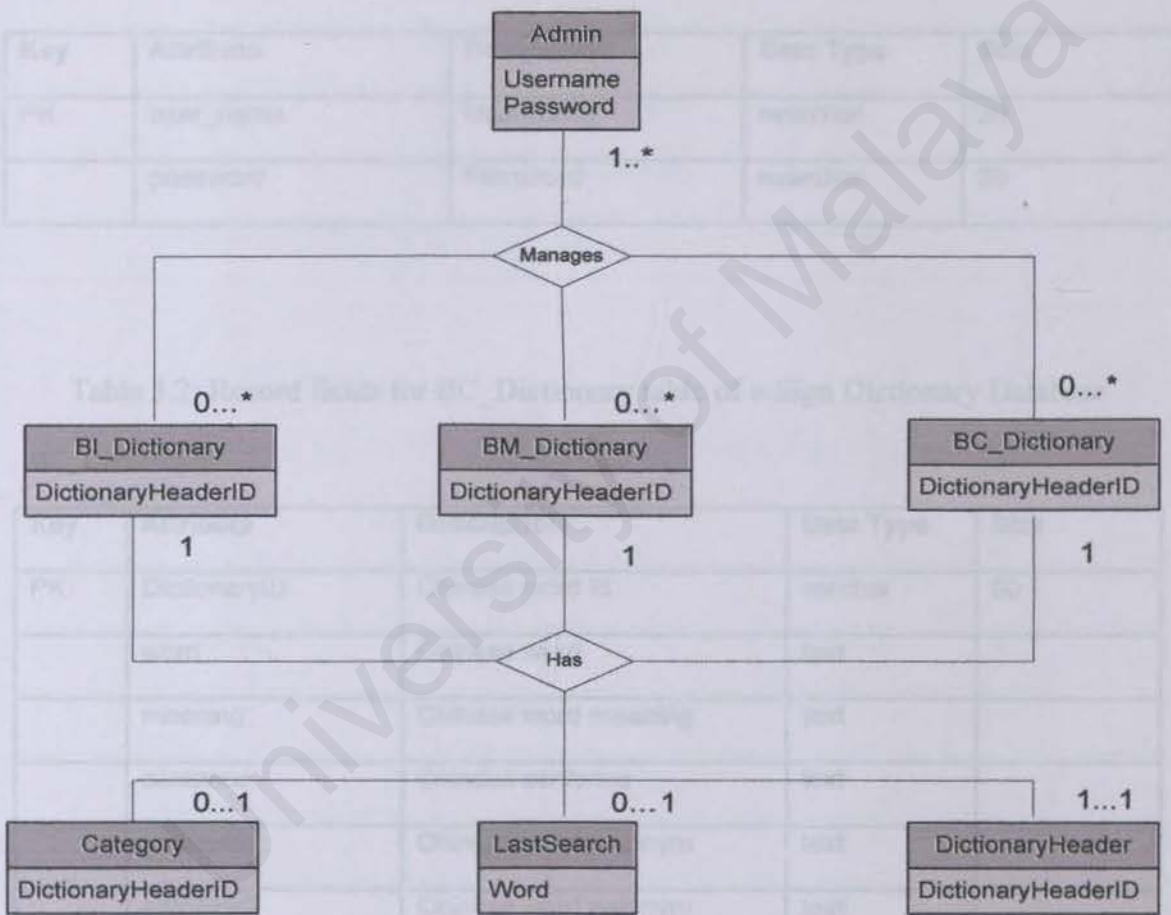
Figure 3.15: The Entity-Relationship Diagram for the e-Sign Dictionary System

3.4.2 Data Dictionary

There are 7 tables created for our system, which are Admin, BC_Dictionary, BI_Dictionary, BM_Dictionary, Category, DictionaryHeader and LastSearch. The fields in each table are as below.

Table 3.1: Record fields for Admin table of e-Sign Dictionary Database

Admin

Key	Attribute	Description	Data Type	Size
PK	user_name	User name	nvarchar	20
	password	Password	nvarchar	20

Table 3.2: Record fields for BC_Dictionary table of e-Sign Dictionary Database

BC_Dictionary

Key	Attribute	Description	Data Type	Size
PK	DictionaryID	Chinese word Id	varchar	50
	word	Chinese word	text	
	meaning	Chinese word meaning	text	
	sentence	Chinese sentence	text	
	antonym1	Chinese word antonym	text	
	antonym2	Chinese word antonym	text	
	synonym1	Chinese word synonym	text	
	synonym2	Chinese word synonym	text	
FK	DictionaryHeaderID	Word Id	varchar	50

Table 3.3: Record fields for BI_Dictionary table of e-Sign Dictionary Database

BI_Dictionary

Key	Attribute	Description	Data Type	Size
PK	DictionaryID	English word Id	varchar	50
	word	English word	text	
	meaning	English word meaning	text	
	sentence	English sentence	text	
	antonym1	English word antonym	text	
	antonym2	English word antonym	text	
	synonym1	English word synonym	text	
	synonym2	English word synonym	text	
FK	DictionaryHeaderID	Word Id	varchar	50

Table 3.4: Record fields for BM_Dictionary table of e-Sign Dictionary Database

BM_Dictionary

Key	Attribute	Description	Data Type	Size
PK	DictionaryID	Malay word Id	varchar	50
	word	Malay word	text	
	meaning	Malay word meaning	text	
	sentence	Malay sentence	text	
	antonym1	Malay word antonym	text	
	antonym2	Malay word antonym	text	
	synonym1	Malay word synonym	text	
	synonym2	Malay word synonym	text	
FK	DictionaryHeaderID	Word Id	varchar	50

Table 3.5: Record fields for Category table of e-Sign Dictionary Database

Category

Key	Attribute	Description	Data Type	Size
PK	CategoryID	Word category Id	varchar	50
	BICategoryDesc	English word category description	text	
	BMCategoryDesc	Malay word category description	text	
	BCCategoryDesc	Chinese word category description	text	

Table 3.6: Record fields for DictionaryHeader table of e-Sign Dictionary Database

DictionaryHeader

Key	Attribute	Description	Data Type	Size
PK	DictionaryHeaderID	Dictionary header Id	varchar	50
FK	CategoryID	Word category Id	varchar	50
	path_picture	Picture for the word	text	
	path_boy	Boy animation main view	text	
	path_girl	Girl animation main view	text	
	path_man	Man animation main view	text	
	path_woman	Woman animation main view	text	
	path_SAVLBoy	Boy animation left view	text	
	path_SAVLGirl	Girl animation left view	text	
	path_SAVLMan	Man animation left view	text	
	path_SAVLWoman	Woman animation left view	text	
	path_SAVRBoy	Boy animation right view	text	
	path_SAVRGirl	Girl animation right view	text	
	path_SAVRMan	Man animation right view	text	

	path_SAVRWoman	Woman animation right view	text	
	path_RVLBoy	Boy animation rotate left view	text	
	path_RVLGirl	Girl animation rotate left view	text	
	path_RVLMan	Man animation rotate left view	text	
	path_RVLWoman	Woman animation rotate left view	text	
	path_RVRBoy	Boy animation rotate right view	text	
	path_RVRGirl	Girl animation rotate right view	text	
	path_RVRMan	Man animation rotate right view	text	
	path_RVRWoman	Woman animation rotate right view	text	

Table 3.7: Record fields for LastSearch table of e-Sign Dictionary Database

LastSearch

Key	Attribute	Description	Data Type	Size
	word	Word	text	
	languageType	Language of the word	text	
	search_date	Date of the word searched	datetime	
	counter	Counter for the word search	integer	4

3.5 Graphical User Interface Design

It is important to follow certain principles when developing user interfaces. The acceptance of the users on a system is normally based on the system's interfaces. Some of the user interface principles that we have followed are user familiarity, consistency, minimal surprise, recoverability and user guidance.

3.5.1 Input and Output Screen Design

The input and output screens for the e-Sign Dictionary system are as follow:
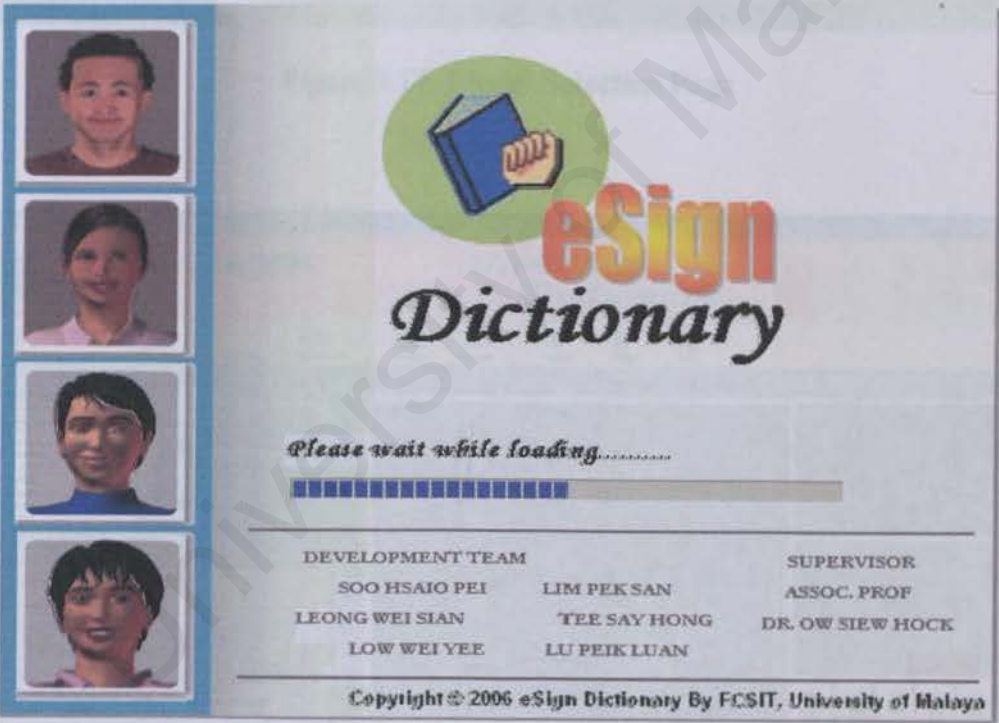
Figure 3.16: Welcome Page
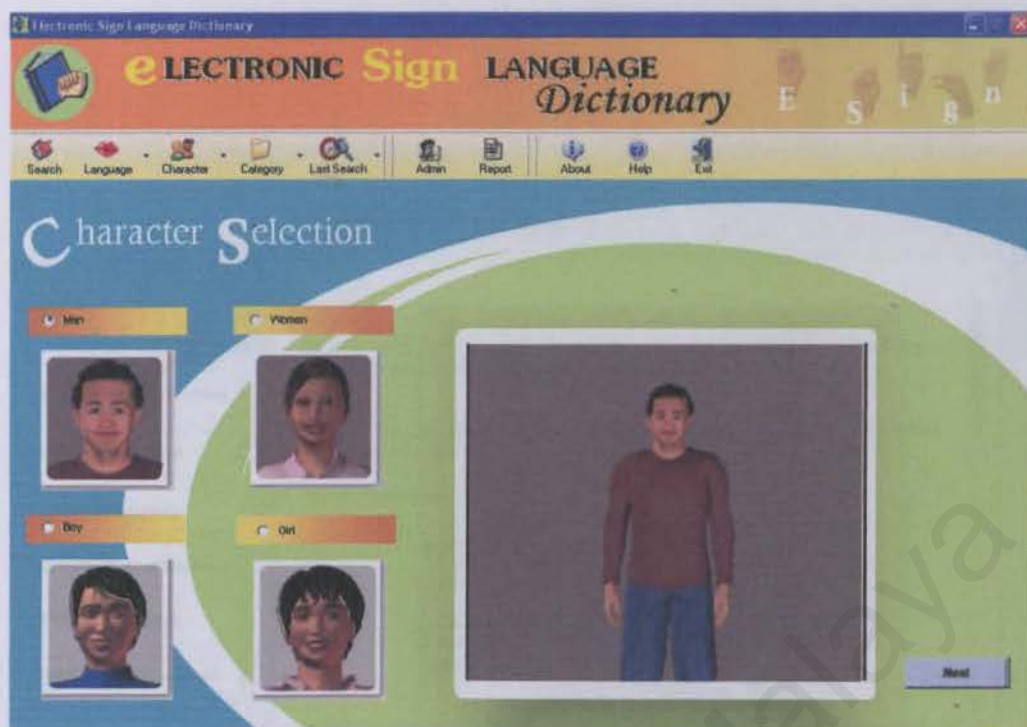
Figure 3.17: Model Selection Page

Figure 3.18: Search Page
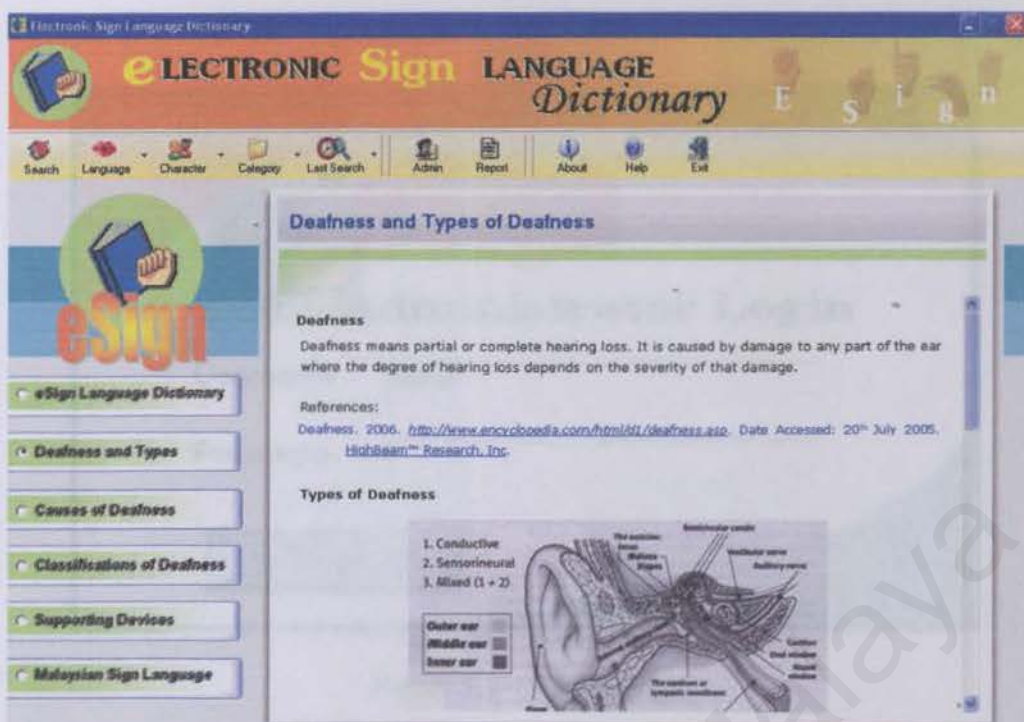
Figure 3.19: About Page
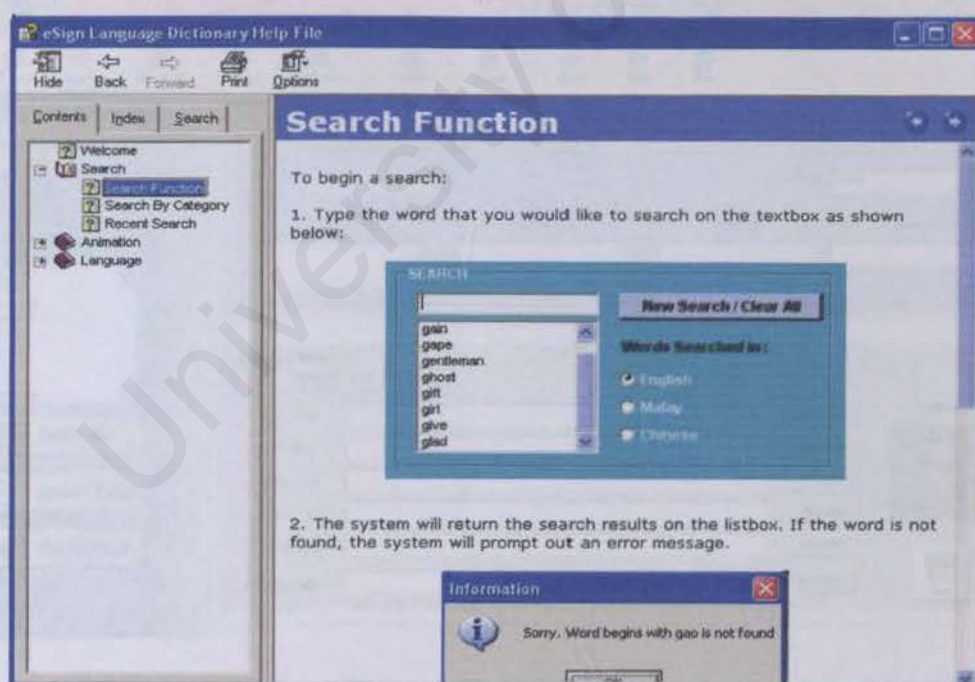
Figure 3.20: Help Page

Figure 3.21: Login Page
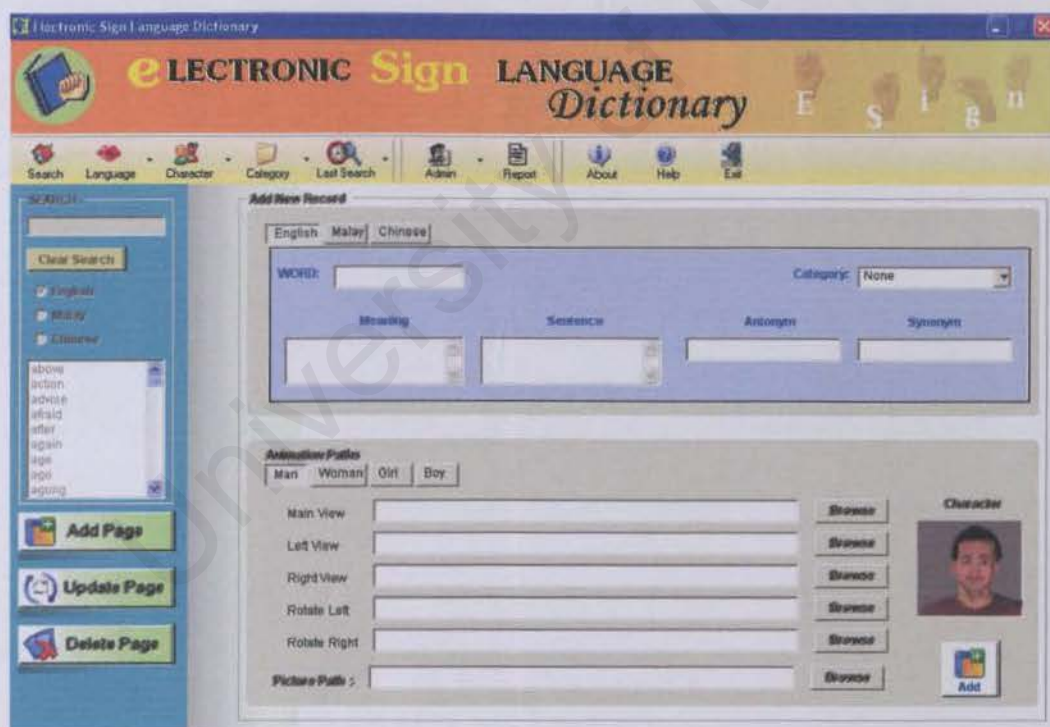
Figure 3.22: Admin Page

Figure 3.23: Change Password Page

Figure 3.24: Report Design

3.5.2 Report Design

In the e-Sign Dictionary system, there is a function that allows the users to search a word using last search. Users will be able to see all the words that had been searched in last month, last week or last 24 hours. The system will generate a report to the user based on their request. The design of the report is as follow:
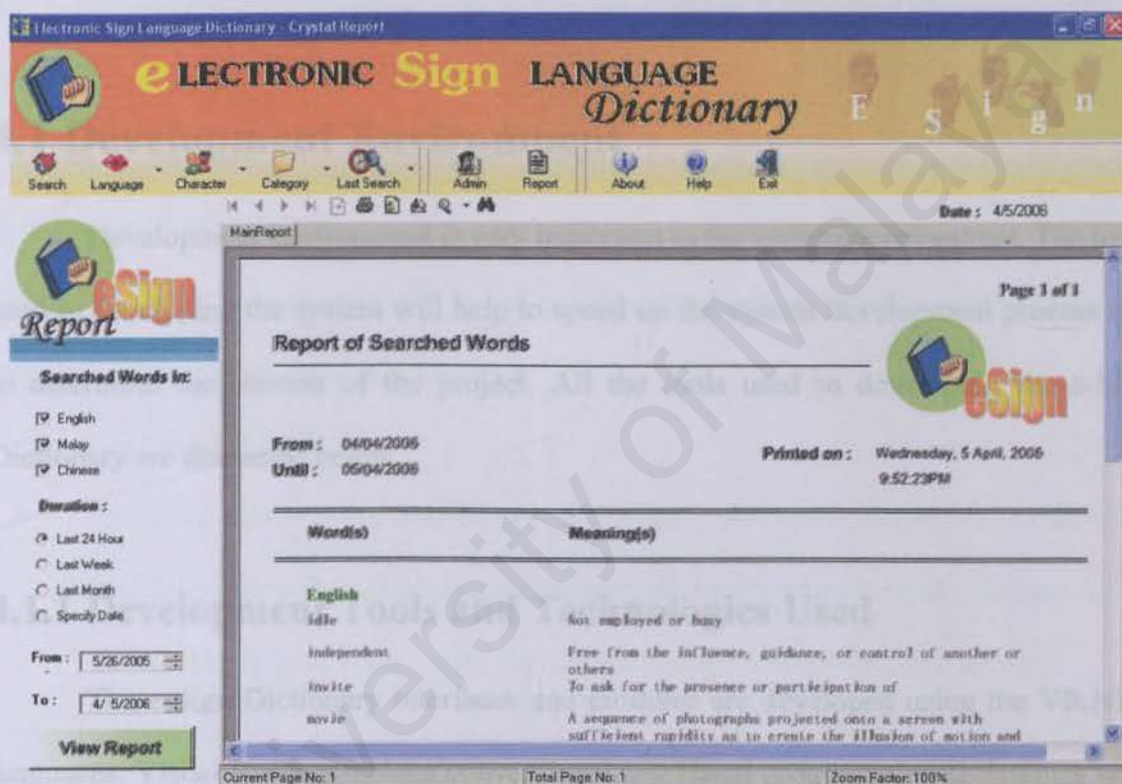
Figure 3.24: Report Design

CHAPTER 4: SYSTEM DEVELOPMENT

System Development is an important phase in software engineering. At this phase, all the requirements that have been agreed earlier are put into action. Coding is the core process in system development. It helps to translate design specifications into source codes that the computer can process. The compiler or interpreter will translate the source codes into executable program.

4.1 Development Environment

Development environment is very important to the system development. The tools used in developing the system will help to speed up the system development process and to determine the success of the project. All the tools used in developing the e-Sign Dictionary are discussed below.

4.1.1 Development Tools and Technologies Used

The e-Sign Dictionary interfaces and modules are developed using the VB.NET languages. Visual Basic combines conventional test based programming techniques with visual techniques using icons and menus. In developing the interfaces for the system, VB.NET provides a drag and drop function that allow user to draw the components (i.e. textbox, combo box and image) directly without writing the codes. It is much like using a paint program. In addition, the buttons, textboxes and other objects that we have dragged into the window form will automatically recognize user actions such as mouse

movements and button clicks. So, it is quite easy for us to design the interfaces as well as writing the source codes to our system.

We used MS SQL Server to develop the database for the e-Sign Dictionary system. Data controls were used to perform database administrative tasks from the interfaces created using VB.NET.

Besides, we also used 3D Poser 6.0 in developing the animation for each word. All the animation files are converted to the .avi type, so that it will be easier for us to incorporate it in our system using VB.NET codes.

Development of Database

There are two processes involved in developing the database for the e-Sign Dictionary system. Below are the processes involved:

- Creating the database

  There are seven tables created in our database, which are Admin, BI_Dictionary, BM_Dictionary, BC_Dictionary, Category, DictionaryHeader and LastSearch.

  o The Admin table stored the administrator's username and password.

  o The BI_Dictionary table stored all the English words together with their meanings, synonyms and antonyms.

  o The BM_Dictionary table stored all the Malay words together with their meanings, synonyms and antonyms.

  o The BC_Dictionary table stored all the Chinese words together with their meanings, synonyms and antonyms.

  o Category table stored all the categories of the words

- o DictionaryHeader table stored all the animation paths for each word.
- o LastSearch stored all the words that have been searched by the users.
- Entering Data

  There are two ways for the administrator to enter the data into the database. Data can be entered directly into the database using MS SQL Server, or through the e-Sign Dictionary Add module.
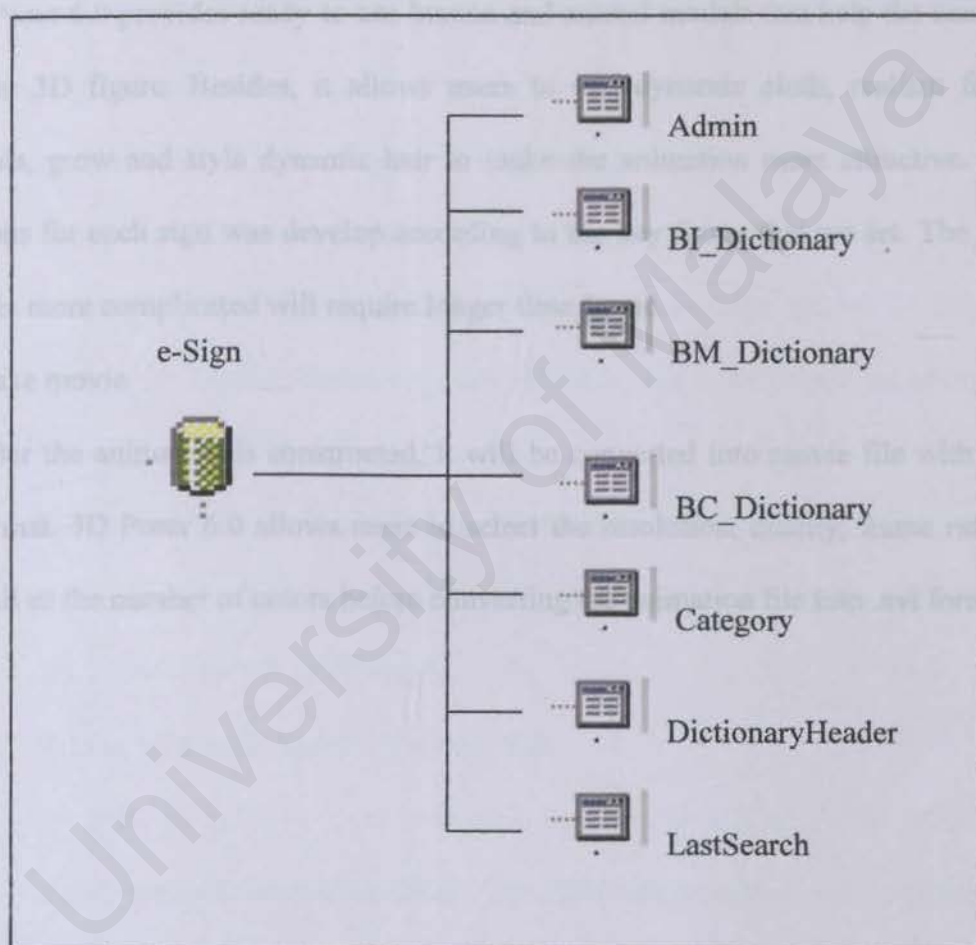
Figure 4.1 Database and tables in e-Sign Dictionary

Development of Animations

e-Sign Dictionary system provides four types of animation models for the user to choose. There are man, woman, boy and girl. All the models and the signs for each word are constructed using 3D Poser 6.0. The processes that involve in developing the animation for e-Sign Dictionary are:

- Construct the animation

  3D Poser 6.0 provides ready-to-use human and animal models that help the users to create 3D figure. Besides, it allows users to add dynamic cloth, realism facial photos, grow and style dynamic hair to make the animation more attractive. The actions for each sign was develop according to the key frame that we set. The sign that is more complicated will require longer time frame.

- Make movie

  After the animation is constructed, it will be converted into movie file with .avi format. 3D Poser 6.0 allows users to select the resolution, quality, frame rate as well as the number of colors before converting the animation file into .avi format.

Development of Audio

In the e-Sign Dictionary system, there is a function that can help the users to learn the pronunciation of the words. This function was developed using Text-to-Speech. Since our system is a trilingual system, we include not only English version of Text-to-Speech but also Malay and Chinese versions. The processes that involved in developing the audio into the e-Sign Dictionary can be divided into three parts.

- Developing English audio

  The steps that involved in developing the English audio are:

  a) Install Text-to-Speech audio

     In order to incorporate the Text-to-Speech into the e-Sign Dictionary system, an English Text-to-Speech engine needs to be downloaded into its folder.

  b) Installing Microsoft Text-to-Speech SDK 2.5 run time support

     To produce spoken output, SDK 2.5 run time support must be installed to the Text-to-Speech (TTS) engine.

  c) Add in Microsoft Agent 2.0 component

     In order to call the Text-to-Speech function in VB.NET, the Microsoft Agent component must be added. The agent can be called in the program by add in some codes.

▪ Developing Malay audio

The steps that involved in developing the Malay audio are:

a) Add in the Text-to-Speech module

The Malay Text-to-Speech module must be added into the e-Sign Dictionary system.

b) Calling synthesize function

Speech synthesize can be performed by calling the synthesize function.

▪ Developing Chinese audio

The steps that involved in developing the Chinese audio are:

a) Add in the Text-to-Speech module

The Chinese Text-to-Speech module must be added into the e-Sign Dictionary system.

b) Calling synthesize function

Speech synthesize can be performed by calling the synthesize function

4.1.2 Programming Tools

Visual Basic.NET was designed to make user-friendly programs easier to develop. With visual programming, the programmer has the ability to create graphical user interfaces (GUI) by dragging and clicking with the mouse. Visual programming eliminates the need for the programmer to write most of the codes for the functions in the system. The programmer creates GUI and writes code to describe what happens when the user interacts (clicks, press a key, double-clicks, etc.) with the GUI. These notifications, called events, are passed into the program by Microsoft's Windows operating system. Programming the code that responds to these events is called event-driven programming. The steps that involved in developing the e-Sign Dictionary system are:

- Design the appearance of the window that the user sees.
- Determine the events that the controls on the window should recognize.
- Write the event procedures for those events.

When the program is running,

- VB.NET monitors the controls in the window to detect any event that an object can recognize (mouse movements, click, keystroke and etc.)
- When VB.NET detects an event, it examines the program to see if the event procedure for that event is written or not.
- If an event procedure is written, VB.NET executes the instructions that make up that event procedure.

90

4.2 Development Process

Development process is a structure imposed on the development of a system. Development processes are composed of many activities, notably the following.

4.2.1 Programs Naming Convention

A standard naming convention and also a standard usage of graphical user interface components is employed in developing the system. Standard naming convention provides programmers with easy identification of variables. While a standard in usage of graphical user interface components provides the users an environment that will not generate much surprise to them. Usages of these standards perform as a mean towards coding consistency and standardization.

In coding the e-Sign Dictionary system, we follow the standard naming convention. Below are the lists of variables that we have used in our system.

Table 4.1 Lists of Variables

Variable	Meaning
lbl	Label
butt	Button
ant	Antonym
syn	Synonym
tb	Textbox
mng	Meaning

stc	Sentence

4.2.2 Design the Programs

After review the system requirements, the following step is to design the program. This is the initial step that put all the requirements discussed earlier into action. The activities involved in designing the programs include

     i. Designing the interfaces of the system.

    ii. Designing the functions of the system.

   iii. Dividing the modules of the system.

4.2.3 Coding Styles

Coding style is an important attribute of source code. An easy way to read source code makes the system easier to maintain and enhance. Elements taken into considerations while coding an easy to maintain and enhance system are internal documentation, standard naming convention and standard graphical user interface.

In coding the e-Sign Dictionary system, we followed certain rules or styles. To make it easier for our team members to identify the variables used in the system, we used the lowercase letter to represent the variables. Besides, we make sure that all logical blocks are formatted in such a way as to be clean and clear to what is occurring. Every If/Then should have an End If. In addition, we avoid single line statement in every instance except possibly the Select Case blocks.

4.2.4 Testing and Debugging

Test the system

During the system testing phase, the whole system is tested to ensure the correctness of the system flow. Besides, all the functions in the system also been tested to make sure that the functions are working or functioning as required. First, the modules are tested individually and then tested as a whole. The testing techniques that we used most in our system are unit testing and integration testing.

Debugging

Debugging is the activity of finding and fixing the errors. There are various types of errors that exist in the system such as compile error, run time error and logic error. The debugging strategies applied in the system are listed as below:

✓ Built in error detection

Visual.NET has a built in error detection. If an error is found during application execution, an error message together with the lines number where the error occurred will be debugged. With this feature, the debugging work becomes easier and faster.

✓ Reviewing the algorithm used

If a program is running well but the information is not what as intended, logic error or database error may occur. Reviewing algorithm and computations for their correctness and efficiency is needed for this purpose. Sometimes, by using different algorithms, the effiency of the program will increase.

✓ Display on screen the passing value

One of the possibilities of wrong information being retrieved is that the wrong value is being passed from one page to another page. To ensure that right value has been passed to the next page for processing, the passing value is displayed on the screen for reviewing.

✓ Check success status

Some processes are dependent where failure in the previous process will affect others processes. In order to avoid chain reaction from this kind of process, a success status is purposely set to return a true or false value. The success status is checked to determine whether to continue process or to exit from the program and display error message.

✓ Using Structured Query Language (SQL)

When there is an error in the database transaction, an error message will be displayed and most of the time, it is caused by incorrect syntax in the SQL statement. Therefore, SQL statement being used to execute the transaction will be tested.

4.3 Programming Techniques

Good programming techniques are very important in determing the success of a system. Inappropriate used of a programming technique in developing a system will not only cause the delay for the system to be delivered but also will increase the development cost.

4.3.1 Modular Programming

Modular programming can be used to break up a large program into manageable units, or to create code that can be easily re-used. A modular program consists of a main module and one or more auxiliary modules. Each module originates from a separate source code file.

We have divided the programs in the e-Sign Dictionary system into seven modules, which are search, animation, report, admin, help, about and login. Each of us is responsible for developing a module. I was responsible in developing the animation module.

The descriptions for each module are stated below.

❖ Search module

There are three types of searching methods provided in the e-Sign Dictionary system which are search by category, alphabetical search and last search.

❖ Animation module

All the controls of the animation such as play, pause and rotate are under this module. Besides, there are four types of animation models provided for

the users (i.e. man, woman, boy and girl). User will need to choose their preference model. In addition, the zooming and animation rotation functions are also under this module.

❖ Report module

Display the words that have been searched by the users at a particular of time (i.e. last 24 hours, last week or last month).

❖ Admin module

Involves add, update and delete function.

❖ Help module

Display the help information to the user when user clicks on the help button.

❖ About module

Display the deafness and Malaysian Sign Language information to the users when they click on the about button.

❖ Login

Verify the status of the administrator before allowing them to enter the admin page.

4.3.2 Module Integration

All the seven modules that we have divided among our team members are integrated to form a complete system. Before the integration take place, each module is tested to ensure that the functions under the particular module are working properly. Integrating the modules into a system is not easy and it requires a great amount of time. The incomplete module will cause problem during the module integration.

CHAPTER 5: SYSTEM TESTING

Testing is the process of executing a program with intention of finding errors. The testing objectives can be divided into two: direct objectives and indirect objectives.

Direct objectives:

- To identify and reveal as many errors as possible in the tested system.
- To bring the tested system, after correction of the identified errors and retesting, to an acceptable level of quality.
- To perform the required tests efficiently and effectively, within budgetary and scheduling limitations.

Indirect Objectives:

- To compile a record of system errors for use in error prevention (by corrective and preventive actions).

Before a system is utilized, it needs to be tested thoroughly. This is achieved by using carefully planned test strategies and realistic data, so that, the entire testing process is methodically and rigorously carried out. In developing a system, the testing process usually involved several stages.
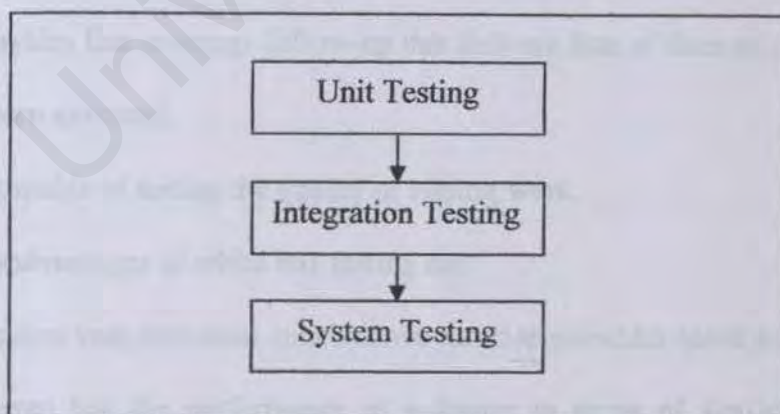
Figure 5.1 Testing Stages

5.1 Testing Techniques

5.1.1 Module Testing

The goal of module testing is to confirm that the module is correctly being coded and it can carry out the function that is originally assigned to it. This type of testing will make sure that the component can perform properly with the expected types of input and output. In this technique, coding and testing are carried out in parallel. This testing technique was also used to ensure that the bugs are fixed without side effects. Functions and procedures in each module are examined carefully after the coding process.

White box testing

White box testing is the testing that takes into account the internal mechanism of a system or component. White box testing examines internal calculation paths in order to identify bugs. Although the term "white" is meant to emphasize the contrast between this method and black box testing, the method's other name- "glass box testing"- better expresses its basic characteristic, that of investigating the correctness of code structure.

The main advantages of white box testing are:

- ✓ It permits direct checking of processing paths and algorithms.
- ✓ It provides line coverage follow-up that delivers lists of lines of code that have not yet been executed.
- ✓ It is capable of testing the quality of coding work.

The main disadvantages of white box testing are:

- ✓ It requires vast resources, much above those required for black box testing.
- ✓ It cannot test the performance of software in terms of availability, reliability, stress, etc.

Black box testing

Black box testing is the testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions. Black box testing is conducted to evaluate the compliance of a system or component with specified functional requirements.

The main advantages of black box testing are:

- ✓ It allows the tester to carry out almost all test classes.

- ✓ For test classes that can be carried out by both white and black box testing, black box testing requires considerably fewer resources.

The main disadvantages of black box testing are:

- ✓ It allows for indentification for coincidental errors as correct.

- ✓ It lacks control of line coverage.

- ✓ It lacks possibilities to test the quality of coding work.

In the module testing process of e-Sign Dictionary system, the following aspects have been considered:

- ❖ Code Review

    At the beginning of the module testing, the program codes are examined to make sure that all the algorithms used in the program are correct. Besides, we had gone through the program codes to identify any data or syntax errors. Throughout the checking process, syntax errors, logic errors, deviations from coding standards and fraudulent code are discovered in each module.

In addition, the specification and the designs of the system also been compared with the specific codes. This is important as it can help us to make sure that all the relevant cases have been considered. Before the test cases are developed to show that the input is properly converted to the desired output, the program is run once to see the result and to eliminate the remaining syntax errors if necessary.

❖ Interface

All the interfaces in e-Sign Dictionary system have been test to make sure that the flow of information from one module to another module are correct.

❖ Local Data Structure

Local Data Structure is examined to make sure that the integrity of the data is maintained during the execution of an algorithm. Besides, the local impact on global data should be ascertained during module testing.

❖ Errors Handling Paths

All errors handling paths are checked to ensure its ability to detect and recover all fatal errors during system execution. In addition, it is also important to ensure that the routine for all the error handling are works correctly according to what it is directed.

❖ Independent Paths

All the independent paths throughout the system structure are tested to
ensure that all the statements in a module have been executed at least
once.

5.1.2 Integration Testing

The process of system integration involves building a system from its components
and testing the resultant system for problems that arise from component interactions. The
components that are integrated may be off-the-shelf components, reusable components
that have been adapted for a particular system or newly developed components.
Integration testing checks that these components actually work together are called
correctly and transfer the right data at the right time across their interfaces.

System integration involves identifying clusters of components that deliver some
system functionality and integrating these by adding code that make them work together.
Sometimes, the overall skeleton of the system is developed first, and components are
added to it. This is called top-down integration. Another approach is called botton-up
integration. This approach integrates infrastructure components that provide common
services, such as network and database access, and then add the functional components.

The integration testing includes structure tests and functional tests. Structure tests
emphasize on exercising all the input and output parameters of each module and
exercising all modules and all calls to utility routines. For functional tests, the goal is to

demonstrate that all functions specified in the requirements and specification documents are completed.

During the integration, all the modules were combined and tested. The testing environment was consistent for all the modules in terms of interfaces and function calling procedures. The program flow of the modules were reviewed and identified. At the end, the program flow for the entire program were reviewed and identified.

In the e-Sign Dictionary system, bottom-up approach is selected to perform the integration testing. All the lower level modules were divided among our team members. Each of us is required to code and test the module that we selected before integration moves upwards. Then, the next higher level modules are integrated and tested with the already tested modules. This step repeated until all the modules are tested.

The criteria that taken into account while conducting integration testing for e-Sign Dictionary system are:

> All the functions in the system are tested to uncover functional errors.
> Check the data passed between interfaces to make sure that if there is any lost of data across interfaces
> Performance tests are conducted to verify the performance of the system.

5.1.3 System Testing

System testing involves integrating two or more components that implement system functions or features and then testing this integrated system. In an iterative development process, system testing is concerned with testing an increment to be delivered to the customer; in waterfall process, system testing is concerned with testing the entire system.

System testing is the final testing procedure. A system test is a series of different tests designed to fully exercise the system to uncover its limitation and measure its capabilities which results from unanticipated interactions of system components or units. The objective of system testing is to test an integrated system and verify that it meets specified requirements, either functional or non-functional. It actually focuses more on behavior.

The functions in the e-Sign Dictionary system were tested when all the modules were integrated. This type of testing required a large amount of time as the system is quite large and consists of many functions. Anywhere, it is important to perform system testing as it can help us to ensure that our system fulfill all the requirements stated before the development of the system.

5.1.4 User Acceptance Test

User Acceptance Testing is a critical phase of any systems project and requires significant participation by the end user. To be of real use, an acceptance test plan should be developed in order to plan precisely, and in detail, the means by which acceptance will be achieved. The final part of the user acceptance test can also include a parallel run to prove the system against the current system.

The user acceptance test plan will vary from system to system but, in general, the testing should be planned in order to provide a realistic and adequate exposure of the system to all reasonably expected events. The testing can be based upon the user requirements specification to which the system should conform.

In order to conduct this testing, we had asked some of the deaf students and teachers at Sekolah Kebangsaan Vokasional Khas, Shah Alam, to test our system. Their satisfaction to our system shows that we had fulfills the users' requirements. Below are the survey results that we had done.

Types of search are sufficient

	Totally disagree	Disagree	Not sure	Agree	Totally agree
Student			37.50%	50.00%	12.50%
Teacher			25.00%	25.00%	50.00%
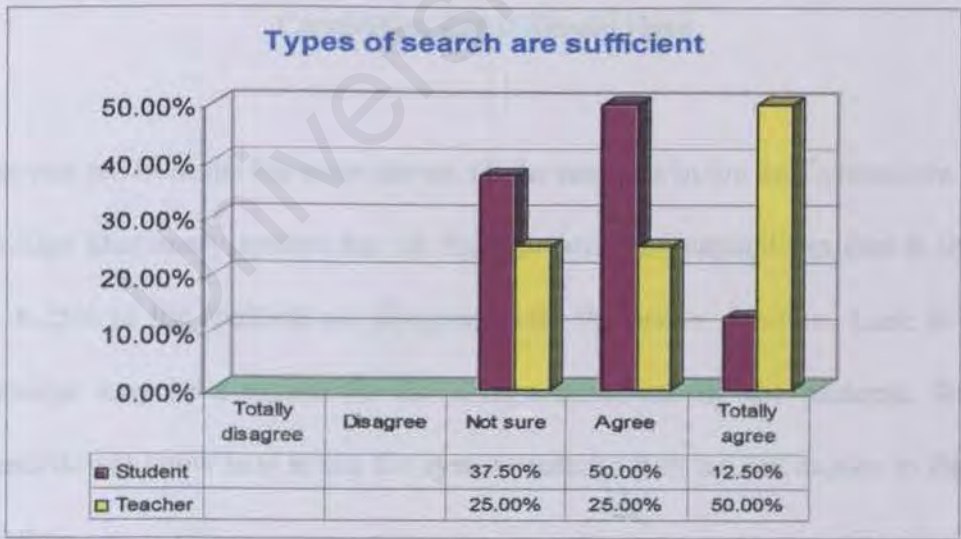
Figure 5.2: Survey Result on Type of Search are Sufficient

From Figure 5.2, we can see that more than 50% of the people who had participated in our survey are agreed that the searching methods in the e-Sign Dictionary system are sufficient. Anywhere, more than a quarter of the participants are not very sure about this function. This happened due to the unfamiliar usage of the system that they are testing.

e-Sign has all the functions and capabilities that it should have

	Totally disagree	Disagree	Not sure	Agree	Totally agree
Student		6.25%	31.25%	50.00%	12.50%
Teacher		0.00%	0.00%	100.00%	0.00%
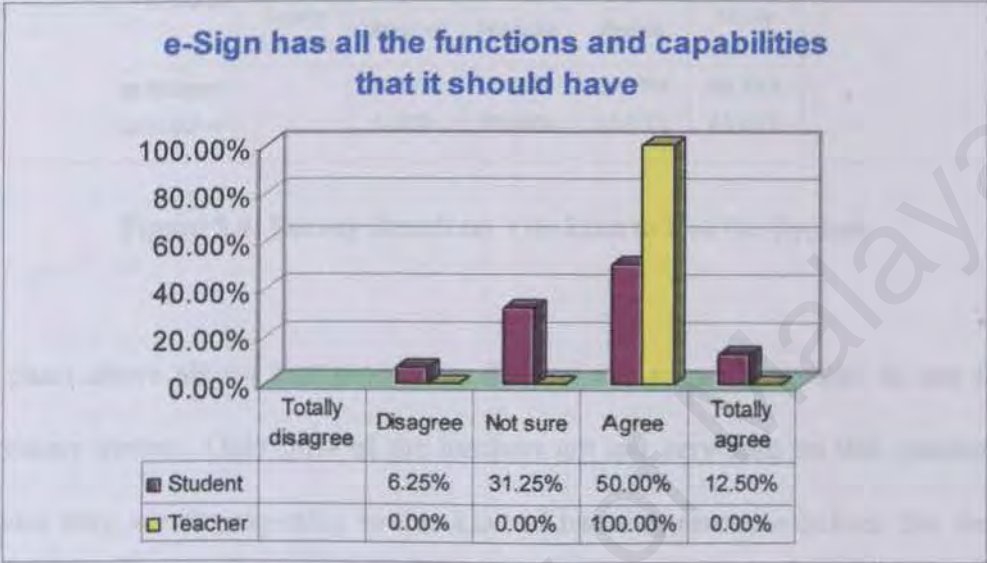
Figure 5.3: Survey Result on e-Sign Has All the Functions and

Capabilities that It Should Have

As we can see from the bar chart above, all the teachers in the deaf school are agreed that the e-Sign Dictionary system has all the functions and capabilities that it should have. Only 6.25% of the students are disagreed with the above question. Lack of computing knowledge may be a reason for the disagreement among the students. Some of the students do not know how to use the system because they are not expose to the use of the computer.

You like to use the system

	Totally disagree	Disagree	Not sure	Agree	Totally agree
Student		6.25%	6.25%	43.75%	43.75%
Teacher		0.00%	50.00%	25.00%	25.00%
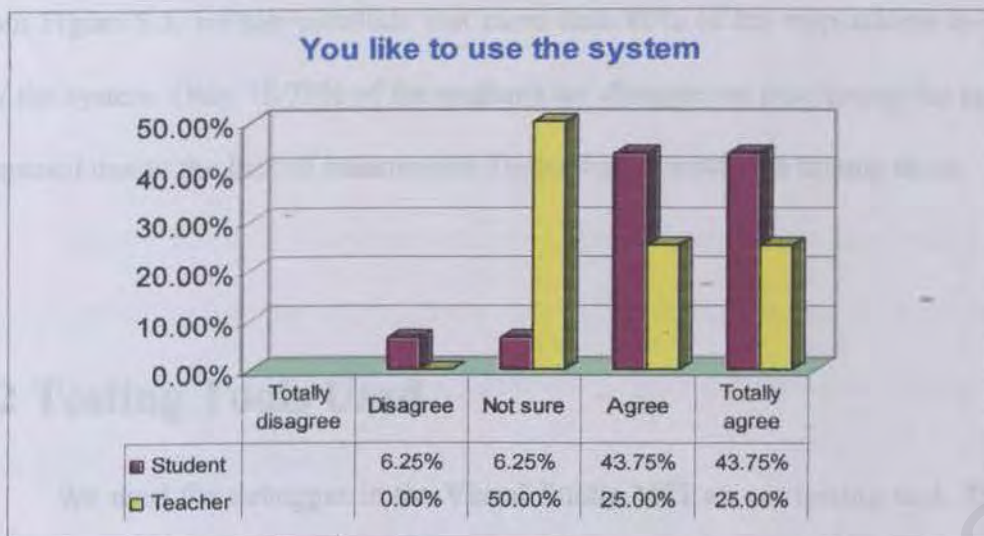
Figure 5.4: Survey Result on You Like to Use the System

Bar chart above shows that more than 80% of the respondents like to use the e-Sign Dictionary system. Only 50% of the teachers are not very sure on this question. This is because they are not exposing to this kind of learning resource before. So, they are not very sure whether this resource is effective or not.

You will buy the system

	Totally disagree	Disagree	Not sure	Agree	Totally agree
Student		18.75%	18.75%	31.25%	31.25%
Teacher		0.00%	25.00%	50.00%	25.00%
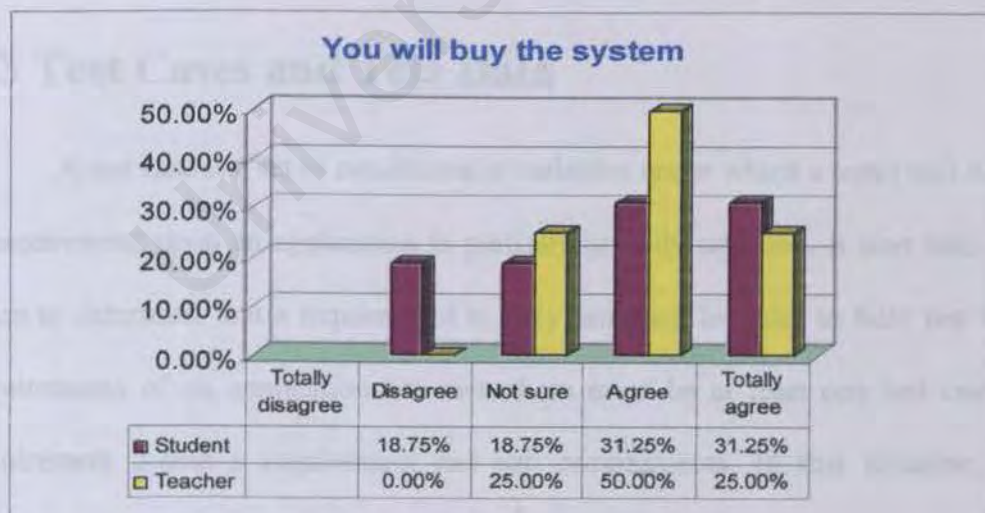
Figure 5.5: Survey Result on You Will Buy the System

From Figure 5.5, we can conclude that more than 80% of the respondents are willing to buy the system. Only 18.75% of the students are disagree on purchasing the system. This happened due to the lack of Information Technology knowledge among them.

5.2 Testing Tools Used

We used the debugger in the Visual Studio.NET as our testing tool. To test each line of code, we mark a breakpoint at the beginning of the line. From there, we can detect whether all the lines of code has been gone through or not. If there is an error in certain line, the debugger will skip the line and display the error message.

Testing using breakpoint is important in determining the value of a variable at certain line. This will be easier for us to review the algorithm that we used. Incorrect usage of algorithm will cause logic error or database error in our system.

5.3 Test Cases and Test Data

A test case is a set of conditions or variables under which a tester will determine if a requirement upon an application is partially or fully satisfied. It may take many test cases to determine that a requirement is fully satisfied. In order to fully test that all the requirements of an application are met, there must be at least one test case for each requirement unless a requirement has sub requirements. In that situation, each sub requirement must have at least one test case.

What characterizes a formal, written test case is that there is a known input and an expected output, which is worked out before the test is executed. The known input should test a precondition and the expected output should test a post condition.

We had divided the test modules among our team members. Each of us is required to come out with some test cases and test data for the module those we responsible. In the e-Sign Dictionary system, I was responsible on the animation module. Below are the test cases for the animation module.

Table 5.1 Test Cases for Animation Module

Step	Test Procedure	Expected Output	Test Result (Passed or Failed)
1	Select the man animation model.	The selected animation model is checked on the man option.	Passed
2	Select the woman animation model	The selected animation model is checked on the woman option.	Passed
3	Select the boy animation model	The selected animation model is checked on the boy option.	Passed
4	Select the girl animation model	The selected animation model is checked on the girl option.	Passed
5	Play the animation for a particular word when the selected animation model is man.	The sign language for the particular word is showed by the man model.	Passed
6	Play the animation for a particular word when the selected animation model is woman.	The sign language for the particular word is showed by the woman model.	Passed
7	Play the animation for a particular word when the selected animation model is boy.	The sign language for the particular word is showed by the boy model.	Passed
8	Play the animation for a particular word when the selected animation model is girl.	The sign language for the particular word is showed by the girl model.	Passed

9	Click the play button located below the animation frame.	The animation for the particular word is showed again.	Passed
10	Click the pause button located below the animation frame.	The animation for the particular word is paused.	Passed
11	Click the stop button located below the animation frame.	The animation for the particular word is stopped.	Passed
12	Click the zoom in button located below the animation frame.	The animation will enlarge.	Passed
13	Click the zoom out button located below the animation frame.	The animation returns to the normal size.	Passed
14	Click the rotate 90° to the right button.	The animation showed on the main frame is rotating 90° to the right side	Passed
15	Click the rotate 90° to the left button.	The animation showed on the main frame is rotating 90° to the left side	Passed
16	Drag the speed control to the slow option.	The speed of the animation showed on the main frame is less than the normal speed.	Passed
17	Drag the speed control to the fast option.	The speed of the animation showed on the main frame is greater than the normal speed	Passed

CHAPTER 6:DISCUSSION AND CONCLUSION

6.1 Problems Encountered and Solutions

Developing a system is not as easy as we think. It involves a lot of stages and phases. Throughout the development of e-Sign Dictionary system, we have encountered several problems. Below are the problems that we encountered during the system development.

6.1.1 Use of New Development Tools

At the beginning of the development process, we faced some difficulty in using the new development tools such as 3D Poser 6.0 and Text-to-Speech.

- 3D Poser 6.0 was used to construct the animation files. All the sign language of the words were constructed or built using 3D Poser 6.0. At the beginning, we have some difficulty in building the animation models for our system. As we do not have any knowledge using this software before, we spent almost 2 weeks to construct a model.

- In e-Sign Dictionary system, we provide a function for the user to learn the pronunciation of a particular word. We used Text-to-Speech to develop this function. At the earlier stage of the development process, we faced some difficulties to incorporate the Text-to-Speech codes and file into our system.

In order to allow us to deliver a complete system on time, we had tried our best to find the solutions for our problems. The solutions that we had found are:

- ❖ Learn how to use the 3D Poser 6.0 software by following the online-tutorial.
- ❖ Refer to the seniors to find out how to incorporate the Text-to-Speech codes into our system.
- ❖ Refer to the online help found in the internet.

6.1.2 Lack of Programming Skills

We used VB.NET to program the e-Sign Dictionary system. As we did not expose to this programming language before, we have some difficulties in coding the functions in our system. In addition, we also faced some difficulties in writing the codes for connecting the system to the database.

Not to deny that, a large amount of time was spent on finding the codes for the functions in our system. To solve this problem, we referred to the VB.NET books to find out the syntax that we do not know. Besides, we also referred to the online help provided on the internet.

6.1.3 Poor Programming Techniques

Poor programming techniques are one of the problems that we encountered during the development of e-Sign Dictionary system. As we do not have any experience in using the VB.NET to develop a system, we failed to write the coding for the system in more efficient way. At the beginning of the system development, we do not know how to reuse some of our codes on different pages.

Anywhere, we had managed to find the solution for this problem. We had consulted our course mates who have experience using VB.NET. Besides, we also referred to the VB.NET books to learn the good programming techniques.

6.2 System Evaluation

Evaluation is the ultimate phase of developing a system. It is also an important phase that we should go through before delivering the system to the end users. System evaluation is implemented by more than just simply comparing the information obtained with the expected information. It was related to user environment, attitudes, information priorities and several other concerns that are to be considered carefully before effectiveness can be concluded. At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on a variety of sources and information.

6.2.1 System Strengths

The strengths of e-Sign Dictionary system cannot be overlooked. This system consists of many useful, unique and important features and functions. The strengths of e-Sign system can be discussed as below:

✓ Trilingual system

e-Sign Dictionary is a trilingual system with instructions in Malay, English and Chinese.

✓ Animation

e-Sign Dictionary system provides a function that allow the users to view the sign language of a word from different angles- from the front, left and right.

✓ Rotation

The animation shown in this system can be rotated 90° to the left or to the right. In addition, e-Sign Dictionary system also provides "zoom –in" and "zoom-out" features which allow the users to adjust the size of the animation according to their preferences.

✓ Four types of models

e-Sign Dictionary system consists a function for the users to choose their favourite animated model, i.e. a man, woman, boy or girl to demonstrate the sign language.

✓ Text-to-Speech

With Text-to-Speech features incorporated into the e-Sign Dictionary system, users can learn the pronunciation of the word that they are searching.

✓ Information about deafness

Information about deafness and Malaysian Sign Language also can be found in the e-Sign Dictionary system. This will actually helps the users to get to know more about deafness and Malaysian Sign Language.

6.2.2 System Weaknesses and Limitations

Not to deny that e-Sign Dictionary system also have some weaknesses and limitations. Below are some of the weaknesses and limitations that we had identified in our system.

✓ Admin technical limitation

In order to add a new word into the e-Sign Dictionary system, the administrator needs to construct the sign for the word first. So, the administrator must not only know the sign language of a word but also must know how to construct the sign using 3D Poser 6.0.

✓ Animation models

Although we can provide four types of animation models to the users but we failed to provide the models that have a similar look like the Malaysian people. The models provided in our system are more to foreigner look.

✓ Text-to-Speech

The Text-to-Speech modules used in the e-Sign Dictionary system are not complete. Text-to-Speech in Malay and Chinese versions only can provide man voice and Text-to-Speech in English version only limited to man and woman voice.

6.2.3 Future Enhancements and Expansions

In future, we still need to enhance some parts of our system to make it more complete and perfect. The limitations that we had identified in our system are the important parts that we would like to enhance in future.

✓ Construct the animation models that look like Malaysian.

Because our system is mainly focused on Malaysian Sign Language, so it will be better to have animation models that look exactly like Malaysian.

✓ Add more language types into the e-Sign Dictionary system.

More language types (such as Tamil and others languages used by the natives in east Malaysia) should be added into our system to fulfill the needs of different users.

✓ Add complete modules of Text-to-Speech into the e-Sign Dictionary system

Since the Text-to-Speech that we had incorporated in our system does not have some of the humans' voice that we required, we have to find the latest version of Text-to-Speech or other software that can provide four types of humans' voice (man, woman, boy and girl) to replace it.

6.3 System Applications

6.3.1 Usefulness of the System

e-Sign Dictionary system can be a good learning tool for the deaf community as well as the public in general. Through this system, users will be able to learn not only the sign language of a particular word but also the meaning of the word in English, Malay and Chinese. The synonym and antonym of the word as well as examples of its usage in a sentence are also provided.

Besides, e-Sign Dictionary system also has a function that can help the users to learn the pronunciation of the words they are searching. This will actually help the user who is weak in pronunciation to pronounce a word correctly.

Not only that, users can learn the sign language and the meaning of a word at any time because our system is a standalone system. So, it is so convenient for the users to learn the sign language.

6.3.2 Benefits Gained

Not to deny that, e-Sign Dictionary system brings a lot of benefits not only to the deaf community but to the general public as well. By using this system, users will be able to:

✓ learn the sign language for a particular word.
✓ learn the meaning, synonym and antonym of a particular word.
✓ get to know more about deafness and Malaysian Sign Language.
✓ learn the pronunciation of a word.
✓ close the communication gap between the deaf community and the general public. No doubts, it is not easy to convey a message to the deaf community because they cannot hear what we are saying. So, by using this system, users will be able to learn the sign language for a particular word. This will actually help to bridge the communication gap between the deaf community and the general public.

6.4 Conclusion

6.4.1 Knowledge and Experience Gained

By developing this system, I gained a lot of knowledges and good experiences. I realized that developing a system is not an easy task. We must have a good planning and appropriate schedules before we can come out with a complete system. Besides, I learned that there are actually a lot of stages and process that we should follow in order to come out with an impressive system. All the steps or stages in software development process cannot be abandoned. It is important to follow the software life cycle which start with requirement definition and end with operation and maintenance process. If we try to skip any stages of the software development process, we will face trouble throughout our development process.

Besides, it is important to fulfill the users' needs. If we come out with a system that does not fulfill our user requirements or our project objectives, our poject or system can be considered as failed.

The most precious experience and knowledge that I gained from this project is the important of teamwork. No doubts, it is important to have a full cooperation from your team members. A project may be failed if all the team members do not play their roles in the development team. Our system may not be able to complete if all the team members do not participate and cooperate with one another. So, I can say that team spirit is very important in determining the success of a project.

6.4.2 Communication and Presentation Skills

At the end of the project, we are required to present our system to our panels and supervisor. So, it is important to have good communication and presentation skills. If our presentation skills are very low, we will not be able to deliver our message clearly. Besides, we may not be able to explain to our panels and supervisor about our system.

I think it is important to have this section (presentation section) at the end of our project because from there we will be able to learn and improve our communication and presentation skills. This will prepare us with the real working environment where we need to present our product or system to our customers or our manager.

6.4.3 Project Outcome

Finally the e-Sign Dictionary system has been successfully completed. Most of the requirements and objectives defined earlier have been achieved.

Throughout the development of this project, a lot of precious knowledge was gained. This project provides an opportunity for me to build an application or system from scratch. It is a great challenge to complete this project on time. In addition, lessons learnt throughout the last three years in the university are able to be applied in this project.

From this project, one can realize that there are still a lot of things to learn especially in this fast growing world of Information Technology age because the knowledge obtained in university is just the foundation for us to face the future challenge. Last but not least, this project has armed me with invaluable knowledge and experience.

REFERENCES

Daniel Galin. 2004. *Software Quality Assurance From Theory to Implementation*. United Kingdom, Pearson Addison Wesley.

Thomas, M.C. and Carolyn, E.B. 2002. *Database System: A Practical Approach to Design, Implementation and Management*. 3rd Edition. England, Pearson Education Limited.

David I. Schneider. 2003. *An Introduction To Programming Using Visual Basic.NET*. 5th Edition. New Jersey, Prentice Hall, Inc.

Deaf and hard of hearing people. 2005. *http://www.rnid.org.uk/*. RNID.org.uk. (Date Accessed: August 21, 2005).

Gerald Kotonya and Ian Sommerville. 2003. *Requirements Engineering*. England, John Wiley & Sons Ltd.

Ian Sommerville. 2004. *Software Engineering*. England, Pearson Education Limited.

Jeffrey A.Hoffer, Joey F.George and Joseph S.Valacich. 2005. *Modern Systems Analysis And Design*. New Jersey, Pearson Education, Inc.

Modular Programming. 2002. http://xbnotes.freehosting.net/. Last Updated: April 11, 2002. Ken Minogue. (Date Accessed: March 31, 2006).

Program Pendidikan Khas untuk Pelajar Pekak di Malaysia. 2004. *http://www.epekak.net.my/*. e-pek@k. (Date Accessed: August 15, 2005).

Registered Deaf Population in Malaysia By End 2000. *http://www.epekak.net.my/*. (Date Accessed: August 15, 2005).

Test Case. 2006. http://en.wikipedia.org/wiki/Test_case. Last Updated: April 2, 2006. Wikimedia Foundation, Inc. (Date Accessed: April 3, 2006).

The Waterfall Model. 2003. *http://scitec.uwichill.edu.bb/cmp/online/_cs22l/waterfall_model.htm*. Last Updated: September 15, 2003. Adrian Als & Charles Greenidge. (Date Accessed: August 25, 2005).

Types of hearing aid. 2005. *http://www.rnid.org.uk/*. RNID.org.uk. (Date Accessed: August 21,2005).

VB.NET Coding Guidelines. 2005. http://addressof.com/blog/. Last Updated: March 16, 2005. Cory Smith. (Date Accessed: April 1, 2005).

What is deafness?.2005. *http://www.teachnet.ie/mhickey/*. Teachnet.ie. (Date Accessed: August 20, 2005).

3D Canvas. 2005. http://www.amabilis.com/. Amabilis Software. (Date Accessed: September 6th, 2005).

APPENDIX A

Survey Form

Purpose: **To gather requirements for the development of an Electronic Sign Language Dictionary for Malaysians in particular, for the Malaysian deaf community.**

Instructions:
Please tick (✓) the relevant box or boxes provided and/or fill in the spaces where appropriate.

Part 1: Personal Information

i.   Occupation : ☐ Deaf Student   ☐ Teacher

ii. Gender   : ☐ Male   ☐ Female

iii. Age group   : ☐ 15 - 18 years   ☐ 18-30 years
                    ☐ 31 - 50 years   ☐ Above 50 years

Part 2: Preferences about the system

1. Have you used any Sign Language Dictionary before?

    ☐ Yes   ☐ No

    (If No, please proceed to Question 3.)

2. What types of Sign Language Dictionary have you used?

    (You may tick (✓) more than one option and/or specify otherwise.)

    ☐ Electronic dictionary (Internet, software, etc.)

    ☐ Paper-based dictionary (Hardcopy.)

    ☐ Others, please specify: _____

3. What are the features do you think a good Electronic Sign Language Dictionary should have? (You may tick (✓) more than one option and/or specify otherwise.)

☐ Easy to use ☐ Attractive graphics

☐ User friendly interfaces ☐ Encourage interactive learning

☐ Others, please specify: _____

4. What do you think a good Electronic Sign Language Dictionary should consists of? (You may tick (✓) more than one option and/or specify otherwise.)

☐ Introduction of Deafness ☐ Search function

☐ Introduction of Sign Language ☐ Multilingual

☐ Facial expression (e.g.: sad, happy, etc.) ☐ Text – to – speech

☐ Meanings of the word ☐ Instruction of hands movement

☐ Hand(s) movements to illustrate the word

☐ Others, please specify: _____

5. What languages do you think should be included in an Electronic Sign Language Dictionary?

(You may tick (✓) more than one option and/or specify otherwise.)

☐ English ☐ Malay

☐ Chinese

☐ Others, please specify: _____

6. What types of searching functions are suitable to be included in an Electronic Sign Language Dictionary?

(You may tick (✓) more than one option and/or specify otherwise.)

☐ Search by keyword ☐ Search by category

☐ Search by alphabetical order ☐ Last search and/or within last week

☐ Last search and/or within last month ☐ Last search and/or within last 3 months

☐ Others, please specify: _____

7. Which of the following should be included in the search by category function? (You may tick (✓) more than one option and/or specify otherwise.)

☐ Alphabets (e.g.: a, b, etc) ☐ Numbers and fraction (e.g.: 1, 1/2, etc)

☐ Body (e.g.: eye, head, etc) ☐ Occupations (e.g.: teacher, policeman, etc)

☐ Buildings (e.g.: house, school, etc) ☐ Relationship (e.g.: parent, friend, etc.)

☐ Colours (e.g.: blue, yellow, etc.) ☐ State in Malaysia (e.g.: Malacca, Perak,

etc)

☐ Emotions (e.g.: happy, sad, etc.) ☐ Vehicles (e.g.: car, bus, etc.)

☐ Fruits (e.g.: apple, orange, etc.) ☐ Time (e.g.: hour, minute, etc.)

☐ Day (e.g.: Monday, Tuesday, etc.) ☐ Month (e.g.: January, February, etc.)

☐ Others, please specify: _____

8. Which of the following functions and features do you think should be incorporated in an Electronic Sign Language Dictionary? (You may tick (✓) more than one option and/or specify otherwise.)

☐ Sign rotation ☐ Zoom in and zoom out

☐ Speed control

☐ Others, please specify: _____

9. How many degrees do you prefer the sign to rotate?

☐ 90 degree ☐ 180 degree

☐ 270 degree ☐ 360 degree

☐ Self-determine angle (e.g.: 110 degree), please specify: _____

10. Which of the following sign character do you prefer?

(You may tick (✓) more than one option and/or specify otherwise.)

☐ Man ☐ Woman

☐ Boy ☐ Girl

☐ Cartoon of human face ☐ Cartoon of animal face

☐ Others, please specify: _____

11. What kind of colours do you prefer?

☐ Bright colour (e.g: yellow, pink)

☐ Dark colour (e.g: grey, black)

☐ Combination of bright and dark colours

☐ Others, please specify: _____

12. Would you buy an Electronic Sign Language Dictionary priced between RM50 – RM60?

☐ Yes   ☐ No

<p style="text-align:center">Thank you for your participation in the survey.</p>

E-SIGN SYSTEM EVALUATION QUESTIONNAIRE

e-Sign: e-Sign Dictionary Usage Indicator

Objective:
This survey is conducted to gather data on the user evaluation of **e-Sign**.

Section 1: Respondent Details
Please ✓ the selected option.

1. Gender group
   ☐ Male   ☐ Female

2. Race
   ☐ Malay   ☐ Chinese   ☐ Indian   ☐ Others, please specify:
   _____

3. Age group
   ☐ < 21   ☐ 21 – 30   ☐ 31 – 40   ☐ 41- 50   ☐ >50

4. Do you use any electronic sign language dictionary before?
   ☐ Yes   ☐ No

Section 2: Evaluation of e-Sign

Choose and circle the most appropriate score for each question given below pertaining to the software, **e-Sign**.

Opinion	Score
Totally disagree	1
Disagree	2
Not sure	3
Agree	4
Totally agree	5

1. Easy to use.   1 2 3 4 5

2. Confident in using **e-Sign**.   1 2 3 4 5

3. All the search functions are sufficient.   1 2 3 4 5

4. Helps in giving the correct content of the word.   1 2 3 4 5

5. **e-Sign** has all the functions and capabilities that it should have.   1 2 3 4 5

6. Help function are useful.   1 2 3 4 5

7. Error messages are clear on how to fix problems quickly and easily.   1 2 3 4 5

8. **e-Sign** responds quickly to inputs.   1 2 3 4 5

9. Organisation of information on the screen is very clear.   1 2 3 4 5

10. The reports are very clear.   1 2 3 4 5

11. Amount of information displayed on the screen is adequate.   1 2 3 4 5

12. Screen sequence is logical.   1 2 3 4 5

13. The wordings are clear and easy to read.   1 2 3 4 5

14. The colour combination is suitable.   1 2 3 4 5

15. Navigation from one screen to another is easy.   1 2 3 4 5

16. Can exit from **e-Sign** at anytime.   1 2 3 4 5

17. You like to use the system.   1 2 3 4 5

18. You will introduce the system to your others.   1 2 3 4 5

19. You will buy the system.   1 2 3 4 5

Section 3: Comments

Other comments about **e-Sign** (if any):

_____

_____

_____

Thank you for your kind participation.

APPENDIX C

SAMPLE SOURCE CODES

Sample of Source Codes for Search Function

```
Private Sub searchwordtb_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles searchwordtb.TextChanged

    ListBox.Items.Clear()

    Dim strSearch As String = searchwordtb.Text

    Dim cnn As SqlConnection = New SqlConnection
    cnn.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn.Open()
    Dim dr As SqlDataReader
    Dim cmd As SqlCommand = New SqlCommand

    If lblSearchLang.Text = "english" Then
        cmd.CommandText = "select * from BI_Dictionary where word LIKE'" &
strSearch & "%' ORDER BY word ASC"
    End If
    If lblSearchLang.Text = "malay" Then
        cmd.CommandText = "select * from BM_Dictionary where word LIKE'" &
strSearch & "%' ORDER BY word ASC"
    End If
    If lblSearchLang.Text = "chinese" Then
        cmd.CommandText = "select * from BC_Dictionary where word LIKE'" &
strSearch & "%' ORDER BY word ASC"
    End If

    cmd.Connection = cnn
    dr = cmd.ExecuteReader()

    Do While dr.Read()
      ListBox.Items.Add(dr.Item("word").ToString())
    Loop
    dr.Close()

    If ListBox.Items.Count = 0 Then
        Dim ans As Integer
        ans = (MsgBox("Sorry. Word begins with " + searchwordtb.Text + " is not found"
+ "" & vbCrLf & "", MsgBoxStyle.Information, "Information"))
```

```
If ans = 1 Then
    searchwordtb.Text = ""
    lblcat.Text = ""

    AxMediaPlayer1.Hide()
    AxMediaPlayer2.Hide()
    AxMediaPlayer3.Hide()

    alphabet.Checked = False
    body.Checked = False
    building.Checked = False
    emotion.Checked = False
    fruit.Checked = False
    relationship.Checked = False
    state.Checked = False
    time.Checked = False
    vehicle.Checked = False
    occupation.Checked = False
    number.Checked = False
    color.Checked = False

    ListBox.Items.Clear()

    If lblSearchLang.Text = "english" Then
        cmd.CommandText = "select word from BI_Dictionary ORDER BY word
ASC"
    End If

    If lblSearchLang.Text = "malay" Then
        cmd.CommandText = "select word from BM_Dictionary ORDER BY word
ASC"
    End If

    If lblSearchLang.Text = "chinese" Then
        cmd.CommandText = "select word from BC_Dictionary ORDER BY word
ASC"
    End If

    cmd.Connection = cnn
    dr = cmd.ExecuteReader()
    Do While dr.Read()
        ListBox.Items.Add(dr.Item("word").ToString())
    Loop
    dr.Close()
```

```
                word.Text = ""
                wordM.Text = ""
                wordC.Text = ""
                mng1.Text = ""
                mng1M.Text = ""
                mng1C.Text = ""
                stc1.Text = ""
                stc1M.Text = ""
                stc1C.Text = ""
                ant1.Text = ""
                ant1M.Text = ""
                ant1C.Text = ""

                syn1.Text = ""
                syn1M.Text = ""
                syn1C.Text = ""

                url1 = ""
                url2 = ""
                url3 = ""
                url4 = ""
                url5 = ""
                url6 = ""
                url7 = ""
                url8 = ""
                url9 = ""
                url10 = ""
                url11 = ""
                url12 = ""
                url13 = ""
                url14 = ""
                url15 = ""
                url16 = ""
                url17 = ""
                url18 = ""
                url19 = ""
                url20 = ""

                searchwordtb.Focus ()


        End If
        End If

    End Sub
```

Sample of Source Codes for Play Animation Function

```
Private Sub Button12_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles playButt.Click

    AxMediaPlayer1.Play()
    AxMediaPlayer2.Play()
    AxMediaPlayer3.Play()
End Sub
```

Sample of Source Codes for Pause Animation Function

```
Private Sub Button13_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles pauseButt.Click
    AxMediaPlayer1.Stop()
    AxMediaPlayer2.Stop()
    AxMediaPlayer3.Stop()
End Sub
```

Sample of Source Codes for Stop Animation Function

```
Private Sub Button14_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles stopButt.Click
    AxMediaPlayer1.Stop()
    AxMediaPlayer2.Stop()
    AxMediaPlayer3.Stop()
End Sub
```

Sample of Source Codes for the Add Function

```vb
Private Sub AddRecordButt_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles AddRecordButt.Click

    Dim randomNum As New Random
    Dim txtDictionaryEngID As String
    Dim txtDictionaryBMID As String
    Dim txtDictionaryBCID As String
    Dim txtDictionaryHeaderID As String
    Dim myCount As String
    Dim result As String
    Dim strSearch1 As String
    Dim addWord As String

    txtDictionaryEngID = randomNum.Next(0, 2000)
    txtDictionaryHeaderID = randomNum.Next(0, 2000)
    strSearch1 = engW.Text

    Dim cnn As SqlConnection = New SqlConnection
    cnn.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn.Open()

    Dim drAdd As SqlDataReader
    Dim cmd As SqlCommand = New SqlCommand


    cmd.CommandText = "select word from BI_Dictionary WHERE word LIKE'" &
strSearch1 & "%' "

    cmd.Connection = cnn
    drAdd = cmd.ExecuteReader()
    Do While drAdd.Read()
        myCount = myCount + 1
    Loop
    result = myCount + 1
    drAdd.Close()
    addWord = engW.Text + "(" + result + ")"
    Dim dr As DataRow = mds.Tables("BI_Dictionary").NewRow()
    dr(0) = txtDictionaryEngID
    If (result = 1) Then
        dr(1) = engW.Text
    Else
```

```
      dr(1) = addWord
End If
dr(2) = engMng1.Text
dr(3) = eng_sen1.Text
dr(4) = engA1.Text
'dr(5) = engA2.Text
dr(6) = engS1.Text
'dr(7) = engS2.Text
dr(8) = txtDictionaryHeaderID

mds.Tables("BI_Dictionary").Rows.Add(dr)
mda.Update(mds, "BI_Dictionary")

Dim dr1 As DataRow = mds1.Tables("BM_Dictionary").NewRow()

txtDictionaryBMID = randomNum.Next(0, 2000)

dr1(0) = txtDictionaryBMID
dr1(1) = bmW.Text
dr1(2) = bmMng1.Text
dr1(3) = bm_Sen1.Text
dr1(4) = bmA1.Text
'dr1(5) = bmA2.Text
dr1(6) = bmS1.Text
'dr1(7) = bmS2.Text
dr1(8) = txtDictionaryHeaderID

mds1.Tables("BM_Dictionary").Rows.Add(dr1)
mda1.Update(mds1, "BM_Dictionary")

Dim dr2 As DataRow = mds2.Tables("BC_Dictionary").NewRow()

txtDictionaryBCID = randomNum.Next(0, 2000)

dr2(0) = txtDictionaryBCID
dr2(1) = bcW.Text
dr2(2) = bcMng1.Text
dr2(3) = bc_Sen1.Text
dr2(4) = bcA1.Text
'dr2(5) = bcA2.Text
dr2(6) = bcS1.Text
'dr2(7) = bcS2.Text
dr2(8) = txtDictionaryHeaderID

mds2.Tables("BC_Dictionary").Rows.Add(dr2)
mda2.Update(mds2, "BC_Dictionary")
```

```
Dim dr3 As DataRow = mds3.Tables("Category").NewRow()
Dim randomCat As New Random
Dim txtCategory As String

txtCategory = randomCat.Next(0, 2000)

dr3(0) = txtCategory
dr3(1) = engCat.Text
dr3(2) = bmCat.Text
dr3(3) = bcCat.Text

mds3.Tables("Category").Rows.Add(dr3)
mda3.Update(mds3, "Category")

Dim dr4 As DataRow = mds4.Tables("DictionaryHeader").NewRow()

dr4(0) = txtDictionaryHeaderID
dr4(1) = txtCategory
dr4(2) = ""
dr4(3) = boyM.Text
dr4(4) = girlM.Text
dr4(5) = manM.Text
dr4(6) = womanM.Text
dr4(7) = boyL.Text
dr4(8) = girlL.Text
dr4(9) = manL.Text
dr4(10) = womanL.Text
dr4(11) = boyR.Text
dr4(12) = girlR.Text
dr4(13) = manR.Text
dr4(14) = womanR.Text
dr4(15) = boy_turnL.Text
dr4(16) = girl_turnL.Text
dr4(17) = man_turnL.Text
dr4(18) = woman_turnL.Text
dr4(19) = boy_turnR.Text
dr4(20) = girl_turnR.Text
dr4(21) = man_turnR.Text
dr4(22) = woman_turnR.Text

mds4.Tables("DictionaryHeader").Rows.Add(dr4)
mda4.Update(mds4, "DictionaryHeader")

MsgBox("Record has been added." & vbCrLf & "", 48, "Confirmation")
DengW.Text = ""
```

```
DbmW.Text = ""
DbcW.Text = ""
DengCat.Text = ""
DbmCat.Text = ""
DbcCat.Text = ""
DengMng1.Text = ""
DbmMng1.Text = ""
DbcMng1.Text = ""
Deng_Sen1.Text = ""
Dbm_Sen1.Text = ""
Dbc_Sen1.Text = ""
DengA1.Text = ""
DbmA1.Text = ""
DbcA1.Text = ""
'DengA2.Text = ""
'DbmA2.Text = ""
'DbcA2.Text = ""
DengS1.Text = ""
DbmS1.Text = ""
DbcS1.Text = ""
'DengS2.Text = ""
'DbmS2.Text = ""
'DbcS2.Text = ""
DmanM.Text = ""
DmanL.Text = ""
DmanR.Text = ""
Dman_turnL.Text = ""
Dman_turnR.Text = ""
DwomanM.Text = ""
DwomanL.Text = ""
DwomanR.Text = ""
Dwoman_turnL.Text = ""
Dwoman_turnR.Text = ""
DgirlM.Text = ""
DgirlL.Text = ""
DgirlR.Text = ""
Dgirl_turnL.Text = ""
Dgirl_turnR.Text = ""
DboyM.Text = ""
DboyL.Text = ""
DboyR.Text = ""
Dboy_turnL.Text = ""
Dboy_turnR.Text = ""

UengW.Text = ""
UbmW.Text = ""
```

```
    UbcW.Text = ""
    UengCat.Text = ""
    UbmCat.Text = ""
    UbcCat.Text = ""
    UengMng1.Text = ""
    UbmMng1.Text = ""
    UbcMng1.Text = ""
    Ueng_Sen1.Text = ""
    ubm_Sen1.Text = ""
    Ubc_Sen1.Text = ""
    UengA1.Text = ""
    UbmA1.Text = ""
    UbcA1.Text = ""
    'UengA2.Text = ""
    'UbmA2.Text = ""
    'UbcA2.Text = ""
    UengS1.Text = ""
    UbmS1.Text = ""
    UbcS1.Text = ""
    'UengS2.Text = ""
    'UbmS2.Text = ""
    'UbcS2.Text = ""
    UmanM.Text = ""
    UmanL.Text = ""
    UmanR.Text = ""
    Uman_turnL.Text = ""
    Uman_turnR.Text = ""
    UwomanM.Text = ""
    UwomanL.Text = ""
    UwomanR.Text = ""
    Uwoman_turnL.Text = ""
    Uwoman_turnR.Text = ""
    UgirlM.Text = ""
    UgirlL.Text = ""
    UgirlR.Text = ""
    Ugirl_turnL.Text = ""
    Ugirl_turnR.Text = ""
    UboyM.Text = ""
    UboyL.Text = ""
    UboyR.Text = ""
    Uboy_turnL.Text = ""
    Uboy_turnR.Text = ""
    engW.Focus ()

End Sub
```

Sample of Source Codes for the Update Function

```
Private Sub Button47_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button47.Click
    Dim cnn As SqlConnection = New SqlConnection
    cnn.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn.Open()
    Dim dr As SqlDataReader
    Dim cmd As SqlCommand = New SqlCommand
    cmd.CommandText = "UPDATE BI_Dictionary SET word = '" + UengW.Text + "',
meaning = '" + UengMng1.Text + "', sentence = '" + Ueng_Sen1.Text + "', antonym1 = '"
+ UengA1.Text + "',synonym1 = '" + UengS1.Text + "' WHERE DictionaryHeaderID ='"
+ HeaderID + "'"
    cmd.Connection = cnn
    cmd.ExecuteNonQuery()
    cnn.Close()

    Dim cnn1 As SqlConnection = New SqlConnection
    cnn1.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn1.Open()
    Dim dr1 As SqlDataReader
    Dim cmd1 As SqlCommand = New SqlCommand
    cmd1.CommandText = "UPDATE BM_Dictionary SET word = '" + UbmW.Text +
"', meaning = '" + UbmMng1.Text + "', sentence = '" + ubm_Sen1.Text + "', antonym1 =
'" + UbmA1.Text + "',synonym1 = '" + UbmS1.Text + "' WHERE DictionaryHeaderID
='" + HeaderID + "'"
    cmd1.Connection = cnn1
    cmd1.ExecuteNonQuery()
    cnn1.Close()

    Dim cnn2 As SqlConnection = New SqlConnection
    cnn2.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn2.Open()
    Dim dr2 As SqlDataReader
    Dim cmd2 As SqlCommand = New SqlCommand
    cmd2.CommandText = "UPDATE BC_Dictionary SET word = '" + UbcW.Text + "',
meaning = '" + UbcMng1.Text + "', sentence = '" + Ubc_Sen1.Text + "', antonym1 = '" +
UbcA1.Text + "',synonym1 = '" + UbcS1.Text + "' WHERE DictionaryHeaderID ='" +
HeaderID + "'"
    cmd2.Connection = cnn2
    cmd2.ExecuteNonQuery()
    cnn2.Close()
```

```vb
    Dim cnn3 As SqlConnection = New SqlConnection
    cnn3.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn3.Open()
    Dim dr3 As SqlDataReader
    Dim cmd3 As SqlCommand = New SqlCommand
    cmd3.CommandText = "UPDATE DictionaryHeader SET path_boy = '" +
UboyM.Text + "', path_girl = '" + UgirlM.Text + "', path_man = '" + UmanM.Text + "',
path_woman = '" + UwomanM.Text + "',path_SAVLBoy = '" + UboyL.Text +
"',path_SAVLGirl = '" + UgirlL.Text + "',path_SAVLMan = '" + UmanL.Text +
"',path_SAVLWoman = '" + UwomanL.Text + "',path_SAVRBoy = '" + UboyR.Text +
"',path_SAVRGirl = '" + UgirlR.Text + "',path_SAVRMan = '" + UmanR.Text +
"',path_SAVRWoman = '" + UwomanR.Text + "',path_RVLBoy = '" + Uboy_turnL.Text
+ "',path_RVLGirl = '" + Ugirl_turnL.Text + "',path_RVLMan = '" + Uman_turnL.Text +
"',path_RVLWoman = '" + Uwoman_turnL.Text + "',path_RVRBoy = '" +
Uboy_turnR.Text + "',path_RVRGirl = '" + Ugirl_turnR.Text + "',path_RVRMan = '" +
Uman_turnR.Text + "',path_RVRWoman = '" + Uwoman_turnR.Text + "' WHERE
DictionaryHeaderID ='" + HeaderID + "'"
    cmd3.Connection = cnn3
    cmd3.ExecuteNonQuery()
    cnn3.Close()

    Dim cnn4 As SqlConnection = New SqlConnection
    cnn4.ConnectionString = "Data Source = (local); Initial Catalog = eSign; Integrated
Security = SSPI"
    cnn4.Open()
    Dim dr4 As SqlDataReader
    Dim cmd4 As SqlCommand = New SqlCommand
    cmd4.CommandText = "UPDATE Category SET BICategoryDesc = '" +
UengCat.Text + "', BMCategoryDesc = '" + UbmCat.Text + "', BCCategoryDesc = '" +
UbcCat.Text + "' WHERE CategoryID ='" + CatID + "'"
    cmd4.Connection = cnn4
    cmd4.ExecuteNonQuery()
    cnn4.Close()
    MsgBox("Record has been updated" & vbCrLf & "", 48, "Confirmation")
    DengW.Text = ""
    DbmW.Text = ""
    DbcW.Text = ""
    DengCat.Text = ""
    DbmCat.Text = ""
    DbcCat.Text = ""
    DengMng1.Text = ""
    DbmMng1.Text = ""
    DbcMng1.Text = ""
    Deng_Sen1.Text = ""
    Dbm_Sen1.Text = ""
```

```
Dbc_Sen1.Text = ""
DengA1.Text = ""
DbmA1.Text = ""
DbcA1.Text = ""
'DengA2.Text = ""
'DbmA2.Text = ""
'DbcA2.Text = ""
DengS1.Text = ""
DbmS1.Text = ""
DbcS1.Text = ""
'DengS2.Text = ""
'DbmS2.Text = ""
'DbcS2.Text = ""
DmanM.Text = ""
DmanL.Text = ""
DmanR.Text = ""
Dman_turnL.Text = ""
Dman_turnR.Text = ""
DwomanM.Text = ""
DwomanL.Text = ""
DwomanR.Text = ""
Dwoman_turnL.Text = ""
Dwoman_turnR.Text = ""
DgirlM.Text = ""
DgirlL.Text = ""
DgirlR.Text = ""
Dgirl_turnL.Text = ""
Dgirl_turnR.Text = ""
DboyM.Text = ""
DboyL.Text = ""
DboyR.Text = ""
Dboy_turnL.Text = ""
Dboy_turnR.Text = ""

UengW.Text = ""
UbmW.Text = ""
UbcW.Text = ""
UengCat.Text = ""
UbmCat.Text = ""
UbcCat.Text = ""
UengMng1.Text = ""
UbmMng1.Text = ""
UbcMng1.Text = ""
Ueng_Sen1.Text = ""
ubm_Sen1.Text = ""
Ubc_Sen1.Text = ""
```

```
        UengA1.Text = ""
        UbmA1.Text = ""
        UbcA1.Text = ""
       'UengA2.Text = ""
       'UbmA2.Text = ""
       'UbcA2.Text = ""
        UengS1.Text = ""
        UbmS1.Text = ""
        UbcS1.Text = ""
       'UengS2.Text = ""
       'UbmS2.Text = ""
       'UbcS2.Text = ""
        UmanM.Text = ""
        UmanL.Text = ""
        UmanR.Text = ""
        Uman_turnL.Text = ""
        Uman_turnR.Text = ""
        UwomanM.Text = ""
        UwomanL.Text = ""
        UwomanR.Text = ""
        Uwoman_turnL.Text = ""
        Uwoman_turnR.Text = ""
        UgirlM.Text = ""
        UgirlL.Text = ""
        UgirlR.Text = ""
        Ugirl_turnL.Text = ""
        Ugirl_turnR.Text = ""
        UboyM.Text = ""
        UboyL.Text = ""
        UboyR.Text = ""
        Uboy_turnL.Text = ""
        Uboy_turnR.Text = ""
        engW.Focus ()


    End Sub
```

Sample of Source Codes for the Delete Function

```
Private Sub Button28_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button28.Click
    Dim answer As Integer

    answer = MsgBox("Are you sure you want to delete the current record?", 32 +
vbOKCancel)

    If answer = 1 Then

        Dim cnn As SqlConnection = New SqlConnection
        cnn.ConnectionString = "Data Source = (local); Initial Catalog = eSign;
Integrated Security = SSPI"
        cnn.Open()
        Dim dr As SqlDataReader
        Dim cmd As SqlCommand = New SqlCommand
        cmd.CommandText    =    "DELETE    FROM    Bl_Dictionary    WHERE
DictionaryHeaderID ='" + HeaderID + "'"
        cmd.Connection = cnn
        cmd.ExecuteNonQuery()
        cnn.Close()

        Dim cnn1 As SqlConnection = New SqlConnection
        cnn1.ConnectionString = "Data Source = (local); Initial Catalog = eSign;
Integrated Security = SSPI"
        cnn1.Open()
        Dim dr1 As SqlDataReader
        Dim cmd1 As SqlCommand = New SqlCommand
        cmd1.CommandText    =    "DELETE    FROM    BM_Dictionary    WHERE
DictionaryHeaderID ='" + HeaderID + "'"
        cmd1.Connection = cnn1
        cmd1.ExecuteNonQuery()
        cnn1.Close()

        Dim cnn2 As SqlConnection = New SqlConnection
        cnn2.ConnectionString = "Data Source = (local); Initial Catalog = eSign;
Integrated Security = SSPI"
        cnn2.Open()
        Dim dr2 As SqlDataReader
        Dim cmd2 As SqlCommand = New SqlCommand
        cmd2.CommandText    =    "DELETE    FROM    BC_Dictionary    WHERE
DictionaryHeaderID ='" + HeaderID + "'"
        cmd2.Connection = cnn2
```

140

```
cmd2.ExecuteNonQuery()
        cnn2.Close()

        Dim cnn3 As SqlConnection = New SqlConnection
        cnn3.ConnectionString = "Data Source = (local); Initial Catalog = eSign;
Integrated Security = SSPI"
        cnn3.Open()
        Dim dr3 As SqlDataReader
        Dim cmd3 As SqlCommand = New SqlCommand
        cmd3.CommandText    =    "DELETE   FROM   DictionaryHeader   WHERE
DictionaryHeaderID ='" + HeaderID + "'"
        cmd3.Connection = cnn3
        cmd3.ExecuteNonQuery()
        cnn3.Close()

        Dim cnn4 As SqlConnection = New SqlConnection
        cnn4.ConnectionString = "Data Source = (local); Initial Catalog = eSign;
Integrated Security = SSPI"
        cnn4.Open()
        Dim dr4 As SqlDataReader
        Dim cmd4 As SqlCommand = New SqlCommand
        cmd4.CommandText = "DELETE FROM Category WHERE CategoryID ='" +
CatID + "'"
        cmd4.Connection = cnn4
        cmd4.ExecuteNonQuery()
        cnn4.Close()
        MsgBox("Record has been deleted." & vbCrLf & "", 48, "Confirmation")
        'MsgBox("Record Deleted")

        ListBox.Items.Clear()
        Me.ResetText()
        txtSearch.Text = ""
        DengW.Text = ""
        DengMng1.Text = ""
        Deng_Sen1.Text = ""
        DengA1.Text = ""
        DengS1.Text = ""
        'DengA2.Text = ""
        'DengS2.Text = ""

        DbmW.Text = ""
        DbmMng1.Text = ""
        Dbm_Sen1.Text = ""
        DbmA1.Text = ""
        DbmS1.Text = ""
```

```
    DbcW.Text = ""
    DbcMng1.Text = ""
    Dbc_Sen1.Text = ""
    DbcA1.Text = ""
    DbcS1.Text = ""
    'DbcA2.Text = ""
    'DbcS2.Text = ""

    DboyM.Text = ""
    DgirlM.Text = ""
    DmanM.Text = ""
    DwomanM.Text = ""
    DboyL.Text = ""
    DgirlL.Text = ""
    DmanL.Text = ""
    DwomanL.Text = ""
    DboyR.Text = ""
    DgirlR.Text = ""
    DmanR.Text = ""
    DwomanR.Text = ""
    Dboy_turnL.Text = ""
    Dgirl_turnL.Text = ""
    Dman_turnL.Text = ""
    Dwoman_turnL.Text = ""
    Dboy_turnR.Text = ""
    Dgirl_turnR.Text = ""
    Dman_turnR.Text = ""
    Dwoman_turnR.Text = ""
    txtSearch.Focus ()


    End If
End Sub
```