Online Crossword Puzzle for Learning Synonyms and Antonyms Front End User System

2 maroon

RB

By
Tay Lik Czek
(WEK 000366)

Under supervision of
Mr. Chiew Thiam Kian
Faculty of Science and Information Technology
University of Malaya

Moderator
Ms. Norazlina Khamis
Faculty of Science and Information Technology
University of Malaya

This project is submitted to the Faculty of Science and Information Technology,
University of Malaya,
in partial fulfillment of the requirement of the Bachelor of Computer Science

Abstract

Learning is an evolving phenomenon. Efficient ways of learning have been explore. Relating puzzles with learning seem to have received great responds. With today's technology, electronic crossword puzzle can make learning of words easier.

The Online Crossword Puzzle for Learning Synonyms and Antonyms an electronic crossword puzzle with the aim of enabling users learn synonyms and antonyms. The development of this system is divided into two sub modules, which are System Administration and Front End User System. This report focuses on the Front End User module.

The Front End User module focuses mainly on the crossword puzzle playing engine. The playing engine enable users to interact with the puzzle and give immediate responses. A timer is allocated where users need to complete the puzzle within the time frame in order to achieve higher score. Features like enabling users to reveal a letter or word in the puzzle, delete unsatisfied answer and user guide is provided for better usability.

The system is built on a iterative and incremented waterfall model.

The system is based on the Internet based three-tier client server architecture and an iterative and incremented waterfall model will be used to develop this project. Hence, it is believe that this interactive crossword puzzle can benefits to everyone who enjoys playing crossword puzzle.

Acknowledgement

One of the greatest pleasures of writing this report is acknowledging the efforts of many people whose names may not appear on the cover, but whose hard work, cooperation, friendship, and understanding were crucial throughout the undergoing of this project.

Firstly I would like to thank Mr. Chiew Thiam Kian, my supervisor for his guidance and advice throughout the entire project. The time that he has spared with me has made a great contribution to the success of the project.

Following that, I would like to thank my moderator, Ms. Norazlina Khamis for her valuable pointers and comment on this project.

In addition, I would like to express my gratitude to my partner, Ms. Lee Ching Kuan for her willingness to share ideas and cooperation during discussions to make this project successful.

Following that, I would like to thank all my friendly and helpful course mates for sharing their knowledge.

Last but not least, I would like to thank my family for their encouragement and patience in the succession of this project.

Table of Contents

Chapter Six System Implementation

Chapter Seven System Testing

Chapter Eight System Evaluation

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Project definition

The fast growing of technologies in the society has indirectly made interactive online learning a significant role in education. Crossword puzzles, which are made into electronic games as well as online games, can act as an interactive education tool to help in learning. Studies shown playing crossword puzzles can help student memorized better and improve language vocabulary.

Online Crossword Puzzle to Learn Synonyms and Antonyms is a Web-based application aimed to provide an alternative way for user to learn synonyms and antonyms. This application can be accessible anywhere and anytime; provided with an Internet access. It focuses on helping user to achieve higher vocabulary level in a free and easy way with an attractive and user-friendly interface.

Apart from the crossword clues, user can obtain tips by revealing a letter or word from the crossword. As for user who loves competition, score for each game played can be recorded. User who wants to set a self-limitation can benefits from the timer that is featured in the crosswords page. User can then view the top player list, which will be uploaded on the Web site. Guide and help file is available to guide first time user on how to manipulate with the features available on the crossword puzzle page.

1.2 Project Objectives

- To develop a Web-based crosswords puzzle as a learning tool for user to learn synonyms and antonyms.
- To design an easy to use and attractive crossword puzzle interface with good features.
- To make Internet word games a more useful and educational activity for Internet surfers.
- To help broadened the vocabulary limit of users and improve their English language command.

1.3 Project Scope

Online crossword puzzle to learn synonyms and antonyms is a web-based application that can divide into two main sub modules, the administration system and the front end user system.

1.3.2 Front End User System

- Create the crosswords puzzle interface with clues uploaded from the database storage.
- Provide help file and instructions to users.
- Allow users to choose the level of difficulties their prefer.
- Set the timer when user starts playing the crosswords for scoring purposes and allow user to submit completed puzzle.
- Calculate the user's score and display onto scoreboard.

- Users can write their comment about the site on the guest book and view other users comment.
- Users can rate the site by voting in the vote form and view the poll result.

1.3.1 Administration System

- Provide an attractive homepage; manage user registration and store members' information into the database.
- Allow members to update information and change password.
- Provide two versions of interface to suit the wide age range of target users where user can choose either the adult or the junior version.
- Provide an easy to use interface for administrator to create new puzzles with synonyms and antonyms as clues to be uploaded to Web site.
- Allow a super administrator authorization with high security clearance to add and manage other administrators' status.
- Displays puzzle list and status of the puzzle.
- Allow administrator to search and view words stored in the puzzles.
- Allow the administrator to maintain the user's home page guest book.
- Provide a help file to administrator.
- Provide other miscellaneous features on a Web site.

1.4 Project Limitation

In the development of this project, there are some limitations that are unavoidable and some features will be excluded from the system.

- Internet Access

  As a Web-based application, connection to the Internet is compulsory to enable full enjoyment of the crossword game.

- No parallel session

  Players are limited to play one crossword puzzle at a session.

- Screen area limitation

  Due to limitation of the vertical length of the display, the size of the crossword puzzle with full features (timer, clues, word or letter reveal button) will be restricted.

1.5 Project Expected Outcome

The outcome of this project for the Front End User module will be an online crossword puzzle play engine that focuses on synonyms and antonyms words. Below are some of the expected outcomes for this module:

- The system should allow only registered user with a login to play the puzzle
- Users should be able to choose the level of difficulties of a puzzle.
- The system should have a timer that will be automatically triggered once the user invoke the play function and when the timer runs out of time, all functions in the playing grid should be disabled.
- The system should be able to calculate score of the each completed puzzle and the final score of a user and display it.
- The system should provide a clear and easy to understand user guide.
- The play engine command icon should be meaningful and easy to use.
- The play engine should be developed in a way that it is easy for future enhancement.

1.6 Project Methodology

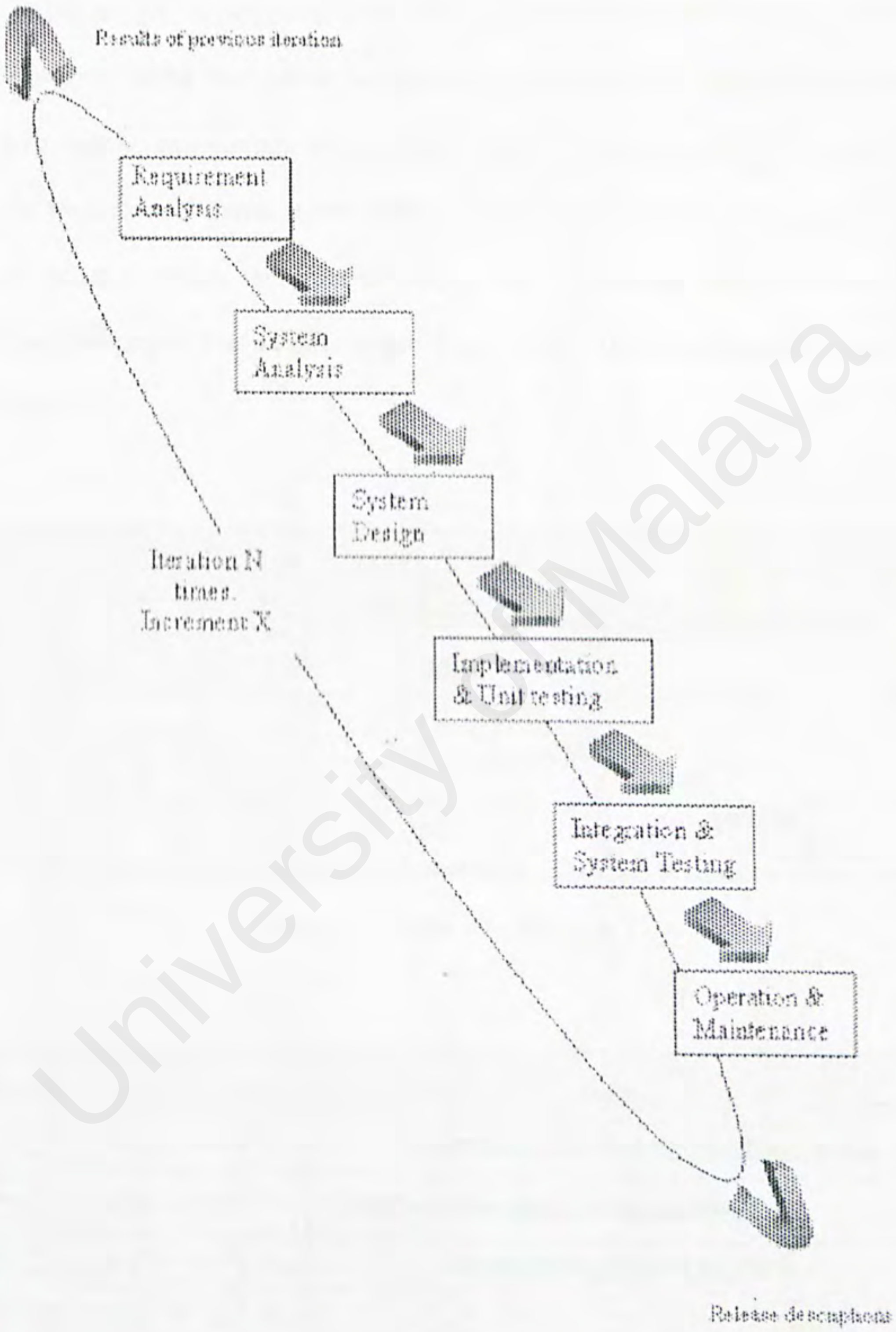
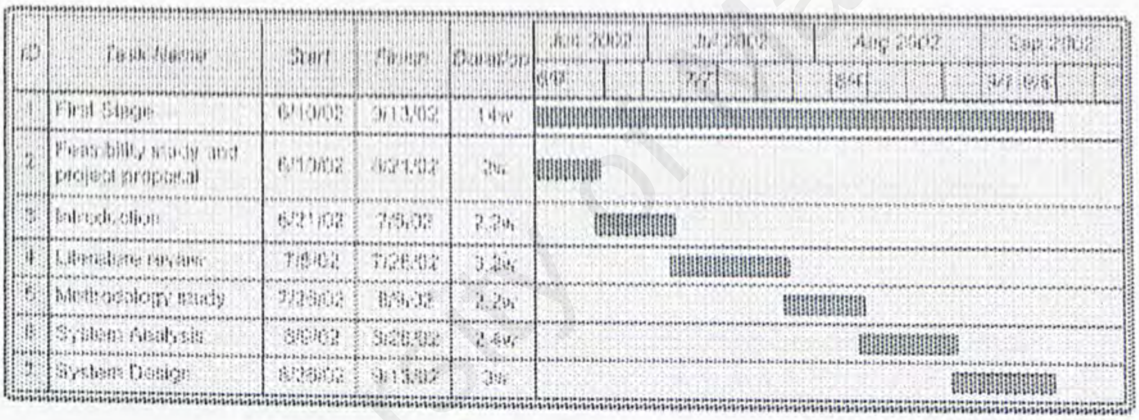


Figure 1.1 The Iterative and Incremented Waterfall Model

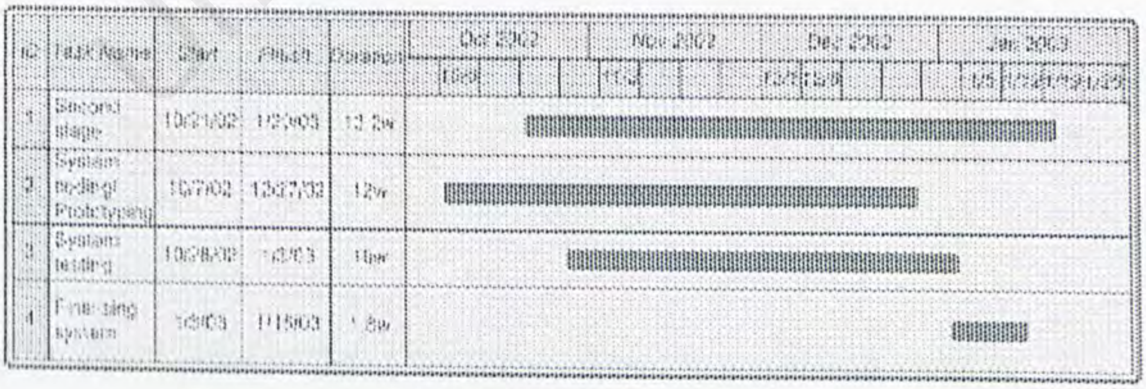
1.7 Project Schedule

This project is split in two phases. The first stage spans within three months. Activities that are done during this period are feasibility study, project proposal, introduction, literature review, methodology study, system analysis, and system design. In short, this stage is to create a blueprint of the project by in depth study of the overall project. The second phase activities are system coding and prototyping, system testing, and finalizing the system. The duration is also three months. The following are the details of the duration.



ID	Task Name	Start	Finish	Duration
1	First Stage	6/10/02	9/13/02	14w
2	Feasibility study and project proposal	6/10/02	6/21/02	2w
3	Introduction	6/21/02	7/5/02	2.2w
4	Literature review	7/8/02	7/26/02	3.2w
5	Methodology study	7/29/02	8/9/02	2.2w
6	System Analysis	8/9/02	8/28/02	2.4w
7	System Design	8/26/02	9/13/02	3w

Figure 1.2 Project schedule stage 1



ID	Task Name	Start	Finish	Duration
1	Second stage	10/21/02	1/9/03	13.2w
2	System coding/ Prototyping	10/7/02	12/27/02	12w
3	System testing	10/28/02	1/3/03	10w
4	Finalising system	1/6/03	1/15/03	1.8w

Figure 1.3 Project schedule stage 2

1.8 Report Layout

Chapter 1: Introduction

This chapter reveal on the overview of the system with the project objectives, scope, limitation and the development schedule.

Chapter 2: Literature Review

This chapter review on some existing systems, system architecture, software technologies and also possible development tools so as to gather more information.

Chapter 3: Methodologies

This chapter discusses the system methodology and techniques used in developing the system.

Chapter 4: System Analysis

This chapter is an analysis on the system functional and non-functional requirements.

Chapter 5: System Design

This chapter explains on the system design with explanation on how the system works.

Chapter 6: System Implementation

In this chapter, the process of converting system requirements and design into program code is being explained.

Chapter 7: System Testing

This chapter discuss on the testing done to the system with explanation on the process and techniques.

Chapter 8: System Evaluation

This chapter reviews the system and evaluates its features, strengths and constrains.

Chapter 2 Literature Review

Before doing any part of the project, the understanding of the topic is important. Hence, doing studies, previews, observations, and researches are necessary. By doing all this, greater understanding and getting concrete evident for the relevant topic. Sources for this purpose are from the Internet, thesis, books, inquires and observations.

2.1 Definition

Below are the definitions of each word of the topic from Chamber's Dictionary

Online - a condition of turned on and being connected to a network of computers or other devices and is frequently used to describe someone who is currently connected to the Internet.

Crossword Puzzle - a puzzle in which words are filled into a pattern of numbered squares in answer to correspondingly numbered clues and in such a way that words read across and down.

Synonyms - one or two or more words or expressions of the same language that have the same or nearly same meaning in some or all senses.

Antonyms - a word of opposite meaning

2.2 History of Crosswords Puzzle

Crossword puzzles are said to be the most popular and widespread word game in the world, yet have a short history. The first crosswords appeared in England during the 19th century. They were of an elementary kind, apparently derived from the word square, a group of words arranged so the letters read alike vertically and horizontally, and printed in children's puzzle books and various periodicals. In the United States, however, the puzzle developed into a serious adult pastime.

The first known published crossword puzzle was created by a journalist named Arthur Wynne from Liverpool, and he is usually credited as the inventor of the popular word game. December 21, 1913 was the date and it appeared in a Sunday newspaper, the *New York World*. Wynne's puzzle(see below) differed from today's crosswords in that it was diamond shaped and contained no internal black squares. During the early 1920's other newspapers picked up the newly discovered pastime and within a decade crossword puzzles were featured in almost all American newspapers. It was in this period crosswords began to assume their familiar form. Ten years after its rebirth in the States it crossed the Atlantic and re-conquered Europe.

2.3 Studies on existing systems

2.3.1 Analysis on The Star Daily Crossword Puzzle

The Star is a local daily newspaper. It provides a daily crossword puzzle within its Section 2. The style of playing it is very traditional by using a pencil and eraser. Players need to look up for the word and match the clue manually by looking along the clue column. Following that, to match the answers, player needs to wait for the next day's copy of the newspaper to match them.

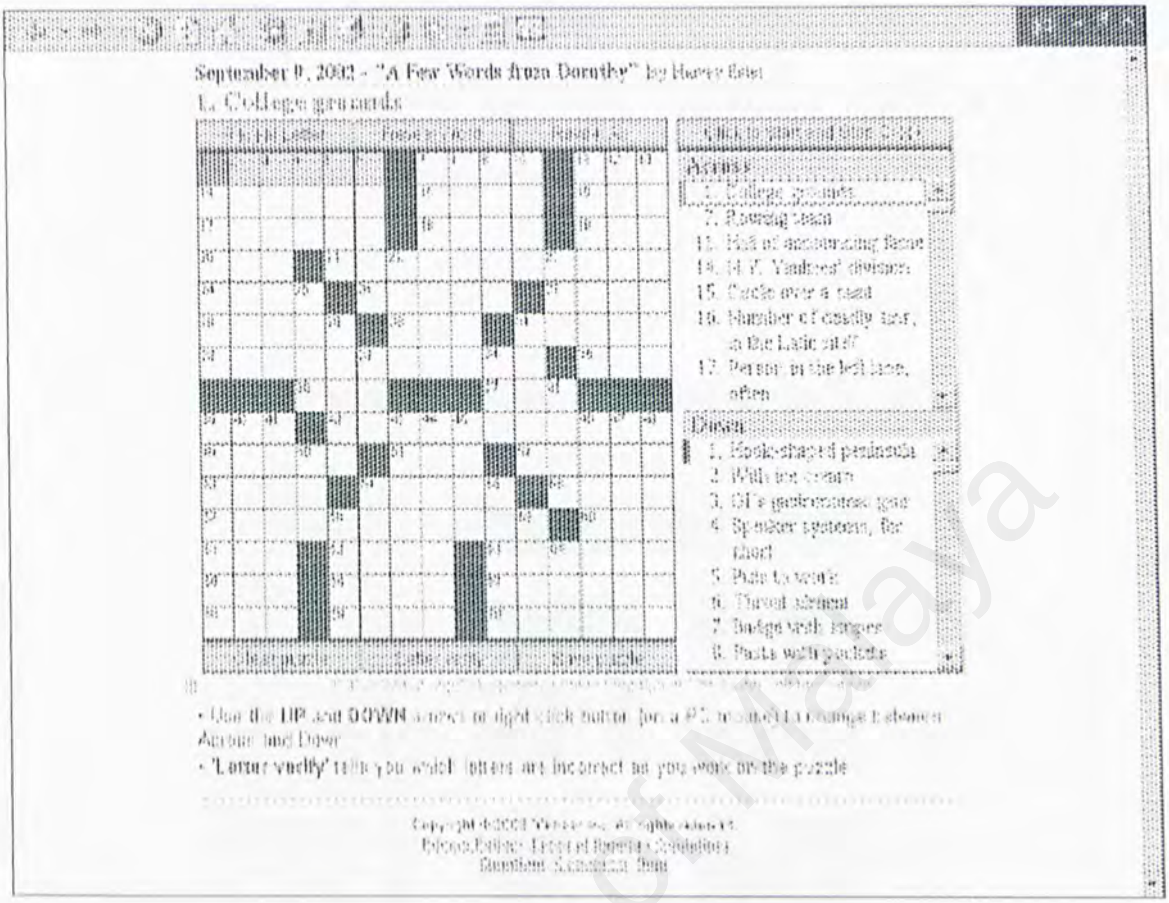
Matching the answer too is a manual and tedious job. The grids are arranged in a compact manner that some times it is hard for beginners to play with it. Hence, reducing the interest in the puzzle. No time limit is fixed on the players. It is more to a leisure puzzle rather than a learning tools as players play during their free time.

2.3.2 Analysis on "Yahoo! Crossword Puzzle"

URL address: http://games.yahoo.com

Yahoo! Games is a web-site where Internet users can play games. The games available range from board game to card games. Players can choose to from single player game (e.g. solitaire) to multiplayer games (e.g. scrabble). The games keep on improving from features, games, and interfacing.

Yahoo! Games crossword puzzle comes in a handy page. The basic instructions needed for playing the games are simple and easy to use.



September 9, 2002 - "A Few Words from Dorothy" by [illegible]

1. College grounds

Across

1. College grounds
7. Rowing team
13. Hall of accounting fame
14. N.Y. Yankees' division
15. Circle over a head
16. Number of deadly sins, in the Latin rite?
17. Person in the left lane, often

Down

1. Hook-shaped peninsula
2. With ice cream
3. GI's gastronomic gear
4. Speaker systems, for short
5. Puts to work
6. Throat element
7. Badge with fingers
8. Pasta with pockets

- Use the UP and DOWN arrows or right click button (on a PC mouse) to change between Across and Down
- 'Letter verify' tells you which letters are incorrect as you work on the puzzle

Copyright © 2002 Yahoo! Inc. All rights reserved.
[illegible]

Figure 2.1 Yahoo! Games Crossword Puzzle

Advantages of this system

- Word/letter reveal function
- Check the word as the play answers
- Easy and simple to use interface
- Built-in timer to check the time needed to complete the puzzle

Disadvantages of this system

- Users are unable to edit or create new puzzles
- Lack of color indicator usage
- No tracking of progression or scoring for each user
- A general crossword without examine on certain areas

2.3.3 Analysis on "Crossword Express v5.6"

Crossword Express is a shareware than enable users to have a game of crossword puzzle. The functions of the application is not just limited to solving a puzzle, hence it can be use as a crossword generator.

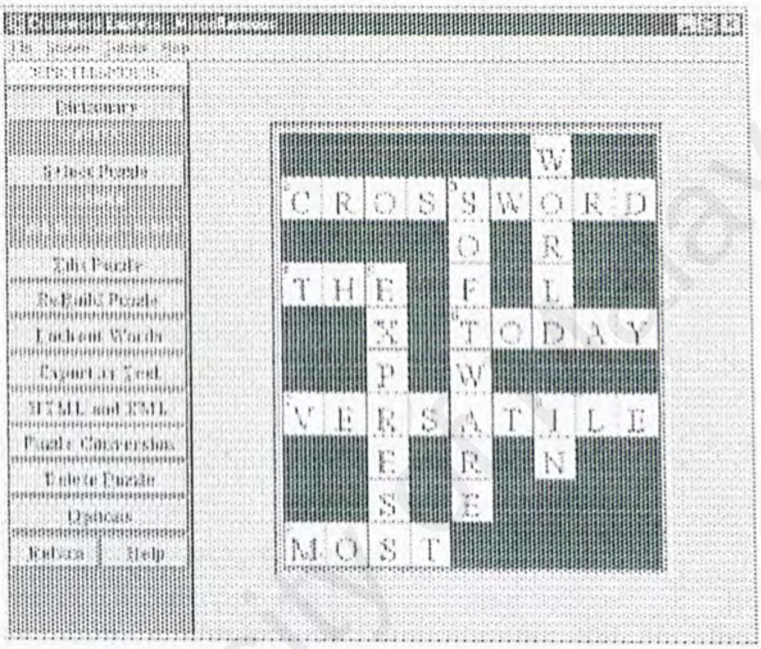


Figure 2.2 Crossword Express

Advantages of this system

- Has a customize dictionary
- Has a crossword generator
- Multiple language support

Disadvantages of this system

- Not a web-base program
- Complicated user interface
- Clue is shown only when related boxed are highlighted

2.4 Software and Technologies

2.4.1 Client/Server

Client/Server computing is a kind of distributed system architecture. In general, client server computing refers to a group of computers link together where one of the computers acts as a service provider. Services provided can be in a form of action, information or messages. The server responds to the message by fulfilling the request after it has done some or no processing at all.

Request made can be a request for database query, application, or remote console procedures. This capability cannot only provide services to clients but share data, application and messages among clients with the server acting as the central hub for all services.

2.4.1.1 Two-tier Client/Server Architecture

A two-tier client/server environment, the clients usually host only the interface. This interface may look impressive but it does not have a back-end engine running behind the scene. The main logical control of the whole process is located at the server. Every data input through the user interface is send over to the server for processing. After all processing had been done, the server returns the desire results to the client. All the client needs to do is the display the information. In some cases, even the formatting of the display output is not necessary.

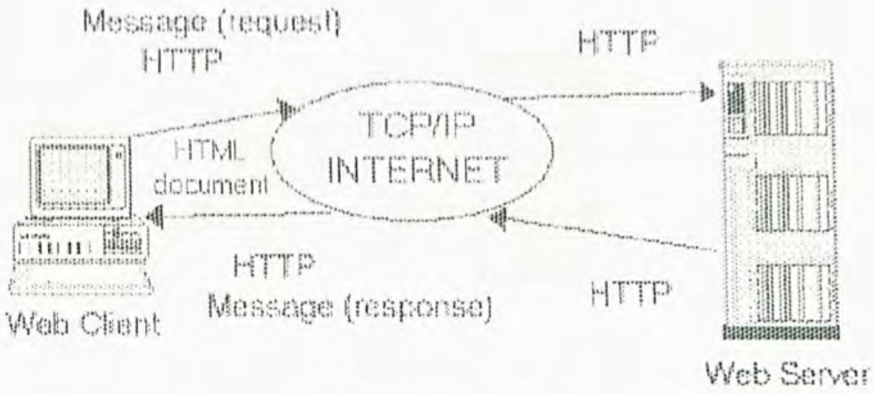


Figure 2.3 View of a Two-tier Architecture

As for the server side, processing had to be done there. These processes include computation of data and results, recalling data from a database, formatting data before recording and entering new data in the database. The server side sometimes needs to format the data output. These formatted interfaces are then send to the clients for presentation. Other than computation done in the server, database storage is also held in the same domain.

2.4.1.2 Three-tier Client/Server Architecture

Three-tier client/server was an evolution from the two-tier client/server architecture. The major addition to the architecture is a middle-tier layer and the detachment of the bottom layers. As the name goes, three-tier architecture consists of three layers, namely presentation layer, application logic layer and data manipulation layer. The corresponding host for this layers are, presentation layer on a client, and application logic and data manipulation layer on separate server. The separate server in this scenario does not mean a physically different server but a conceptual idea.

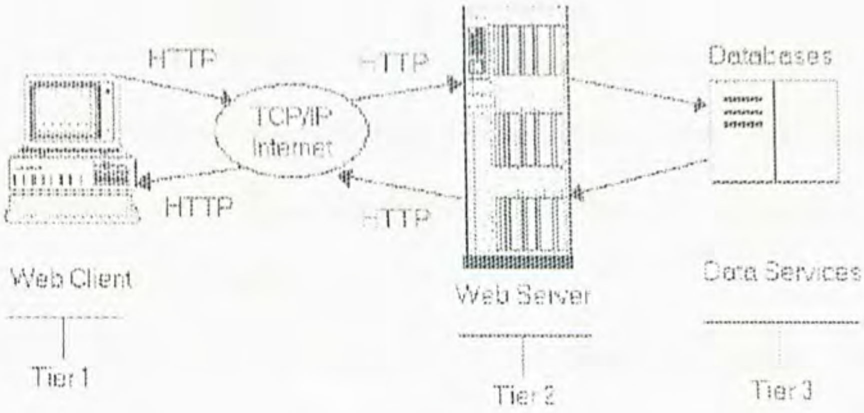


Figure 2.4 View of Three-tier Architecture

The presentation layer held by the client still plays the same role as it was in the two-tier architecture. However, the application logic layer now focuses solely on computation and processing of data in and out of the database. This provides a more powerful yet scaleable architecture. Resources on this layer become flexible on this layer to suit its needs. The data manipulation layer's duties are to store data, checking them for consistency and formatting them for storage.

2.4.2 Markup Language

2.4.2.1 HyperText Markup Language (HTML)

HTML is a set of source code that contains tags. These tags are noted by angle brackets with the ending tag having an extra slash indicating the end of marking. Texts encoded between the start and end tags are then publish as a webpage base on the command stated in the tag. The controlling body to standardize the tags is World Wide Web Consortium (W3C). Popular browsers like Microsoft's Internet Explorer and Netscape's Navigator are fully compliance with W3C tags but do support additional tags.

2.4.2.2 Dynamic HTML (DHTML)

Dynamic HTML (DHTML) brings a livelier page than HTML. Although HTML is the mother of all webpage language, looking at a static page does not serve the purpose of the fast pace modern world. The word dynamic shows that contents in the page can be dynamic rather than the static HTML.

The term dynamic, span through data, presentation, and publishing. Information that is published can be updated constantly without complicated involvement form web authors. The down-side of DHTML is that it only support newer version of web browsers.

2.4.2.3 Extensible Markup Language (XML)

XML (Extensible Markup Language) is a new markup language than is gaining huge popularity. However, the birth of XML does bring death to HTML. XML is just a common markup language that works hand in hand with HTML or other HTML dialects.

HTML describes how the data are published on the web. Where else XML describes what the data contains. The usage of XML encourages data sharing not limiting the sharing to web pages but also application. Application with the capability of getting information from XML web pages extracts the exact information that is required rather than filtering of the information that is published.

2.4.3 Web Application Programming Languages

2.4.3.1 Active Server Pages (ASP)

As the demands of the Internet increase especially the booming of electronic commerce, static pages turns dull. Microsoft came out with a solution to provide a dynamic page, namely ASP. ASP allows pages to change dynamically without intervention from users. The technology combines the capabilities of ActiveX components, which opens a wide range of function from database to printing. ASP's capability does not stop there on the server side but allows client side scripting like VBScript and JavaScript.

2.4.3.2 ASP.NET

ASP.NET came along year 2001, providing ease to program for Internet applications. The .NET framework stress on Common Language Infrastructure (CLI), which main goal was to allow user "execute on different operating systems and on different hardware platforms without having to recompile the high-level source code" (Grundgeiger, 2002). In short, this encourages developers of different platform or programming languages to program without the need to "relearn" a new language.

2.4.3.3 Java Server Pages (JSP)

Java solution to address dynamic web pages was JSP. JSP came from the Java 2 Enterprise Edition (J2EE). Hence, the programming style in JSP is similar to those implemented in Java and is fully object-oriented. The core of JSP is servlets where the server side processing is done before pages are published to clients. As for the client side, JavaScript is used to enhance the interface.

2.4.3.4 Applet

Applets are small application that is written in Java. These application has limited capabilities but sufficient enough to do small algorithms including simple animation. Applets can be bundle together with HTML files and brought over to the clients. This proven to save computing time as the applets are able to do simple computing without sending a users request back to the web server.

2.4.3.5 ActiveX

ActiveX is the name Microsoft has given to a set of "strategic" object-oriented programming technologies and tools. The main technology is the Component Object Model (COM). Used in a network with a directory and additional support, COM becomes the Distributed Component Object Model (DCOM). The main thing that users create when writing a program to run in the ActiveX environment is a component, a self-sufficient program that can be run anywhere in user's ActiveX network (currently a network consisting of Windows and Macintosh systems). This component is known as an ActiveX control. An ActiveX control is roughly equivalent to a Java applet.

2.4.4 Scripting Languages

2.4.4.1 Client-Side Scripting

Client-Side scripting is often used for validation, interactivity, enhancing a web page. These scripts runs on the client end to provide response without the need of sending and receiving information back from server. This can be advantage for validation of data before being send to the server. Currently famous client-side scripting are JavaScript and VBScript but JavaScript seems to have the edge because of cross browser support.

2.4.4.2 Server-Side Scripting

Server-Side scripts plays a different role not mainly because it resides on the server but the needs on a server is totally different. The need for database connectivity, processing power, and formatting requires the script to react differently. Processing could include responses like customization to the client browser, authentication, and security. Unlike the client-side scripts, server-side scripts are not sent over to the clients after it has been processed.

2.4.4.3 JavaScript

JavaScript is a fully object-oriented scripting language for web pages. The script works on the client-side of the system. Using JavaScript provides a better interfacing with the user as the web page replies to the user's actions but still able to communicate with the server.

2.4.4.4 VBScript

VBScript works similarly to JavaScript. VBScript are able to call ActiveX objects. Therefore, the internal working of VBScript is related to the more famous Visual Basic. To enjoy the capability of VBScript, web browser needs to be Microsoft Internet Explorer.

2.4.5 Web Server

2.4.5.1 Internet Information Server

Internet Information Server (IIS) is a BackOffice family product by Microsoft. The strong integration with Windows 2000 increases the stability of the system. IIS does not only include support for hosting Hypertext Transfer Protocol (HTTP) requests, but also File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP). The HTTP functionality is not on static pages, but include support for dynamic pages like DHTML and Active Server Pages. These livelier pages are connected together through the usage of Application Programming Interface (API) and ActiveX objects. Security on the IIS is base on the Windows 2000 technology. Which means publishing on the intranet does not pose a problem, but just a matter of click.

2.4.5.2 Apache

Apache came along from a UNIX platform environment. These platform include Linux, Sun Solaris and HP-UNIX. It popularity was gain through the open source license. Open source attracted programmers around the world to develop Apache for different operating systems. Today, Apache is used on platforms like BeOS, Windows 2000, Amiga, and many others. Being free, it usage increase throughout the whole world. Users who are not satisfied with it can contribute to the system by sending in their own modification. This tends to reduce the bugs that exist.

2.4.6 Web Browser

2.4.6.1 Microsoft Internet Explorer

Internet Explorer is a graphical web browser that is bundle together with Windows family. Its capability not on limits to accessing HTTP but it also accessing FTP and other ports. As the demand grows, Internet Explorer is able to merge with other Microsoft applications by calling in other application or vice versa. The functionality expand when plug-ins are added in.

2.4.6.2 Netscape Communicator

Netscape Communicator start off as Mosaic, the forefather of all graphical web browsers. The evolution of Netscape grew with the Internet. Netscape, as it was know at the start, started off as a simple web browser then grew to Netscape Communicator, that provides extra functions like a mail client and web page editor. Today, Netscape is wholly own by Internet Service Provider, America Online (AOL).

2.4.7 Web Application Development Tools

2.4.7.1 Visual InterDev

Visual Interdev belongs to the Visual Studio family, which is a complete development kit under Windows environment. As with the word "Visual", designing the web page becomes easy by dragging and dropping the desire component on to screen. This "what you see is what you get (WYSIWYG)" frees the programmer to do the hard coding for the interface. All the programmer needs is to think more on the back-end processing of the web pages. ActiveX controls and Java applets are able to be integrated on to the web pages through Visual InterDev. The help provided by Visual InterDev is complete with examples and syntax through Microsoft Developer Network (MSDN).

2.4.7.2 Macromedia Dreamweaver UltraDev4

Dreamweaver UltraDev4 is a complete web page design environment. Web page enhancements are done easily by a number of clicks. These enchantment include automatic inclusion of JavaScript, building Flash graphics, and much more. The compatibility format range from HTML, DHTML to dynamic pages like ASP, JSP and CFML. Although it comes without strong programming support for dynamic pages, it can be useful tool when it comes to enhancing the looks of the web page.

2.4.7.3 Paint Shop Pro 7.0

Paint Shop Pro is a powerful image editor that has filters, hue controls, and layering functions. It support multiple graphic picture format namely, Bitmap (BMP), (TIFF),JPG, CompuServe (GIF). Plug-ins that are Adobe Photoshop compatible can be install easily increase it functions. Filters here are able touch up picture to give a realistic feel or turns an image to 3 dimensional view.

2.4.8 Database

2.4.8.1 Database Connectivity

2.4.8.1.1 Open Database Connectivity (ODBC)

ODBC is the layer between the application logic layer and the database manipulation layer. It creates a conceptual layer which enables communication between the two layers without the application layer needing to know the complexity of the database layer (Embong, 2000).

ODBC was design by Microsoft as an application-programming interface(API), mainly in use of Microsoft products. ODBC supports Microsoft Access, dBase, DB2, MySQL and many others more.

2.4.8.1.2 Java Database Connectivity (JDBC)

JDBC play the same role as the ODBC. The different is the development environment that it is used. JDBC was created to support application written in Java. JDBC drivers for various database management system are easily available.

2.4.8.2 Relational Database Management System

2.4.8.2.1 Microsoft SQL Server 2000

Microsoft SQL Server came a long way back in 1980s. This long journey has enhanced its functionality and improvement in the ease of use. The building up of the application base on SQL statements furthermore increase it popularity. The usage as a web server database has been proven over the past years. The latest improvement was the capability to export XML files.

2.4.8.2.2 MySQL

MySQL is open source database systems that can be customized to suit users need. Although the application was develop in C++, and mainly used on Linux platform, now it has several versions running on other platforms. The language use in communicating with the database is SQL statement. MySQL is noted mainly for its speed, reliability and flexibility (whatis.com, 2002).

2.5 Summary

Literature review proves to be fruitful after all work had been done. Starting from reviewing existing systems, we know what current the market has and what is lack off. Following that, tools of development are review to give us an insight of what strength and weakness of each tool. This give us valuable information on what and how each tool perform and how it can be use to suit the project's needs.

Chapter 3 Methodology

3.1 Definition of Software Development Methodology

A methodology is build up by a set of software development models or a sequence of processes, and it combines with software development techniques with the idea of developing an application. A few techniques are functional oriented, prototyping and object-oriented. These techniques are used in realizing the development model like waterfall model, spiral model, and incremental and iterative model. However, methodologies are not static. They are dynamic as they are customized to suit the need of the software development.

The chosen methodology for developing the Front End User module for the Online Crossword Puzzle for Learning Synonyms and Antonyms is a combination of the incremental and iterative model and a waterfall model.

3.2 Iterative and Incremented Waterfall Model

(Software Development Process Modeling)

3.2.1 Overview

The Iterative and Incremented Waterfall Model comprises of a set of stages of the waterfall model that describes the steps to follow during development of the Front End User model. In within all the stages, the iterative method is being applied for features enhancements that will result in an increment. Unified Modeling Language is used to prepare all the blueprints of the system development.

Like the Unified Process, this method is also use-case drive, architecture centric and is deployed iteratively and incrementally. Use case driven in the sense that the functionalities of the system are being captured in use cases with the actors. The software architecture is being described as different views of the system being built. Iterations of each mini project will result in an increment where the product grows.

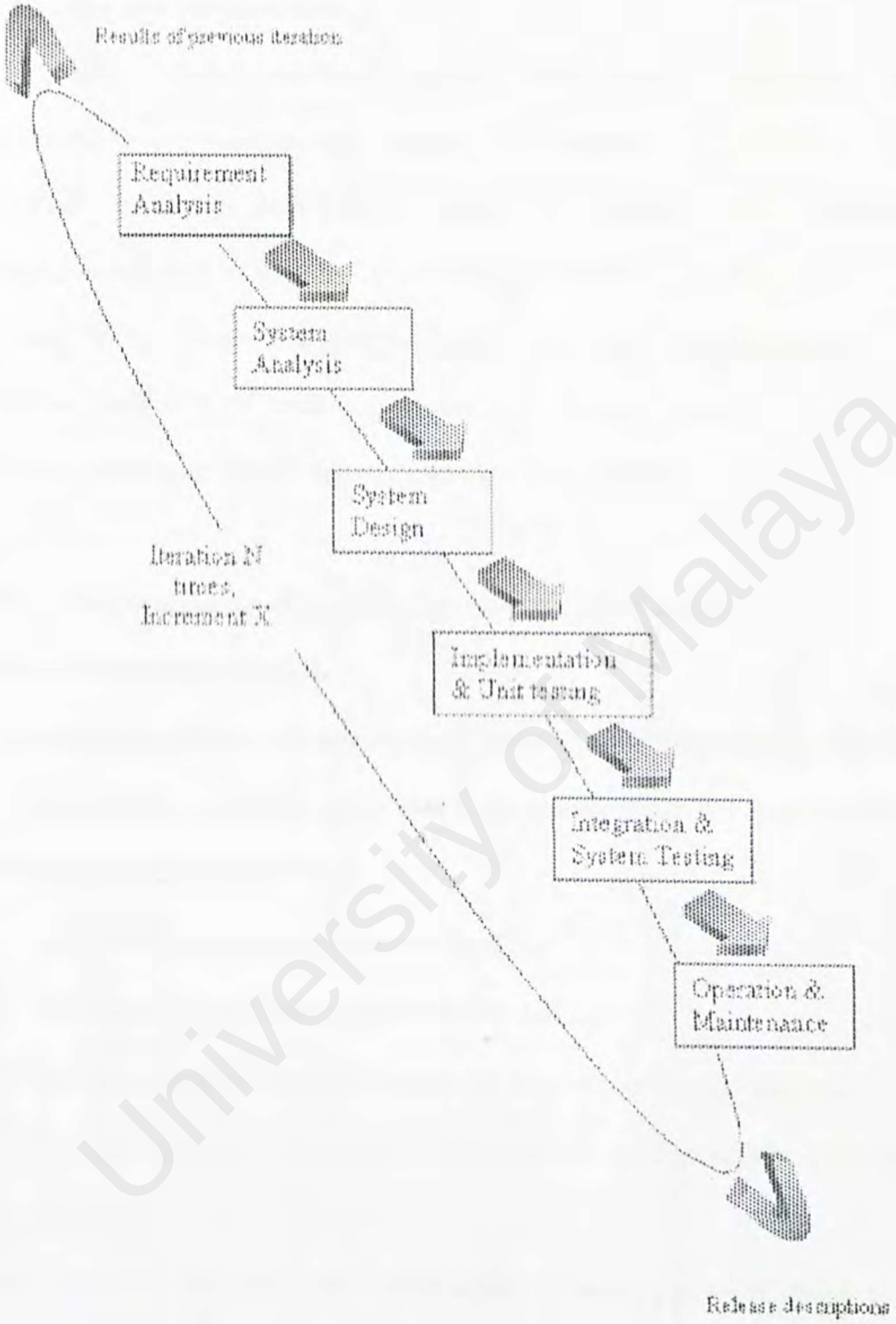

Results of previous iteration
Requirement Analysis
System Analysis
System Design
Iteration N times, Increment X
Implementation & Unit testing
Integration & System Testing
Operation & Maintenance
Release descriptions

Figure 3.1 Iterative and Incremented Waterfall Model

3.2.2 Iterative and Incremented Life Cycle

The System Administration module uses the Iterative and Incremented life cycle. In certain developing stage, the main system can be divided into subsystems to be built separately as iteration. Each iteration results in an increment. When a subsystem of a stage does not meet its goal, the previous stages must be revisit with a new approach. A release of the system that contains added and/or improved functionality over the previous iteration is the result with an increment. Iterations might focus differently to a greater or lesser extend depending on the activities of each stage.

3.2.3 Stages of the Iterative and Incremented Waterfall model

3.2.3.1 Requirement Analysis

Requirement analysis is the primary stage where the feasibility studies, objectives and scopes definition as well as project plan is established. Tasks that are performed during requirement analysis are as follow:

- Delimiting the scope and objectives of the system
- Outlining crucial subsystems that is risky and difficult
- Identifying and prioritizing the risks that can affect the building of the system
- Planning the project schedule and reviewing all possible software tools and skills required.

The output of this stage is the specifications document that defines what the system does and not how the system works.

3.2.3.1 System Analysis

The primary goal of this stage is to analyze the specifications document for completeness so as to guide the development of the system into the design stage. In this stage all the functional and non-functional requirements are identified. The tasks involved in analyzing the System Administration module are as below:

- Capturing the functional requirements and non-functional requirements of the system
- Identifying candidate solutions as the target for the system design.
- Choosing the necessary software development tools for the system

In this stage of the development, a use case model is used to capture all functional requirements as critical use cases. The analysis model is another important model that involves in this stage to help in refining thé requirements.

3.2.3.3 System Design

The purpose of the design stage is to transform the requirements statements from the analysis stage into design specifications for the stage. The system design stage in the System Administration module outline the architecture of the system as well as the interface design. The most likely tasks during this stage are:

- Defining what a user must accomplish from the system
- Determining the order a user most likely use to accomplish the tasks

The documentation of this stage consists of the design model, which serves as the blueprint for the implementation model.

3.2.3.4 Implementation and Unit Testing

In the implementation, the System Administration module applications are being coded using selected languages and application tools defined in the analysis and design stage. Details of the operations and web interfaces are implemented in this stage. The integration of new code with the existing one from the previous iterations is gradually being implemented during construction. Few of the major tasks involved are:

- Implementing the design classes as file components like source code
- Integrating the new code and code from previous iterations
- Verifying each unit with its specification
- Performing unit tests of all the components

The implementation stage is the focus of the iteration life cycle where modification and increment of the system is more likely to be rotated. Following this, models are divided into subsystem to make future integration easy to handle. The implementation model that denotes the actual implementation of the system in terms of components and subsystems is used.

3.2.3.5 Integration and System Testing

During this stage, the individual subsystem programs are integrated and tested as a complete system to ensure that the system met its requirements. The main task in this stage is to create test cases that specify what to test and the procedure of how to test.

3.2.3.6 Operation and Maintenance

In the whole software development life cycle, the operation and maintenance stage consumed the longest phase. The system is deployed and put into practical use. Maintenance involves correcting errors, which were not discovered in earlier stages of the life cycle; improving the implementation of the system and enhancing the services as new requirements are discovered. Proper documentation can help to reduce the amount of time spend on training.

3.2.4 Why use Iterative and Incremented Waterfall model?

- In a complex stage, a big module can be divided into sub modules where each iteration will results an incremented of the module. This process can be repeated until the whole module is complete.
- Additional functionalities can be added easily into system for future enhancement.
- This model allows early verification on finished sub module for error so that problems can be solved earlier.
- A Web-based system is always updated so iteration can be done from time to time.

3.3 Unified Modeling Language (UML)

"The Unified Modeling Language (UML) is a standard modeling language for software –a language for visualizing, specifying, constructing and documenting the artifacts of a software-intensive system." [Jacobson et al, 1999] The UML is a graphical notation with sets of rules and semantics that enables developers to describe the static and dynamic aspects of the system as a blueprint. The UML can be generally divided into three categories: things, relationships, and diagrams. Below are the basic structures of each category of the language. [Jacobson et al, 1999]

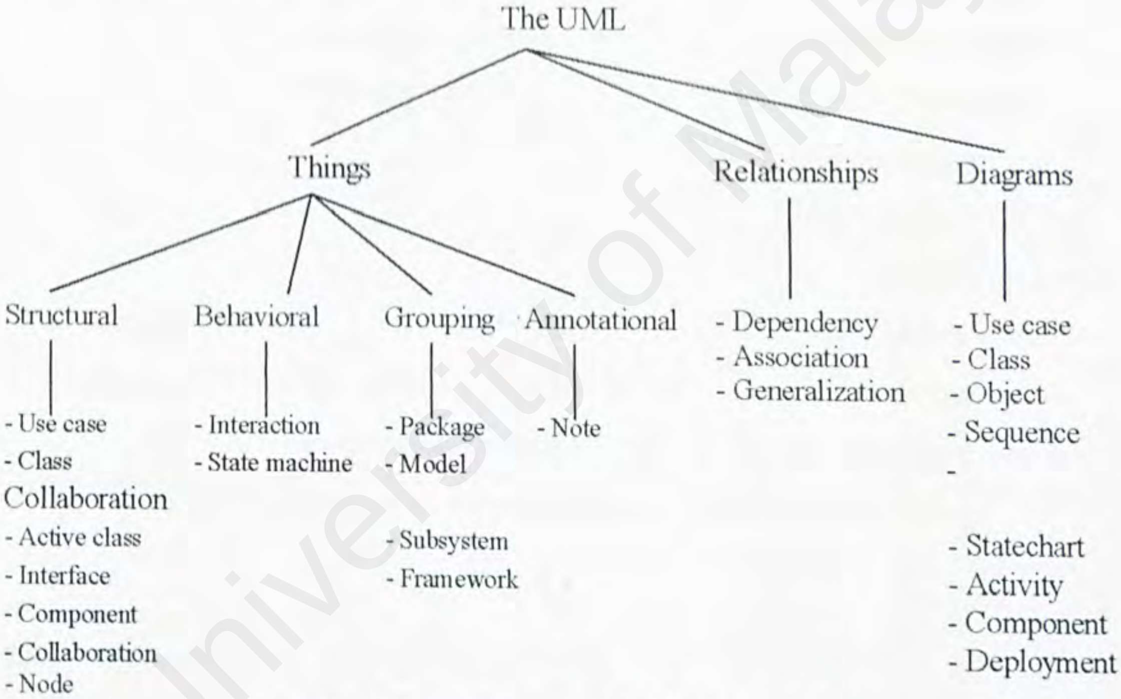


Figure 3.2 Tree diagram of the UML vocabulary

The summary of the major phases of system life cycle and its deliverables models are described in the following table:

Stage	**Purpose**	**Deliverables**
Requirements Capturing	Capturing functional and non-functional requirements	Use case model: • use case diagram
System Analysis	Analyzing the collected requirements and identifying analysis model	Analysis model: • class diagram • collaboration diagram
System Design	To design a structure for the system that includes the various subsystems, and their interrelationships.	Design model: • class diagram • sequence diagram • activity diagram • state diagram Deployment model: • deployment diagram
Stage	Purpose	Deliverables
Implementation	To construct the increments and integrations of iterations	Implementation model: • component diagram

Table 3.1 Summary of the major phases and its deliverable models

3.4 Summary

Each development model needs to be study carefully to determine whether it is suitable or not. After intensive studies, and reviews, the development model that is chosen is a hybrid of iterative and incremental model and the waterfall model. The combination of both this model provides a good solution in development an evolving system like the web page. To complete the methodology, UML notation is used to record the procedures.

Chapter 4 System Analysis

In system analysis, the main aim of the system is noted in general. The steps taken for this purpose is requirement specification followed by the stated what tools are used in developing the system. In requirements specification, functional requirements and non-functional requirement are stated clearly and concise. This is to prevent any misinterpretation. In the following part, development tools justified on why there are chosen.

4.1 Requirements Specifications

The task of capturing the functional requirements is made easier with the usage of Unified Modeling Language (UML) as UML is use-case driven. The use case model captures all the requirements as use cases and serves as the foundation for all other development work. Functional Requirements of the Online Crossword Puzzle for Learning Synonyms and Antonyms – Front End User Module are basically based on the system administrator module and user module.

4.1.1 Functional Requirements

The following are the Use Case Diagrams for the Main Package and the Play Package. After that, elaboration on each use case is stated.

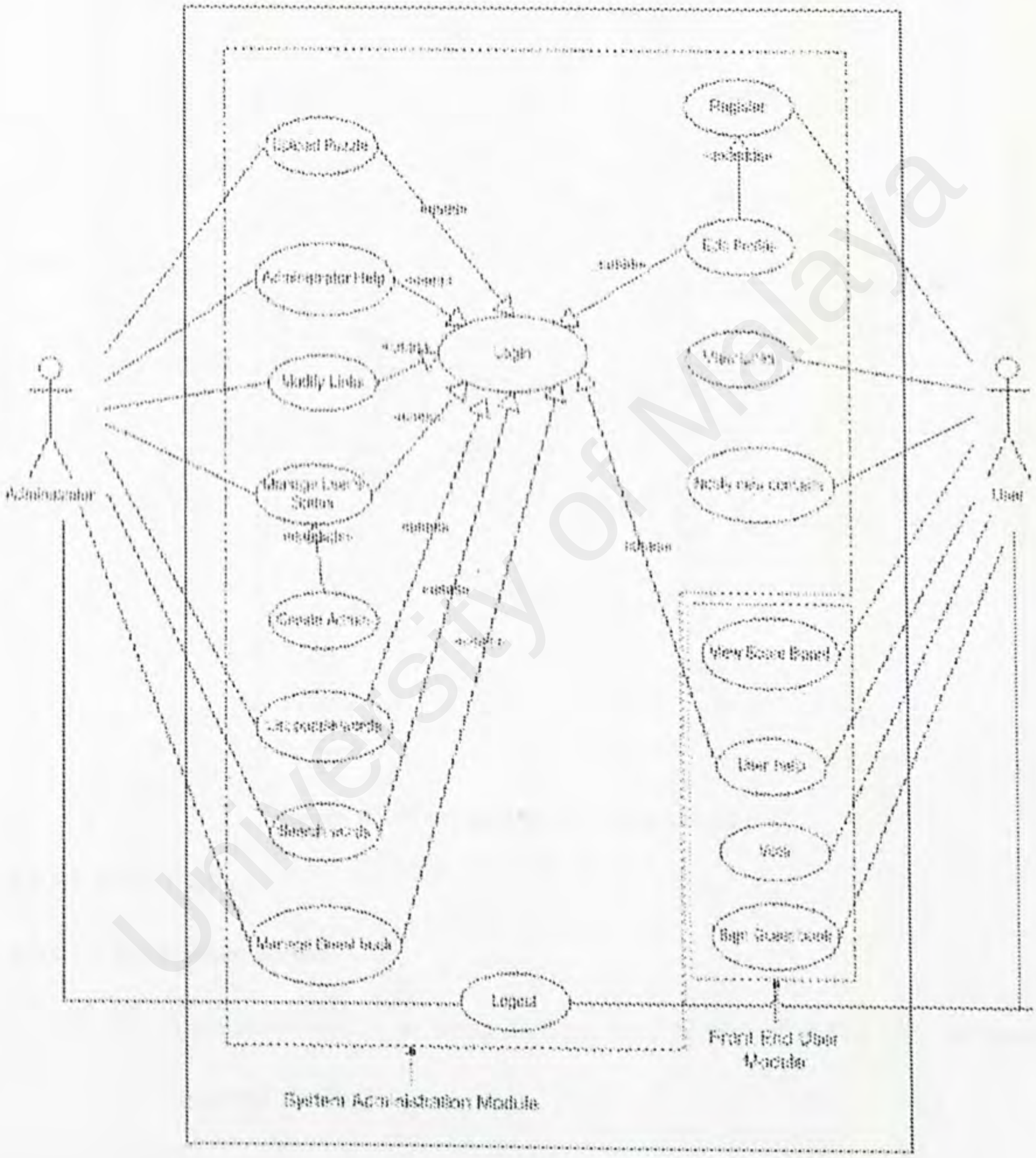


Figure 4.1 Main Package Use Case Diagram

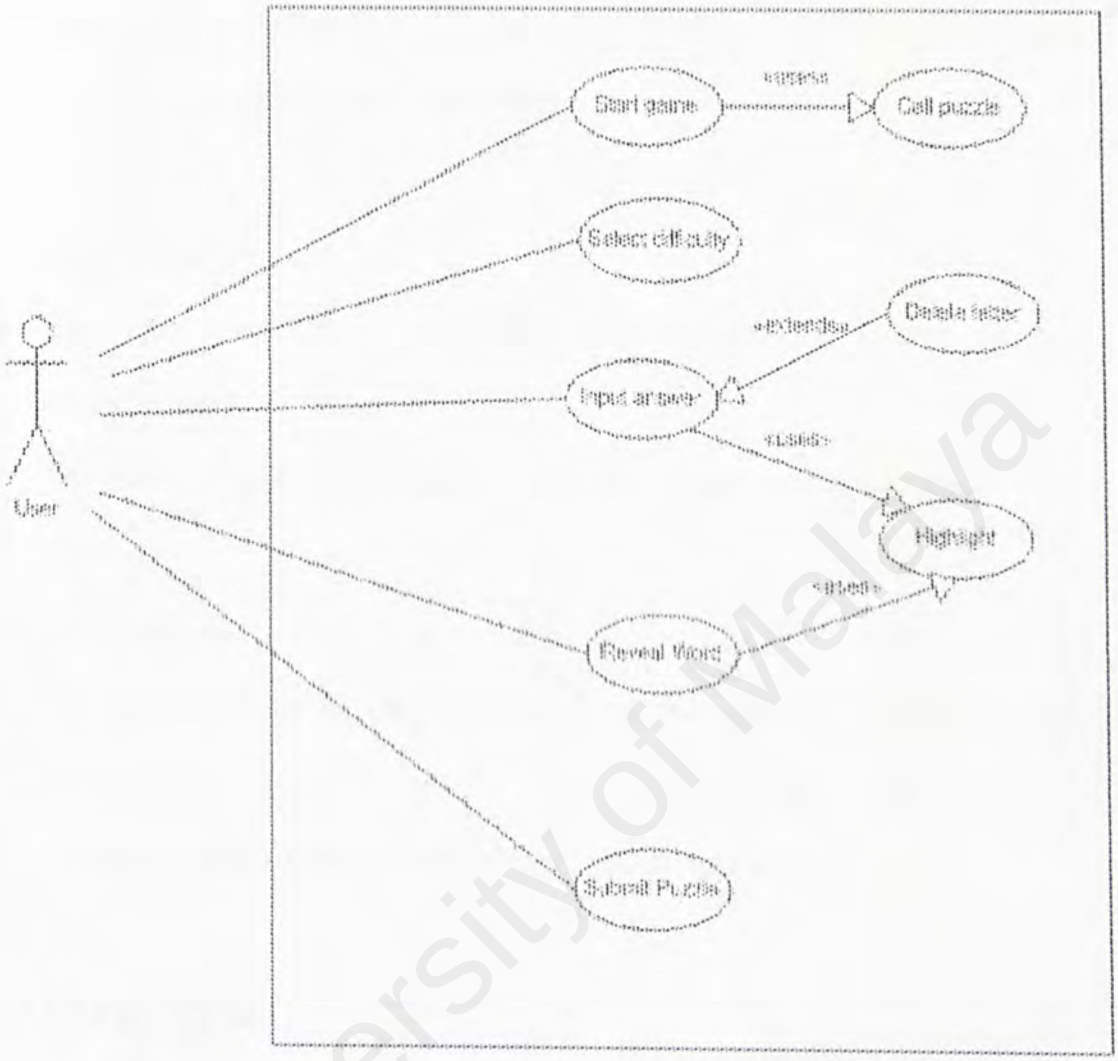


Figure 4.2 Play Package Use Case Diagram

4.1.1.1 Main Package

4.1.1.1.1 View Score Board

- The system should provide a scoreboard of the top few players so that user can keep track of their current standing.
- The system should allow user to view the top players scoreboard via hyperlink.
- The system should update the scoreboard to the latest detail.

4.1.1.1.2 User Help

- The system should have a user guide manual on how to play the crossword puzzle as well as rules and regulation of the puzzle.

4.1.1.1.3 Guest book

- The system should have a guest book for users to put their comments on the crossword puzzles or site.
- The system should allow the users to view all the comments on the guest book.

4.1.1.1.4 Vote Box

- The system should provide a voting box for the users to vote for the website performance.
- The system should display the results to the users after the voting process.

4.1.1.2 Play Package

4.1.1.2.1 Start Game

- The system should trigger the timer once the user invokes the function to start solving the crossword puzzle.
- The system should request for login before a user can start solving the puzzle if the user have not login.

4.1.1.2.2 Input Answer

- User can key answers into the grid without having to follow the chronological order of the grid number.
- User can key in a letter into the grid that is highlighted and subsequently to the next grid unless the user highlight another grid box.

4.1.1.2.3 Modify Answer

- The system should allow the user to delete or change letters in the grid as long as it is in the allowable time.
- The system should enable the user to clear the whole puzzle without resetting the timer.
- The system should not allow user to change the answer once he click on Submit or after the given time is over.

4.1.1.2.4 Choose level of difficulty

- The system should allow the user to the level of difficulty of the puzzle that they wanted to play before the game starts.

4.1.1.2.5 Reveal word

- The system should reveal only the right answers on the grid and show the wrong answers in a different colour.
- Users are allowed to scroll the clue box to review the right answers and its companion clue.

4.1.1.2.6 Submit Puzzle

- Once the timer has run out of time, the system should be able to detect and submit the score of the user into the database and the scoreboard.
- User can also submit the solved puzzle provided it is completed before the timer run out of time.

4.1.2 Non-functional Requirements

Non-functional specifications are the constraints under which a system must operate and the standards which must be met by the delivered system [Sommerwille, 1995]. The new Online Crossword Puzzle for Learning Synonyms and Antonyms-Front User Module must includes certain web application qualities like easy to use interfaces and others non-functional requirements like reliability, flexibility, efficiency and modularity.

4.1.2.1 Easy to Use

The system should provide an easy to use interface that is understandable by users who are not familiar with computer. All the features in the play engine should be clear and meaningful to help users familiarized with the system.

4.1.2.2 Reliability

The system should make the correct respond and provide the appropriate output when requested by the users. The system should give a consistent and accurate output for every process under any environment.

4.1.2.3 Flexibility

The system should be flexible to enable upgrades and modification to be done easily for features enhancement. The existing features should be able to integrate with the newly added feature easily.

The system must ensure efficiencies in system execution and data storage. The loading time of the puzzle should no longer than 1 minute. Words and/or letters revealing should be displayed almost instantly after the function execution.

4.1.2.4 Modularity

The system should be developed in parts and sections so that functions of objects could be distinct from one another. This is important to ensure that testing and maintenance can be done easily. Modularity of the system should be applied from the very beginning so that it will lead to easy modification in the future.

4.2 Web Technologies and Development Tools Specification

4.2.1 Web Server

Internet Information Server 5.0 with .NET framework was chosen to play the role of a web server. The point was it has improved much since the earlier version. Integration with the .NET framework provides faster responds to the clients. Security feature that are applied on Internet Information Server is similar to Windows 2000. This gives assurance of the security provided.

Apache was not chosen because Apache required a hefty amount of configurations. Secondly, Apache does not support the .NET framework which it the key point in providing quick respond to client with ASP.

4.2.2 Database System Management

Microsoft SQL Server 2000 was selected because of the quick processing speed of each request. The SQL is the fastest among all in a benchmark test done by Transaction Processing Performance Council (www.tpc.org). This speed was achieved with Windows 2000. Clearly it proves to provide the need of The Online Crossword Puzzle for Learning Synonyms and Antonyms.

MySQL was not selected because it was hard getting the configuration done. The setting up of the server was quite tedious on a Windows 2000 platform. Microsoft Access was also not picked because of the limitation in concurrent access.

4.2.3 Database access technology

ODBC was chosen as the database access technology because of the strong support and usage in Visual Studio development. Microsoft encourages developer of Visual Studio to use ODBC as a connection to database because of the availability of the drivers. JDBC on the other hand, does not provide a concrete link between ASP.NET and Microsoft SQL 2000.

4.2.4 Programming Language

The web programming language that was chosen is ASP.NET. Not because it is the talk of the town, but it proves its capabilities through a series of testing by Nile @Bench. A few success in using the .NET framework is also publish on the Microsoft website. The .NET framework allows users to call function and procedure from other languages easily.

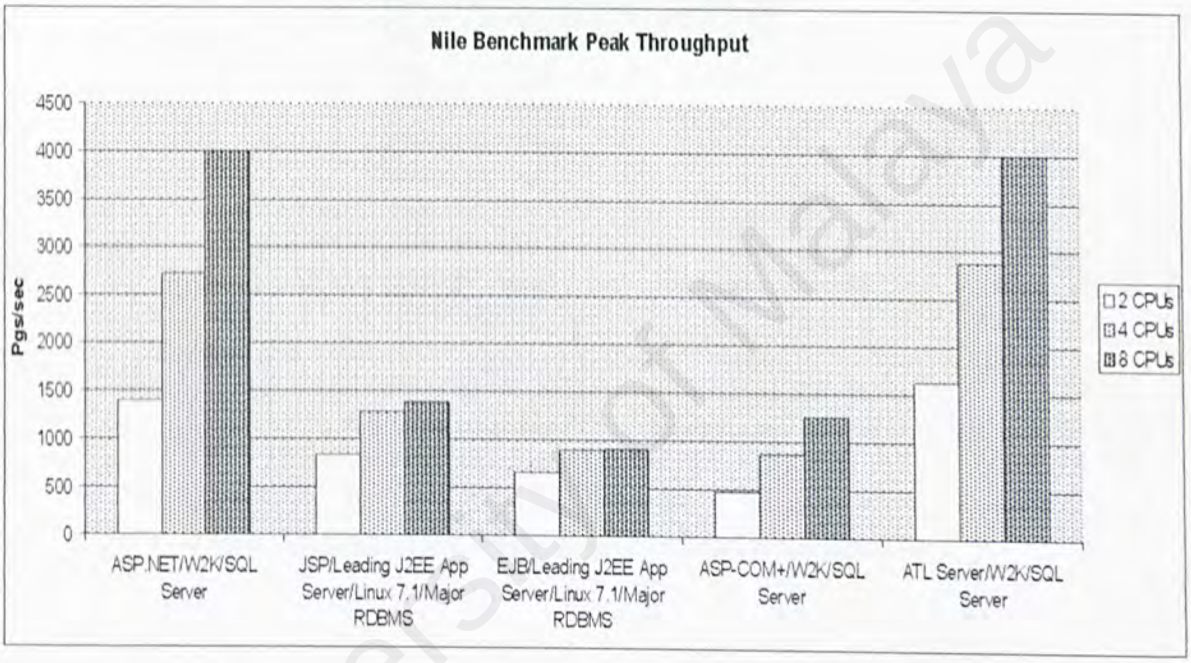


Figure 4.3 Bar chart showing the benchmark of a few programming languages

4.2.5 Visual Studio .NET

Visual Studio.NET is the recommended developer environment for .NET framework. It is easy to use and provides a great amount of help, hints, and pointer through the Microsoft Developer Network (MSDN). Visualizing tools help developers to design better and easier. Developer are allow to program in any text-base editor, but when it comes to debugging, the problem arise.

4.2.6 Web browser

ActiveX components require Internet Explorer. This is not the only reason Internet Explorer is chosen but ASP.NET too requires Internet Explorer. Benchmark testing by tecChannel shows Internet Explorer is stronger by two-fold compare to other web browsers.

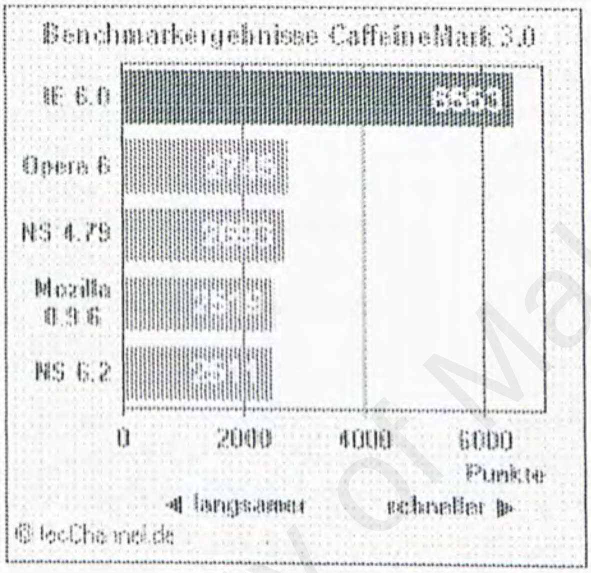


Figure 4.4 Benchmark results on web browser by tecChannel

4.3 Hardware and Software requirements

The table shows the summary of hardware and software requirements that have been considered for this project.

	Server Requirements	**Client Requirements**
Hardware Requirements	• Pentium or AMD with 500MHz and above computer • At least 64 MB RAM • Minimum 1 GB Hard Disk (depend on number of users) • Network connection with recommended bandwidth at 10 Mbps or more • Standard input and output devices	• Pentium or AMD with 133Mhz and above computer • At least 16MB RAM and above • Standard input and output devices
Software Requirements	• Microsoft Windows 2000 Server • Microsoft Internet Information Server with .NET Framework • Microsoft SQL Server 2000	• Any operating with graphic user interface • Microsoft Internet Explorer 4.0 and above

Table 4.1 Summaries of Hardware and Software Requirements

4.4 Summary

In this chapter, all the requirements for the system are captured and map out in use case diagrams. These requirements will provide a strong foundation to the following chapter. Software are justified not to limit the requirements but to know the limitation that will pose by it.

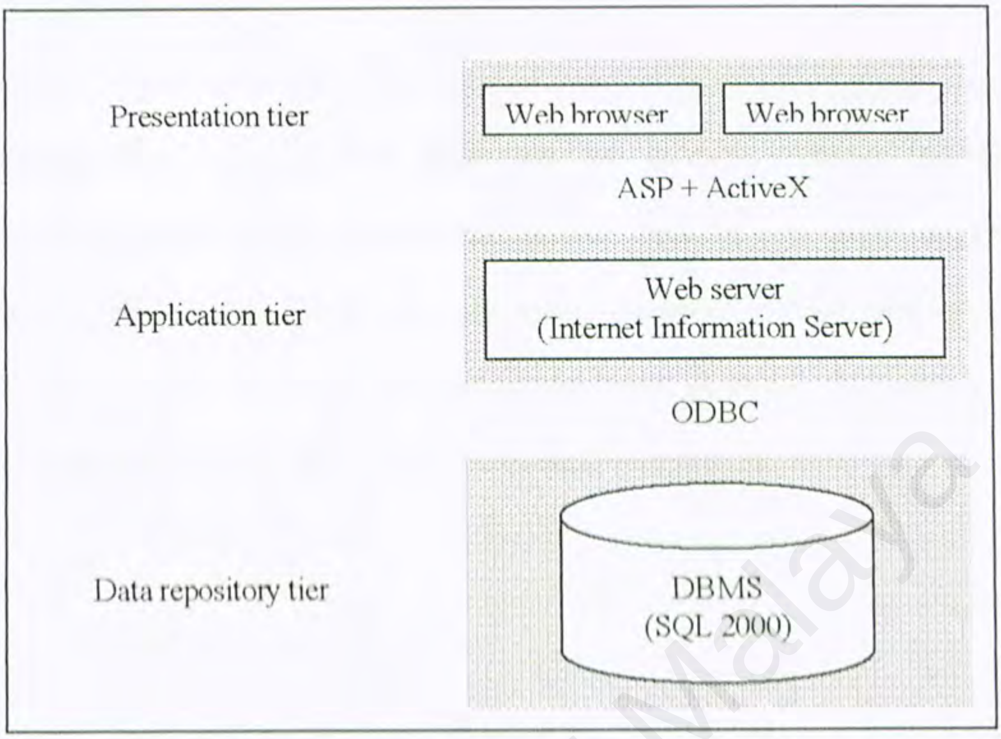


Figure 5.1 System Architecture of Front End User module

All the web page contents and the ActiveX object comes from the middle tier. In the application logic layer, a Web server resides in it. The Web server does all the computations and data retrieval from the database. After all processing is done, the results are formatted and send to the clients.

Before the final layer, a database connecter bridges the two layers. This "bridge" is called the ODBC. It provides independent between what happened in both layers. It helps to program much easier. This brings us to the final layer where the Database Management System (DBMS) resides. The DBMS engine is Microsoft SQL Server 2000. It responsibility is to feed the application layer with data which is later transform into information for the clients.

5.2 Deployment Model

Deployment diagrams show the configuration of run-time processing elements and the software components, processes, and object that live in them. Software component instances the represent run-time manifestations of code units. In most cases, component diagrams are used in conjunction with deployment diagrams to show how physical modules of code are distributed on various hardware platforms. In many cases, component and deployment diagrams can be combined (Paul and Mark, 1997).

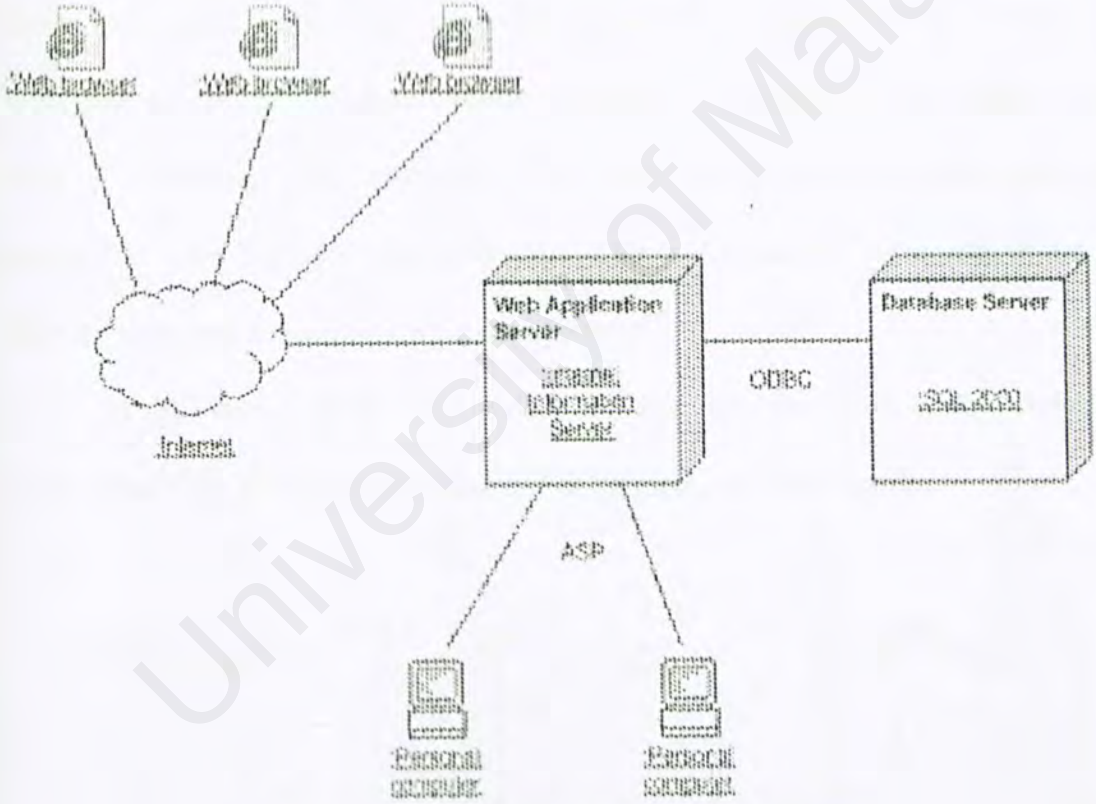


Figure 5.2 Deployment diagram of the system

5.3 Process Design

5.3.1 Class diagram

To design a system, the basic needs to come from the static design. As for UML notation, static entities are map on to a class diagram. These diagram shows the basic relationship between classes, be it associated, generalize or aggregated. Further explanations are added by using stereotypes, tag values and constraints to increase the understanding of the class diagrams (Scott, 2001).

As developing a web application needs extra notation in UML, the server page, client page and HTML forms requires supplementary stereotypes. The HTML form stereotype consist only attributes. Theses attributes are available to client pages that the form is contained. The «submits» association stereotype defines the relationship between a client page and the server page. This is because the processing of the data from the form page is done over at the server page. (Conallen, 1998)

In the context of the Front End User, only one class is necessary. Below are some of the UML extensions class diagrams of the module by Jim Conallen.

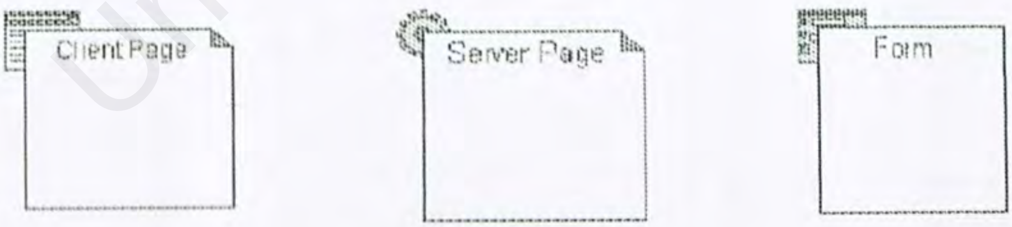


Figure 5.3 UML extensions used

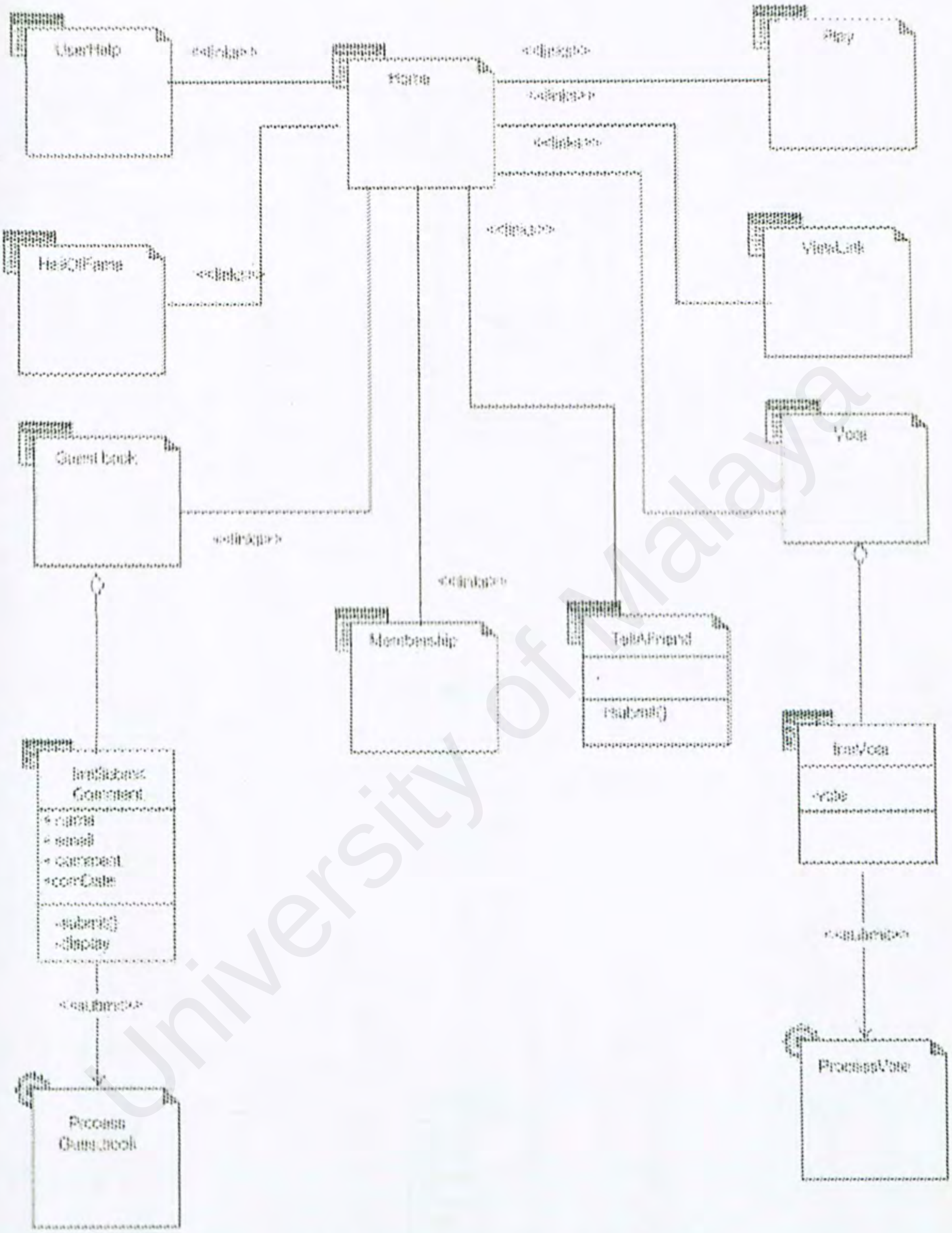


Figure 5.4 User Home Page Class Diagram

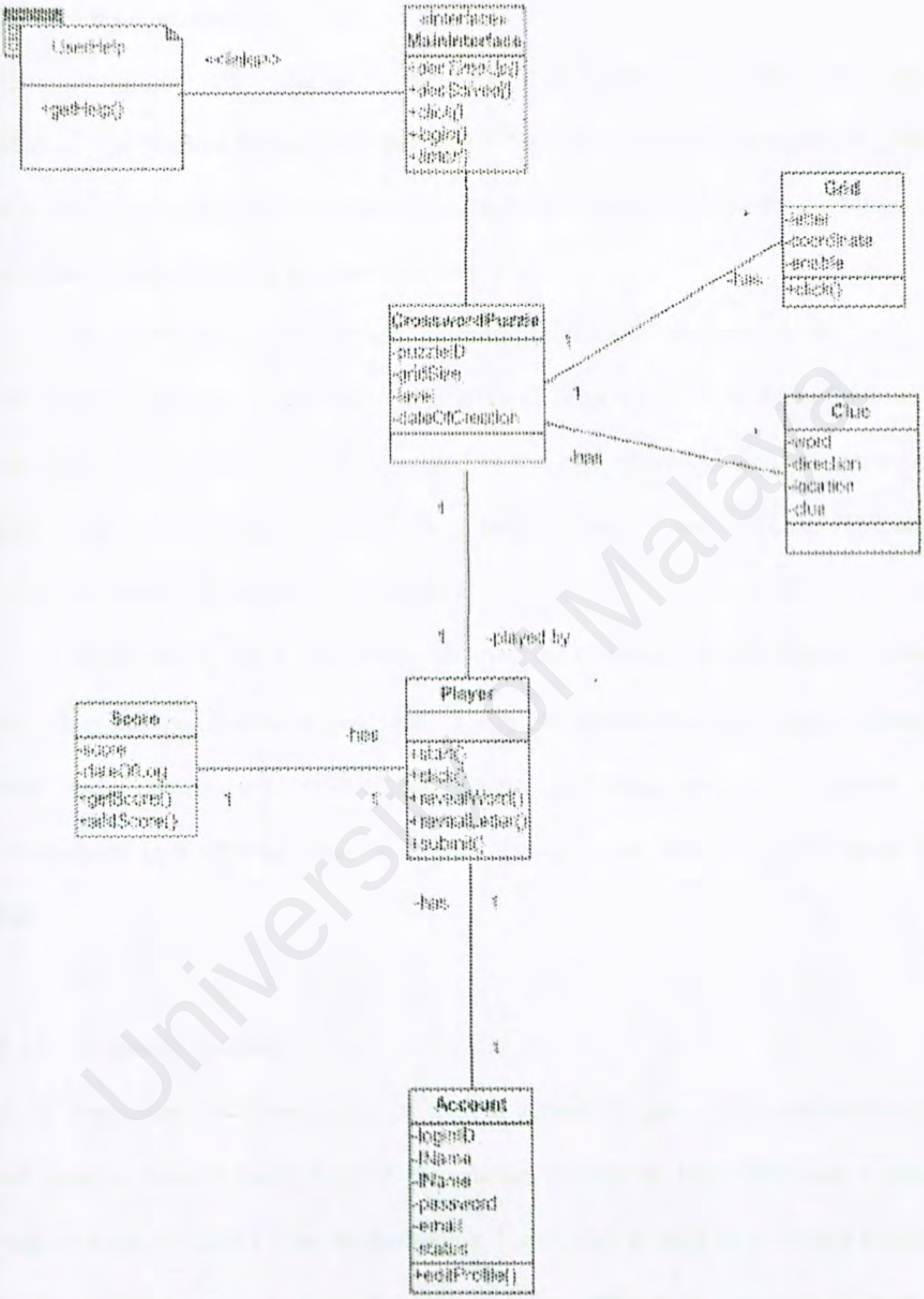


Figure 5.5 Play class diagram

5.3.1.1 Play class diagram

The core of play class diagram is the Crossword Puzzle. This puzzle has a main interface that contains login, timer, and submit button. In addition to the interface, there is a link where users are able to get help. Inside the link page, a friendly guidelines are provided to the users how to get around the homepage.

A Crossword Puzzle consists of many grids. The attributes in the grid are coordinate, letter, and enable. The enable attribute plays a role in disabling the blacked out grids. The Crossword Puzzle is related to the clue where the word, direction, and location are stored. A general score sheet is assign to each Crossword Puzzle. This score sheet is update once every puzzle is submitted.

Every player has a few commands over the Crossword Puzzle and each player can have only one puzzle at one time. These commands which are puzzle related is start, click, reveal word or letter and submit. Following that, user's accounts are encapsulated in a different class of its own. Personal particulars are stored inside this class.

5.3.2 Sequence Diagram

A use cases states the analysis of a problem in general. To get a detail analysis from the use cases, a more in depth form of presentation is required. But, visualizing a smooth flow of event on papers alone is challenging. Hence, this is done by providing scripts or an interaction diagram to show the time or event ordering of messages as they are evaluated.

In sequence diagrams, detail analysis of the use case is map out on a diagram. The diagram illustrates the lifeline and the messages pass between users, objects, entities, and controls. Lifeline can be created and destroyed as the flow of sequences goes in a particular scenario. The following are the sequence diagrams following the analysis from the use cases.

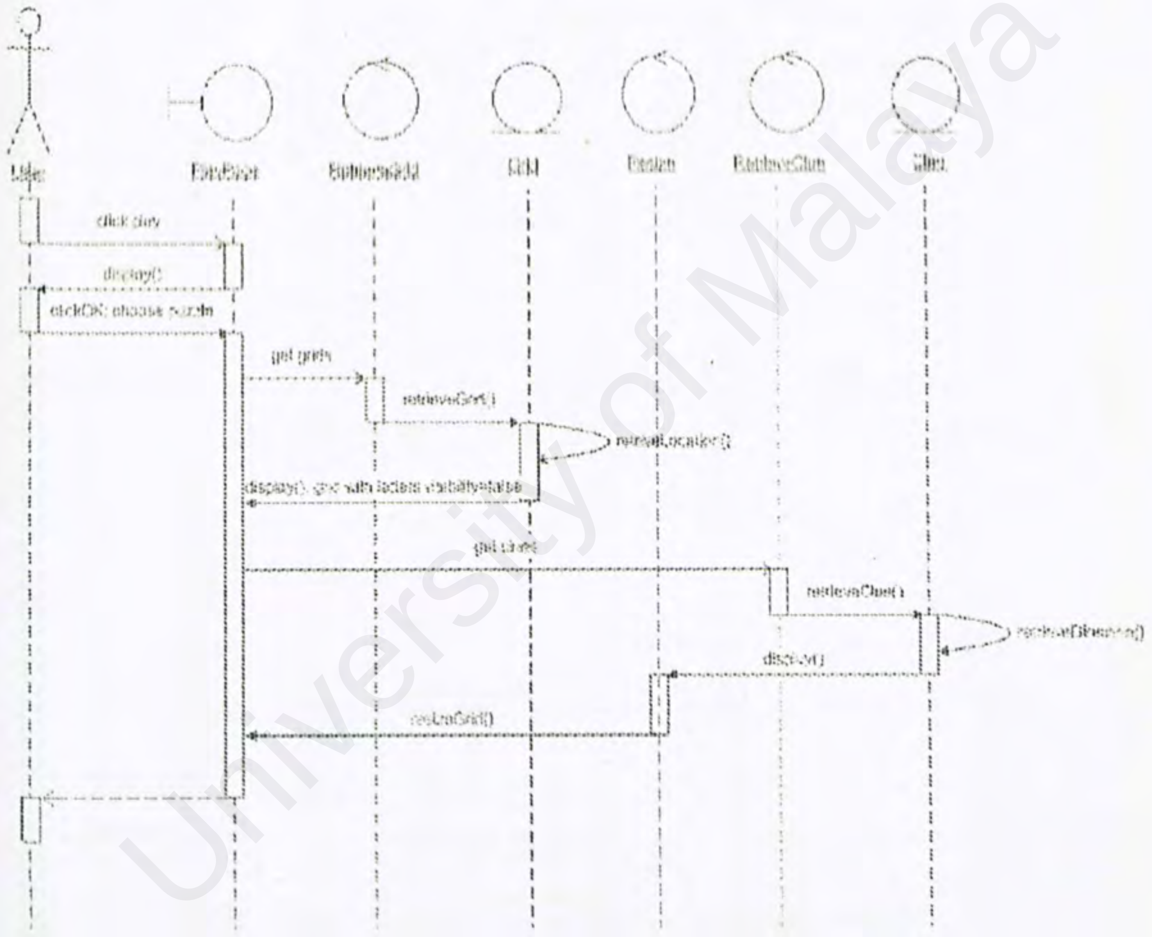


Figure 5.6 Call Puzzle Sequence Diagram

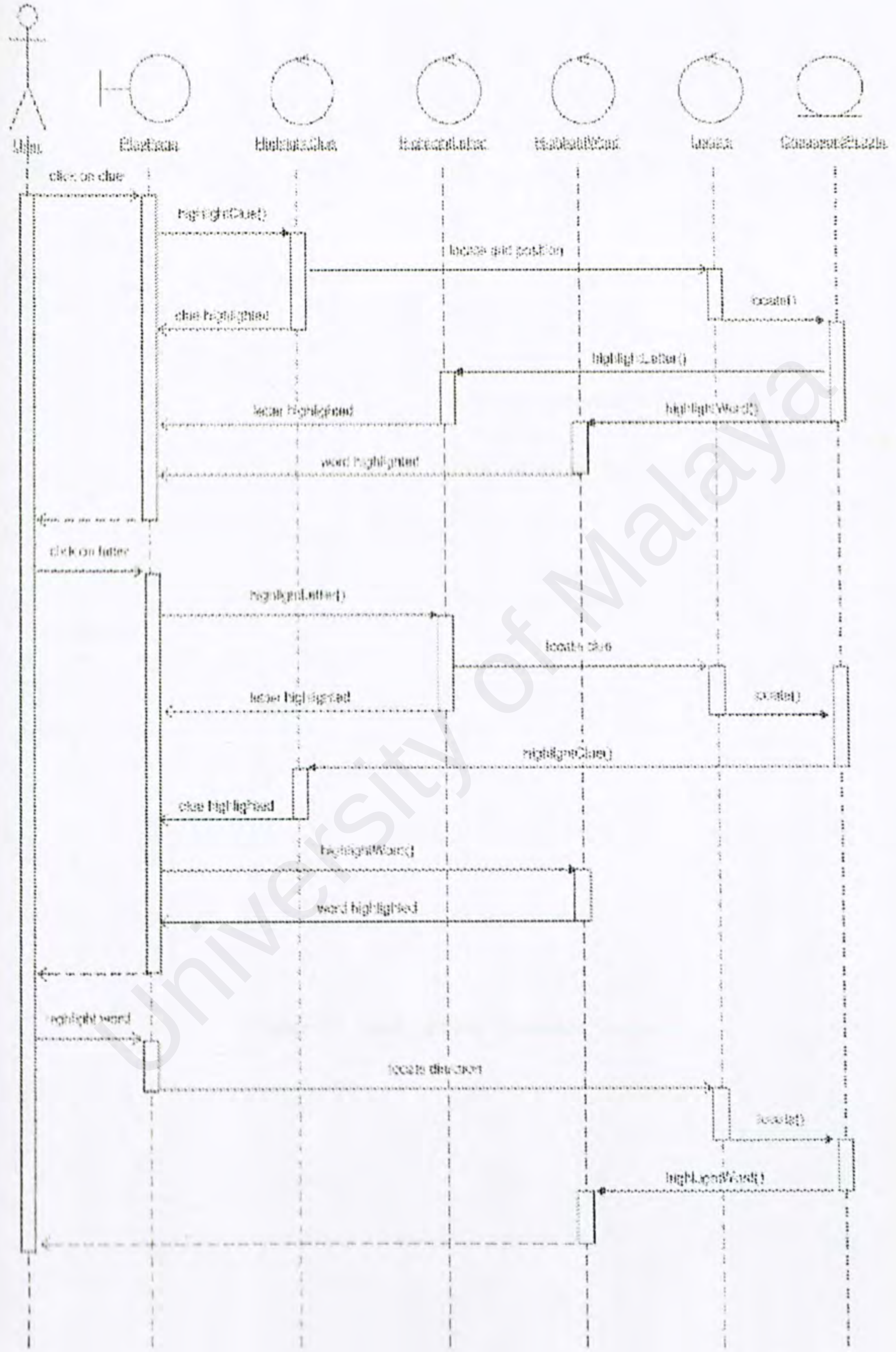


Figure 5.7 Highlight Sequence Diagram

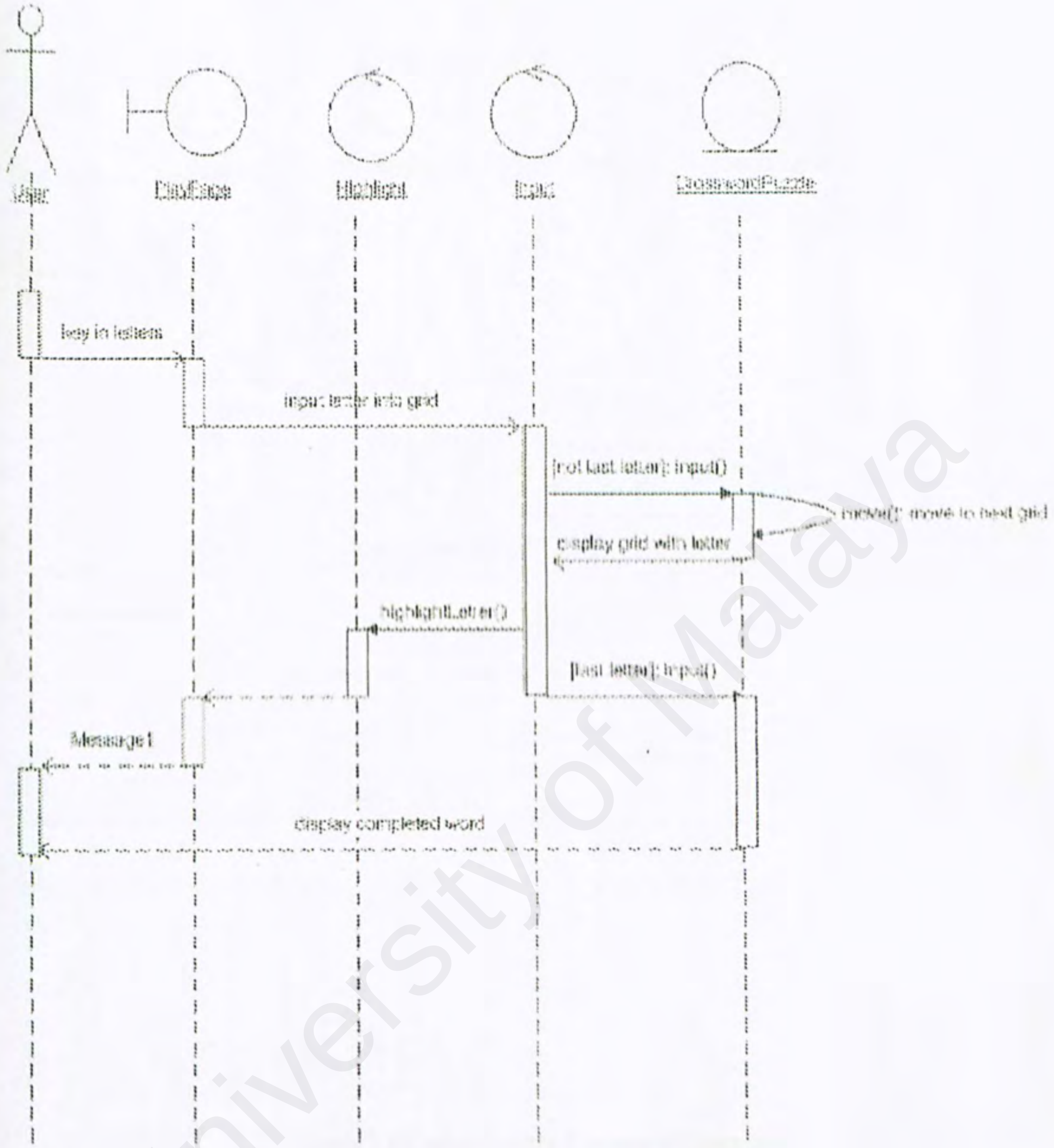


Figure 5.8 Input Answer Sequence Diagram

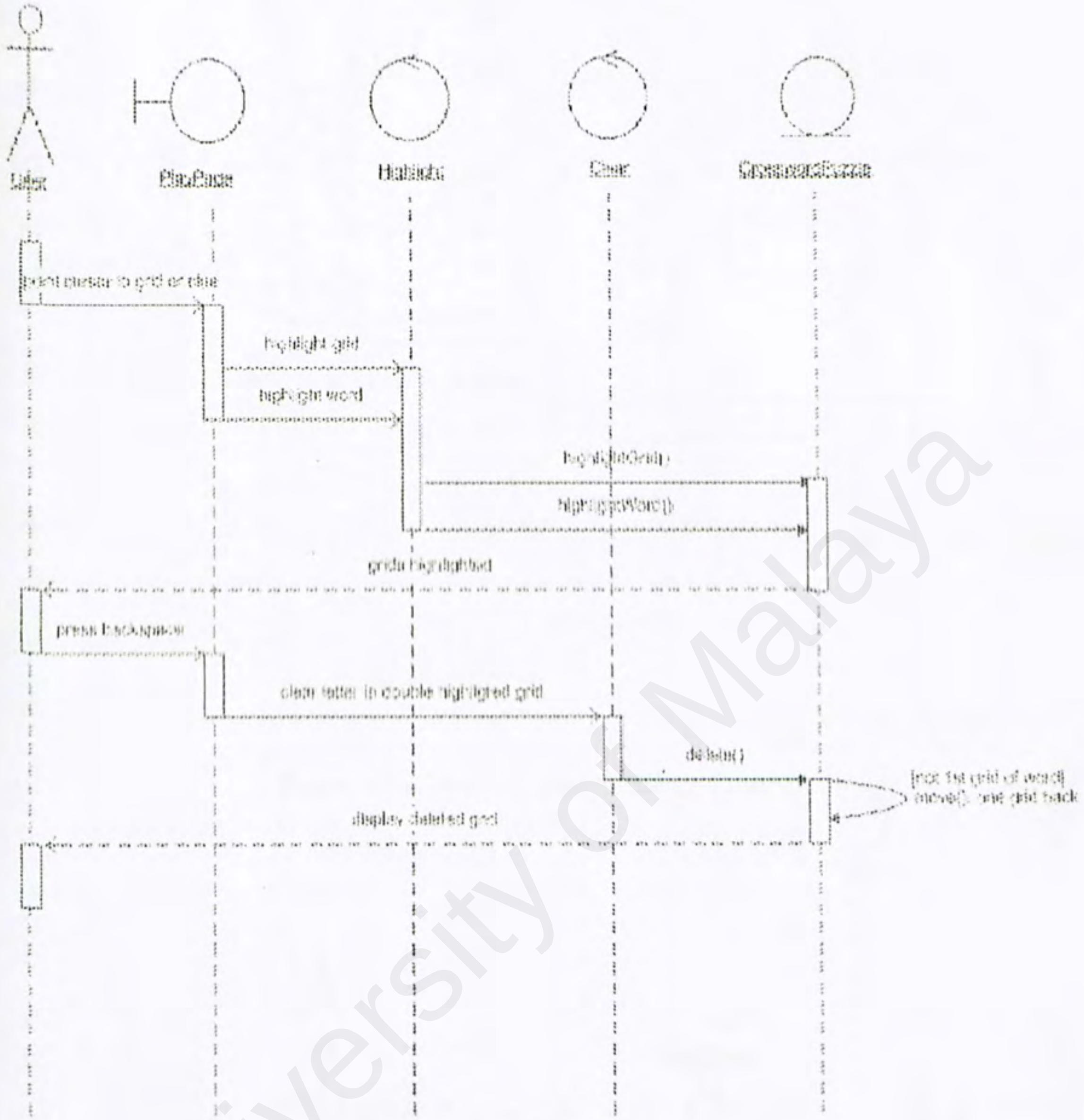


Figure 5.9 Delete Letter Sequence Diagram

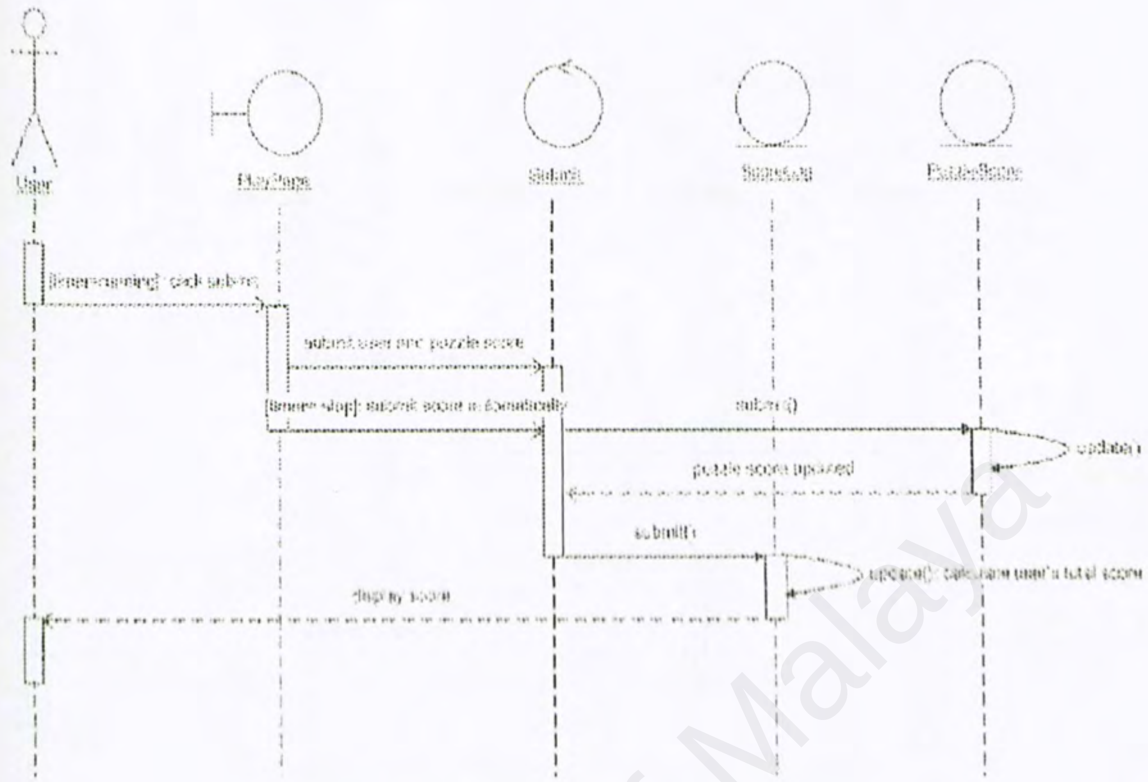

User
PlayPage
Submit
PuzzleScore
[timer=running]: click submit
submit user and puzzle score
[timer= stop]: submit score automatically
submit()
update()
puzzle score updated
submit()
update(): calculate user's total score
display score

Figure 5.10 Submit Puzzle Sequence Diagram

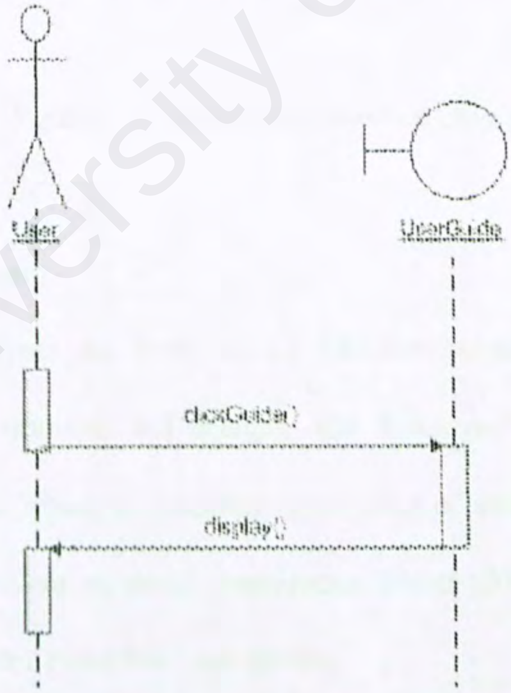


Figure 5.11 User Guide Sequence Diagram

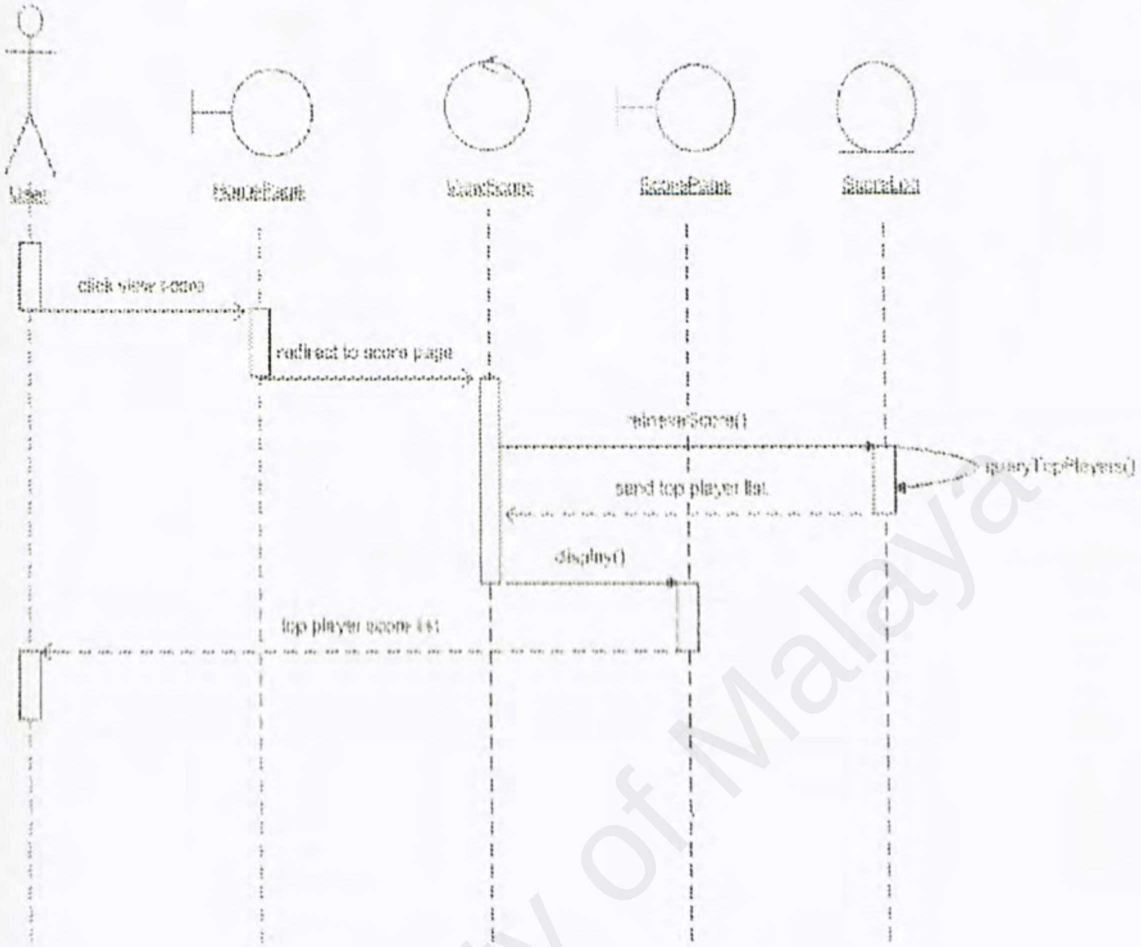


Figure 5.12 View Score Sequence Diagram

5.3.3 Activity diagram

An activity diagram shows the flows among activities associated with a given object, including transitions, branches and merges, and forks and joins. Activity diagrams represent two contexts, which is modeling some kind of workflow, and describing the working of some algorithm or detail computation (Scott, 2001). The following is the play activity diagram of the Front End User module:

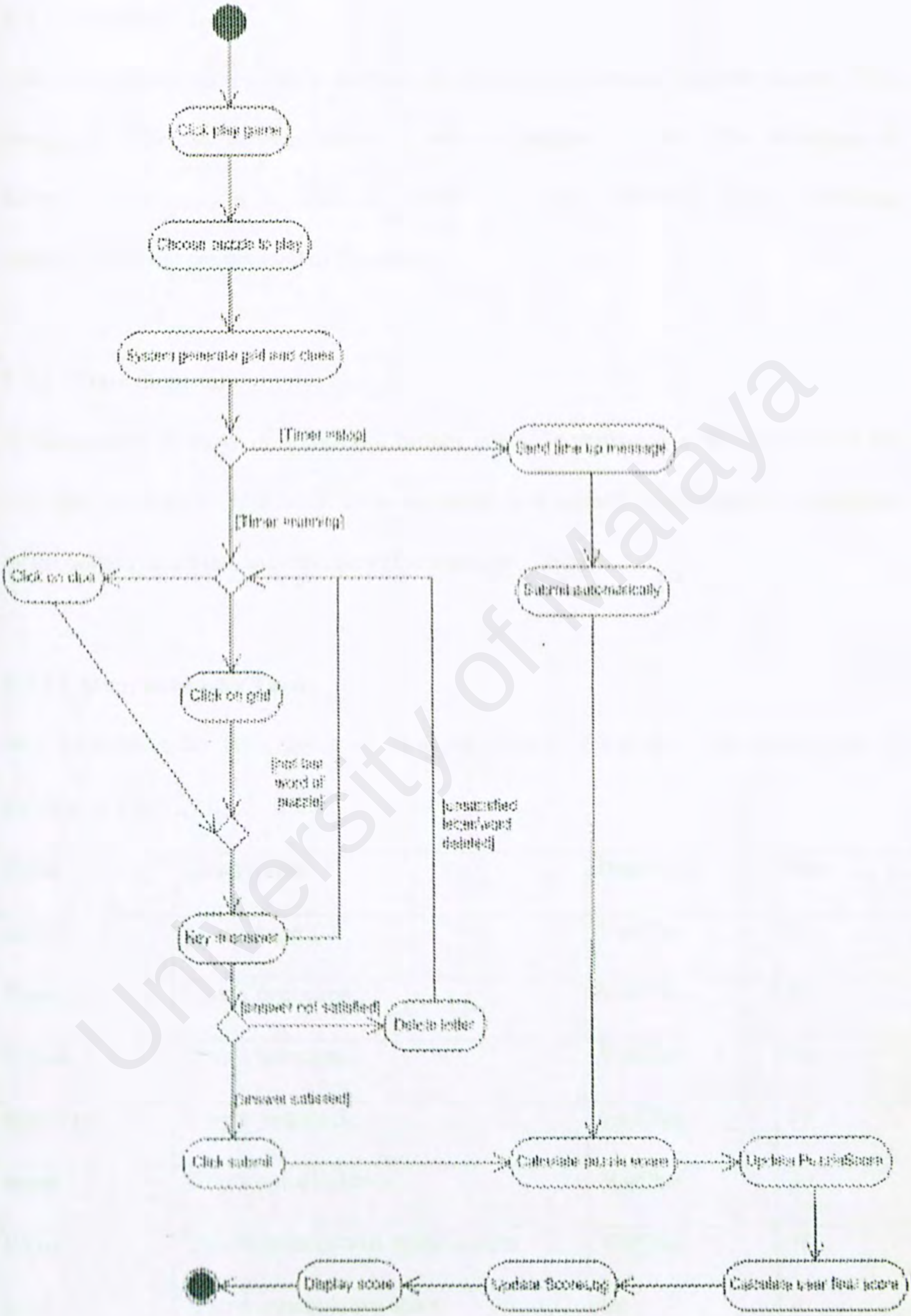


Figure 5.13 Play activity diagram

5.4 Database Design

The Front End User module's database is base on a relational database design. This design has relations between tables to enforce integrity of data. The advantage of enforcing relationship is to create a database free from duplication, hence eliminating redundant data and preserving data formatting.

5.4.1 Data dictionary

A fundamental property of a database system is that is maintains a description of al the data that it contains. This information is stored in a collection of relations, maintained by the system called the data dictionary (Ramakrishan, 1998).

5.4.1.1 MemberMaster Table

This MemberMaster table stores all basic information of the user. The primary key of the table is `loginID.`

Field	Description	Data type	Size
userID	User's login ID	VarChar	20
fName	User's first name	VarChar	20
lName	User's last name	VarChar	20
password	User's password	VarChar	16
email	User's e-mail address	VarChar	30
status	User's current status in the system	VarChar	10
score	User's accumulative score	Int	4

Table 5.1 Data Dictionary for Member Master Table

5.4.1.2 PuzzleMaster Table

This PuzleMaster table stores all basic information of the puzzle. The primary key of the table is `puzzleID`.

Field	Description	Data type	Size
puzzleID	Puzzle ID	Int	Auto
level	Level of difficulties of the puzzle	VarChar	15
dateOfCreation	Creation date of the puzzle	VarChar	20

Table 5.2 Data Dictionary for PuzzleMaster Table

5.4.1.3 Puzzle Table

The Puzzle table stores all the content related to the puzzle. The primary key of the table is `no`, which is auto generated.

Field	Description	Data type	Size
No	Index	Int	Auto
puzzleID	Puzzle ID	Int	4
Word	Word in the puzzle	VarChar	13
location	Location of the first letter of the word	Int	4
direction	Direction of word in the puzzle	VarChar	6
clue	Clue for the word in the puzzle	VarChar	20
wordNumber	Clue number	Int	4

Table 5.3 Data dictionary for Puzzle Table

5.4.1.4 LinkMaster Table

The LinkMaster table contains all the hyperlinks information for the system. The primary key of the table is no, which is autogenerated.

Field	Description	Data type	Size
no	Index	Numeric	Auto
title	The site title	VarChar	100
url	The URL address of the site	VarChar	100
category	Category of link	VarChar	16
dateUploaded	Date and time the URL is last updated	DATETIME	8

Table 5.4 Data dictionary for LinkMaster Table

5.4.1.5 CommentMaster Table

The CommentMaster Table contains all the comments and the guests' information. The primary for this table is the index no that is auto generated.

Field	Description	Data type	Size
No	Index	Int	Auto
comDate	Date of comment	DateTime	8
Name	Name of the guest	VarChar	30
comment	Comment that is posted	VarChar	200
Email	Email of the guest	VarChar	30
Version	Comment of which version of interface	VarChar	6

Table 5.5 Data dictionary for CommentMaster Table

5.4.1.6 VoteMaster Table

The VoteMaster Table stores all the related information of the voting poll. The primary key in this table is the VoteNo.

Field	Description	Data type	Size
VoteNo	Index	Int	Auto
vote	Puzzle ID	Int	4
dateOfVote	Word in the puzzle	VarChar	13

Table 5.6 Data dictionary for VoteMaster Table

5.4.1.6 Relationship

The relationship among the tables state above is as Figure 5.23

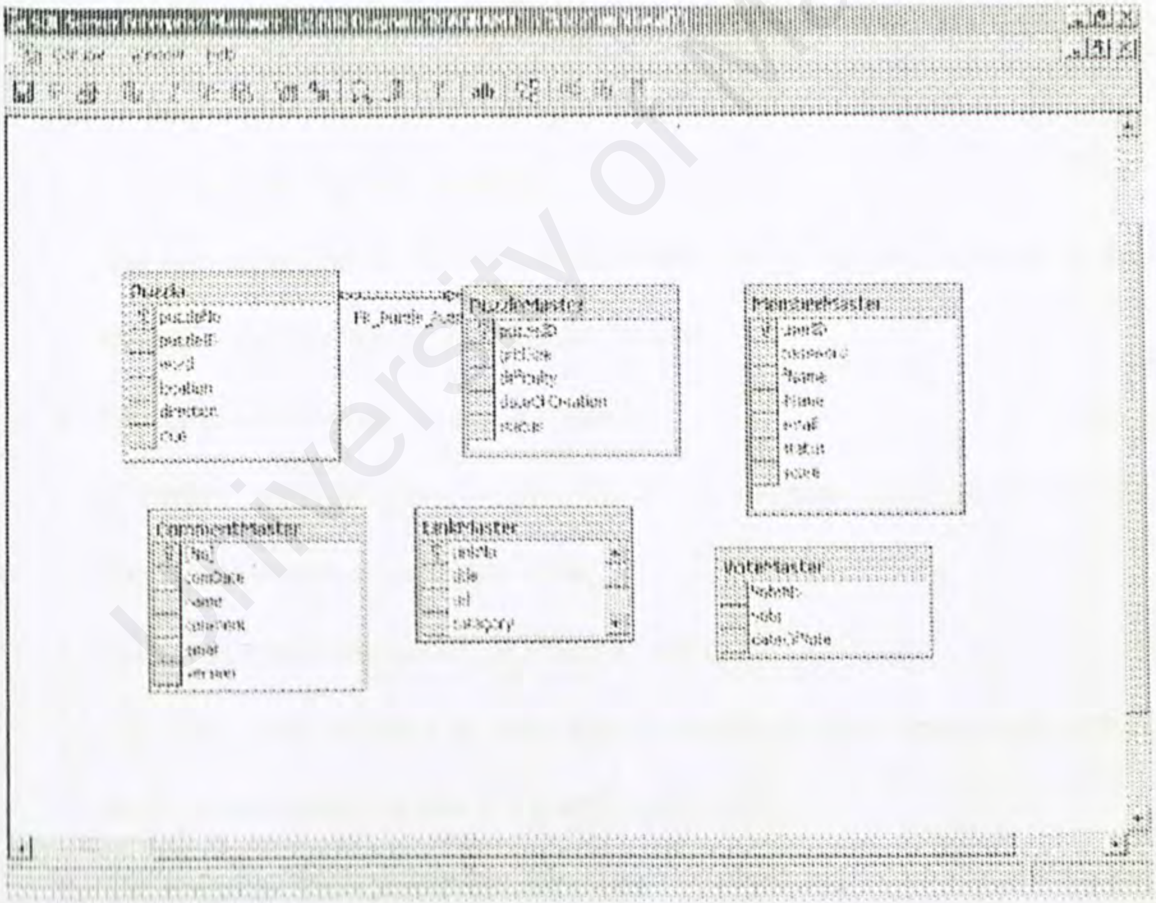


Figure 5.14 Relationship among tables of Front End User module

5.5 User Interface Design

"The architectural design provides a software engineer with a picture of the program structure. Like the blueprint for a house, the overall design is not complete without representation of doors, windows, and utility connections for water, electricity and telephone. These 'doors and windows' for computer software comprise the interface design of a system" (Pressman, 1997).

The guidelines for a good interface design by Roger S. Pressman are:

- Be Consistent

  Use a consistent formats for menu selection, command input, data display, and the myriad other function that occurs in a Human Computer Interaction (HCI).

- Reduce memorizing between actions

  The user should not be expected to remember a list of numbers or names so that he or she can reuse them in a subsequent function.

- Use simple action verbs to name commands

  A lengthy command name is more difficult to recognize and recall. It may also take up unnecessary space in menu lists.

- Display relevant information to the current context

  The user should not have to wade through extraneous data, menus, and graphics to obtain information relevant to a specific system function.

- Use consistent labels, and predictable colors

  The meaning of a display should be obvious without reference to some outside source of information.

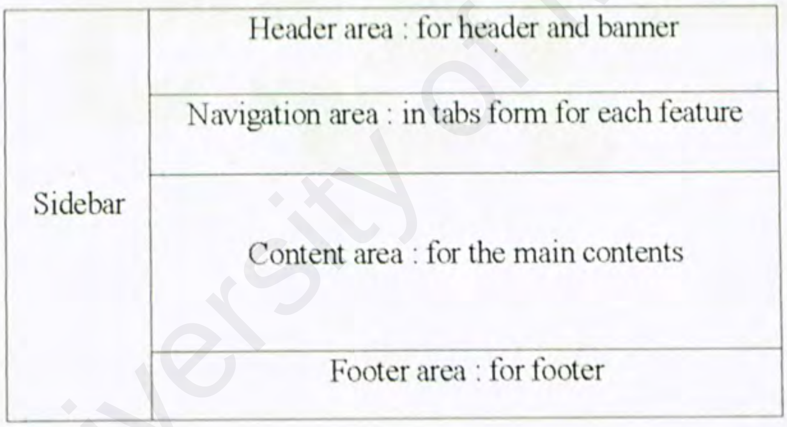
- Provide help facilities that are context sensitive

  This reduces the time required for the user to obtain help and increases the "friendliness" of the interface.

- Let the user control the interactive flow

  The use should be able to jump unnecessary action, change the order of required action and recover from error condition without exiting from the program.

The user interfaces for the Online Crossword Puzzle for learning Synonyms and Antonyms is a web-based document basically divided into 4 parts:



Sidebar

Header area : for header and banner

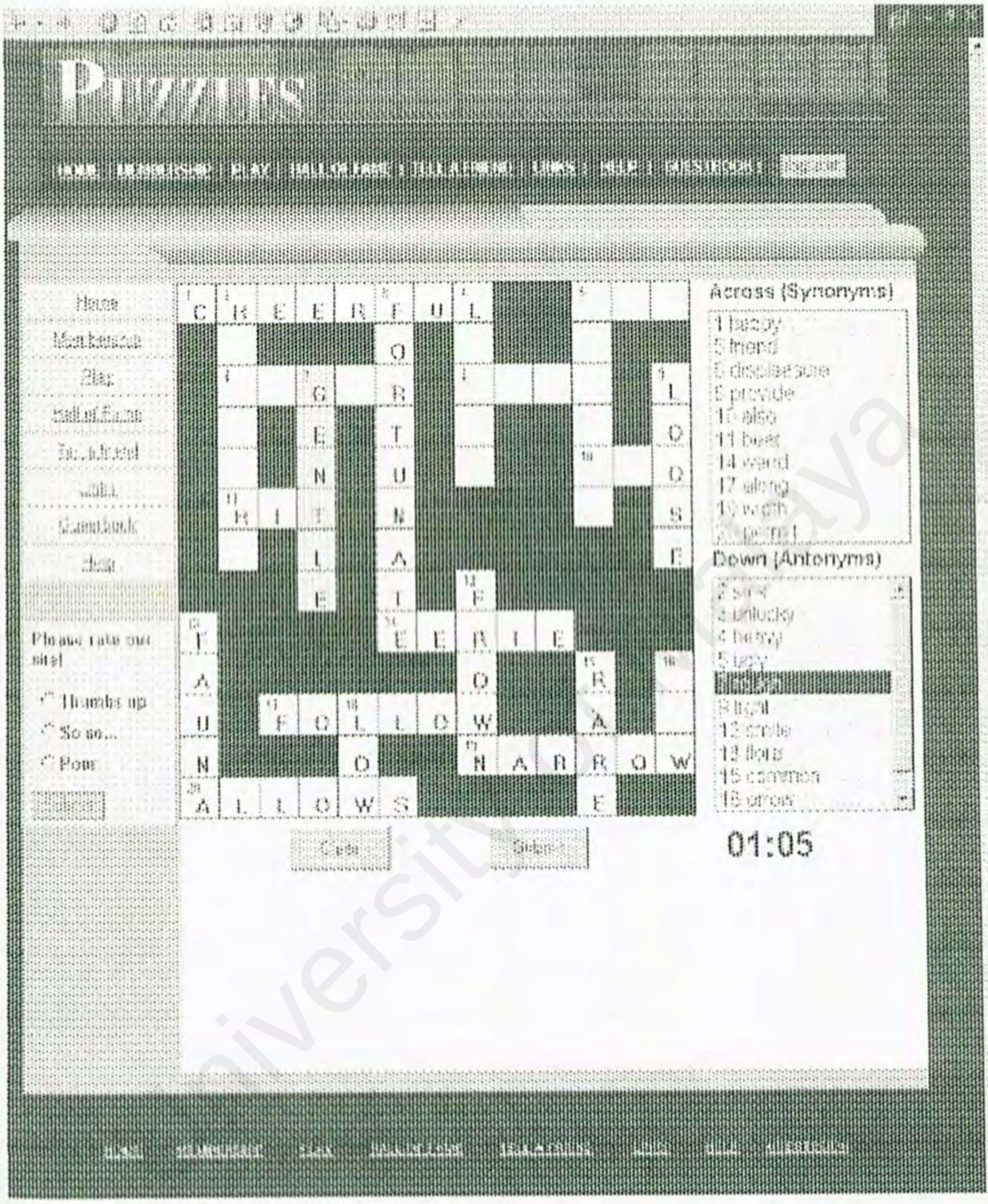
Navigation area : in tabs form for each feature

Content area : for the main contents

Footer area : for footer

Figure 5.15 Web Page Design Outline

The following pages show the sample screen shots of the actual user interface design.


PUZZLES
HOME | MEMBERSHIP | PLAY | HALL OF FAME | TELL A FRIEND | LINKS | HELP | GUESTBOOK |
Home
Membership
Play
Hall of Fame
Tell a friend
Links
Guestbook
Help
Please rate our site!
○ Thumbs up
○ So so...
○ Poor
Clear
Submit
HOME MEMBERSHIP PLAY HALL OF FAME TELL A FRIEND LINKS HELP GUESTBOOK

Across (Synonyms)

1 happy
5 friend
6 displeasure
8 provide
10 also
11 beat
14 weird
17 along
19 width
20 permit

Down (Antonyms)

2 sick
3 unlucky
4 heavy
5 lazy
[illegible]
9 tight
12 smile
13 flora
15 common
16 enemy

01:05

Figure 5.16 Grown up user playing page

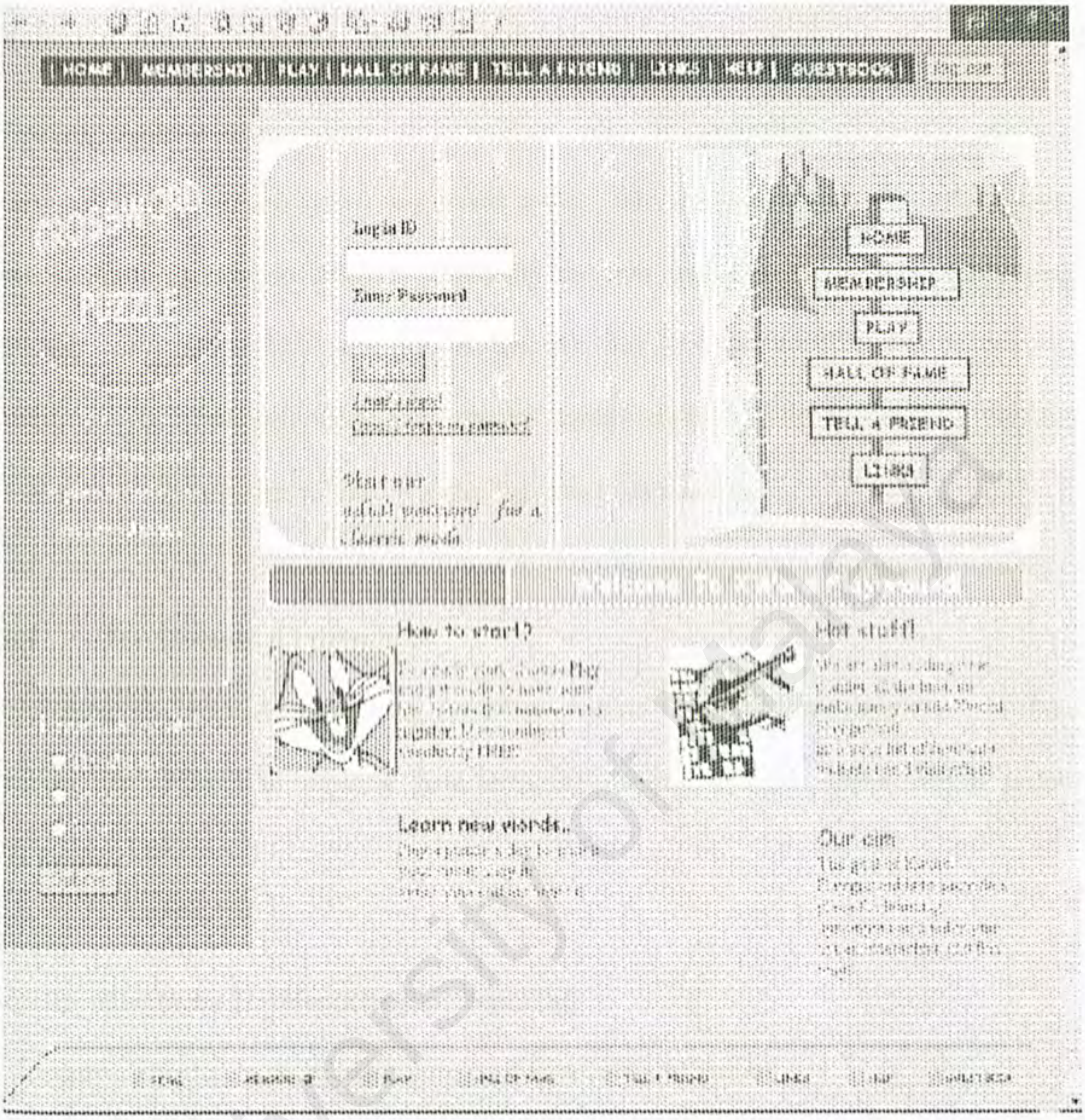


Figure 5.17 Junior version login page

5.6 Summary

A good system design will act as a good foundation to the system. Further progression in developing the system will be in order and smooth sailing. The designs here are base on requirements that were stated in the previous chapter. As the understanding of the needs increase, some detail maybe uncovered. The iterative and incremental waterfall model does allow us to do the minor modification.

Chapter Six System Implementation

System implementation plays an important role of converting previously analysis, design and requirements into a real world system. The designs done in the earlier stages were meant to provide ease in combining the sub modules into a fully functioning system during implementation. As the Iterative and Incremented Waterfall Model was used, designs previously done may require certain changes. The approach of building this system is bottom-up; follow up to the high-level sub modules.

6.1 Development Environment

Choosing the right hardware and software criteria enable a full utilization of both capabilities. This will able to help ensuring quality and the speed during system implementation.

6.2.1 Hardware Development Environment

Beneath are the hardware specifications used to develop the system

- Intel Pentium III 733 Mhz
- 256MB SD RAM
- 20.0 GB Hard Disk
- 100 Mbps Network Interface Card
- 17" colour monitor capable of 1024 x 768 resolution
- 1.44 MB Floppy Drive
- 52x CD-ROM Drive
- Other standard computer peripherals

6.2.2 Software Development Environment

The following is a list of software tools used in the development of this system.

Software	**Usage**	**Description**
Microsoft Windows 2000 Server	System Development	Operating System
Internet Information Server 5.0 with .NET framework	System Requirement	Web Server Host
Microsoft SQL Server 2000	Database	Database Management System
Microsoft Visual Studio .NET	System Development	Integrated Development Environment
Internet Explorer 6.0	System Development	Web Browser
Jasc Paint Shop Pro 7.0	Interface Design	Graphics Editor
Jasc Animation Shop 3.0	Interface Design	Graphics Editor
Microsoft Word	System Development	Documentation

Table 6.1 List of software tools used

6.3 Data Preparation

6.3.1 Images and animated graphics

Web applications are usually judge by their appearance. This makes attractiveness compulsory criteria. Hence, adding images and animated graphics as background and buttons are done to attract visitors. Commonly picture format used in the web pages are .GIF and .JPG. Paint Shop Pro is used to create images, while Animation Shop is used to create animated images.

6.3.2 Database

The database for Crossword for Learning Synonyms and Antonyms is created on a Microsoft SQL Server 2000. Query Analyzer was use to create and modifying the tables, views and relationships.

ADO.NET is used to provide the .NET Framework for interacting with data from data store (Ulman C et al). In ADO.NET, data are manipulated as disconnected data. This means that a copy of the data is copied from the database. The database is updated after we have finish with the manipulation. This provides efficiency and scalability. If disconnected data are not used, connections need to be kept open until the end of each user's session. The structure of ADO.NET illustrated below.

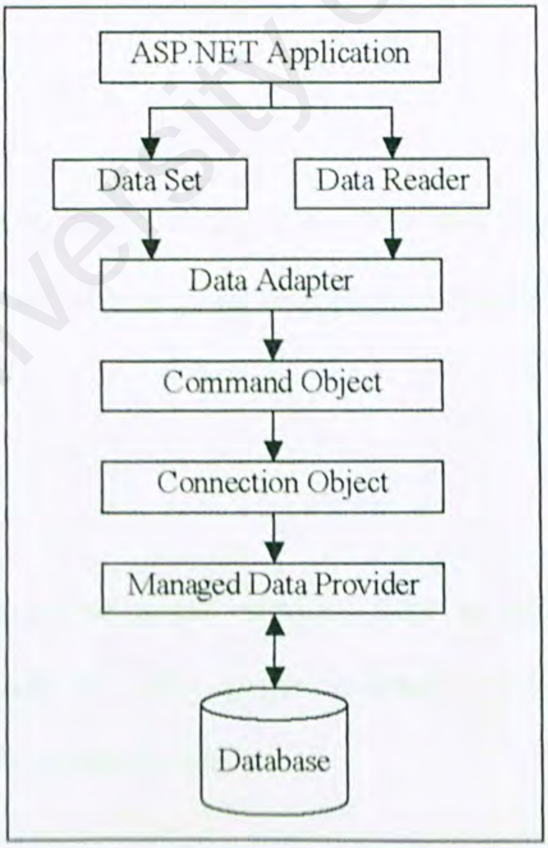


Figure 6.1 Structure in ADO.NET

6.4 Program Coding

Program coding is the most tedious part of implementing the system. From the designs made, algorithms are written to tackle the problem or to convert the designs to computer understandable language. Programming requires a creative and flexible approach from the programmer to create a simple and neat coding. But, there are certain guidelines and strategies that can help implement a high quality programmed code.

6.4.1 General Guidelines and Coding Principles

6.4.1.1 Modularity

Proper partition of functions helps a developer to reduce the complexity of problem. In this way, reusability is also enhanced, as functions are more independence from the main problem.

6.4.1.2 Reusability

Functions that are written should be able to be reuse after slight or no modification at all. This ease the process of coding and contributes to reducing the time and space of redundant lines of codes.

6.4.1.3 Readability

Coding done on the system should provide reader an easy understanding of the program. This is done by using proper indentation for control structures and standardizes function and variable declaration.

6.4.1.4 Maintainability

Developer should be able to read, revise and correct program that are written by previous developers. This not just encourages reusability but also make future enhancement easier. Codes should be loosely coupled and written in simple and separate modules.

6.4.2 Internal Documentation

Internal documentation is explanatory text embedded into the code giving the programmer a better understanding of the scenarios. These texts are located at places where the programmer feels needed.

6.4.2.1 Header Block

Header Block provides an overview of the function as whole, the parameters involve, algorithm of the function, and the return values. The following lines show a sample block code used during coding.

```
/*------------------------------------------------------------
*     Filename : play.aspx
*     Author   : Tay Lik Czek
*     Function : To provide a playing platform combining web
*                elements with ActiveX object
*     Date     : 30 November 2002, 3:42PM
------------------------------------------------------------*/
```

6.4.2.2 Variable Names and Function Names

The standard used for variable names are variable are name by using the camel case. This is by starting the variable name with a lower-case letter and continuing in lower-case, except for the first letter of each concatenated word, which should be in upper case. For example `wordNumber`, which means the variable contains the number of the word.

The naming of function is base on Pascal case. In this case the first letter of each word concatenated will start with an upper-case letter. For example, `GridClear`, which describe the function of clearing the grid.

All constants, declared in the function would be written in upper case letters. For example, `GRIDSIZE`, which is constant.

6.4.2.3 Indentation

Indentation is applied to control structure to ease reading and tracing of codes. Codes within a control structure are indent once on first level, twice on second level and so on. When indentation reaches too deep, modularization is considered to reduce the complexity of the function. The following is an example of the indentation style used in coding.

```
private void Page_Load(Object sender, EventArgs E)
{
   if (Session["userId"]!= null)
   {
      objDataReader = objCommand.ExecuteReader();
      while(objDataReader.Read() == true)
      {
         strUserId += objDataReader["userID"];
         strEmail += objDataReader["email"];
      }
      objDataReader.Close();
   }
}
```

6.4.3 Application Constant Parameters

The parameters of a few commonly used functions are save in a web.config file. This file is located at the root of the virtual directory, but it is not accessible by client browsers. This creates a secure area for accessing confidential parameters like a connection to the database.

The web.config file is save in XML format. Each parameter name is declared as a key follow by the value of the parameter. Below is the extract of the web.config file used.

```
<configuration>
   <appSettings_>
      <add key="SqlDataSource"
         value="server=localhost;uid=sa;pwd=;database=3182" />
      <add key="CABSource"
         value="http://localhost/CAB/" />
      <add key="Servlet"
         value="http://localhost/" />
   </appSettings>
</configuration>
```

6.4.4 Database Connection

To establish a connection to the database, 4 important elements are required. Theses element were illustrated in Figure 6.1, namely Data Reader, Data Set, or Data Adapter, Command object.

Command object is an object that contains the SQL statement, and the connection involved. Different functional SQL statement requires reconfiguration of the Command object.

Data Adapter creates however is more flexible that command object. This object requires only a single SQL SELECT statement to produce relevant SQL statement for INSERT, UPDATE, and DELETE.

Data Set is consider as a collection of disconnected data from the database server. In a Data Set, users can create multiple data tables and establish relation between them as in the DBMS.

Data Reader provides an efficient way to retrieve the data for publication purpose. The down side it only provides a quick one directional process of retrieving the data to the web page but not updating the data.

6.5 Algorithms

Algorithms are where the logics, and formulas are presented. The detail studies and dissection of the problem gives a clear picture on how the problem is solve ease the coding of the function. The following are a few special cases noted for their complexity.

6.5.1 Across and Down Words

The problem appear when the language does not provide a two dimensional array for objects. This creates a problem in placing the words vertically. The problem was sorted out by using matrix algorithm of transpose. The first assumption was of the current array of objects is arrange in a square matrix order.

Transpose is an algorithm where the elements of the matrix are mirrored along the perpendicular line from the top left corner to the bottom right corner. After applying the transpose algorithm, words that are horizontal turn vertical, and vice versa.

T	A	B	L	E
		O		
		O		
		K		
		S		

Before transpose

T				
A				
B	O	O	K	S
L				
E				

After transpose

Figure 6.2 Example of transpose process

An array of variable is defined to provide reversal calculation of the transpose position to actual position on the gird. The following is the pseudo code for declaring the variable with the actual position on the grid.

```
Initialize size to size of square matrix
Initialize element to number of elements in square matrix
Initialize reference element to one

While still in range of elements (size*size)
      Set element to reference value

      If element divisible by size
            Set reference element to current element plus size
      Else
            Set reference element to integer of element divided by
size plus one
Loop
```

In programming language, the pseudo code will be translated to the codes below.

```
    j = 1
    For i = 1 To size * size
        refer(i) = j
        If i Mod size Then
            j = (j + size)
        Else
            j = Int(i / size) + 1
        End If
    Next
```

The results of the code with size of five is stated below

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Before Transpose

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

After Transpose

Figure 6.3 Matrix showing the relative position of the crossword

6.5.2 Scoring algorithm

The scoring algorithm for the crossword puzzle is base on the number of letters each word contains. This means a correct four-letter word will get four points. If the letters in the word exceeds four, an addition three mark will be added to number of letters. This means a correct six-letter word will get six points plus three, totaling 9 points.

Different level of difficulty will produce different multiplication scoring. Easy level points do not multiply. But intermediate level will multiply the points by two and hard level will multiply the points by three.

6.5.3 Servlet In ASPX

A connection between a client computer to a database server over the internet requires high bandwidth. This cause the client web page to take a longer period of time to process especially during sending of information to the server. Placing an ASPX page on the server solves this problem. The purpose of this page is to receive input string that is needed to retrieve data from the server or input strings that is needed to store data on the server.

By placing an ASPX page on the server, the database processing is done over on the web server instead of from the client directly to the server. Besides that, the firewall would not pose a problem, as the connection will only be done through the port 80, namely HTTP port.

In ActiveX environment, ActiveX Internet Transfer Protocol is used to provide such a connection to the ASPX page. The strings are pass to the ASPX page using the GET method, while the responses are returned in a form of text file. Below is an example of retrieving response from the server followed by the processes done on the ASPX page.

```
ID=Inet1.OpenURL http://localhost/plserver?difficulty=hard

<%@ Page language="c#"  %>
<%
        if(Request.QueryString.Get("difficulty")=="hard")
```

```
            {
                    ...
                    Response.Write(ID);
            }
    %>
```

6.6 Summary

The whole process of implementation begins from the process of setting up the required hardware and software. This is followed by the need to solve technical issue like database connection, and designs of images. The whole process required proper documentation for future reference. In this part important algorithms are describe to solve complex solutions.

Chapter Seven System Testing

Testing is one of the critical phases in a Software Development Life Cycle as it determines how well the final outcome of the system. Testing was conducted during and after the system implementation as defined in the development methodology in so that errors can be detected earlier. It represents the complete and extensive review and challenge on the application design, specifications and codes. Testing also provides a method to uncover logic errors and to test the system reliability.

There are four basic concepts related to software testing (Jarvis and Crandall, 1997):

- Error detection that helps to identify errors either by inspection, walk-through or other type of error detection approaches.
- Error removal where the process of debugging and removing identified error was conducted.
- Error tracking to find the cause of the error and fix the flaw.
- Regression testing where testing is conducted to find out whether the fixed error is working properly and/or the rework of the code actually fixes the error or fixes it in one part and fails another part of the code.

7.1 Front End User Testing

A good and organized testing plan is essential to ensure that the testing is done appropriately and thoroughly. The basic step in testing the Online Crossword Puzzle for Learning Synonyms and Antonyms (Front End User Module) is the unit testing that was to ensure that the code implemented the design properly. This was followed by integration testing on all the units and integrations of all integrated unit clusters and later followed by system testing. The steps for testing this project are shown in the figure below.

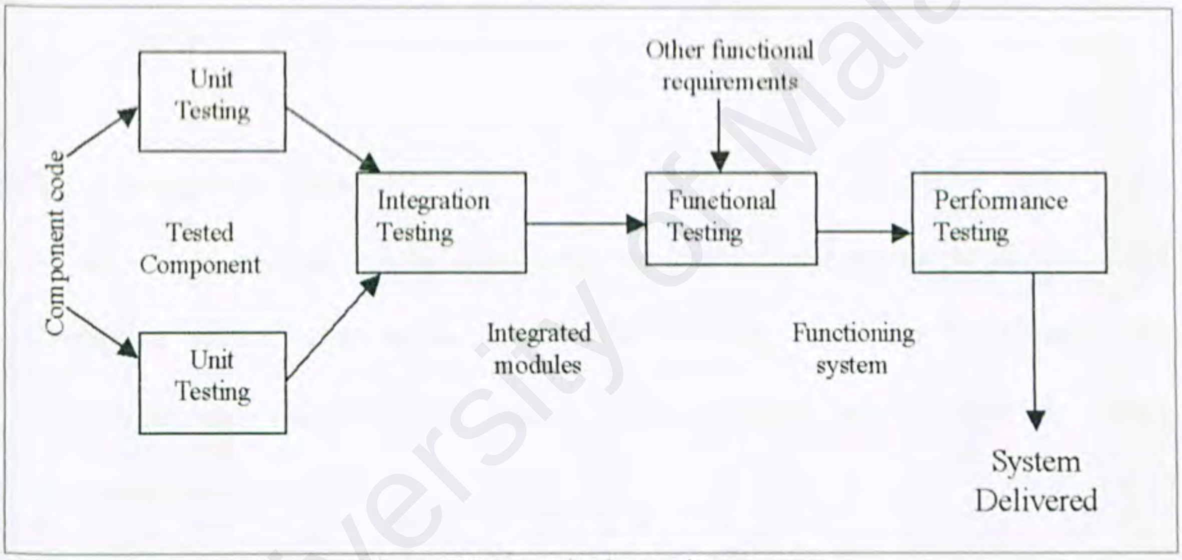


Figure 7.1 Steps in the testing process

7.2 Unit Testing

Unit testing is started off with testing the functionality of each component as an individual. This means the object is isolated from the rest of the system and tested as an entity. For example, the combo box is tested as an individual like how it passes the value and the reaction to event occurred. Three methods have been used for this purpose. They are:

- Ad Hoc approach tests the object to the extreme by trying to bring the object to a halt. This requires trial and error method, which pushes the object to the limits.
- White-box Testing is done by implementing the tests internally within the program. This means that input by users will be validated before being input into the systems. The example of such implementation would be the need to validate the email address before being store or process in the database.
- Black-Box Testing is completed by running a series of test with unlimited and unknown input. The outputs will be analyzed to match the expected outputs. The system (Pfleeger, 2001).

7.3 Integration Testing

At the end of unit testing, the components are gathered and arrange to provide a full functioning system. But, before this system is being tested for functionality, the components are integrated and tested. The integration testing helps to ensure components gels well.

The bottom-up integration is adopted in this system testing. The tests started with the testing of several components in small groups. An example of it will be the guest book module. This module is tested for it storage of the content written on the guest book. This is done as an individual set of components. The rest of the modules follow its footsteps next. Figure 7.2 illustrates the integration of the unit components in the Playing Modules as whole.

After these small groups of components are tested and verified, they are then integrated into Playing Module as whole. The escalation moves on to the integration of both Administrative module and Playing Module to provide life to the whole system.

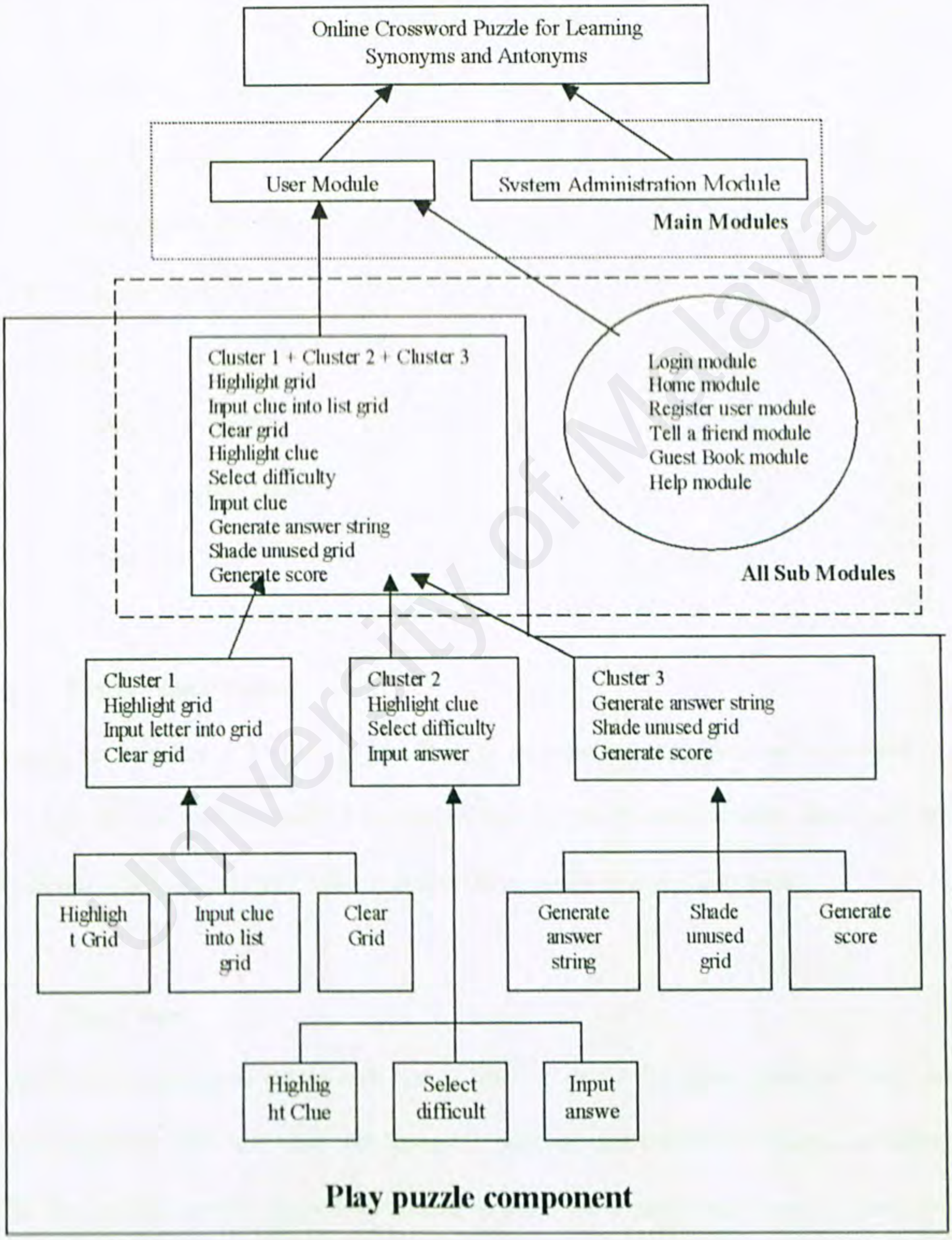


Figure 7.2 Flow of bottom-up testing of the system

7.4 Functional Testing

The purpose of functional testing is to be certain of the characteristic of the system works as it was stated in the system requirements and analysis specification. The following are some of the main function tested:

User Homepage

- Login and logout
- Playing the crossword
- Calculation of score
- Selecting difficulty
- Signing on guest book
- Viewing on guest book
- Voting

7.5 Performance Testing

Test on performance is a series of test done to measure the non-functional requirements. This includes the test of speed of building a random puzzle and the reply time given by the servlet. The test cases are built to measure the responds time of the system.

7.6 Test Cases

Test Cases are scenarios where each test is listed in detail the inputs, series of tasks and results expected. The test cases are design to perform unit testing till integration testing with the specific results. Repetitive testing is done on a single test case to prove the consistency of the results.

7.6.1 Sample Test Cases

The following are some sample test cases on certain sub-modules taken from Playing Package of the whole system. Selected module shown here is the Play Puzzle module.

7.6.1.5 Unit testing for sub-module: Play Puzzle

Condition to Test	Data/Steps to perform	Expected results	Actual Results
Items in combo box can be selected.	Choose and click on an item in the combo box.	Item is selected.	Item is selected.
Play button functioning	Click Play button, after selecting difficulty	Background image disappear. Crossword grid appears.	Background image disappear. Crossword grid appears.
Selected word can be highlighted for letter input.	Click on a grid.	On clicked grid is highlighted in blue and the whole word is highlighted in green.	On clicked grid is highlighted in blue and the whole word is highlighted in green.
Highlighted grid can accept letter input.	Type in words into the highlighted grid.	Grid is filled with inputted letter. The following letter area is highlighted in blue.	Grid is filled with inputted letter. The following letter area is highlighted in blue.

Condition to Test	**Data/Steps to perform**	**Expected results**	**Actual Results**
Grid control by directional keys.	Click on the grid, move around the grid using arrow keys as navigator. Press spacebar to change direction.	Highlighted rows moved around according to keypad navigation. Highlighted row changed direction when spacebar is press.	Highlighted rows moved around according to keypad navigation. Highlighted row changed direction when spacebar is press.
Backspace and delete key and functioning	Backspace button and delete key deletes current highlighted letter	When backspace button press, blue grid will move backward and delete the letter in the previous grid if exist. Delete key clear current blue grid letter.	When backspace button press, blue grid will move backward and delete the letter in the previous grid if exist. Delete key clear current blue grid letter.
Clear button is functioning.	Click on the clear button.	Grid is being clear off without removing word numbers and blackout areas.	Grid is being clear off without removing word numbers and blackout areas.
Submit button is functioning.	Click on the Submit button after filling up answers.	Score is shown after button is instigated. Grid is disabled.	Score is shown after button is instigated. Grid is disabled.

Condition to Test	Data/Steps to perform	Expected results	Actual Results
Timer working well	Start puzzle until timer is up	Timer update itself every second. Timer turns red when time is less than 10 seconds. When timer is up, submit is automatically instigated.	Timer update itself every second. Timer turns red when time is less than 10 seconds. When timer is up, submit is automatically instigated.

Table 7.1 Sample test cases for unit testing on Upload Puzzle Sub-module

7.6.1.6 Integration testing of all components of sub-module: Play Puzzle

Condition to Test	Data/Steps to perform	Expected results	Actual Results
Playing a new puzzle.	Highlight clue, select difficulty and insert clue. Click finish and confirm pop-up message box.	Random selection of puzzle base on level of difficulty. Accuracy of score calculation and final score. Crossword is disabled.	Random selection of puzzle base on level of difficulty. Accuracy of score calculation and final score. Crossword is disabled.
Mouse and keypad navigation functioning simultaneously	Click on the grid by using mouse, navigate the grid using keypad arrow keys then double click on grid	Grid movement and highlighting of row is done smoothly.	Grid movement and highlighting of row is done smoothly.

	using mouse and/or press spacebar to change direction.		
Highlight grid, input letter and clear grid.	Highlight a grid, input letter into grid, and clear grid after input.	All functions working smoothly.	All functions working smoothly.
Scoring and total cumulative score.	Fill in answer with wrong answers.	Score produce should be as stated in algorithm. Accumulative score is shown.	Score produce should be as stated in algorithm. Accumulative score is shown.

Table 7.2 Sample test cases of integration testing for Upload Puzzle Sub-module

7.7 Summary

The process of testing Playing module for the Online Crossword Puzzle for Learning Synonym and Antonyms are stated in this chapter. An extract of the test cases done on the whole system is also shown to show the handling of the test case. Being develop base on the iterative and incremental waterfall mode, errors found in the test cases are able to be corrected and retest. If a solution cannot be found, slight amendment on the design is being done. The cycle of testing is time consuming but it helps ensure the quality of the final product.

Chapter Eight System Evaluation

The process of evaluation is conducted to identify the strengths, limitations and future enhancement of the Online Crossword Puzzle for Learning Synonyms and Antonyms (Front End User Module). Besides that, this phase also discussed the problems encountered during the system development life cycle and its solutions.

8.1 Problems Encountered and its Solutions

There are several error and problems encountered throughout the system development stage. Most of the problems were solved so that the system will function properly and meet its requirements. Understanding the problems and research in the Internet as well as discussion with course mates, help to solve these problems. Some of the problems encountered are listed below.

8.1.1 Development Tools Selection

One of the major problems encountered at the early stage of development was the selection of the appropriate development tools and language for the object. As the most important sub-modules of the Front End User module is the crossword puzzle play engine, the language of the object is very important and determined the strength of the whole game. Besides that, the chosen language must be compatible with the web programming language that was chosen. Therefore, a thorough study and analysis was done in order to make the best decision for the system.

8.1.2 Lack of Knowledge in the programming language

The combination of ActiveX components with dynamic content creates a new challenge in programming. During implementation, a brand new picture comes into the perspective. This widens the scope of learning the implementation logic behind the scene. Building an ActiveX component also was some thing new, especially building it for the purpose of Internet. ASP.NET brought a whole new method in designing the web. The function that came with the .NET framework took time to learn the necessary function.

8.2 Systems' Strengths

8.2.1 Focuses on Synonyms and Antonyms

As most of the crossword puzzle available is based on synonyms concept, this system provides an advantage by presenting a puzzle in both synonyms and antonyms words and clues. This point can ensure that users not only learn word of a same meaning but also word of an opposite meaning.

8.2.2 Ease of control

The usage of the keyboard and mouse can be interchange at anytime during the game. Users are able to use the directional key and the spacebar key to help movement in the grid. For users that prefer mouse control, they are able to click on the list of clue to highlight the word on the grid, and vice versa.

8.2.3 Dual interface

The interface provided by the system caters for both junior users and grown up users. This feature allows user to choose the interface of their liking and still enjoy the same facility of the system.

8.2.4 Provide a score function

The Online Crossword Puzzle for Learning Synonyms and Antonyms are different from convention puzzles where it is develop with a combination of a game and a learning tool where a scoring system is developed to keep track of the achievement of each user.

8.2.5 Timer Control Game

A timer is there to encourage users to react fast. When they react fast, their association of the word to synonyms and antonyms increase. This provides a good way of improving their vocabulary.

8.2.6 System transparency

The system itself is built in such a way that the users do not feel the complexity of the system. The movement of the data from the database to presentation and the interaction of the ActiveX with the back end is totally transparent. The medium of communication to the users is the graphical user interfaces.

8.3 System Constraints

8.3.1 Only rights answers will be displayed

The play engine detects all the answers inputted on the grid and highlight the right answers but user will not be given the answers to the clue that they answered wrongly. In this case, user needs to look for other resources and play the puzzle all over again until the correct answer is obtained.

8.3.2 Word pronunciation support

The clue or word listed in the crossword puzzle cannot be read out to the users. This limitation may cause player being unable to learn the right pronunciation of the words.

8.3.3 Web browser limitation

The playing engine of the crossword was created as an ActiveX object. Currently, only Internet Explorer provides a smooth integration of ActiveX object. Other leading web browsers like Netscape and Opera needs an additional plug-in before being able to load the ActiveX object.

8.3.4 Static grid size

The crossword grid size is fixed to 13 squares across and 13 squares down. This limits the player to play any larger size of puzzle that is set by the administrator. Crossword puzzles that cover smaller area will be filled with a dark area.

8.4 Future Enhancement

8.4.1 Audio support

This is to open the game to those who are blind. Furthermore, this feature helps the users to have an opportunity of learning the right way of pronouncing the word or new words learn during the game.

8.4.2 Forum corner

A forum corner will provide user to create a topic on what is going on or problems face in playing the puzzle. An example of it will be the players maybe posting a topic related to a specific puzzle, then discuss about it. The forum will allow follow-up to the topic that area available.

8.4.3 Dynamic grid size

The size of the grid will shrink and grow base on the puzzle that is chosen by the player. This will provide users a more pleasant interface by removing area that are not relevant to the current puzzle.

8.4.4 Multi-lingua support

The support of such facility enlarges the scope of users as the Internet covers globally and the world does not speak only one language. This implementation covers the aspect of the game play and the whole user's interface.

8.5 Summary

As a conclusion of the whole system, the development of the whole system faced a few problems and challenges, which was calmly overcome. The system has a few advantages of the currently existing system. Hence, there are still area where the system lacks compare to other systems and the vast improvement of technology. This brings the need to estimate the future and current technology that can be embedded into the system to provide the users a more comfortable and meaningful system.

References

Conallen J. (1999, March 9). *Modeling Web Applications with UML.* Website for Conallen Inc. www.conallen.com/whitepapers/webapps/ModelingWebApplications.htm. Visited on 2002, August 3.

Deitel H. M., Deitel P.J. & Nieto T.R. (2000). *Internet & World Wide Web How to Program*. 2nd ed. Prentice Hall, Inc.

Galitz W.O. (1997). *The Essential Guide to User Interface Design: An introduction to GUI Design Principles and Techniques*. 1st ed. John Wiley & Sons, Inc.

Knight B. (2002). *SQL Server 2000 Benchmark Results; Yukon Continues Dramatic Change*. Website for SWYNK.com. www.swynk.com/friends/knight/sql2000benchmark.asp. Visited on 2002, September 8.

Online Dictionary. (2002). www.webopedia.com. Visited on 2002, July15.

Parker G. (2000). *Using Games and Activities For On-line Learning* : Some Basic Principles. www.glennparker.com/Freebees/article-online-learning.html. Visited on 2002, August 1.

Pfleeger S. L. (2001). *Software Engineering Theory and Practice*. 2nd ed. Prentice Hall, Inc.

Product Overview for Visual Studio.NET. (2002, June 4). Website for Microsoft Corporation. http://msdn.microsoft.com/vstudio/productinfo/overview.asp. Visited on 2002, August 3.

Rowe T.A. *Webmaster Resouces.* Website for YCOLN.com. www.ycoln-resources.com. Visited on 2002, August 3.

Schwarz, et al. (1993). *The Chambers Dictionary.* British Library Cataloguing in Publication Data.

Scott K. (2001). *UML Explained.* 1st ed. Addison-Wesley.

Taylor C. *Applying Software Engineering Principles to a Web Site Development Process.* http://facweb.cs.depaul.edu/ctiphd/ctirs99/online/taylor.html. Visited on 2002, August 3.

Website for TechTarget Enterprise. www.searchwin2000.com. Visited on 2002, August 3.

Website for TechTarget Enterprise. www.whatis.com. Visited on 2002, July 27.

Bibliography

Chambers J., CEO Cisco System 1999, Comdex Keynote, November 1999.

Rufas D. C.(2001, Jun 15). *Database design in UML.*
www.histeresys.com/eng/articles/UMLDBDesign.htm.
Visited on 2002, August 20.

Jacobson I., Booch G. & Rumbaugh J. (1999). *The Unified Software Development Process.* 1st ed. Addison- Wesley Longman, Inc.

Whitten J. L., Bentley L. D. & Dittman K.C. (2000). *System Analysis & Design Mehods.* 5th ed. McGraw- Hill Irwin.