INTERACTIVE MATHEMATICS LEARNING SYSTEM

(SMARTMATH)

BY

REGINA TAY LI LING

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

UNIVERSITY OF MALAYA

KUALA LUMPUR

2001/2002

Abstract

Interactive Mathematics Learning System (Smartmath) is a Web-based Multimedia Learning-teaching system for lower primary school level. Primary school students, teachers, and parents are the target users of this system. It can be used as a teaching tool or as a self-learning program.

The strength of Smartmath is that it provides lessons to educate users about basic mathematic operations according to users age group. Lessons are provided for children aged 7 and 8 which can also be used by preschool children. There are interactive quizzes that test the student's understanding after each lesson. Games based on mathematic operations and numbers are created for students to test their skills.

This project aims to create an effective and interesting learning tool to enable students learn and enjoy mathematics at the same time. Smartmath, as a web-based application could promote and expose the use of computers to students at a tender age thus enhancing computer literacy.

The waterfall model with prototyping will be development model used to develop Smartmath. The main advantage of this system is that the process can be tailored to meet the specific requirement yet possibly changing needs of any application.

As Smartmath is an interactive multimedia web-based system, tools that will be used for the development of the system comprises of Macromedia Dreamweaver UltraDev and Adobe Photoshop. HTML, JavaScript, and Java are the programming languages that will be used to develop the system.

The main characteristics of this system are user-friendly, attractive interface, easy to navigate, interactive, and interesting with a talking wizard which serves as a guide.

Acknowledgements

As with any thesis, this one wouldn't have been possible without the contributions of many people in one way or another. I would like to take this opportunity to express my gratitude to the following parties:

Firstly, I would like to thank the management of Sacred Heart Convent for allowing me to conduct a survey there. I wish to thank the students and teachers who have given me cooperation during my visit. Their kind cooperation enables me to gather precious information.

Secondly, I would like to thank my supervisors, Pn. Hannyzzura Affal @ Pal and Pn. Sri Devi Sudharshan and moderator, Pn. Sameem Abd. Kareem who has given me precious guidance throughout the development of Smartmath. Their advices and critical comments allowed me to be more creative and analytical when designing the system of Smartmath.

Next, I would like to thank my course mates who gladly shared their knowledge and experiences. The discussions held allowed us to realize the weaknesses and strength of each other's system. With that, we are able to improve the weakness and improve the quality of our system.

Lastly, I would like to thank my family and friends for giving me support and encouragement. Without their motivation and encouraging words, this proposal wouldn't have been made possible.

Once again I would like express my gratitude to all the parties mentioned above. Their contribution is very much appreciated. To those that I missed out, I would like to apologize and your kindness will also be remembered.

Thank you.

Table Of Contents

Chapter 7: System Evaluation

List Of Figures

Chapter 5

Chapter 1 - Introduction

1.1 The Future of Multimedia in Education

Many predictions have been made about the future of education, the demise of the classroom-based teacher, and an information technology lead revolution in schools, universities and organizations. There is little doubt that a significant percentage of educational activity will take place online. By mid-1996 there were over 2000 courses already offered on the World Wide Web *(Tapscott, 1996)*. That number has been growing steadily as strategic plans of universities and educational governance bodies worldwide have been pushing for alternative solutions to dwindling education budgets.

1.2 Project Definition

On December 8, 1980 the Minister of Education at the moment, YB Datuk Musa Hitam, announced that the government will implement a new curriculum in schools. The new curriculum stress on basic education in 3 areas: read, write, count *(3M - membaca, menulis, mengira)* in primary school level *(Awang, 1981)*. The project was launched in 1985, and has been practiced in the country since 1987 *(Ministry of Education, 1993)*.

KBSR *(Kurikulum Bersepadu Sekolah Rendah)* is divided into two *(Ministry of Education, 1993)*:

a) Level 1 (Standard 1 - 3)

b) Level 2 (Standard 4 - 6)

In the Mathematics learning area, students learn about mathematics, what it is and how it is used in making decisions and solving problems. Mathematics involves observing, representing and investigating patterns and relationships in social and physical

phenomena and between mathematical objects themselves: Mathematics is often defined as the science of space and number … [but] a more apt definition [is that] mathematics is the science of patterns. The mathematician seeks patterns in number, in space, in science, in computers, and in imagination. Mathematical theories explain the relations among patterns … Applications of mathematics use these patterns to 'explain' and predict natural phenomena … *(Steen, L.A. (1988), "The science of patterns", Science, 240, 29, 616.)*

Mathematics computation is part of our daily lives. Mathematics could exist in the simplest way such as reading the time to calculating the total amount that we have to pay for what we buy. Everybody will be introduced to the wonders of mathematics once they step into schools for education. Usually it begins with learning numbers.

Traditionally, mathematics is a subject that's being taught in schools using books. The advancement of computer has made learning online possible. Online learning doesn't require one to be in the classroom and it can be done anywhere as long as there is a computer connected to the Internet. This has overcome the weakness in traditional learning which requires coordination between the teacher and students.

Smartmath is a web-based multimedia learning-teaching projects for lower primary school. This can be used as a teaching material in classes or a self-learning program for the students. It also allows parents to give extra guidance to their children at home.

The strength of Smartmath is that it has a talking wizard that catches the student's attention. The wizard works as a tutor that guides the students along the way. It appears on every site to give brief explanation about it. Besides that, Smartmath is a bilingual site.

You can choose to surf the site with English or Bahasa Melayu. Smartmath also provides lessons and quizzes to test the student's capability. Users can learn the basics of mathematic operations according to topics through these lessons. The lessons are divided into 2 main parts:

a) Standard 1 (for kids aged 7)

b) Standard 2 (for kids aged 8)

There are fifteen topics in each level. After each lesson, there will be an interactive quiz section whereby students can practice what they have just learnt. Starting and ending of the quiz is up to the student's choice. There is a timer so that they can know the amount of time they took to answer a certain amount of questions. Students can also test their skills and understanding by playing the games which are based mathematic operations and numbers.

This site contains hyperlinks to other educational sites available. If the users wish to tell their friend about this site, it can be done by sending them an email containing the URL of the site. That way, more people will know about this site.

1.3 Aim & Objectives

1.3.1 Aim

This project aims to create an interesting and effective mathematics teaching and learning tool for lower primary level. It is also important to encourage the young generation to get in touch with the computer and information technology thus enhancing computer literacy among students and teachers.

1.3.2 Objectives

The objectives of this project are as follow:

a) To design and develop a web-based multimedia teaching and learning mathematics system for lower primary school.

b) To develop an effective and interesting learning tool.

c) To create a better learning and teaching approach with the addition of graphics and animation to text.

d) To expose the usage of computers to students at a tender age.

e) To improve the computer literacy level among students.

f) To bring the e-level of education one step further by providing the more effective way of teaching and learning in order to prepare them for greater challenges in the future.

1.4 Project Scope

Several considerations will be made during the development of this project. The main parties that would taken into consideration are the lower primary school students and lower primary school teachers. It will also be based on the mathematics subject syllabus that is being practiced by primary schools in Malaysia. This project is being developed for:

a) The lower primary (standard 1-2) students aged between 7 to 8 years old. It can also be used by pre-school children. It is design to make math learning more interesting and effective. It also to ensure that children are exposed to information technology at an early age.

b) Primary school teachers. It is a teaching tool that can be conducted in classes. It is an advance teaching approach that is going along with the rapid changes in the technology world.

c) Parents of primary school children. Parents will be able to give extra guidance to their children at home.

1.5 Assumptions

Several assumptions have been made in order to develop this system:

a) Access to the Internet is available as this is a web-based application which only can be retrieved online from the Internet.

b) Students and teachers have got basic knowledge in handling a computer.

1.6 Limitations

Smartmath has certain limitations as follow:

a) This system is meant for primary school or preschool children aged 7-8.

b) The targeted users of this system are children, teachers, and parents that visit this web site, which means they must have access to the Internet.

c) Guidance from the parents might be needed as there is a possibility that the children will misuse the system and surf other unnecessary sites instead.

1.7 Report Layout

Chapter 1 - Introduction

This chapter gives an overview of the system. It gives the reader an overall perception of the system proposed.

Chapter 2 - Literature Review

Literature review consists of the various researches done prior to the design and implementation of the application. Decision made on this application will be based on the research results accumulated in this chapter.

Chapter 3 - Methodology

This chapter will discuss the various modules, system properties, architecture, and decisions of the software as well as hardware intended for the development of this application.

Chapter 4 - System Design

System design describes the different designs used during the project development. It covers the Structural Design, Data Flow Diagrams (DFD), and the user interface design.

Chapter 5 - System Implementation

The overview implementation of the application will be thoroughly explained in this chapter. System implementation describes the environment, tools, coding, and development of the individual modules.

Chapter 6 - System Testing

System testing covers the techniques and methods of testing the completed application. All system must go through a series of testing before it is deployed as a fully functional application.

Chapter 7 - System Evaluation

System evaluation is the final chapter whereby the entire application is being evaluated. This chapter outlines the strengths and weakness of this application. Suggestions of future enhancements of this application will be given. Lists of problems faced during implementation and experiences gained throughout the project would also be included.

Conclusion

This is the conclusion of the whole report.

1.8 Project Schedule

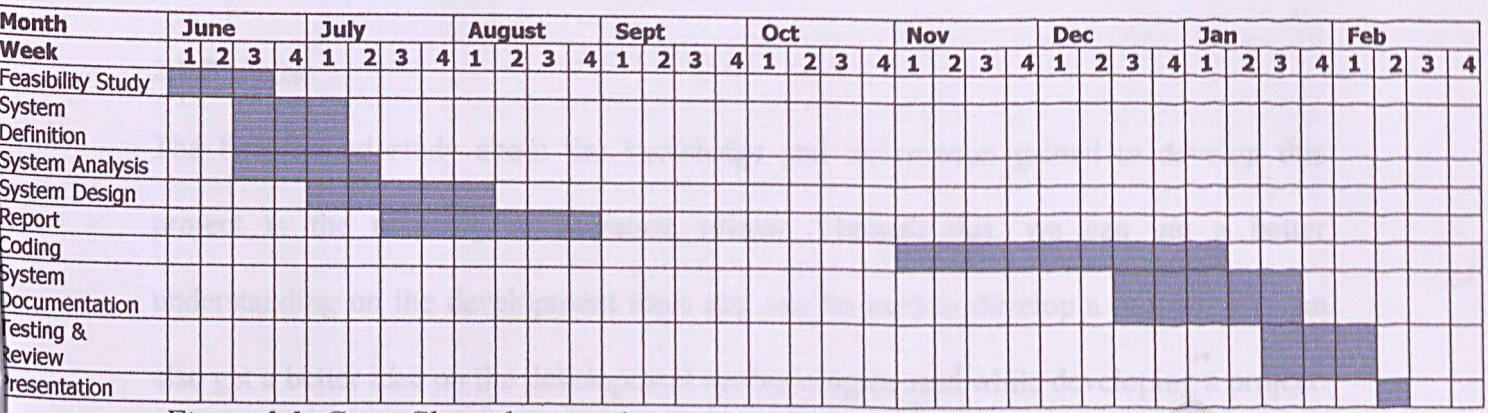


Figure 1.1: Gantt Chart showing the starting and ending for each activity in the project

Chapter 2 - Literature Review

2.1 Purpose

The background study about the knowledge and information gained to develop this project is the meaning of literature review. Through this, we can get a better understanding on the development tools that can be used to develop a project. We can also get a better idea on the development methodologies used while developing a project. Besides that, it allows the developer to study existing or past-develop projects and find out the weakness and strength of it. From there developers can find solutions to curb the weaknesses and improve the existing strength of these projects to make sure that the project that is to be developed is of high quality.

References, related article and journals, existing interactive multimedia software in the market, and existing online web-based learning application have been analyzed to gain the necessary information. These information are important to ensure that the proposed system is much better compared to the existing ones. It is very important that the same weaknesses will not be repeated in the new proposed system.

2.2 What is Web-based learning?

2.2.1 An Overview of Web-based learning

Web-based learning (a major subcomponent of the broader term "e-learning") is one of the tools with which education is delivered. In traditional academic institutions, web-based learning systems are generally housed administratively in a "distance education" department alongside other at-distance delivery methods such as correspondence, satellite broadcast, two way videoconferencing, videotape and CD-ROM/DVD delivery systems.

All such systems seek to serve learners at some distance from their learning facilitator. Many such systems attempt to serve learners interacting with the learning source at different chronological times (for example, email). Distance Education, then, is often referred to as those delivery modalities that seek to reduce the barriers of time and space to learning, thus the frequently used phrase "anytime, anywhere learning" [1].

The simplest definition of Web-based learning is the delivery of interactive training or education over the Internet or Intranet. It is the structured transfer of skill or knowledge that takes place using the World Wide Web as the distribution channel. The way this interactive learning is designed and implemented varies greatly. A full service learning community offering will likely have to support many approaches to on-line learning design and delivery.

2.2.2 Advantages of Web-based learning

a) Geographic independence - in a Web-based classroom, learning is no longer restricted to the physical buildings of the learning institutions, and the problem of overcrowding start to disappear.

b) Temporal independence - learners can study when it is convenient and when they are free. There is no longer any need for the teacher and learners to synchronize their timetables and meet at the same place and time. There is no longer any reasons for a learner to miss a class.

c) Computer-mediated - all information and communication in a Web-based classroom passes through or is stored in a Web-based classroom can be changed at any time and become available to the learners immediately.

d) Increased communication - the Web allows learners to talk with each other, individually or as a group, and to send questions or hold discussion with their instructor. Indeed, it is commonly reported that people will "talk" more electronically than they do in a face-to-face situation.

e) Learn in a convenient location and at their own pace - using internet, distance learning can take place in many convenient locations, such as office or home or anywhere with access to internet. This is means that people can take the course more conveniently. Learners can take a course during a traditional term or training sessions, or they can take their time to complete learning activities. They can go over materials whenever it is convenient for them

f) Effectiveness and retention - In an independent study done as shown in the following, students normally lost 87% of what they have learnt within a month. It is human nature to learn by repetition. As a result, Web-based learning can allow you to refresh or repeat as many times as you like until you master the material before you move on to the next topics.

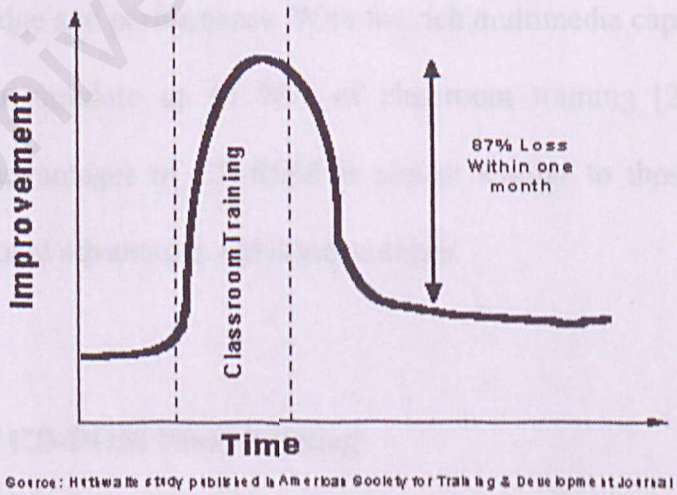


Figure 2.1: Learning improvement against time

2.2.3 Disadvantages of Web-based learning

a) Web-based distance learning has different set of cost associated with it than the traditional classroom. These cost include purchase or implementation of Web-based delivery platform. The cost associated with server support , and additional instructor time required to lead an online course.

b) Online course often only deliver information rather than foster the kind of interaction that leads to effective learning normally only delivery information but not delivery learning.

c) Web material can't identify individual student problem - although web-based learning today is that it can reach large numbers of students, it can't identify individual problems.

2.3 What is CD-ROM based learning?

It refers to the use of CD-ROM based technologies to deliver a broad array of solutions that enhance knowledge and performance. With the rich multimedia capability, CD-COM Based Learning can simulate up to 98% of classroom training [2]. Basically, the advantages and disadvantages of CD-ROM is almost similar to those of Web-based. Below are the additional advantages and disadvantages:

2.3.1 Advantages of CD-ROM based learning

a) The cost to produce a CD is inexpensive. With the wide usage of CD, the price of a CD-ROM package available in the market is very cheap.

b) Do not need a network connection. A CD drive on a normal PC is sufficient.

2.3.2 Disadvantages of CD-ROM based learning

a) Information stored in a CD is static and cannot be changed or updated. Therefore, the developer needs to be sure about the content that will be included in the CD. If changes or additional information needs to be done, a new CD must be produced.

b) No form of communication. Users cannot interact or have discussion with anybody.

2.4 What is Multimedia?

Multimedia is not a single technology. Multimedia stands for the convergence of several streams of development in the computing, audio, video, and communication industries. The term "multimedia" is often used but hard to define. As a buzzword, it is used to advertise different products such as video games, computer-based training, sales presentations, etc. Basically, multimedia can be understood as the integration of more than one medium. In this broad interpretation, current media systems of our society such as newspapers and television fall within the scope of multimedia. The integration of static media such as text, graphics, and still images are well known.

The term multimedia has become somewhat of a trendy, hyped word in the present computer age. When distilled to its root elements its meaning can be devised: *multi* ('much' or 'many') and *media* (from *medium* meaning 'means of communicating'). Using multimedia, then, is simply using a variety of media, whether visual or auditory,

with the intent of communicating. Computers have become such a pervasive force because of their unique ability to help humans organize information, and to that end, communicate more effectively. With today's inexpensive, powerful personal computers and many easy-to-use authoring languages, multimedia production has never been more accessible.

2.4.1 Typical Multimedia Application Areas

The spectrum of multimedia development tools can be further subdivided into three typical multimedia application areas:

a) Text-Based Applications
b) Interactive Applications
c) Wide-Area Applications

Depending on the application which is to be developed, what information is to be conveyed, who the audience will be, and how much interaction there will be between the application and the user, an appropriate tool can be chosen. Some of the typical multimedia applications areas, and the specific packages which would cater to each development area are discussed below.

a) Text-Based Applications

Many multimedia applications provide efficient navigation through a large resource of primarily text-based information. These applications need to be searchable so that relevant information can be found easily and quickly. Development tools which cater to

this type of application generally provide hypertext capabilities. Hypertext is similar to regular text, except that it contains information pointing to another point in an application. Microsoft Windows Help is an example of a hypertext, searching program. Some form of an overview, table of contents, or map of the information available in such an application helps the user to navigate efficiently. These applications can also often handle embedded images, sounds, and movies, which makes them true multimedia applications.

There are specific tools which provide good development environments for text-intensive applications. Microsoft's Multimedia Viewer 2.0 is a sophisticated information viewer with multimedia, hypertext, and sophisticated search capabilities. Adobe Acrobat is another text-based package which is hypertext-capable, but has limited search capabilities. Both of these packages provide an overview of the content, to guide the reader through the maze of information, and allow importing existing word processor documents. All multimedia applications are capable of storing text and moving through quantities of it, but some tools are specifically designed to work more efficiently with large volumes of it.

b) Interactive Applications

The majority of multimedia applications fall into the category of interactive, graphical applications. These tools are fully capable multimedia tools which can handle all media formats, as well as providing interactivity with the user. This is often desirable in an education setting as it provides the ability to allow specific feedback to a user, keep track of results, and customize the application to a specific user as a function of responses.

Although most tools provide these capabilities, some are better suited to complicated, interactive applications than others.

The level of sophistication of graphical, interactive applications is often related to their cost. IconAuthor 4.0.2 from AimTech Corporation, Authorware Professional 2.0.1 from Macromedia, the Apple Media Tool and Programming Environment from Apple, and Course Builder 4.0.9 from Discovery Systems are four professional-quality, sophisticated multimedia packages, but are also quite expensive. Development packages like Asymetrix Corp's Multimedia ToolBook 3.0, and Claris Corp's Hypercard 2.2 are very capable development tools which cost significantly less. The goals of the multimedia project must provide the specific criteria for choosing between several development tool alternatives. This often requires first-hand experience with the development environment to assess the tool's capabilities and example applications.

c) Wide-Area Applications

A new area of multimedia applications is emerging with the purpose of providing information to an audience over a wide geographical area. This is in part being made possible via the Internet in conjunction with new technologies such as the World Wide Web (WWW) and Mosaic. These new technologies compose an information distribution system providing services to 10-20 million people from commercial and academic organizations. Mosaic is a WWW browser, and is capable of retrieving information from all over the world via the Internet in the form of text, graphics, sounds, and movies. One of the important capabilities of the World Wide Web is its support of hypertext, which allows users to maneuver quickly from one WWW site to another with the click of a

button. There is an enormous wealth of information available on the Internet, and contributing to this body of information is, in essence, providing multimedia access to information. One of the serious drawbacks of this wide-area technology is its lack of organization. There is a tremendous amount of information available, but finding information you are interested in can be difficult. If a multimedia application is to be implemented with a geographically diverse, academic audience as its recipients, this technology is very suitable.

Information is made accessible on the World Wide Web using a mark-up language called HTML (HyperText Markup Language). This language provides the common protocol for providing rich-formatted text, embedded graphics, sounds, movies, and hypertext. More recently there has been the development of image map, and forms fill-out technology. Image mapping allows selected regions on an image to contain links which, when clicked, take the user to another document. The fill-out forms function allows user feedback through fields, buttons, and drop-down menus. This information is relayed to the originating server where it is subsequently processed. To provide documents on the Internet with these capabilities requires setting up a World Wide Web server, and composing documents in HTML. One of the tremendous benefits of this system is that a user can gather information free of charge, as long as Internet access exists. The cost to the information provider is the hardware cost of the server itself, and the time devoted to creating and updating HTML documents. This is quickly becoming the standard method for providing many types of information to a wide-area audience.

2.5 Interactive Multimedia

2.5.1 What is Interactive Multimedia?

Interactive multimedia has been called a "hybrid technology." It combines the storage and retrieval capabilities of computer database technology with advanced tools for viewing and manipulating these materials. Multimedia has a lot of different connotations, and definitions vary depending on the context. Interactive multimedia is defined by these three criteria:

- Interactive Multimedia is any package of materials that includes some combination of texts, graphics, still images, animation, video, and audio;
- These materials are packaged, integrated, and linked together in some way that offers users the ability to browse, navigate and analyze these materials through various searching and indexing features, as well as the capacity to annotate or personalize these materials;
- Interactive multimedia is always "reader-centered." In interactive multimedia, the reader controls the experience of reading the material by being able to select among multiple choices, choosing unique paths and sequences through the materials. One of the key features of interactive multimedia is the ability to navigate through material in whatever ways are most meaningful for individual users.

Interactive multimedia is synonymous with another frequently used term: hypermedia. Hypermedia is the multimedia version of the term hypertext. A hypertext is defined as any non-sequential, electronic text, assembled not as a seamless sequence of material with a beginning, middle and end, but as a web of interrelated "chunks" of text. In a

hypertext, the reader controls the sequence of reading by choosing how to navigate among these chunks of text by various electronic links.

The term hypermedia was coined to mean a hypertext that uses multiple media. In other words, hypermedia is a collection of multimedia materials with multiple possible arrangements and sequences. Hypertext and hypermedia are "electronic" concepts that can only exist in a computer-based environment. Only in a computer-based environment can materials can be linked and organized in multiple ways simultaneously, and searched, sorted and navigated in hundreds of possible combinations by different users.

2.5.2 Why use interactive multimedia?

We've seen that interactive multimedia, by definition, has the capacity to deliver large amounts of materials in multiple forms, and to deliver them in an integrated environment that allows users to control the reading and viewing experience. How then do these defining characteristics and virtues translate into benefits in an educational environment?

First of all, multimedia programs bring to education the extraordinary storage and delivery capabilities of computerized material. This is especially important for schools, libraries, and learning institutions where books are difficult to obtain and update. Multimedia is a powerful and efficient source for acquiring learning resources. Multimedia can also provide educational institutions access to other kinds of inaccessible materials, such as hard to find historical films, rare sound recordings of famous speeches, illustrations from difficult to obtain periodicals, and so on. Multimedia can put primary and secondary source materials at the fingertips of users in even the remotest locations from major research facilities.

Secondly, it is not just sheer access to these materials that makes multimedia a powerful tool, but the control over those materials that it gives to its users. Interactive multimedia programs enable the user to manipulate these materials through a wide variety of powerful linking, sorting, searching and annotating activities. Each of these activities can be made to reinforce and inculcate various intellectual skills, in addition to satisfying certain cognitive needs for quality learning, such as the ability to follow through links at the immediate moment when curiosity is aroused, and the ability to view different forms of the same information side-by-side.

Furthermore, interactive multimedia programs usually integrate some combination of orientation tools, such as timelines, graphs, glossaries, and other pedagogical guides. These kinds of tools further point to the third major benefit of multimedia: the personalization or individualization of the learning experience.

By allowing users to control the sequence and the pacing of the materials, multimedia packages facilitate greater individualization in learning, allowing students to proceed at their own pace in a tailored learning environment. Furthermore, interactive multimedia can be a powerful learning and teaching tool because it engages multiple senses. Students using multimedia are reading, seeing, hearing, and actively manipulating materials. As one educator enthusiastically put it,

As humans, we seem hard-wired for multiple inputs. Consider that we remember only about 10% of what we read; 20%, if we hear it; %30, if we can see visuals related to what we're hearing; %50, if we watch someone do something while explaining it; but almost 90%, if we do the job ourselves. In other words, interactive multimedia if properly developed and properly implemented, could revolutionize education. *(Menn, 1993)*

Although "revolutionize" may be a bit optimistic, interactive multimedia is a promising medium for reinforcing, extending, and "supplementing" what goes on in the classroom with print materials, lectures and classroom discussions.

Incorporating multimedia into the curriculum does not mean "throwing out the printed books." Most teachers who incorporate some kind of interactive multimedia into their teaching do so to enhance printed materials and the core course content.

Multimedia materials help students and teachers by way of reinforcement and extension, not substitution. What hypermedia provides is access to materials and unique personalized control over them. In other words, interactive multimedia isn't about replacing books, but about replacing the absence of books; hypermedia doesn't do what books do, but what books can t do.

2.5.3 Types of media included to create a multimedia production

Every once in awhile a new tool is developed that can have great impact on the way things are. Interactive multimedia is such a tool. Yet the elements used in multimedia have all existed before. Multimedia simply combines these elements into a powerful new tool. Interactive multimedia can weave five basic types of media into a multimedia production:

Text

Out of all of the elements, text has the most impact on the quality of the multimedia title. Generally, text provides the important information. But too much text on a page can be a detriment. Readers will soon head for the nearest exit button. About 4 or 5 sentences per screen works well. Text acts as the keystone tying all of the other media elements

together whereas a picture is only worth a thousand words. It is well written text that makes a production wonderful. Suddenly, editing becomes more than an exercise, it becomes a necessity.

Graphics

Graphics provide the most creative possibilities for a title. They can be photographs, drawings, graphs from a spreadsheet, pictures from CD-ROM, or something pulled from the Internet. (within the boundaries of the copyright law.) With a scanner, hand-drawn work can be included. In developing a multimedia system the selection of the graphic is crucial. The graphic should complement the text on the page. By searching for the best graphic a whole new sense of visual literacy can be developed.

Sound

Sound is probably the most misunderstood part of a multimedia title. Sometimes sound is being neglected because it is thought that sound is unnecessary. Sound can be used to provide emphasis or highlight a transition from one page to another. A project on India demands a few bars of northern classical sarod music. With the insertion of the sarod music in the project, it would be easier to achieve the purpose of this project.

Animations

Animations are primarily used to demonstrate an idea or illustrate a concept. Video is usually taken from life, whereas animations are based on drawings. There are two types of animation: Cel based and Object based. Cel based animation consists of multiple drawings, each one a little different from the others. When shown in rapid sequence, the drawing appear to move. Cel animation can be used to show, for example, how an

engine's crankshaft works. Object based animation (also called slide or path animation) simply moves an object across a screen. The object itself does not change. Object animation can be used to illustrate a point - imagine a battle map of Gettysburg where troop movement is represented by sliding arrows.

Video

When it comes to making an impact, video is right there at the top of the list. It takes a lot of computing power to incorporate video into a production, but it takes even more visual skill. Great ideas can be found by analyzing the videos shown on television. The video shown must captured, edited, compressed and stored. A video file is an expensive resource. It can take up a huge amount of space on a hard drive. The images must tell their story quickly, yet -completely.

When combined in a clear and organized manner, the final production - called a title - becomes interactive. A reader can choose what to investigate next. An interactive multimedia title is not linear. You don't start on page one and read to the end. An interactive multimedia title is more like a spider's web, with one idea linked to another, allowing choices in the reader's path.

2.5.4 The Components of Interactive Multimedia

Below are the separate components of Interactive Multimedia:

Computer Assisted Instruction (CAI)

Computer Based Instruction (CBI) employs computer technology to assist the instructor to instruct or the guide the learning program of individual students. Its main components

consist of CAI and CMI, each of which may employ a variety of media. Computer Assisted Instruction employs instruction modes of tutorial, review and practice and simulation.

Computer Managed Instruction (CMI)

Computer Managed Instruction includes diagnostic assessment and prescriptive study assignments.

Static Visual Displays (SVD)

These are often called visuals or graphics. Their function is to provide a non-text based representation of some object, process, concept or skill to be learned, although they are often accompanied by text. They do not move through space or time and vary in the amount of detail and realism they contain, from a simple line drawing created with a draw/paint program to a photograph.

Audio

Audio broadcasts information or data in a format which can be heard and may include instruction which is redundant with text or unique. It may also include warning sounds and sounds which are used to determine the state of things, for example a badly-tuned automobile engine.

Screen Design/Color

Screen design refers to how information is spatially organized for presentation to the learner. Some separate screen design issues are the use and placements of fonts, color and

numerous design issues such as balance, borders, and so forth. Color has been the object of numerous studies dating back to the middle of the century.

Animation (Dynamic Visual Displays)

Animations are visual images which represent motion through space or time. Like SVDs, they may vary in the amount of detail and realism they contain, from a simple line drawings through to video. They are usually created using special authoring software, such as Authorware, Director. Alternatively, they may be video sequences which have been captured in analog or digital format. The set of motion visuals that includes animation and video has been called dynamic visual displays *(Park & Hopkins, 1993).*

Multi-Channel Learning

Interactive multimedia enables simultaneous delivery of instruction via several senses or channels. It might appear that the same information, delivered through multiple channels simultaneously (e.g., visual and audio) would enhance learning in a measurable way.

Navigation

Navigation refers to the process of acquiring information from a rich multi-media database which has no obvious organizational pattern. The World Wide Web is an example of the latter. It is intuitive and attractive to believe that navigation as a learning system will result in significantly better learning than highly structured learning. To date, there is little research in this area. The conclusion which seem to be emerging is that the effectiveness of navigation cannot be assumed present for all learning situations. While research on navigation is quite new and as yet limited, it is expected to increase dramatically in the next few years.

Instructional Television

Instructional television refers to the use of televised media, whether broadcast, analog tape or videodisc or digital imagery which is employed in an instructional setting to supplement or supplant other forms of instruction. As with other forms of media, ITV originally required massive production facilities, personnel and budgets. With the advent of the transistor, microprocessors and chips, the equipment for capture and editing has become much less expensive and has migrated to the desktop. The basics of good production have remained the same, although styles have been driven by the popular media, styles such as the short sound bite, rapid flashes of imagery, and special graphics effects.

Interactivity

Except for the computer driven by a microprocessor, multimedia are by and large designed for transmission of information about content from a knowledgeable source to the student (information transmission) and is incapable of interaction. There are many definitions of interaction but they generally require that two things be able to carry out activities which elicit a response from one another. Perhaps the highest level of interactivity is a series of activities which result in the student learning or processing information at a cognitive level which is higher than rote memorization.

2.6 Multimedia in Education

In recent years, research activities on uses of computers in schools in western countries such as United States of America and Britain, in particular in K-12 education, have

increased greatly. However, the use of computers in education in Malaysian schools is not so promising. Therefore it is very important to create interactive multimedia learning modules suitable for Malaysian school children to encourage the use computers in their education. This is because this is a new learning technique which has growing popularity and efficiency. Besides, it is also to ensure that our young generation is also exposed to current technologies and not just learning the traditional way.

The notion that children learn by constructing their own knowledge is highly popular among educational theorists. Children ought to be active, not passive, in the learning process. They ought to be doing something, not merely watching it. Multimedia technologies offer children the opportunities of learning "actively" by allowing them to construct knowledge as interactive multimedia documents (e.g. multimedia stories).

Definitive research on the positive impacts of multimedia in education has not yet been assembled. While many would argue that the jury is still out, here are some important reasons to utilize this technological tool in education:

- It facilitates student-centered learning allowing choice in the pathways for learning and the rate at which new material is introduced.
- It can address several learning styles and modalities - providing a rich variety of instructional approaches, which can teach in most of the ways that students learn best.
- It motivates student interaction, experimentation, and cooperative learning.
- Students often work together on computer projects as they never did on paper-and-pencil projects.

- It facilitates "storylines" or thematic learning - where a pathway for exploration can easily be woven around a particular concept dynamics.
- It promotes the "constructivist" view of learning.

2.7 Approaches used in gathering information

A system is a collection of objects and activities, plus a description of a relationship that tie the objects and activities together. Typically, a system definition includes, for each activity, a list of inputs required, actions taken, and outputs produced. A system can be developed in different ways. Before developing a system, information about the characteristics and purpose of the system to be developed, the procedures involved to develop the system, and the methodologies used to develop the system need to be gathered. There are many sources which these valuable information can be obtained.

Each source will provide different information and facts depending on the keyword or phrases used to obtain the information. Information can be obtained from system users through survey and questionnaires, the Internet, books, reviews of existing systems, and so on. For the gathering of information to develop Smartmath, the resources includes electronic media, printed media, survey and questionnaires, guidance from the lecturer, and experience gained during industrial training.

The Internet which is the electronic media provided a lot information regarding methodologies for the system development, information about the most suitable hardware and software to use, and development tools, Besides that, because Smartmath is a web-based learning program, examples of existing program is being reviewed to specify the

necessary requirements. Various search engines were used in the process of gathering the information namely Google, Yahoo, Altavista, MSN search, and www.whatis.com. The specific keywords used for the search depends on the type of information that I'm looking for.

As for the printed media, books were used to get the details about the development models, authoring tools, and the process of capturing requirements to develop a system.

A survey was carried out to gather information from the end-users which are the primary school students and teachers. The result from the survey was analyzed and will be taken into consideration when developing the system. Besides doing a survey on school children and teachers, a survey on current multimedia mathematics learning packages was carried out too.

Besides the mentioned above, precious information were also gathered from the guidance of my lecturer, Pn. Hannyzzura Affal @ Pal and Pn Sri Devi Sudharsan, and moderator, Pn Sameem Abd Kareem. Experience when working in Asia Travel Network during my industrial training allows me to have the basic skills in developing a web site.

2.8 Findings

As mentioned earlier, all the information gathered for Smartmath can be divided into electronic media, printed media, survey and questionnaires, guidance from the lecturer, and experience gained during industrial training. Printed media comprises of books while electronic media includes sites which are found on the World Wide Web using specific keywords. Below are all the findings in detail:

2.8.1 Printed Media

a) Inside Adobe Photoshop 5.5

(Authors: Gary David Bouton & Barbara Bouton, Publisher: New Riders Publishing)

This book gave an insight of what Adobe Photoshop is all about. It gives an in-depth understanding of how to create high quality graphics. It also teaches animation, image compositing, and many more to enhance web output.

b) Software Engineering Theory and Practice

(Author: Shari Lawrence Pfleeger, Publisher: Prentice Hall)

This book explains the various types of development models. The steps models were explained thoroughly and it lists out advantages and disadvantages of each models. This book was also referred to understand the process of capturing the requirements to develop Smartmath.

c) Systems Analysis and Design *(Fourth Edition)*

(Authors: Kenneth E. Kendall, Julie E. Kendall, Publisher: Prentice Hall)

This book was referred to understand the Systems Development Life Cycle (SDLC). The information was used to prepare the third and fourth chapter of this proposal.

d) Teach Yourself HTML 4 *(Author: Cottrel Bryant, Pub: IDG Books Worldwide, Inc)*

HTML *(Authors: Thomas A. Powell, Dan Whitworth, Pub: Osborne/McGraw-Hill)*

These two books stated above were used to learn the basics of HTML *(Hypertext Markup Language)* which will be used to develop the web pages.

2.8.2 Survey and Questionnaires

Survey

A survey was done to find the existing interactive multimedia mathematics learning system existing in the market. The system available can divided into two categories which are CD-ROM based learning and web-based learning.

For CD-ROM learning, there are two systems which have been analyzed:

a) Jump Start Math for Second Graders

b) Awesome Animated Monster Maker-Math

For Web-based learning, these are examples of sites found on the Internet:

a) http://www.funbrain.com/

b) http://www.aplusmath.com/

c) http://www.kids-korner.com/

d) http://www.coolmath4kids.com

e) http://www.edu4kids.com/

f) http://www.iknowthat.com/

g) http://www.c3.lanl.gov/mega_math/

Questionnaires

To gather information from the end user of this system, a survey was conducted in one of the primary schools in Malacca (Sacred Heart Convent). A questionnaire was distributed among 35 students. The objective of this survey is to find out the popularity of web-based learning system among primary school children. It is also to find out the topics that

should be included in the system that would be developed. Here are the questions asked and the results obtained from the survey conducted:

1) What is your favourite subject?

   Mathematics (10), English (17), Bahasa Melayu (8)

2) Have you used a computer-based (CD-ROM/web-based) learning application before?

   Yes (6), No (29)

3) Do you have a computer at home?

   Yes (16), No (19)

4) Would you like to try learning online?

   Yes (35), No (0)

5) Would you prefer the system to be in English or Bahasa Melayu?

   English (28), Bahasa Melayu (7)

Charts of the relevant results.

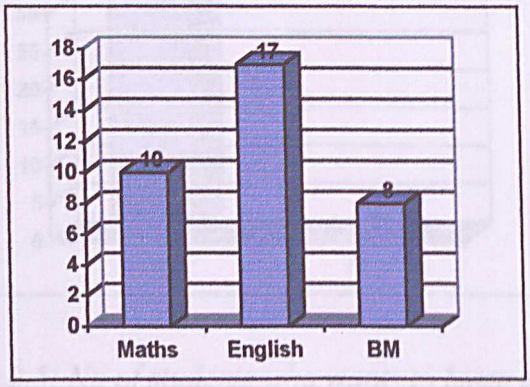


Figure 2.2: Favourite subject in school

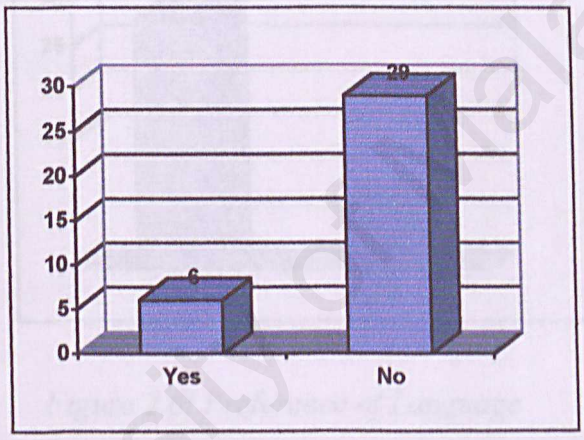


Figure 2.3: No of students used a computer-based learning application before

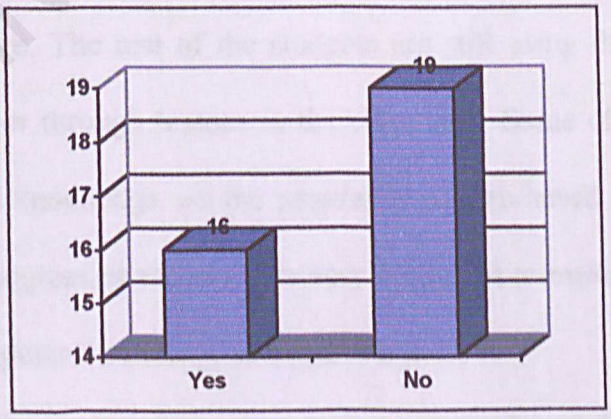


Figure 2.4: No of students with a computer at home

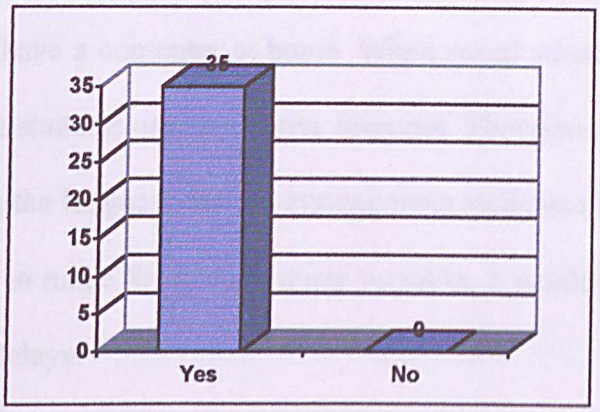


Figure 2.5: No of students who wants to learn online

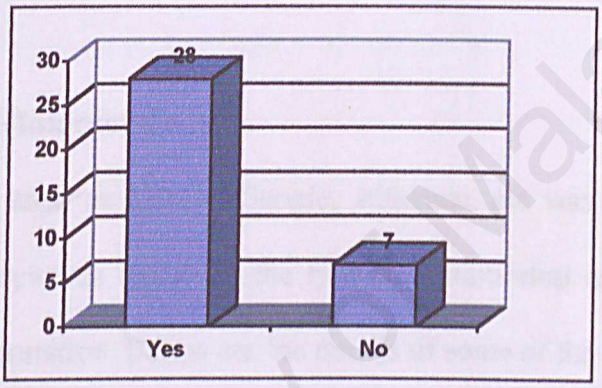


Figure 2.6: Preference of Language

From the results, I found out that students is Malaysia are still not familiar with web-based or CD-ROM based learning as out of the total, only 6 students came across these learning method before. The rest of the students are still using the traditional method which is from books or through lessons in the classroom. Some of them go for tuitions for extra practice and knowledge. As the popularity of web-based learning is increasing and the technology progressing rapidly, it is very important to make sure that our society is exposed to the computer technology at a tender age.

Mathematics is not the favourite subject among the students. This is because students think it is difficult to understand and lessons in class is not sufficient. From the

total, 16 students do have a computer at home. When asked whether they would like to learn online, all of the students gave a positive response. Therefore, Smartmath should be developed. Regarding the language for the system, most students (28) would like it to be in English. However, to make Smartmath more versatile, it would be developed in both English and Bahasa Melayu.

As for the subjects that's going to be included in the system, it's based on the syllabus that's being taught in schools. Teachers teaching lower primary were interviewed and they too agreed that topics to be included in the system should be the syllabus.

2.8.3 Electronic Media (Internet Search)

Various search engines such as Yahoo, Google, Altavista, etc was used to search for information. Specific keywords based on the type of information needed was used to search the necessary information. Below are the details of some of the sites visited:

2.8.3.1 Sites on Web-based learning and Interactive Multimedia

a) http://www.georgetown.edu/crossroads/mltmedia.html

This site provides all the information about Interactive Multimedia. As Smartmath is an interactive multimedia package, this site is very useful in defining interactive multimedia, why is it used, and the types of multimedia available. It also explains how multimedia can be incorporated into education contexts and the technical requirements for using multimedia.

b) http://www.outreach.utk.edu/weblearning

This site defines web-based learning. It gives descriptions about software tools that are suitable to develop web-based courses and links to existing online courses offered. Articles regarding web-based can also be found here. Issues that should be considered for the development of web-based courses and the explanation of each one are listed here too.

2.8.3.2 Sites on web designing

a) http://tutorials.beginners.co.uk/view/cobrand/searchmiddleware/i/6

This site provide useful tutorials on web developing authoring tools such a Macromedia Flash, Macromedia Dreamweaver, Microsoft FrontPage, and so on. It also provide tutorials on JavaScript, HTML, ASP, Visual Basic, XML, the Internet, networking, web development, web marketing and so on.

b) http://snow.utoronto.ca/Learn2/design.html

This site provides information about web-based instructional design. It provides information about choosing an overall site structure, models of site structure, defining general page layout, preparing course content and resources, developing a communication strategy, developing resources to support learners and much more that's very important in developing a web-based learning system.

c) http://www.lmu.ac.uk/lss/staffsup/desmeth.htm

This site provides a collection of resources to help generate on-line learning materials and references to be installed in the proposed system. Its resources are divided into three categories which are design and delivery, reference tools and guides, and graphics and multimedia. The resources provided include design and methodology for a web-based learning environment, authoring tools, templates and examples, JavaScript basics, usage of sound and videos, etc.

d) http://www.med.monash.edu.au/informatics/techme/authorsoft.htm

This site provides information on what is an authoring tool and considerations to be made when choosing a software. It also lists a number of websites that discusses the issue of selecting the right software to develop a multimedia system. It gives detail explanation regarding certain authoring software such as Authorware, IconAuthor, Toolbook, and Macromedia Director.

e) http://scis.nova.edu/~henkeh/story1.htm

This site is an article about evaluating a Web-based Instruction Design (WBI). It gives the explanation of WBI, describes the importance of WBI, define the design issues, methodology design, top ten web design mistakes, and interface design for a computer-based learning environment.

f) http://www.macromedia.com

This site was surfed to get information about design and development tools such as Dreamweaver 4, Flash 5, Fireworks 4, Authorware, etc.

g) http://www.adobe.com

This site provides the information about web design tools such as Adobe WebCollection, Adobe GoLive, Adobe LiveMotion, Adobe Illustrator, Adobe Photoshop, etc. Adobe AfterEffects, Adobe Premiere, and Adobe Streaming Media Collection are examples of multimedia tools.

2.8.3.3 Sites on development process

a) http://www.med.monash.edu.au/informatics/techme/whyuse.htm

This site gives explanation regarding the phases of development in a development model. It explains in detail two types of development model namely the waterfall model and prototyping.

2.8.4 Guidance from the lecturer

Discussion with the lecturer was done along the process of preparing this proposal to ensure that the content of the system proposed is accurate and relevant. Helpful tips on how to gather the information were given by the lecturer. The lecturer also gave opinions and advice regarding the system design of Smartmath.

2.8.5 Experience gained while working in Asia Travel Network

As a trainee in Asia Travel Network, I gained precious experience on how to develop web sites as it is an online travel reservation company (http://www.asiatravelmart.com). There, I learned the basics of creating web sites and techniques to optimize web graphics to ensure that the download time is reasonable.

2.9 Relational Development Models

There many types of development models. Some are *prescriptions* for the way a system development should progress, and others are *descriptions* of the way system development is done in actuality. In theory, the two kinds of models should be similar or the same, but in practice, they are not. Building a process model and discussing its subprocesses helps the team understand this gap between what should be and what is [b].

2.9.1 Waterfall Model

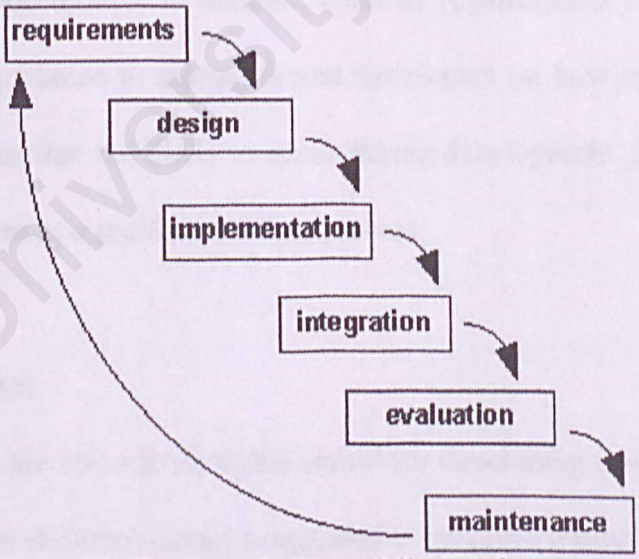


Figure 2.7: Waterfall Model

In a waterfall model, the stages are depicted as cascading from one to another. As the figure implies, one development stage should be completed before the next begins. Thus,

when all of the requirements are elicited, analyzed for completeness and consistency, and documented in a requirements document, system design activities will be carried out. The waterfall model presents a very high-level view of what goes on during development, and it suggests to developers the sequence of events they should expect to encounter.

The waterfall model can be very useful in helping developers lay out what they need to do. Its simplicity makes it easy to explain to customers who are not familiar with the system's development; it makes explicit which intermediate products are necessary in order to begin the next stage of development. Many other, more complex models are really just embellishments of the waterfall, incorporating feedback loops and extra activities.

However, there are two major drawbacks concerning the waterfall model. Firstly, it shows how each major phase of development terminates in the production of some artifact (such as requirements, design, or code) and there is no insight into how each activity transforms one artifact to another, such as requirements to design. Thus, the model provides no guidance to managers and developers on how to handle changes to products and activities that are likely to occur during development. Secondly, the model fails to treat the system as a problem-solving process.

2.9.2 Prototype Model

Prototyping methods are considered highly useful for developing educational technology. There are a number of different names being used to describe similar design/development methods, including prototyping, rapid application development, rapid prototyping and so on. There are two main categories of prototyping technique, as outlined below.

a) Rapid prototyping

Rapid prototyping is used to discover flaws in a design in a short amount of time. The initial design is tested and corrected then tested and corrected again and so on, until a certain level of satisfaction is achieved. Sometimes prototypes are in a much simpler form than the end product- for example paper can be used to prototype a screen design. Other names for this technique include rapid application development. The emphasis is on quick, fast, iterative design.

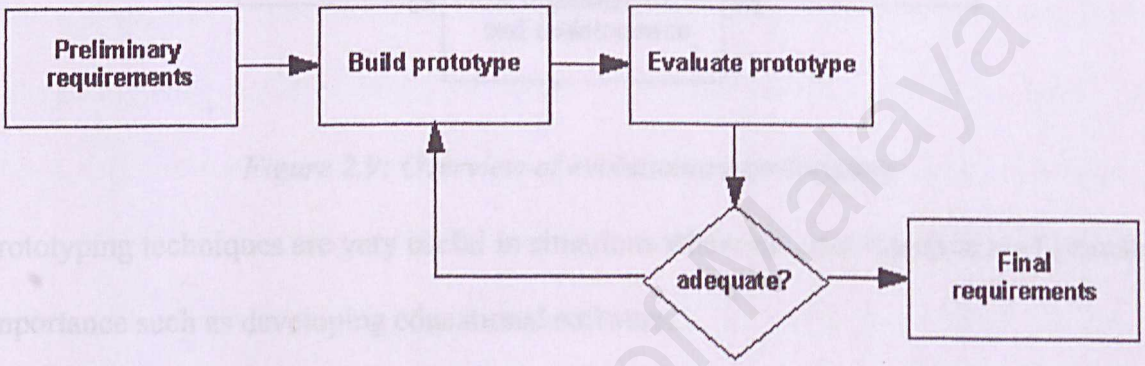


Figure 2.8: Overview of Rapid Prototyping

b) Evolutionary prototyping

Evolutionary prototyping or software prototyping can use rapid techniques, but the emphasis is more on creating a prototype in software, that will (not necessarily rapidly) form the basis of the final product. In a strict sense, once a satisfactory prototype has been created, the project continues on to a more 'waterfall' like method of development. In reality, the 'stricter' software engineering path is rarely followed when creating interactive multimedia.

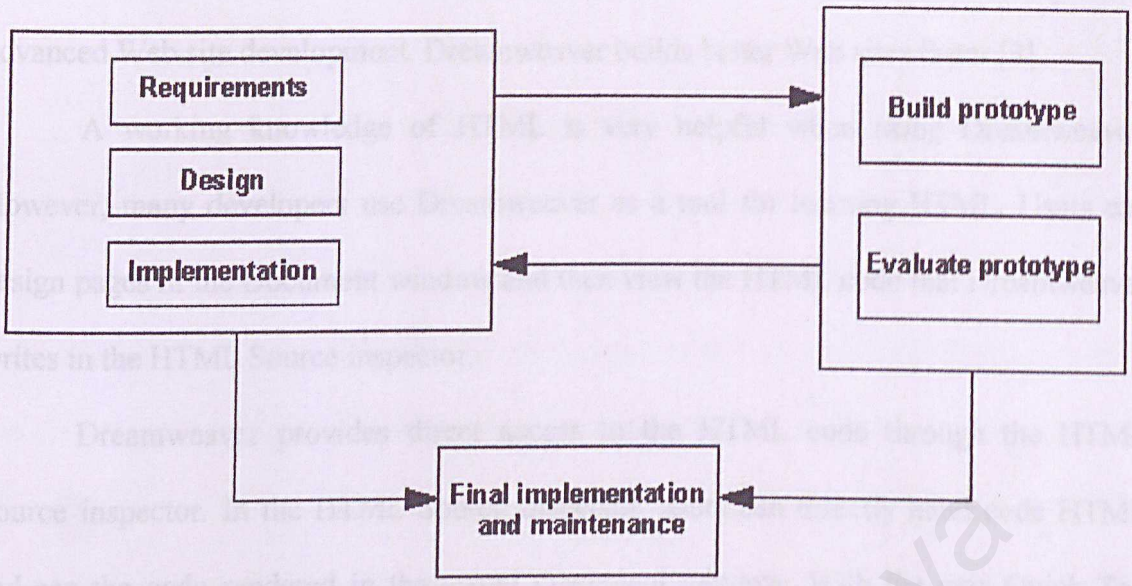


Figure 2.9: Overview of evolutionary prototyping

Prototyping techniques are very useful in situations where the user interface is of primary importance such as developing educational software.

There are problems with prototyping methods. At some point the prototyping has to stop, and the project continue. It is important the iterations be managed appropriately, and not continue on into actual development, where correcting mistakes is difficult and time consuming.

2.10 Relational Authoring Tools and Programming Languages

2.10.1 Macromedia Dreamweaver

Dreamweaver is the professional visual design solution for creating groundbreaking Web sites. Dreamweaver's powerful features allow users to automate production and enhance team efficiency. Dreamweaver facilitates workflow through integration with other Web applications, Microsoft Office, and leading e-commerce and application servers.

Moreover, Dreamweaver can be customized using HTML, JavaScript, and XML for advanced Web site development. Dreamweaver builds better Web sites faster [3].

A working knowledge of HTML is very helpful when using Dreamweaver. However, many developers use Dreamweaver as a tool for learning HTML. Users can design pages in the Document window and then view the HTML code that Dreamweaver writes in the HTML Source inspector.

Dreamweaver provides direct access to the HTML code through the HTML Source inspector. In the HTML Source inspector, users can directly hand code HTML and see the code rendered in the visual Document window. With the new Quick Tag Editor, users have direct access to the code without leaving the visual Document window.

Dreamweaver writes a subset of HTML 4.0 (for DHTML) and HTML 3.2 for maximum compatibility. Dreamweaver writes JavaScript which ranges from 1 to 1.2, using the most widely compatible code. Dreamweaver Templates allow developers to better manage the overall design of their sites by separating page content from page design. By defining editable areas on a page, developers can restrict changes that can be made to the layout of a particular HTML page. Content contributors can then add and edit content in these editable regions without compromising the site's design. Moreover, changes can be made to the overall design of the site quickly by revising the template file directly.

Dreamweaver Library items allow users to save sections of HTML code from an existing page for later use. These Library items can be added to pages in the site from the Library palette with drag and drop ease. Libraries also make it easier than ever to update

code across many pages, as edits to a Library item will be reflected in all pages that reference it throughout the site.

Authoring pages in Latin-based languages is possible providing that the fonts of that language are available on the system. Dreamweaver 3 on the Macintosh can create pages that use double-byte fonts, such as Japanese and Chinese characters, when the appropriate language kit is installed on the system. Dreamweaver on a Windows system can create pages that use double-byte fonts, provided the user is working on the localized operating system for the desired language. On a Windows system, the user must also have the appropriate localized version of Dreamweaver.

While Dreamweaver is primarily an HTML authoring tool, it allows for editing pages with non-HTML markup languages. Support for ASP, JSP, and CFML is included with built-in data translators. Extended support can be added for editing any non-HTML markup language through extensibility. Dreamweaver permits disabling automatic HTML correction when working on pages which contain non-HTML markup language. Users may wish to consider Dreamweaver UltraDev when working with ASP, JSP, or CFML.

2.10.2 Microsoft Agent Scripting Helper (MASH)

Microsoft® Agent is a set of programmable software services that supports the presentation of interactive animated characters within the Microsoft Windows® interface. Developers can use characters as interactive assistants to introduce, guide, entertain, or otherwise enhance their Web pages or applications in addition to the conventional use of windows, menus, and controls. Microsoft Agent enables software developers and Web

authors to incorporate a new form of user interaction, known as conversational interfaces, that leverages natural aspects of human social communication. In addition to mouse and keyboard input, Microsoft Agent includes optional support for speech recognition so applications can respond to voice commands. Characters can respond using synthesized speech, recorded audio, or text in a cartoon word balloon. The conversational interface approach facilitated by the Microsoft Agent services does not replace conventional graphical user interface (GUI) design. Instead, character interaction can be easily blended with the conventional interface components such as windows, menus, and controls to extend and enhance your application's interface.

MASH is an easy-to-use program that lets you to compose and playback entertaining Microsoft Agent presentations. You can specify what you want them to say and do. MASH is used for these inspiring technologies such as talking websites, interactive presentations, self-running kiosks/demos, tutorials, tour guides, clipboard and text file readers, and just plain fun. It can be used to create presentations for Web Sites, Email Messages, Visual Basic and Office Applications, Windows Scripting Host, or MASH's own Desktop Scripts and Executable programs [4].

2.10.3 HTML

Hyper-Text Mark-up Language (or HTML for short) is the language used to specify the construction of Web pages. Web pages are a form of HyperText and include text, graphics and links to other HTML documents.

Web pages are stored as standard ASCII (American Standard Code for Information Interchange) files. Web pages may be viewed by a variety of different Web browsing tools, each of which may have different abilities. However, since Web pages are text files, each Web browser can read it and format the document in accordance with its abilities.

HTML is a standard which enables you to request a Web browser to format and display your Web page in a particular way. HTML allows you to mark areas of your document that will become for example: titles, new paragraphs or italic text. Since the Web page is specified as an ASCII file the codes or "elements" (as they are known) have also to be ASCII.

The elements can broadly be divided into two main categories: those that describe the format of the Web document, i.e. what it looks like, and those that define information about the document, i.e. its title [5].

2.10.4 JavaScript

JavaScript is a general-purpose programming language designed to let programmers of all skill levels control the behavior of software objects. The language is used most widely today in Web browsers whose software objects tend to represent a variety of HTML elements in a document and the document itself. But the language can be and is used with other kinds of objects in other environments. For example, Adobe Acrobat Forms uses JavaScript as its underlying scripting language to glue together objects that are unique to the forms generated by Adobe Acrobat. Therefore, it is important to distinguish JavaScript, the language, from the objects it can communicate

with in any particular environment. When used for Web documents, the scripts go directly inside the HTML documents and are downloaded to the browser with the rest of the HTML tags and content.

How is JavaScript different from Java?

JavaScript was developed by Brendan Eich of Netscape; Java was developed at Sun Microsystems. While the two languages share some common syntax, they were developed independently of each other and for different audiences. Java is a full-fledged programming language tailored for network computing; it includes hundreds of its own objects, including objects for creating user interfaces that appear in Java applets (in Web browsers) or standalone Java applications. In contrast, JavaScript relies on whatever environment it's operating in for the user interface, such as a Web document's form elements.

JavaScript was initially called LiveScript at Netscape while it was under development. A licensing deal between Netscape and Sun at the last minute let Netscape plug the "Java" name into the name of its scripting language. Programmers use entirely different tools for Java and JavaScript. It is also not uncommon for a programmer of one language to be ignorant of the other. The two languages don't rely on each other and are intended for different purposes. In some ways, the "Java" name on JavaScript has confused the world's understanding of the differences between the two. On the other hand, JavaScript is much easier to learn than Java and can offer a gentle introduction for newcomers who want to graduate to Java and the kinds of applications you can develop with it.

2.10.5 Adobe Photoshop

As the industry standard for digital image manipulation software, Adobe Photoshop has revolutionized the photography and prepress industries and has provided commercial and fine artists with an exciting new medium for photographic editing. Adobe has integrated into Photoshop a design based upon traditional photo manipulation technique, where tools and processes directly correspond with those used in `physical' photography. Photoshop introduces features and enhancements which go far beyond the capabilities of the darkroom technician, thanks to digital technology; yet through an interface based on traditional technique, Adobe ensures a relevant, familiar, but powerful program environment.

Pixels

With all the talk of bitmapping and rasterization, it should be somewhat clear now that the primary unit in Photoshop is the pixel, where each pixel represents a unit of color information (hue, saturation and brightness). With the exception of **floating selections**, all changes in Photoshop occur at a two-dimensional level; dragging the paintbrush across a picture causes the replacement of those pixels in its path with new ones. The only way in which to recover the replaced pixels at this point is to choose the Undo or Revert commands or by using the Rubber Stamp/Eraser tools with their From Saved options.

Selections

Selections in Photoshop work similarly to the same function in most other Mac applications, where selections can be copied, cut, pasted, and cleared. While changes can be made to a picture as a whole, selecting a certain area first restricts changes inflicted

through a filter or other editing command to the selected area only. The rest of the picture is protected. Three tools handle the task of selecting: the Marquee tool, which enables the simple box/rectangle method of selection; the Lasso tool, which enables the user to section off portions of the picture in a free-hand manner; and the Magic Wand tool, which uses the intelligence of the computer to make selections based on pixel color similarity, the parameters of which are defined by the user.

Once a selection is created, it can be dragged to another place in the picture or layer, Option-dragged (copying and dragging the copied selection), copied and pasted to another picture altogether, edited, deleted-you name it. The important thing to remember, however, is that once deselected, the object replaces those pixels beneath it; they are forever lost, save through the revert procedures mentioned above.

Layers

The closest Photoshop comes to the versatility afforded object-based software is through the use of layers, which are not unlike cells used in the physical graphics industry. An indefinite number of layers can be created, accessed, stacked, or deleted using the *Layers* palette, available under the Windows/Palette menu. Each picture automatically contains a background layer; the currently highlighted layer on the Layers palette is considered the *target layer*, which is the only editable layer at any given time. Areas on a layer containing image data contain pixels, which are opaque; areas *without* image data (and thus pixels) are transparent, so that underlying layers show through. The advantage to this feature is the ability it gives the artist to designate and seperate components of the composite picture, and then edit them individually without effecting the other layers.

Channels and Modes

Each Photoshop document is considered in terms of one or more channels depending upon the mode; in RGB mode, there are three color channels (one each for Red, Blue and Green) and a fourth composite channel. If the user selects any one of these color channels from the **Channels** palette, the document window reflects a greyed-out version of the image; note that different areas of grey are more/less intense depending on which color channel is selected. These gray images reflect intensity values of that particular basic color in any area of the image: in the case of *additive* color, lighter areas reflect the presence of that isolated color (as in the RGB color model); for *subtractive* colors, the opposite is true, where lighter areas in the isolated channel indicate an absence of the channel's color and darker areas.

Color modes, as discussed earlier, are the various ways in which image data can be considered by Photoshop; the three most common examples are RGB color (consisting of three channels, Red, Blue, and Green), CMYK color (Cyan, Magenta, Yellow, and Black), and Grayscale (a single channel of Black). Modes and channels are in some ways the most complex elements of Photoshop. The more complex the color/image data (i.e., the more channels a picture is working with), the greater the file size, as Photoshop interprets all data in terms of its channels. Thus, a picture saved in the RGB mode will be *three times* the size of a Grayscale image; likewise, a CMYK image will be four times as large.

2.11 Evaluation on Existing Systems

2.11.1 Web-based learning

a) http://www.funbrain.com

This site is game-based learning system. Overall, this system is divided into three main parts which are:

- kids & games
- teachers
- parents

Kids & games:

For this section, all the games that can be played is listed. The games are based certain categories such as numbers, words, universe, culture, etc. The games here tests the kid's knowledge in mathematics, science, geography, sports stars, history, grammar, memory, IQ, puzzles, counting money, telling time, etc. Users can search for games based age or categories. Users can also write comments and it will be stored in the writers' bloc which can be viewed by teachers or parents.

Teachers:

For this section, teachers are allowed to design their own quizzes to be included in the quiz lab. First of all, they need to register themselves as a member. Once registered, they can design their own quizzes and have access to the other quizzes prepared by teachers that's already there. The quizzes prepared by the teachers can be sent to their students via email. Once the quiz is being answered, the quiz lab grades the quizzers and email the results to the teachers.

Parents:

As for this section, parents are given opportunity to participate in the children's education. Here, parents can look for the type of games that they think would be suitable for their children. Besides that, certain games allows two players which means parents are able to play the games with their children in the 'Parent-Kid Challenges' category. There are also some quizzes meant for parents that enable them to understand their children more. The quizzes are based on school age children, preschoolers, toddlers, and infant behavior.

Analysis and comments:

Advantages:

- many types of games available.
- simple and easy to use.
- covers various subjects and topics for a wide age group.

Disadvantages:

- no tutorials are provided. Users are expected to have the knowledge beforehand to play the games available.
- due to wide range of subjects available, it doesn't cover mathematics in detail.

b) http://www.aplusmath.com/

This site focuses on mathematics. There are four categories namely flashcards, games, homework helper, and worksheets. The layout of this site is very simple. It has students

record that displays the scores of the games. There is homework helper section whereby students can type in questions and the answer will be displayed.

Advantages:

- easy to navigate/simple instructions
- students record
- homework helper

Disadvantages:

- interface not attractive
- homework helper (students might take advantage of this tool and not try out their homework themselves first)

2.11.2 CD-ROM based learning

a) Jump Start Math for Second Graders

This is a CD-ROM learning package that is very popular in the current market. It is a game-based program whereby the user answers mathematic questions in an adventure journey. The main attraction of this software is the colourful graphics and interesting animation. However, the interface which is too graphical is rather confusing because it is difficult to navigate the system.

Advantages:

- very attractive interface and animation
- focused topics
- provides progress report and printable workbook

Disadvantages:

- instructions not clear
- must complete the game

b) Awesome Animated Monster Maker Math

This is also a CD-ROM learning package. It focuses on mathematic operations.

Advantages:

- can customize practice with parental control
- automatic leveling responds to the user's performance
- user's manual provided

Disadvantages:

- instructions clear / difficult to navigate
- graphics not very pleasant looking

Chapter 3 - Methodology

3.1 System Analysis

The definition of system analysis is a systematic investigation of a real or planned system to determine the functions of the system and how they relate to each other and to any other system. Depending on the context and the constraints of the package, developers should be able to clarify the interactive multimedia project rationale, to define the program scope, and to set up methods of evaluation by creating a preliminary analysis [3].

Typical constraints in the design of an interactive multimedia application include:

- media configuration and performance, e.g. developers might have multi-platform development strategy aimed at producing a good multimedia application
- for the publishing market, or a very tailored approach for a specific purpose – a teaching and learning tool
- the availability of expertise about the subject,
- the accessibility of related multimedia documentation,
- the budget and the deadline.

The look and feel, interface and functionality of existing interactive multimedia applications should be evaluated. It often helps to make a chart of comparative features of existing applications. Looking at these applications will reveal the puzzle that multimedia design and production represents. Production is always governed by the delivery requirements, hardware limitations, storage capacities, and the speed of the programs that present the information. All the existing products analyzed will demonstrate the trade-offs the developers had to deal with in order to bring the project to the perceived market.

The processor speed, the hard disk storage and access, and memory limitations have all been juggled to create the best application for the investment.

3.2 Modeling the Process and Life Cycle

3.2.1 What is a process?

A process is a series of steps involving activities, constraints, and resources that produce an intended output of some kind [b]. A process usually involves a set of tools and techniques and has the following characteristics:

- the process prescribes all of the major process activities.
- the process uses resources, subject to a set of constraints (such as a schedule), and produces intermediate and final product.
- the process may be composed of sub-processes that are linked in some way. The process may be defined as a hierarchy of processes, organized so that each sub-process has its own process model.
- each process activity has entry and exit criteria, so that we know when the activity begins and ends.
- the activities are organized in a sequence, so that it is clear when one activity is performed relative to the other activities.
- every process has a set of guiding principles that explain the goals of each activity.
- constraints or controls may apply to an activity, resource, or product.

When the process involves the building of some product, it is referred as a life cycle. A life cycle usually involves the following stages:

- requirements analysis and definition
- system design
- program design
- writing the programs (program implementation)
- unit testing
- integration testing
- system testing
- system delivery
- maintenance

Below are the reasons for modeling a process:

- it forms a common understanding of the activities, resources, and constraints involve in a system development.
- helps to find inconsistencies, redundancies, and omissions in the process and in its constituent parts. As these problems are noted and corrected, the process becomes more effective and focused on building the final product.
- the model should reflect the goals of development, such as building high-quality system, finding faults early in development, and meeting required budget and schedule constraints.

Every development model includes system requirements as input and a delivered product as output.

3.2.2 Proposed development model for Smartmath

To develop Smartmath, the model chosen is a combination of the waterfall model and the prototype model.

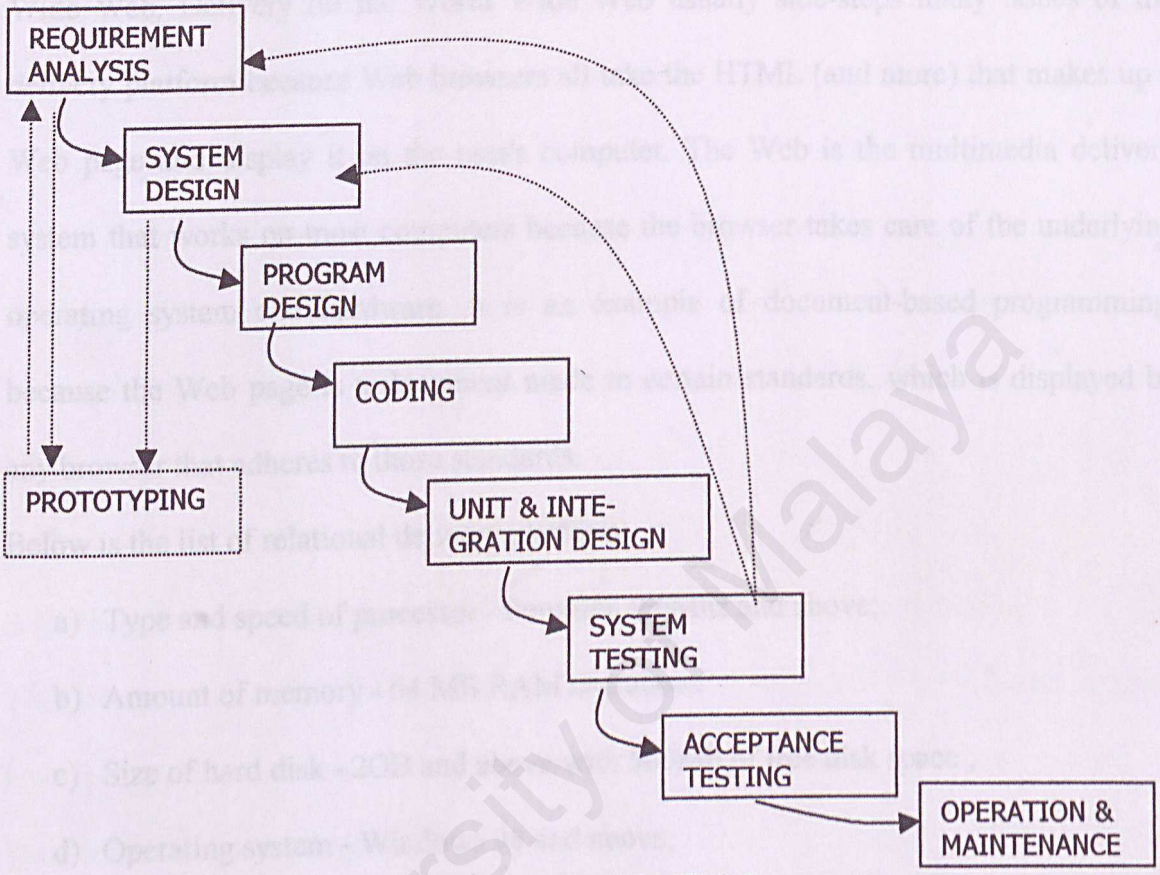


Figure 3.1: Waterfall Model With Prototyping

In a waterfall model, each development stage has to be completed before proceeding to the next. For example, in the first stage all the requirements are elicited, analyzed and documented before designing the system. Overall, this model is a well documented process. With the addition of the prototype model as the sub-process, certain aspects of the system can be reviewed and tested to check its functionality and whether it meets the specific requirements. The means that process can be tailored to meet the specific requirement yet possibly changing needs of any application. This way, major problems can be avoided as errors can be detected at an early stage.

3.3 Delivery Platform and Medium

3.3.1 Delivery platform

As Smartmath is a web-based application, the delivery platform would be the World Wide Web. Delivery on the World Wide Web usually side-steps many issues of the delivery platform because Web browsers all take the HTML (and more) that makes up a Web page and display it on the user's computer. The Web is the multimedia delivery system that works on most computers because the browser takes care of the underlying operating system and hardware. It is an example of document-based programming, because the Web page is a document made to certain standards, which is displayed by any browser that adheres to those standards.

Below is the list of relational delivery platform:

a) Type and speed of processor - Pentium 366Mhz and above;

b) Amount of memory - 64 MB RAM and above

c) Size of hard disk - 2GB and above with 500Mb of free disk space ;

d) Operating system - Windows 98 and above;

e) Access to online systems (local network, Internet, World Wide Web, and so on);

f) Speed of network connection - 56K and above;

g) Resolution of the screen - 800 by 600 pixels;

h) Number of colours on the screen - 256 and above;

i) Sound handling - 8-bit and above.

3.3.2 Delivery medium

Online delivery would be the delivery medium for Smartmath. There are four main issues namely speed of access, distance, updating, 'unlimited' size of the data space that can be provided online.

The speed with which Smartmath's potential users can access the application is unpredictable because it depends on the access speed of the Internet.

Online delivery allows people from all over the world to access the application anytime. All they need is a connection to the Internet. This method overcomes the barrier of time and space. It also provides huge data space that can potentially be presented to the user.

3.3.3 Why Smartmath is made Web-based and not a CD-ROM application

The advantages and disadvantages of Web-based learning and CD-ROM based learning have been discussed earlier in Chapter2. Web-based learning is chosen to develop Smartmath. These are the reasons why:

- Smartmath is a learning and teaching application. Therefore, there is a possibility that the information on the site and stored in the database be updated. Updates on web pages can be done easily whereas information on a CD is static and cannot be changed.
- A web-based application is stored in a server that is connected to a network locally or globally, this allows it be accessed anywhere without the geographical limitations. This has been proven by distance learning programs, which has been introduced by institutions globally. CD-ROM application can only be used if the

users can get hold of the particular CD. This means the usage of the application depends on the distribution of the CD.

- With web-based learning, multiple users can access it simultaneously whereby the teacher and the students' can experienced the interactive learning and teaching environment. One CD can only be used by one user at a time unless the application is saved into the PC which usually takes up a lot of disk space.

3.4 Proposed Development Tools

Below are the tools that will be used for the development of Smartmath. The advantages stated are the reasons why these tools are chosen.

3.4.1 Authoring Tools

3.4.1.1 Macromedia Dreamweaver

Advantages of using Dreamweaver

These are the qualities that makes Dreamweaver different from other high-end Web site tools:

- Roundtrip HTML(tm)
- Customising options and page layout power
- Cross-browser compatibility
- Site management features.

Roundtrip HTML

Most traditional Web page tools are either tag based--good control, but difficult to use--or have a WYSIWYG interface. Tools with a WYSIWYG interface usually create their own

"brand" of HTML that doesn't always conform to the World Wide Web Consortium's HTML standards. So WYSIWYG tools, which shield users from hand editing tags, are convenient and quick to use. HTML purists don't like the HTML these tools produce. Another problem, if more than one editor is used or the pages are worked on by several people using different editors, the source from one falls to bits or is reformatted when it is opened with another. Dreamweaver's Roundtrip HTML(tm) preserves the formatting you set, no matter which source editor you use.

Customising options and page layout power

With Dreamweaver, many things can be customized such as the look of your workspace, your choice of HTML source editor, even the way your HTML source is displayed.

Launcher - Launcher can be used to open and close Dreamweaver's pallets and inspectors. There's a mini launcher at the bottom right edge of the document window. It has the same icons as the launcher, but no labels. Once you get used to the icons, you can dispense with the launcher, freeing valuable desktop workspace for other items.

Object Pallet - This pallet reproduces selections within the Insert menu. It is used to insert page elements. The default groupings for page elements are, Common, Forms and Invisibles. Groupings can be customized in almost any way, from which elements appear in which group to which order groups or elements within a group appear. New objects can be created and added to any group. Objects can be almost any HTML element, such as images, tables, layers, rules, Applets, plugins, forms, scripts, comments, to snippets of code. Once a new object is created and added to the object pallet, it can be added to any page.

Property Inspector - The context sensitive Property Inspector switches dynamically to display the properties of the HTML element that's being worked on. Although it might not be immediately obvious, this is an extremely useful feature. When the user is working on an image, the Image Property Inspector is displayed. If the user want to edit an image, it can be done visually, using the mouse to select and drag the image dimensions. Or change image dimensions by entering new height and/or width values in the inspector window. When the user is working on text, they can change the font or style by selecting it with the mouse and applying the code they want, they can use the HTML Inspector to edit tags, or call up a preferred external editor with a single click, and change the tags there. Whichever way the change is being made, the document is updated automatically in the other views.

Cross-browser compatibility

There are only two main protagonists in the browser wars, but the range of versions, each with its own feature set, makes designing for cross-browser compatibility a nightmare. Dreamweaver has several features to make this essential task easier.

A target browser, or browsers can be selected. Check a document against the target(s) and a list of tags and/or attributes that are not supported by the target browser(s).will be displayed. Predefined profiles for Netscape Navigator 2.0, 3.0, and 4.0, and Microsoft Internet Explorer 2.0, 3.0, and 4.0, or any combination of these. can be checked

Site management features

Creating a single page is one thing, maintaining a site is another. There are links to check (within pages, within sites and across sites), HTML to tweak, and styles to update. Good

sites are internally consistent, consistency helps visitors orient themselves within the site. Changing the way lists are displayed or changing the navigation bar on every page is a pain. Dreamweaver's Target Browser Check and CSE HTML Validator (bundled with Dreamweaver) help you control and manage your HTML. The Check Link Feature (from the File Menu) checks links for a single document or the entire site.

Library Elements - Library Elements can be bullets, backgrounds, logos, addresses, etc, and all their attributes. The elements are dragged or inserted into the page. Within Dreamweaver, Library Elements are displayed differently from ordinary page elements, but they appear normally in the browser. If a library item is changed, a decision decide whether to have Dreamweaver update all pages right then or later must be made When the site is updated all references to the library item are changed in all pages that contain the item. So it's easy to maintain consistency across the site even when a common style is changed, frequently used e-mail address, or any page element that's being set up as a Library Element.

An individual instance of a Library Element can also be edited. Users will first be reminded that it is a Library Element and asked they want to edit the element or this instance. When editing a single instance, only that instance changes. That item will no longer be linked for automatic updates when the library element is changed.

File Check In/Out - It is useful to be able to identify which files have been checked in/out and by whom, especially if there are problems with data loss and version control, two common hazards when more than one person works on a site.

Layers - If the design is for version 4 browsers, layers can be used to position graphics, text and other HTML objects at specific pixel coordinates. Layer's properties can also be modified--including its size, shape, position, visibility, colour and position within a group of layers.

Style Sheets - Styles within a page can be set and controlled--using individual custom styles or style sheets--or across pages (even an entire site) using Library Elements or style sheets. Style sheets can be individual internal styles, placed within the <Head> element of each page, or stand-alone. Stand-alone style sheets are external to individual pages, but linked to the pages to which they apply. Styles are created with the Style Definition Menu and applied to any text in a document. Browsers that don't support styles simply ignore style tags.

JavaScript behaviours - A number of preset behaviours, which include events, for example onClick, onMouseOut, and onMouseOver , and actions, for example, show layer, hide layer, and play sound are built-in to Dreamweaver. Behaviour can be applied to a link, image, form element, layer, almost any HTML element. Just select the element, open the palette, select from the list of events, then select the action desired. Personal customised behaviours can be created and customised if scripting and which events and actions are allowed for different HTML elements and browser versions is understood.

Animation - Very impressive animations can be created by manipulating and combining layers with JavaScript behaviours along a time line. This is one example of what is called Dynamic HTML, or DHTML. Dreamweaver's time-line interface makes it easy to manipulate layers and behaviours, along a time line or time curve. Images can be

swapped, sounds added, layer properties changed, even go to another URL within the animation.

3.4.1.2 Microsoft Agent Scripting Helper (MASH)

MASH is used to generate the talking wizard. The output of the presentation can be of many types of formats including VBScript for HTML, JavaScript for HTML, Email Stationery, Visual Basic, VBA for Office documents, and Windows Scripting Host.

As Smartmath is built with HTML and JavaScript, I chose JavaScript for HTML as the output. From the codes generated, I did some modification to increase the capability of the wizard. Control statements are included to make the wizard more responsive.

3.4.2 Programming Languages

3.4.2.1 HTML

As Smartmath is a web-based system, HTML needs to be used because it is the language used to specify the construction of Web pages.

3.4.2.2 JavaScript

Advantages of using JavaScript

JavaScript's greatest potential gift to a Web site is that scripts can make the page more immediately interactive, that is, interactive without having to submit every little thing to the server for a server program to re-render the page and send it back to the client. For example, consider a top-level navigation panel that has, say, six primary image map links into subsections of the Web site. With only a little bit of scripting, each map area can be

instructed to pop up a more detailed list of links to the contents within a subsection whenever the user rolls the cursor atop a map area. With the help of that popup list of links, the user with a scriptable browser can bypass one intermediate menu page. The user without a scriptable browser (or who has disabled JavaScript) will have to drill down through a more traditional and time-consuming path to the desired content.

On their own, Web pages tend to be lifeless and flat unless animated images are added or more bandwidth-intensive content such as Java applets or other content requiring plug-ins to operate (ShockWave and Flash, for example). Embedding JavaScript into an HTML page can bring the page to life in any number of ways. Perhaps the most visible features built into pages recently with the help of JavaScript are the so-called image rollovers: roll the cursor atop a graphic image and its appearance changes to a highlighted version as a feedback mechanism to let the user know precisely what they're about to click on. But there are less visible yet more powerful enhancements to pages that JavaScript offers.

Interactive forms validation is an extremely useful application of JavaScript. While a user is entering data into form fields, scripts can examine the validity of the data--did the user type any letters into a phone number field?, for instance. Without scripting, the user has to submit the form and let a server program (CGI) check the field entry and then report back to the user. This is usually done in a batch mode (the entire form at once), and the extra transactions take a lot of time and server processing power. Interactive validation scripts can check each form field immediately after the user has entered the data, while the information is fresh in the mind.

JavaScript allows a Web page to perform "if-then" kinds of decisions based on browser version, operating system, user input, and, in more recent browsers, details about the screen size in which the browser is running. While a server CGI program can make some of those same kinds of decisions, not everyone has access to or the expertise to create CGI programs. For example, an experienced CGI programmer can examine information about the browser whenever a request for a page is made; thus a server so equipped might serve up one page for Navigator users and a different page for Internet Explorer users. Beyond browser and operating system version, a CGI program can't know more about the environment. But a JavaScript-enhanced page can instruct the browser to render only certain content based on the browser, operating system, and even the screen size.

Scripting can even go further if the page author desires. For example, the author may include a preference screen that lets the user determine the desired background and text color combination. A script can save this information on the client in a well-regulated local file called a cookie. The next time the user comes to the site, scripts in its pages look to the cookie info and render the page in the color combination selected previously. The server is none the wiser, nor does it have to store any visitor-specific information.

In Smartmath, JavaScript is used to generate all the quizzes and games questions. The questions are generated randomly and scores are stored temporarily. JavaScript is also used to generate the talking wizard.

3.4.3 Graphics

3.4.3.1 Adobe Photoshop

Advantages of Adobe Photoshop

Before going farther, however, there is a crucial distinction to be clarified between Photoshop and most other digital graphic applications. To allow for Photoshop's versatility and power in emulating the virtual photographic editing process, it was necessary to design the software to handle images in a ***rasterized*** format, as opposed to the object-oriented, or ***vector*** format used by such applications as Adobe Illustrator, Canvas, MacDraw, or any CAD program. These applications handle curves, lines, color-indeed, all image information-in terms of mathematical functions, where the defining mathematical principle of any graphic element is preserved despite any manipulation performed by the user.

An excellent applied example of this is text, where any amount of rotating, resizing, discoloring, deforming, or general manipulation causes a re-computation of the graphic definition of the text, resulting in a smooth and mathematically perfect final product. An *object* has been manipulated as a package, where the terms which define the contents of that package remain unchanged and directly effect the manner in which the object's manipulation is mathematically calculated.

Rasterized image data, on the other hand, is nothing but a two-dimensional, single-layered bitmap, where, say, curves are considered an arbitrary assembly of dots as opposed to a graphic object defined by a formula depicting the relationship of those dots to one another. In the case of the text example, manipulation will most likely yield

imperfect results, as there exists no rule by which text perfection is preserved. Bending, rotating, even resizing text is considered by Photoshop as a manipulation of an assembly of pixels.

In conclusion, there is no such thing as an `object' in Photoshop, other than a temporary selection; once a selection is deselected, it takes the place of those pixels it covers and merges with the flatness of the composite image. This is important to realize, as Photoshop lacks the versatility afforded to object-oriented programs to rearrange graphic elements at any time.

3.5 The Requirements Process

Before developing a system, it is very important to capture all the necessary requirements. A requirement is a feature of the system or a description of something the system or a description of something the system is capable of doing in order to fulfill the system's purpose *(Pfleeger, 1998)*. As for this project, the requirements were gathered through research on the Internet and books, analyzing on the results from the survey conducted, and review of the existing systems.

3.5.1 Requirements Elicitation

Requirements elicitation is an especially critical part of the process. A variety of techniques must be used to determine the user's needs and what they really want in the system. Requirements can be separated into three categories:

- requirements that absolutely must be met
- requirements that are highly desirable but not necessary

- requirements that are possible but could be eliminated

The figure above shows the process of determining requirements.

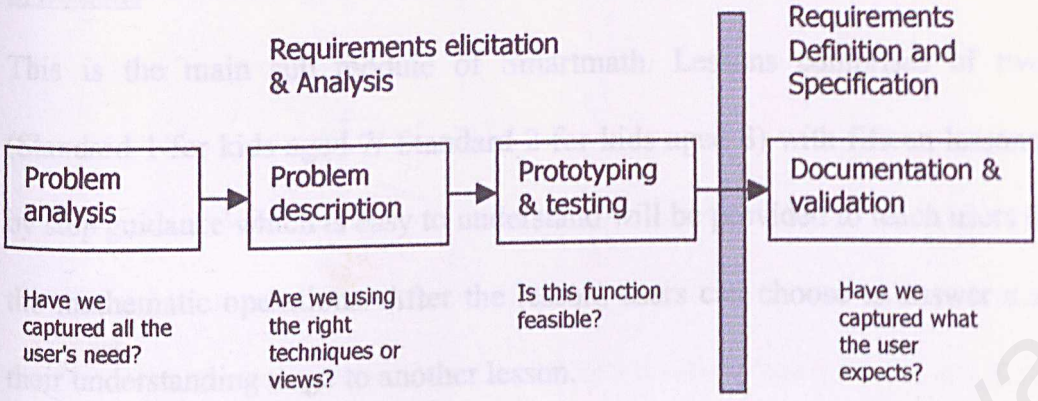


Figure 3.2: The process of determining requirements

3.5.2 Requirements for Smartmath

Through the information gathered, the outline for Smartmath is prepared. Basically, requirements are divided into two namely functional requirements and nonfunctional requirements.

3.5.2.1 Functional Requirements

A functional requirement describes an interaction between the system and its environment. It also describes how the system should behave given certain stimuli. As mentioned earlier, Smartmath is divided into two main modules:

1) English

2) Bahasa Melayu

Users can choose which language they prefer. After choosing the language, Smartmath is then divided into six sub modules:

a) Lessons

This is the main sub module of Smartmath. Lessons comprises of two standards (Standard 1-for kids aged 7/ Standard 2-for kids aged 8) with fifteen lessons each. Step by step guidance which is easy to understand will be provided to teach users the basics of the mathematic operations. After the lesson, users can choose to answer a short to test their understanding or go to another lesson.

b) Games

There are three types of games created. The first game is based on mathematic operations namely addition subtraction, multiplication, and division. Users can choose the type of operation and level that they wish to play. Each time an answer is given, feedback will be given to motivate the users to answer carefully.

The second and third game are about numbers. Users can choose to guess the numbers that the computer is thinking or choose to play the game whereby the computer will try and guess the number that the user is thinking.

c) Forward

The purpose of this module is to promote this web site to as many people as possible. Users of this site can send the URL of the current page that they are viewing to their friends. When they select forward, a new Microsoft Outlook mail will be created. All that they need to do is type the e-mail address of the receiver and click Send.

d) Link

This module provides the link to other educational sites. Users can select the link listed and the site will be displayed as a pop-up window.

3.5.2.2 Nonfunctional Requirements

A nonfunctional requirement or constraint describes a restriction on the system that limits our choices for constructing a solution to the problem. These constraints usually narrow our selection of language, platform, or implementation techniques or tools; however, the selection is made at the design stage, after the requirements have been specified. Below are the nonfunctional requirements that is needed for Smartmath:

a) User - friendly

Users are able to browse the website without any problem. It is important to make sure that users are comfortable and do not encounter difficulties while using the system. The system is easy to use, with its graphical user interface user can point-and click their way around easily. The talking wizard also acts as a guide that gives brief explanations in every page.

b) Attractive Interface

As Smartmath is designed for primary school children, the interface of the system would be very graphical and colorful in order to attract and maintain the children's interest and attention. Graphics that would be used will be based on cartoon characters and a reasonable amount of animation would be implemented.

c) Easy to Navigate

As mentioned earlier, this system is designed for primary school children. So, the navigation for this system is made as simple as possible. The navigation buttons and icons are either graphics or symbols which is easy to understand.

d) Interactive

This Web system, with its pages, enables interactivity between users and the system. The most common form of interactivity is clicking on hyperlinks to navigate around the system. Some pages have input boxes into which the user can enter textual information. As for the input boxes where the user enters the answer to the questions, immediate feedback will be given.

e) Learnability

Learnability refers to the ease with which new or occasional users may accomplish certain tasks in using the Smartmath system. Users are quickly able to understand the most basic comments and navigation options and use them to locate wanted information. In addition to easily understanding functionality of Smartmath, Smartmath will be easy to remember. The casual users should have no problems in remembering how to use and navigate in the system after periods of non-use.

f) User Satisfaction

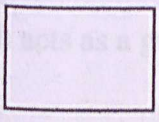
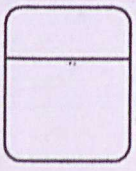
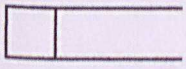
Smartmath is designed to be enjoyable to use and pleasing to users. User satisfaction will be within acceptable levels of user cost in terms of tiredness, discomfort, and individual effort so that the satisfaction causes continued and enhance the usage of the Smartmath system.

Chapter 4 - System Design

Design is the creative process of transforming the problem into a solution; the description of a solution is also called design [b].

A system model is a representation of an in-place or proposed system that describes the data flow throughout the structure. The model describes the points where data or information enters a system and the places where it will be processed, as well as the actions taken and the points where the data will be an output. Design diagrams include data flow diagrams (DFD), structured charts, decision trees, and other items.

For Smartmath, DFD and structured charts will be used to model the system. Structured charts will be used to show the outline of the system. DFD will give the graphical illustration that shows the flow of data and logic within the system. DFD comprises of four basic symbols:

Symbols	Name	Description
	Entity	An external entity that can send data to or receive data from the system. Interacts with the system but considered as outside of the boundaries of the system.
	Flow of Data	Used to show the movement of data from an origin to a destination with the head of arrow pointing towards the destination.
	Process	It represents the transformation or processing of information within a system.
	Data Store	Shows a depository for data that allows addition or retrieval of data.

4.1 Designing Smartmath

The design of Smartmath is based on all the information gathered which has been explained in Literature Review. The content of the system is designed based on the requirements of the end users. The advantages and disadvantages of the existing systems are taken into consideration when designing Smartmath. Basically, these are the characteristics of Smartmath:

- user friendly
- simple and easy to understand instructions
- lessons focused on topics (addition, subtraction, multiplication, division, money, and time)
- games based on mathematic operations and numbers
- attractive interface
- well organized

Smartmath will be developed in both English and Bahasa Melayu. A talking wizard, which acts as a guide will appear on every site.

4.2 System Structural Design and Data Flow Diagram (DFD) for Smartmath

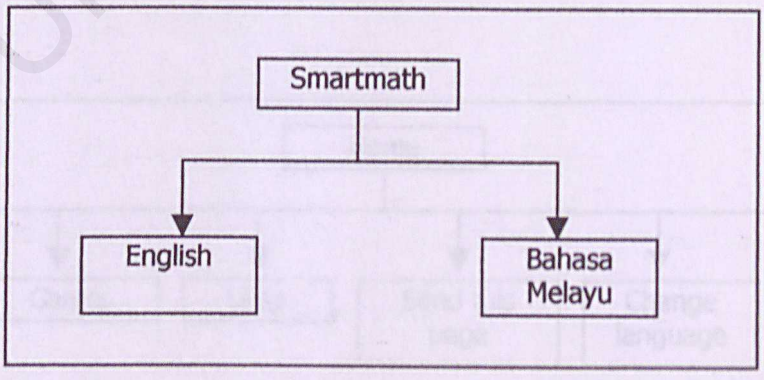


Figure 4.1: Main Structural Design for Smartmath

Smartmath is divided into two main parts. Users can choose to view the site in English or Bahasa Melayu.

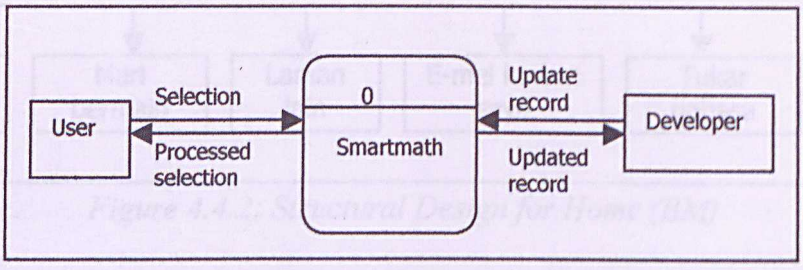


Figure 4.2: Context diagram for Smartmath

Figure 4.2 shows the context diagram for Smartmath. The main entities are the users and the developer of this site. Smartmath processes the user's selections and the updates done by the developers.

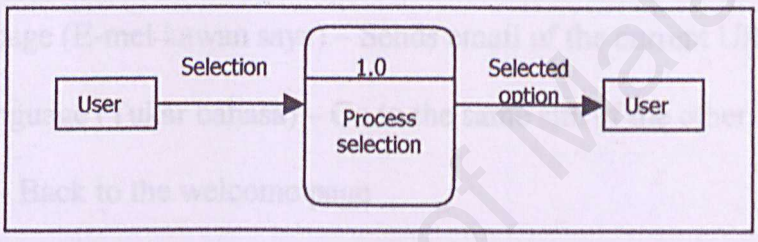


Figure 4.3: Data Flow Diagram for Main

Figure 4.3 shows the DFD for the main page of Smartmath. The two main options are the languages. If the user can't hear the talking wizard, step by step instruction will be given for them to down load the necessary application in order to hear the wizard.

4.2.1 Home

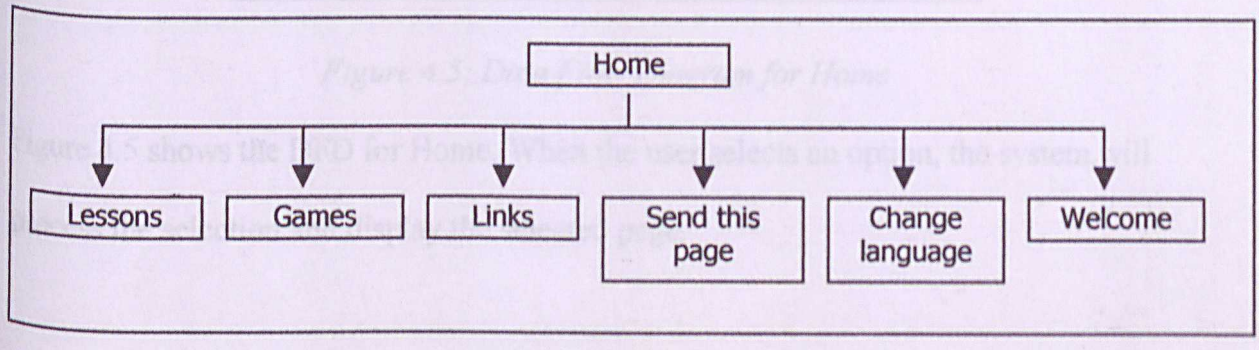


Figure 4.4.1: Structural Design for Home (English)

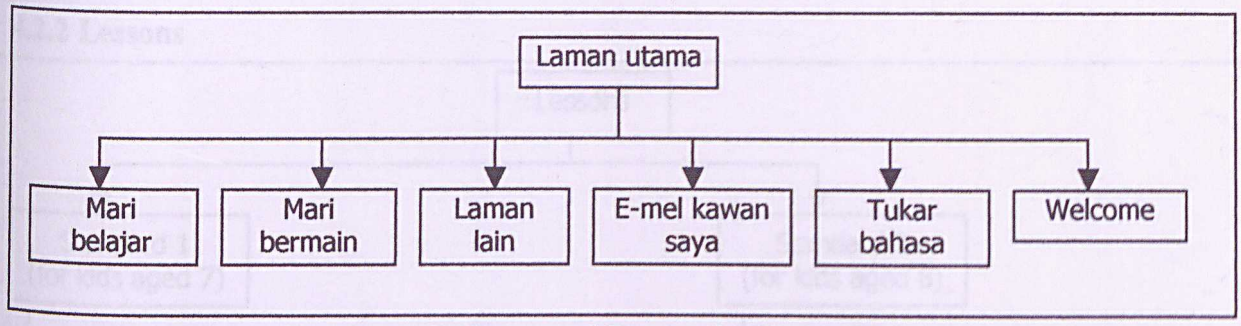


Figure 4.4.2: Structural Design for Home (BM)

In Home, there are 6 main modules:

1. Lessons (Mari belajar) – List of lessons available
2. Games (Mari bermain) – Lists types of games
3. Links (Laman lain) – Links to other educational sites
4. Send this page (E-mel kawan saya) – Sends email of the current URL to a friend
5. Change language (Tukar bahasa) – Go to the same site of the other language
6. Welcome – Back to the welcome page

These modules are the links that the user will see in the Smartmath homepage. User can click at any link and the site will move on to the selected page. Some of the modules have sub-modules whereas some of the modules are HTML text pages only.

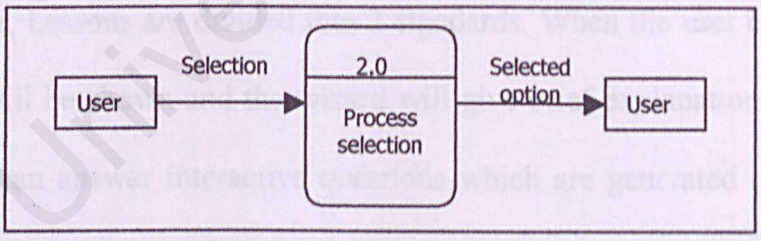


Figure 4.5: Data Flow Diagram for Home

Figure 4.5 shows the DFD for Home. When the user selects an option, the system will process the selection and display the selected page.

4.2.2 Lessons

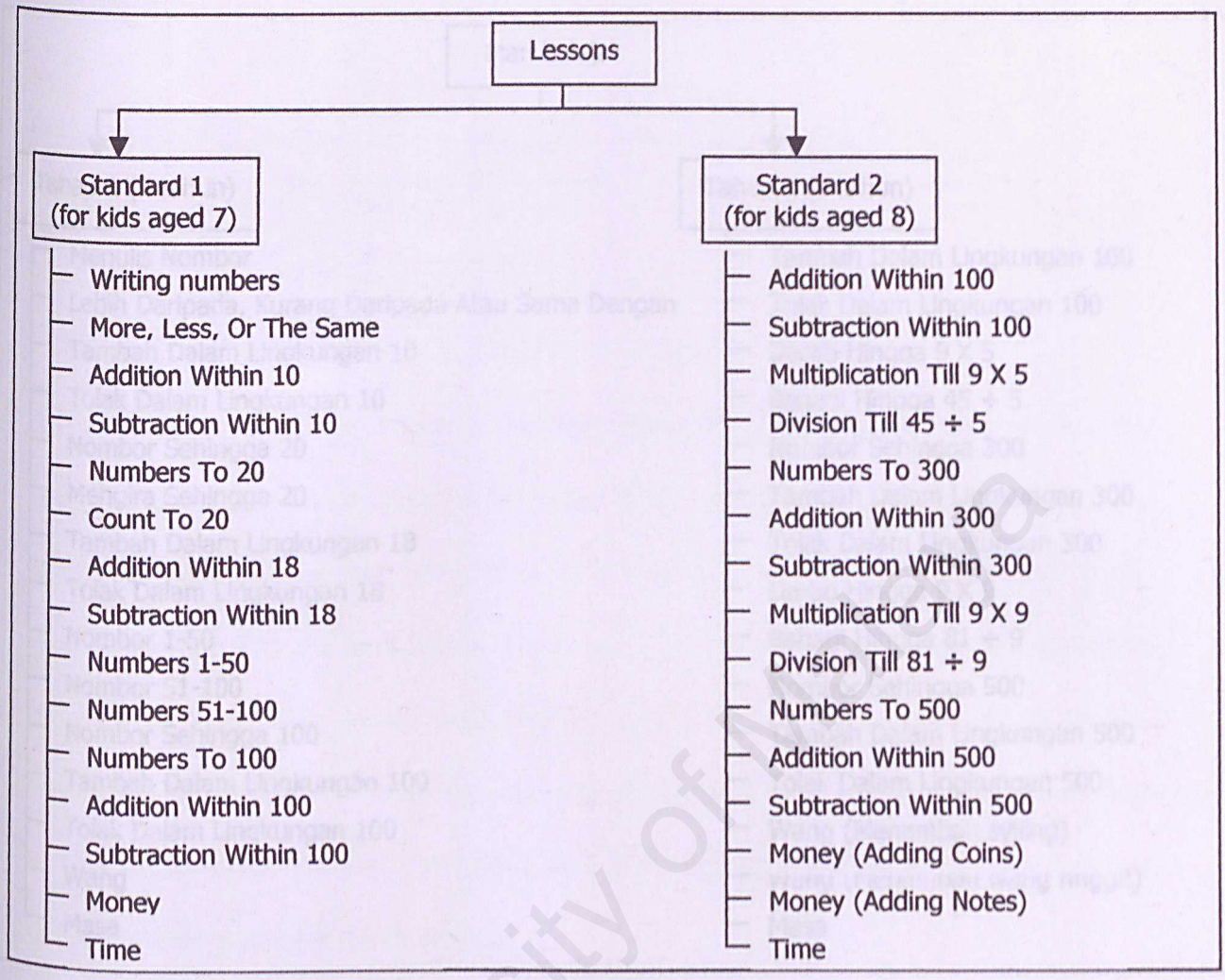


Figure 4.6.1: Structural Design for Lessons (English)

As shown above, Lessons are divided into 2 standards. When the user click on a certain topic, the site will be shown and the wizard will give brief explanation about the topic. Then, the user can answer interactive questions which are generated randomly. Figure 4.6.2 below shows the structural design for lessons in BM.

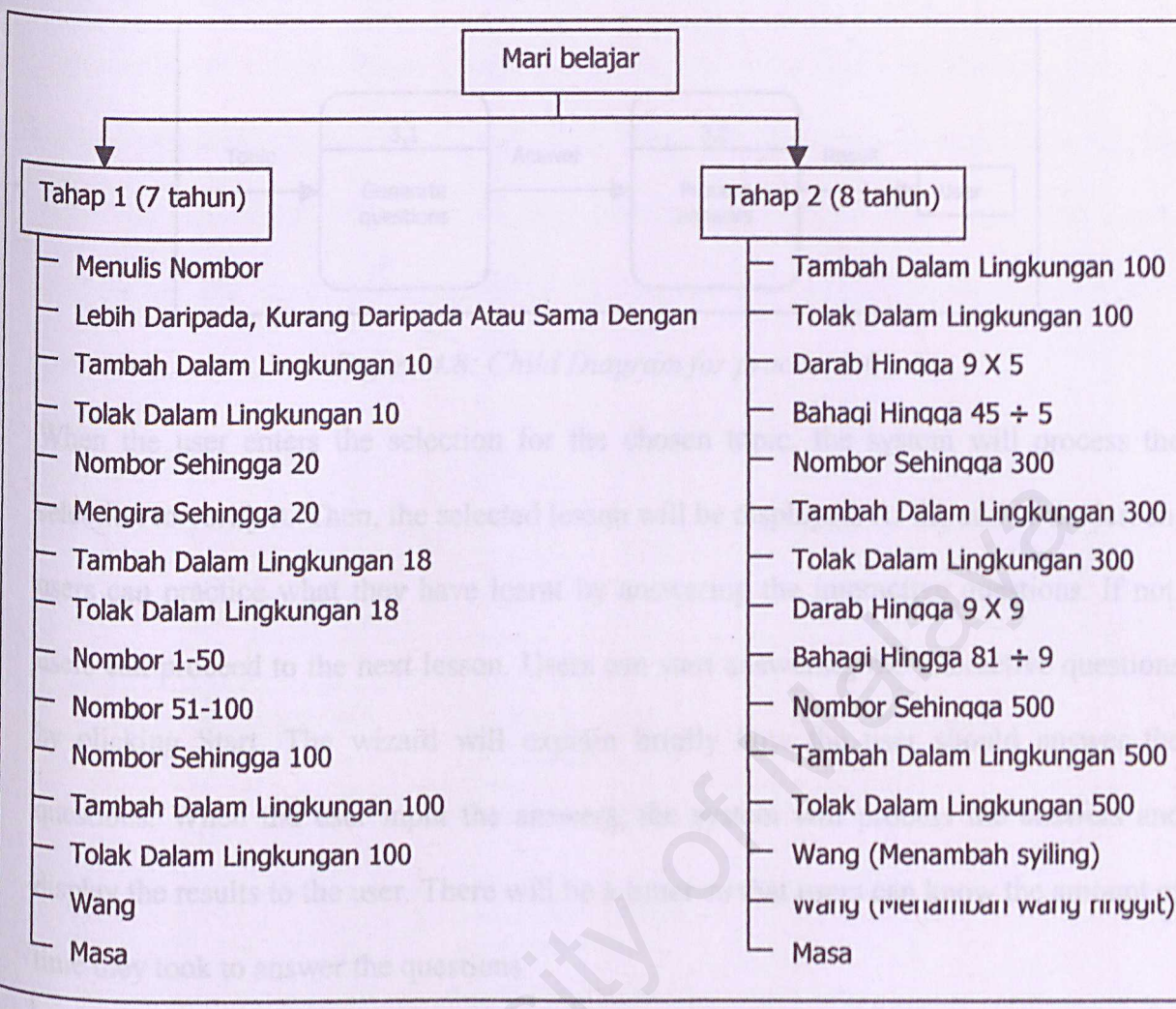


Figure 4.6.2: Structural Design for Lessons (BM)

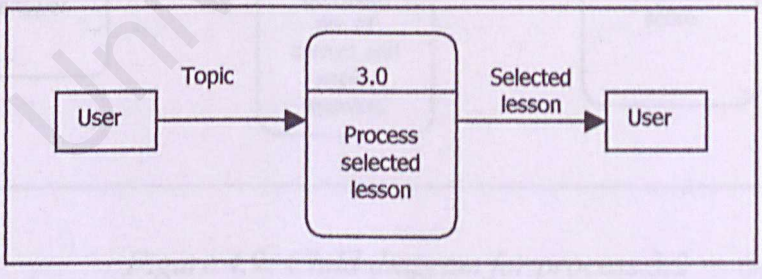


Figure 4.7: Data Flow Diagram for Lessons

Figure 4.7 shows the DFD of Lessons. When the user selects a topic, the system will process the selected lesson and display it to the user.

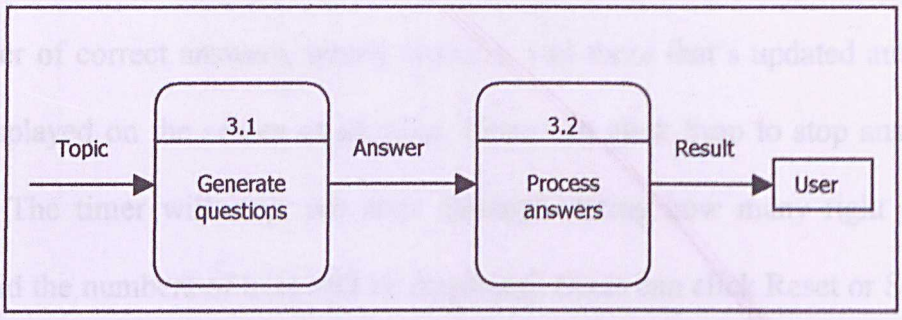


Figure 4.8: Child Diagram for process 3.0

When the user enters the selection for the chosen topic, the system will process the selection to verify it. Then, the selected lesson will be displayed. At the end of the lesson, users can practice what they have learnt by answering the interactive questions. If not, users can proceed to the next lesson. Users can start answering the interactive questions by clicking Start. The wizard will explain briefly how the user should answer the questions. When the user input the answers, the system will process the answers and display the results to the user. There will be a timer so that users can know the amount of time they took to answer the questions.

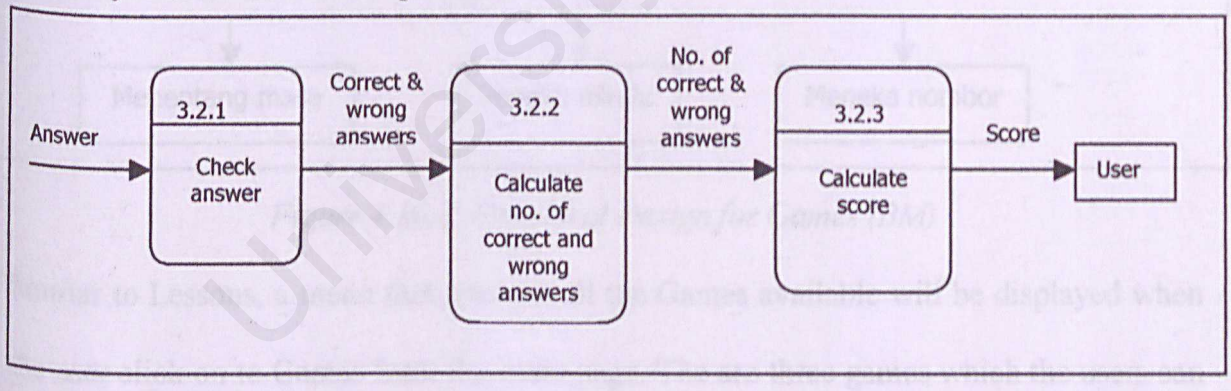


Figure 4.9: Child diagram for process 3.2

The diagram above shows the details for processing the answers. When the user inputs an answer, the system will check whether the answer is accurate or not and total up the number of correct answers and wrong answers. At the same time, the score will also be calculated.

Score (%) = (No. of correct answers / No. of tries) X 100

The number of correct answers, wrong answers, and score that's updated automatically will be displayed on the screen at all time. Users can click Stop to stop answering the questions. The timer will stop. An alert message stating how many right and wrong answers and the numbers of tries will be displayed. Users can click Reset or Start to start answering questions again or proceed to the next lesson.

4.2.3 Games

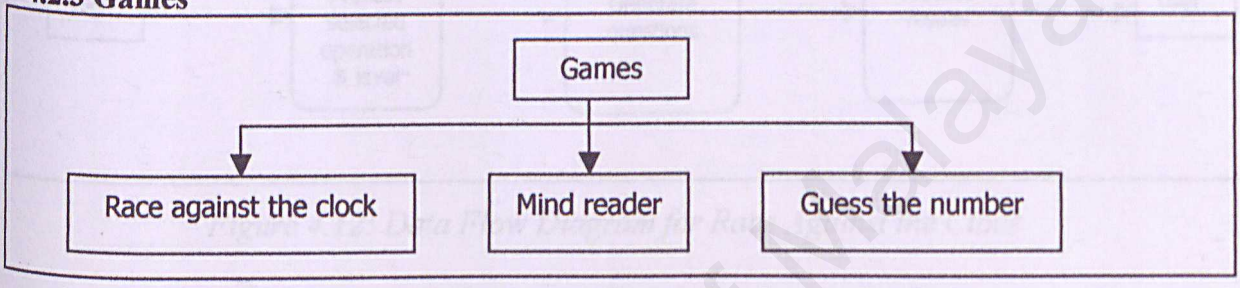


Figure 4.10.1: Structural Design for Games (English)

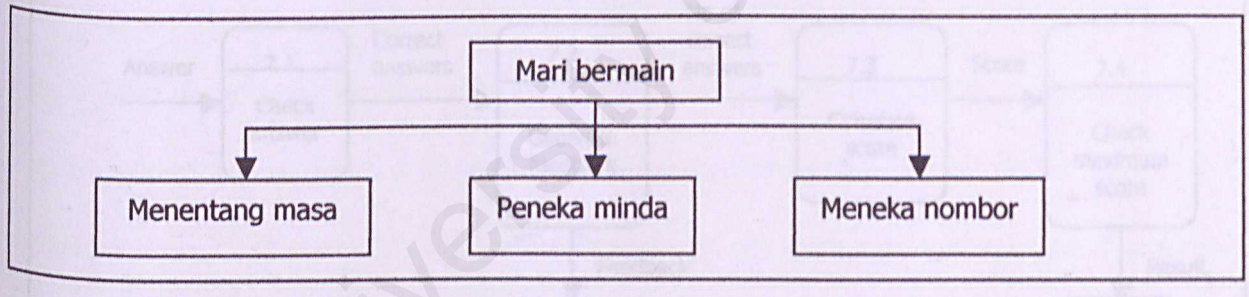


Figure 4.10.2: Structural Design for Games (BM)

Similar to Lessons, a menu that lists out all the Games available will be displayed when the user click on to Games from the main page. The are three games which the users can choose from. 'Race against the clock' is a game that tests the students understanding about the lessons. Users have to answer as many questions as possible in 60 seconds. The users can choose the level that they want to play. 'Mind reader' and 'Guess the number' are games about numbers. Below are the DFD's for Games and for each game respectively.

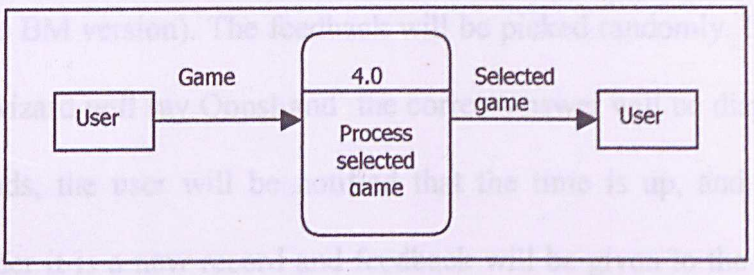


Figure 4.11: Data Flow Diagram for Games

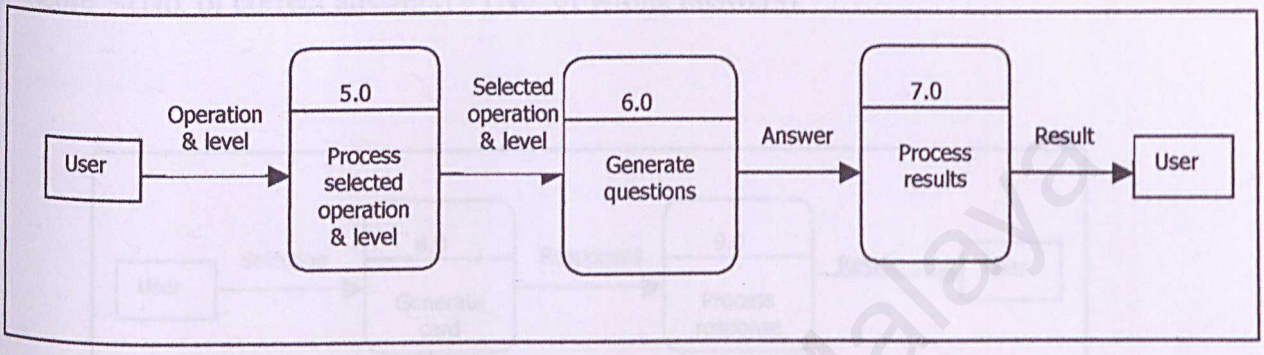


Figure 4.12: Data Flow Diagram for Race Against the Clock

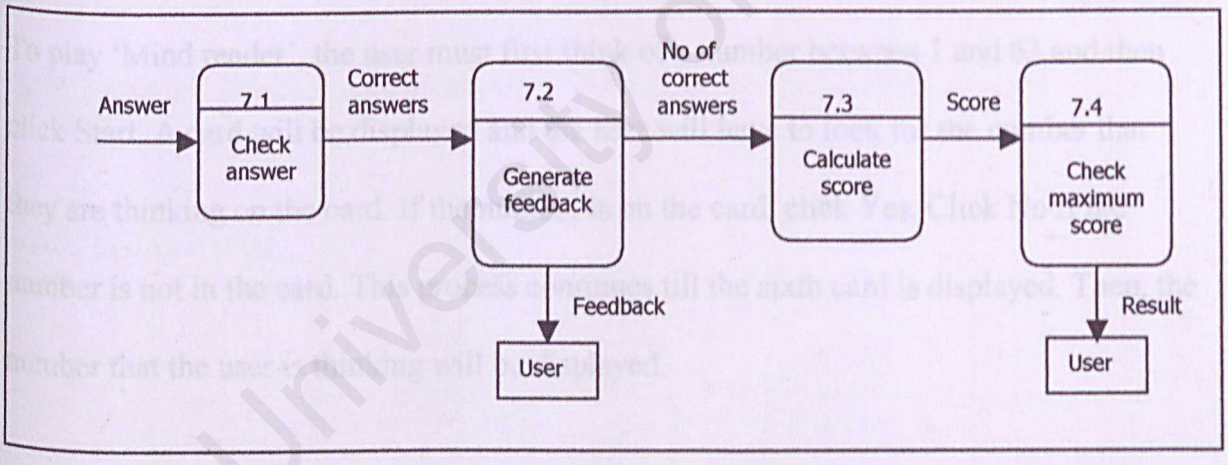


Figure 4.13: Child diagram for process 7.0

When a user chooses to play 'Race against the clock', the user will have to choose the type of operation and level. Then, the selected page will be displayed. Users will be given 60 seconds to answer as many questions as they can. Feedback will be given to each answer given by the user. If the user answers correctly, there are 4 types of feedback which are You are right!, Correct!, Well Done!, or Good Job! (Syabas!, Betul!, or

Hurray! for the BM version). The feedback will be picked randomly. If the user answers wrongly, the wizard will say Oops! and the correct answer will be displayed. At the end of sixty seconds, the user will be notified that the time is up, and the score will be checked whether it is a new record and feedback will be given to the user. The score is calculated this way,

Score = (No. of correct answers) – (No. of wrong answers).

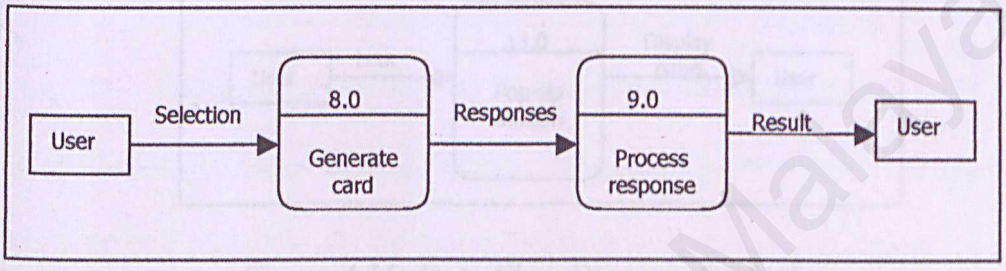


Figure 4.14: Data Flow Diagram for Mind Reader

To play 'Mind reader', the user must first think of a number between 1 and 63 and then click Start. A card will be displayed and the user will have to look for the number that they are thinking on the card. If the number is on the card, click Yes. Click No if the number is not in the card. This process continues till the sixth card is displayed. Then, the number that the user is thinking will be displayed.

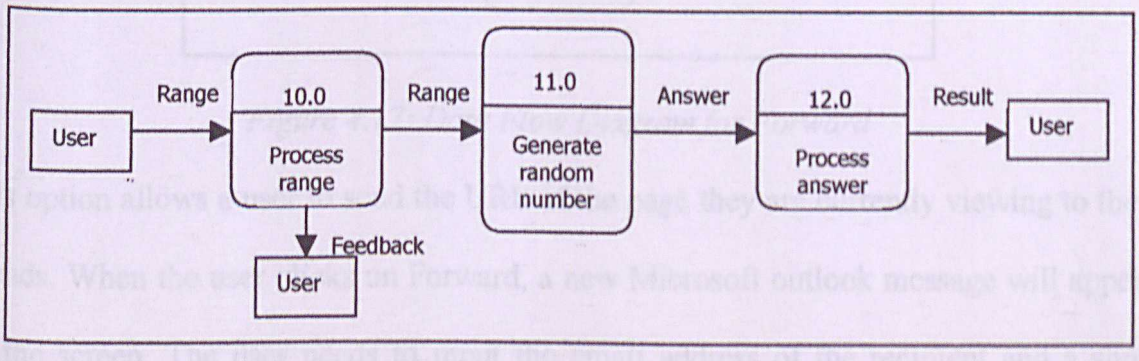


Figure 4.15: Data Flow Diagram for Guess the Number

In 'Guess the number', users will have to try and guess the number that the computer is thinking. Users start off by setting a range of numbers. If the user input the range incorrectly, an alert message box will appear together with the feedback from the wizard. The user will have to set another range. After that, the computer will generate a random number within that range. That is the number that the user will have to guess.

4.2.4 Links

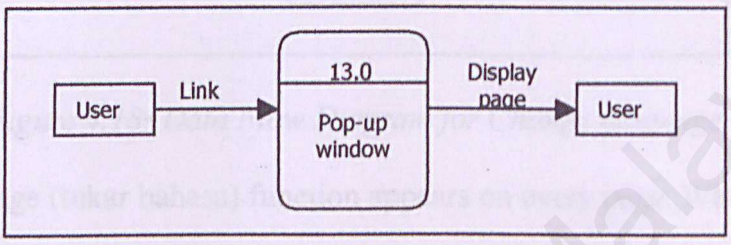


Figure 4.16: Data Flow Diagram for Links

In the Links option, a list of link to other educational web-sites will be displayed. User can click on any of the links and the site will be displayed as a pop-up window.

4.2.5 Send this page

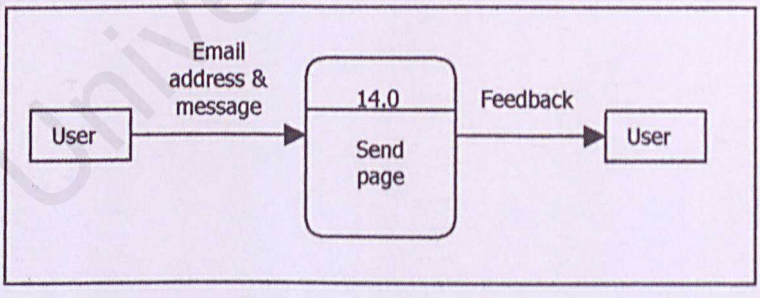


Figure 4.17: Data Flow Diagram for Forward

This option allows a user to send the URL of the page they are currently viewing to their friends. When the user clicks on Forward, a new Microsoft outlook message will appear on the screen. The user needs to input the email address of the recipient and a short

message which is optional. After completing the necessary information, the URL of the current page that the user is viewing will be sent to the stated email address.

4.2.6 Change Language

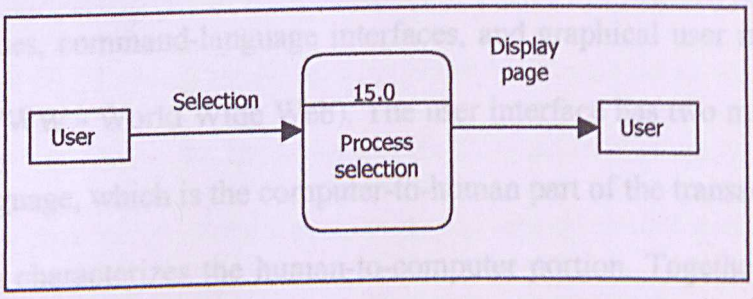


Figure 4.18: Data Flow Diagram for Change language

The change language (tukar bahasa) function appears on every page. When users click on this function, the site will go to the other language version of the same page. For example, when a user that is currently viewing the homepage of Smartmath in t he English version, the site will move to the BM version of the Smartmath homepage.

4.3 Interface Design

The interface is the system for most users. However well or poorly designed, it stands as a representation of the system and reflects the competence of the developer. Types of interfaces includes natural-language interfaces, question-and-answer interfaces, menus, form-fill interfaces, command-language interfaces, and graphical user interfaces (GUIs) and the Web (WWW - World Wide Web). The user interface has two main components: presentation language, which is the computer-to-human part of the transaction, and action language, which characterizes the human-to-computer portion. Together, both concepts cover the form and content of the term *user interface*.

As Smartmath is a web-based system developed for children, the interface would be designed to be colourful and attractive. The interface design will be designed to be user friendly, attractive, and easy to navigate. Graphics and animation that will be included would be at a reasonable amount to ensure that the download time for the site is not too slow. The main attraction of this site is the talking wizard which acts as a tutor guiding the children throughout the lessons and games.

Chapter 5 – System Implementation

5.1 Introduction

This system implementation involves the system development environment and program coding. In this system implementation phase, the system requirements and design are converted into program code. This phase always involves some modifications to the previous design due to the limitations of the programming language used.

Each web page of Smartmath was developed separately and later integrated into a fully functional system once each page has been tested successfully.

5.2 Development Environment

5.2.1 Hardware used in developing the system

- Intel Pentium 4 Processor 1300 MHz
- Memory 128 MB RAM
- Hard Disk 20GB space
- Other standard desktop PC compliance

5.2.2 Software used in developing the system

- Windows ME as the operating system
- Internet Explorer 5.5 as web browser to view the web pages design
- Macromedia Dreamweaver UltraDev to generate the HTML pages
- MASH (Microsoft Agent Scripting Helper) used to generate the talking wizard
- Notepad and Wordpad to edit the coding
- Microsoft Word 2000 to write the documentation

5.3 Implementation

At the beginning stage, Smartmath is planned to be developed using Macromedia Flash. However, due to the ability of JavaScript of generating interactive questions and the talking wizard, the development of Smartmath is changed to JavaScript and Java.

5.4 Web Pages Development

Among the criteria that were applied during the selection of the programming language are:

- Availability of the development tools
- Knowledge of the software development tools
- Nature of the system to be developed

Therefore, languages used to develop the web pages need to be chosen appropriately to ease the development of the web pages. Languages used to develop the web pages are HTML, JavaScript and Java. HTML is the basic language for home page designing whereas JavaScript and Java are used to perform and user interaction. This includes the interactive quiz questions, games, and the talking wizard.

The designs of the web pages are done by drawing sketches on papers and then created in Adobe Photoshop. Adobe Photoshop is mainly used to create the buttons and images. Then, the pages are built with Macromedia Dreamweaver. JavaScript that generates the quiz questions are then inserted in the HTML documents. Once the JavaScript is inserted, any modification is done using Notepad or WordPad only. This is to ensure that the page is error free.

5.5 The Flow of Smartmath

Shown below is the basic flow of Smartmath that has been implemented:

Figure 5.1: Smartmath's Welcome page

Here, users can choose to view the site in English or Bahasa Melayu.

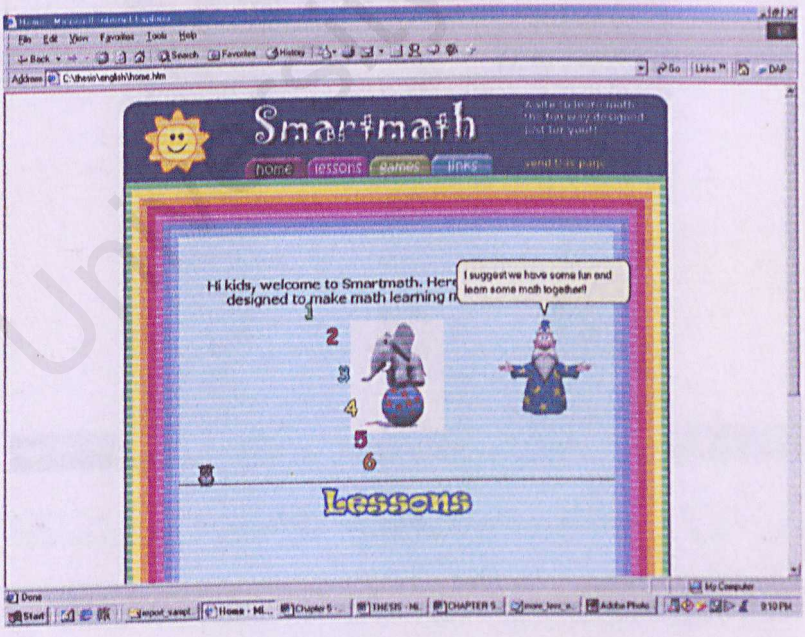


Figure 5.2: Homepage

This page gives a brief explanation about Smartmath and users can proceed to all the other modules from here.

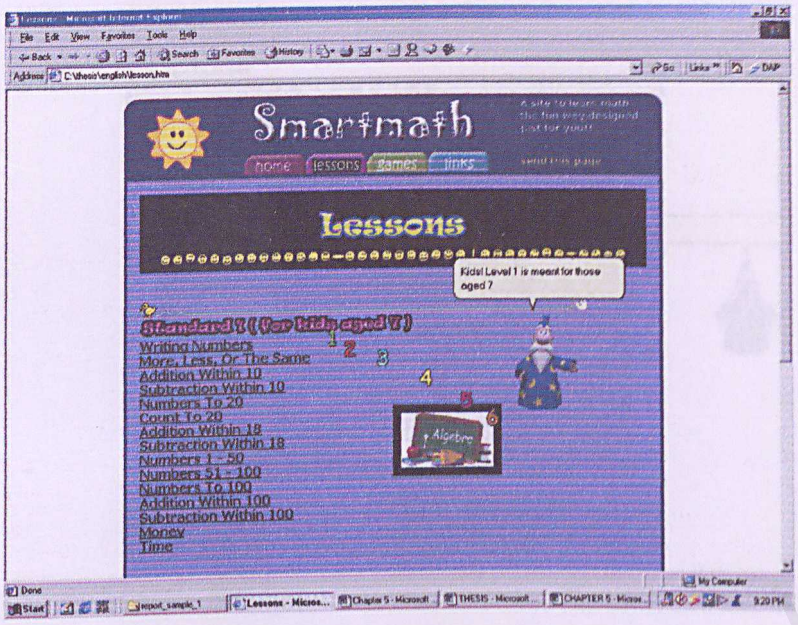


Figure 5.3: Lessons

This page lists out all the topics available.

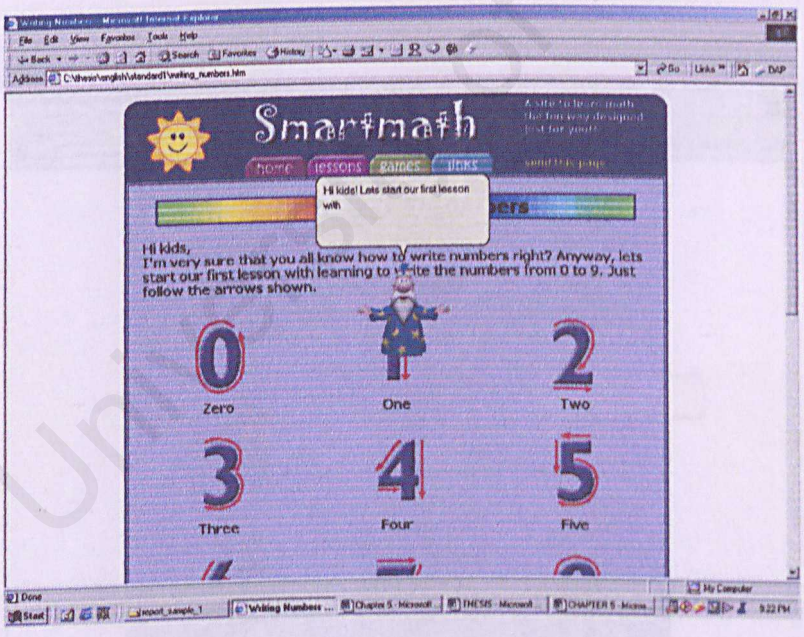


Figure 5.4: Example of a lesson

The figure above shows an example of the lessons.

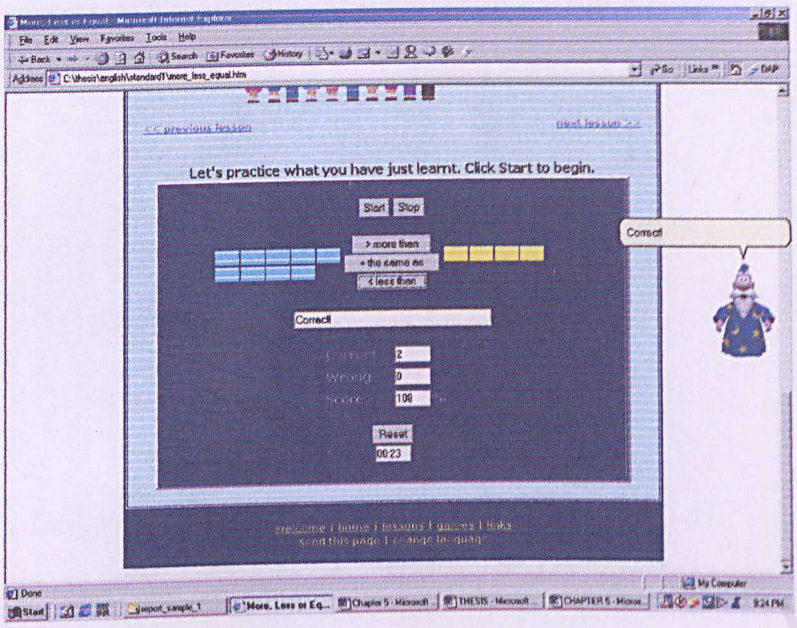


Figure 5.5: Interactive questions

The figure above shows an example of the interactive quiz questions. Almost every lesson has its own different set of quiz questions.

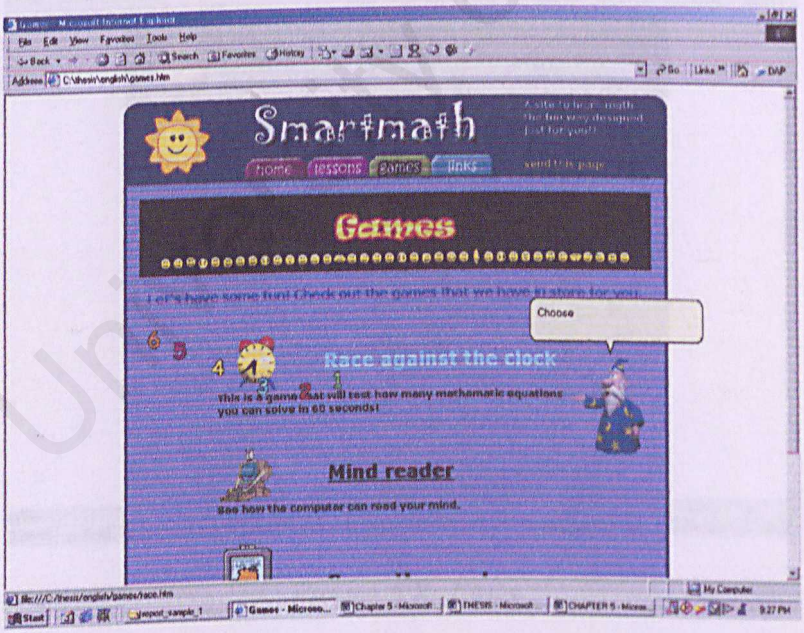


Figure 5.6: Games

This page lists out all the games available.

Figure 5.7: Example of a Game

The figure above shows an example of the games available, Mind Reader.

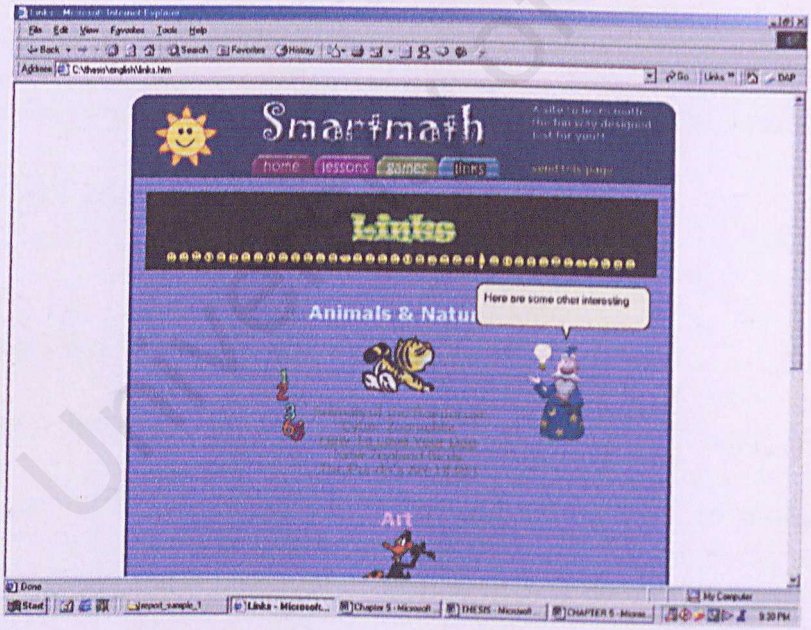


Figure 5.8: Links

This page shows the links to other educational sites. When the user clicks on the link, the site will be opened as a pop-up window.

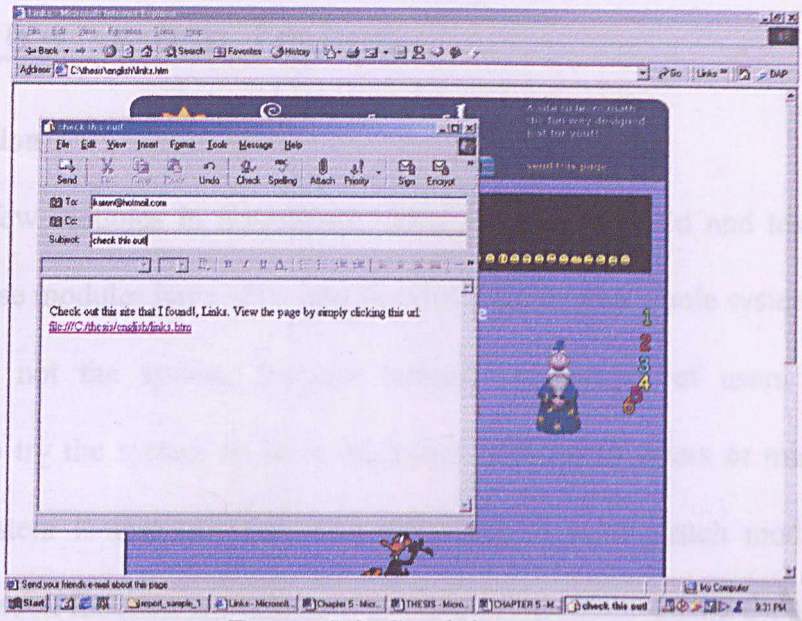


Figure 5.9: Send this page

When the user click on 'Send this page' on any page, a new Microsoft Outlook message will appear. The user should just input the recipient's email address and the URL of the current page would be sent.

Chapter 6 – System Testing

6.1 Introduction

There are a few modules in Smartmath. Each module is coded and tested separately. After that, these modules have to be integrated and tested in a whole system.

The tester is not the system designer himself. A number of users are given the opportunity to try the system so as to trace any unforeseen errors or misunderstanding before the system is implemented. The tester has to ensure each module is running smoothly and each function is performed perfectly. Therefore asking the tester to try out the system will test the usability of the user interface, whether the interface is self-explanatory or not, or whether the tester know what should be the steps taken to run the system. If the tester felt uneasy or confused while testing out the system, the user interface should be revised and improved. Advice that is asked from the tester is to improve the usability of the interface. Sometimes the misuse of the wordings or language in the system might mislead the users.

6.2 The Testing Process

Except for small programs, systems should not be tested as a single, monolithic unit. Large systems are built out of sub-systems, which are built out of modules, which are composed of procedures and functions. The testing process should therefore proceed in stages where testing is carried out incrementally in conjunction with system implementation. Following are the steps in testing Smartmath:

6.2.1 Unit Testing

This is a small unit testing where testing are done on individual components of the system to ensure that they operate correctly. Each web page is tested independently, without other system components. The units that were tested individually are usually the interactive quiz questions, games, and the wizard.

a) **Interactive quiz questions**

It is very important that the questions are generated correctly according to the topic of the lessons. The checking of the answers and calculation of the scores are also tested to make sure of its accuracy.

b) **Games**

The games are tested carefully to ensure that it functions properly. The games are tested over and over again with various answers and possibilities to find out the possible errors and flaw. Functions are included to overcome these weaknesses.

c) **Wizard**

The wizard appears on every web page. So, every paged is viewed to ensure that the wizard can be seen, heard, and animates correctly. The make sure the wizard really works, the web pages are tested on various computers. Usually, the problem that might occur would be the wizard couldn't be seen or heard. This problem could be fixed by installing the necessary applications, which are Microsoft Agent Character file and Text-to-speech engines.

6.2.2 Integration Testing

After every web page is tested individually, the web pages are integrated into a whole system and tested again. This time, testing is emphasized on the links. Each link on a page is tested repeatedly to make sure that it links to the right page.

6.2.3 System Testing

Finally, system testing would be the last stage in testing. The testing process is concerned with finding errors, which result from anticipated interactions between sub-systems and system components. It is also concerned with validating that the system fulfills the functional and non-functional requirements. System testing can be categorized into a few types:

a) **Stress Testing**

This is to determine the program fulfills the requirements defined for it. It is equally important to ensure that the program works, as it should under extreme conditions. In order to perform stress testing, execute the system in a manner that demands resources in abnormal, quantity, frequency, or volume.

b) **Performance Testing**

For real-time and embedded systems, software that provides required function but does not conform to performance requirements is unacceptable. Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process.

6.3 Types of Testing

Generally, there are 2 types of testing:

6.3.1 Statistical Testing

Statistical testing is a software testing process in which the objective is to measure the reliability of the software rather than to discover software faults. Tests are designed to reflect the frequency of actual user inputs. After running the tests, an estimate of the operational reliability of the system can be made. Program performance may be judged by measuring the execution of the statistical tests.

6.3.2 Defect Testing

Defect testing is intended to find areas where the program does not conform to its specification. Tests are designed to reveal the presence of defects in the system. There are 2 approaches to defect testing:

a) **Black-box Testing**

Black-box testing, also called behavioral testing, relies on the specification of the system or software. Tests are used to demonstrate each function is fully operational while at the same time searching for errors in each function.

b) **White-box Testing**

White-box testing, sometimes called glass-box testing, uses the control structure and implementation of the procedural design to derive test cases. Tests are conducted to ensure that the internal operations are performed according to its specifications and all internal components have been adequately exercised.

Chapter 7 –System Evaluation

7.1 System Strength

Here are some of the advantages of Smartmath:

a) User Friendly

Because Smartmath is developed for children, the navigation is made as simple and user friendly as possible so that users do not get confused while surfing the site. Simple instructions are given to ensure users are comfortable and at ease while using the system. Apart from that, the talking wizard also acts as guide for the users and does animations that will keep the children entertained and draws their attention.

b) Attractive Graphical User Interface

The layout of Smartmath's web pages is very colourful and attractive. Images and graphics chosen for the site are suitable for children. Links and buttons are named appropriately with simple words so that children will not have problems understanding it. Lessons and games are also conducted with very simple and easy to understand sentences and phrases.

c) Bilingual Site

This site is made bilingual to cover a larger scale of users. Any users who understand English or Bahasa Melayu can use this site.

d) Scope of Lessons

The lessons provided in Smartmath follows the topics which are taught in Malaysia's primary school. So, it's very suitable for primary school children who wishes to practice mathematic on their own or have some fun with learning math. It can also be used by school teachers to conduct lessons in class or parents to give extra guidance to their children at home.

7.2 System Constraints

a) Simple Lessons

The lessons in Smartmath are very simple and not very complete. It gives brief explanations on each topic and not in detail. The explanation given by the wizard is also very simple.

b) Scope of the Lessons available

Due to the time constraint, the developed lessons are meant for kids in Standard 1 and Standard 2 only.

c) The Wizard

The pronunciation of the words by the wizard in Bahasa Melayu is not very accurate so it might sound slightly confusing sometimes.

7.3 Future Enhancement

a) Detailed Lessons and Scope

Explanations in lessons should be made in detail with more examples and exercises to improve the users understanding. Scope of the lessons available should be broaden so that Smartmath can be used by more children of various ages.

b) Smartmath in other languages

Smartmath could be developed in other languages such as Japanese or Mandarin to provide its benefits to even more students.

c) Creating a Database

Having a database to store the student's progress in answering the interactive questions and games so that they can keep track and improve themselves.

7.4 Knowledge and Experience Gained

Knowledge gained throughout the development of Smartmath is really valuable. Knowledge was gained on website development as well as programming coding and concepts. Here, theories and knowledge gained during the course of Information Technology like *System Analysis and Design*, *Graphical User Interface*, and *Multimedia System* just to name a few were literally put into practice. Experience gained while working as a trainee in *Asia Travel Network* was also put into practice.

Besides that, the knowledge gained from this project is the awareness of the user's needs and the flow of a system. It was found that users need a user-friendly environment, a

readable homepage and clear instruction and guidance. The programmer needs to find and organize the information acquired and analyze the behaviour.

There are improvements in skills of finding information, classifying fields, solving problems and independently plan and accomplish the project on schedule without much supervision.

No doubt that experience has been gained and new knowledge has been acquired. More importantly, the process of doing it has been an exposure to me on how to really plan and work on a project. One important thing I realized is that it is really useful and important to have an up-to-date knowledge and information in keeping up with the fast and ever changing technology of information technology. This project has proved very beneficial for me when I step into the working environment in future.

7.5 Problems Encountered and Solutions

Various problems were encountered throughout the development of Smartmath:

a) Web Page Coding

Problem:

Basically the problem encountered in web pages coding involves the HTML coding, Java scripting, and JavaScript programming especially the study of JavaScript, which is very crucial to the development of the system. Most of the coding and scripting problem was encountered in the early stage of the project development. This is due to the ambiguity and lack of understanding in the early stage.

Solution:

The solution to overcome is to adopt a divide-and-conquer approach by first concentrating and understanding the basic concepts of the programming. As the development of the project goes on, the understanding gradually builds up and most of the problems encountered in the earlier stage were overcame easily.

b) Developing the Real System

Problem:

Due to the lack of knowledge and exposure in developing a website, some parts of the system proposed earlier couldn't be developed as a real system. When developing the real system, I realized that certain parts of the system need to be modified. Also, I decided to develop Smarthmath as a bilingual site. Therefore, the scope of the lessons needs to be streamed down.

Solution:

As a solution to this problem, modification was done to the parts, which couldn't be done as proposed and replaced with the best alternatives available.

c) Tools to Develop the System

Problem:

The tool that was proposed to develop Smartmath was Macromedia Flash. The generation of the talking wizard involves either JavaScript or VBScript. The scripting for Flash is Action scripting. Due to inexperience in all these languages, I do not know how to integrate JavaScript or VBScript into Action scripting.

Solution:

I made the decision to develop Smartmath with HTML, Java language, and JavaScript.

d) The Inability of the Wizard to Recognize Bahasa Melayu

Problem:

The talking wizard was generated with software called MASH (Microsoft Agent Scripting Helper), which consist of JavaScript coding. The use of a text-to-speech engine enables the wizard to talk. Unfortunately, this engine couldn't recognize Bahasa Melayu. So, the BM version of Smartmath couldn't have the animation of a talking wizard.

Solution:

To overcome this problem, I used the text mapping method. To enable the wizard to speak in Bahasa Melayu, I map each syllable of a word to sound like a malay word.

Expected Outcome

The Interactive Mathematics Learning System (Smartmath) is expected to be:

- an effective, interesting, and user friendly learning tool
- able to give users a basic understanding in mathematic operations and cultivate eagerness and interest in learning mathematics
- able to promote the use of web learning as a preferred teaching approach

Conclusion

Building a web-based application package is a very challenging task. Lots of research, time, and effort have been involved in making this project successful and in fulfilling the task requirements. A comprehensive knowledge in building a web-based application and the relational development tools is also necessary and important. Understanding the user's needs and requirements also contributes a lot in the development of this application.

As a conclusion, it can be said that Smartmath has actually reached its objectives as a multimedia teaching and learning package. The system is interactive, attractive, and most importantly easy to use as Smartmath is developed for children. The system is easy to learn and use and users can master it within a short learning time.

Bibliography

Books

[a] Bouton, G. D., Bouton, B. "Inside Adobe Photoshop 5.5." New Riders publishing, 2000.

[b] Pfleeger, S. L. "Software Engineering Theory and Practice." Prentice-Hall, 1998.

[c] Kendall, K. E., Kendall, J. E. "System Analysis and Design (Fourth Edition)." Prentice-Hall, 1999.

[d] Bryant, S. C., "Teach Yourself HTML 4." IDG Books Worldwide, Inc., 1999.

[e] Powell, T. A., Whitworth, D. "HTML." McGraw-Hill, 1998.

URL

[1] http://www.outreach.utk.edu/weblearning/

An Overview of Web-Based Learning

[2] http://www.epsys.com.sg/implementing_cd.htm

Implementing CD-ROM based learning solution (White Paper)

[3] http://www.macromedia.com/support/dreamweaver/ts/documents/presalesfaq.htm

Common questions about Dreamweaver

[4] http://tutorials.beginners.co.uk/view/cobrand/searchmiddleware/i/89

Introduction to HTML

[5] http://www.bellcraft.com/mash/main.htm

The Microsoft Agent Scripting Helper

[6] http://www.funbrain.com/

[7] http://www.aplusmath.com/

[8] http://www.kids-korner.com/

[9] http://www.coolmath4kids.com

[10] http://www.edu4kids.com/

[11] http://www.iknowthat.com/

[12] http://www.c3.lanl.gov/mega_math/

[13] http://www.georgetown.edu/crossroads/mltmedia.html

[14] http://snow.utoronto.ca/Learn2/design.html

[15] http://www.lmu.ac.uk/lss/staffsup/desmeth.htm

[16] http://www.med.monash.edu.au/informatics/techme

[17] http://scis.nova.edu/~henkeh/story1.htm

[18] http://www.adobe.com

[19] http://info.med.yale.edu/caim/manual/contents.html

[20] http://home.earthlink.net/~castelnet/info/web-basedlearning.html

[21] http://www.curriculum.wa.edu.au/pages/framework/framework08a.htm

[22] http://developer.netscape.com/viewsource/husted_js/husted_js.html

[23] http://www.webword.com/interviews/goodman.html

[24] http://www.mcli.dist.maricopa.edu/show/interact/js.html

[25] http://www.clarkson.edu/~dubrovvj/400/javascript/jstutorial/jsintro.htm

[26] http://www.xs4all.nl/~ganswijk/chipdir/about/js.htm

[27] http://www.melbpc.org.au/pcupdate/9807/9807article11.htm

[28] http://www.windows2000.au.com/software/page1.html

[29] http://www.clearform.com/microsoft_access.htm

[30] http://informationr.net/ir/5-1/paper66.html

[31] http://unr.edu/homepage/araby/webservices.html

User Manual

Table of Contents

List of Figures

User Manual

Chapter 1 - Introduction

Smartmath is a web-based multimedia teaching and learning package developed specially for children in Standard 1 and Standard 2. It provides lessons, interactive quizzes, and games to make math learning simple and interesting.

1.1 Run-time Requirements

Hardware configuration requirements to run Smartmath

a) Type and speed of processor - Pentium 366Mhz and above;

b) Amount of memory - 64 MB RAM and above

c) Size of hard disk - 2GB and above with 500Mb of free disk space ;

d) Operating system - Windows 98 and above;

e) Access to online systems (local network, Internet, World Wide Web, and so on);

f) Speed of network connection - 56K and above;

g) Resolution of the screen - 800 by 600 pixels;

h) Number of colours on the screen - 256 and above;

i) Sound handling - 8-bit and above.

Software configuration requirements to run Smartmath

a) Windows 98 and above

b) Internet Explorer 5.5 and above

1.2 User Manual Overview

Chapter 1 – Introduction

Brief description about Smartmath and the hardware and software configuration requirements.

Chapter 2 – Getting Started

Gives a simple explanation about how to get started with Smartmath.

Chapter 3 – Lessons, Games, Links, and Send this page

Explains the modules of Smartmath.

Chapter 4 – List of terms and definitions

Lists out the relational links, terms, and definitions in the Bahasa Melayu version.

Chapter 2 - Getting Started

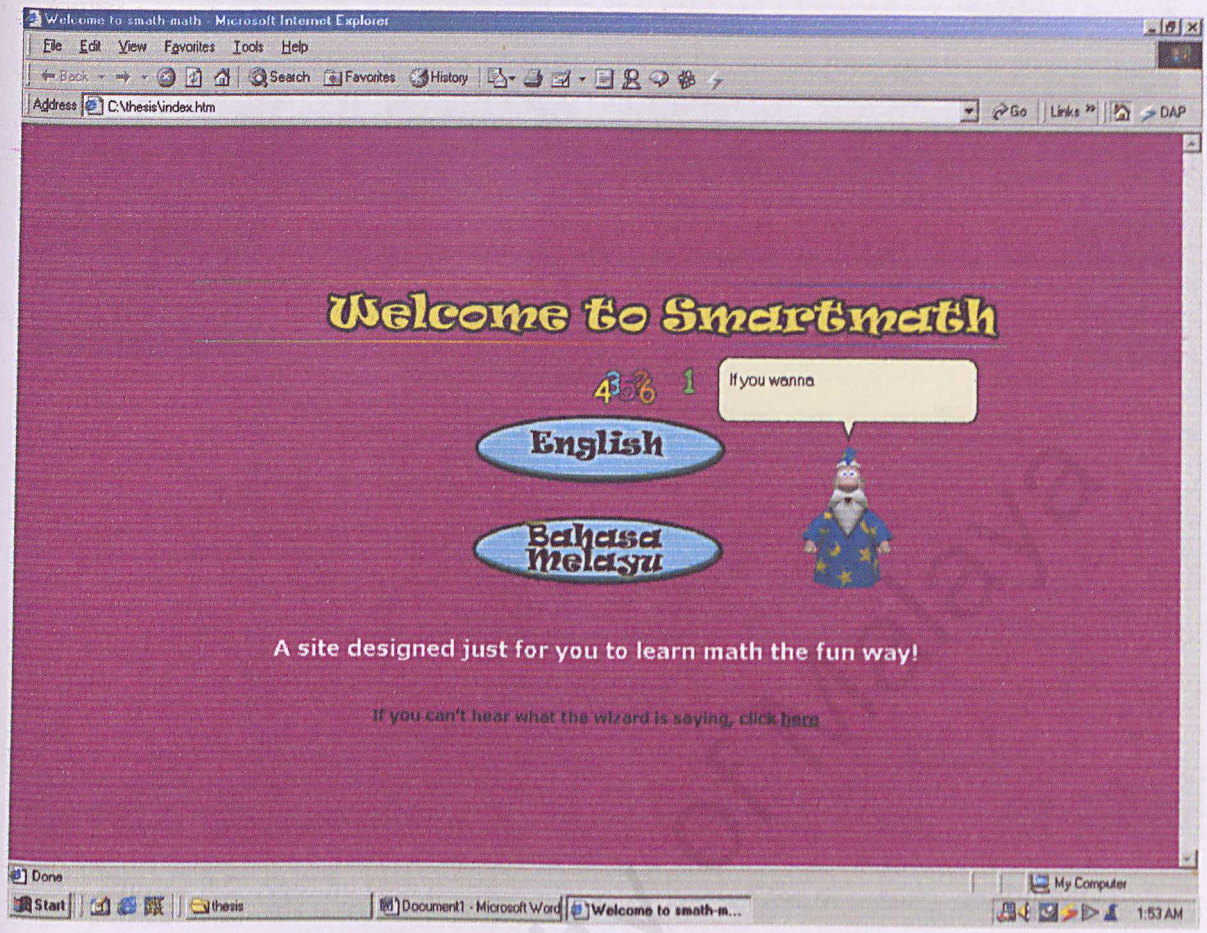


Figure 2.1: Welcome page of Smartmath.

The figure above shows the Welcome page of Smartmath. From here, the user can choose to surf the site in English or Bahasa Melayu. If the user couldn't see the wizard or hear the wizard speak, they can click on the necessary link and they will be brought to the next screen.

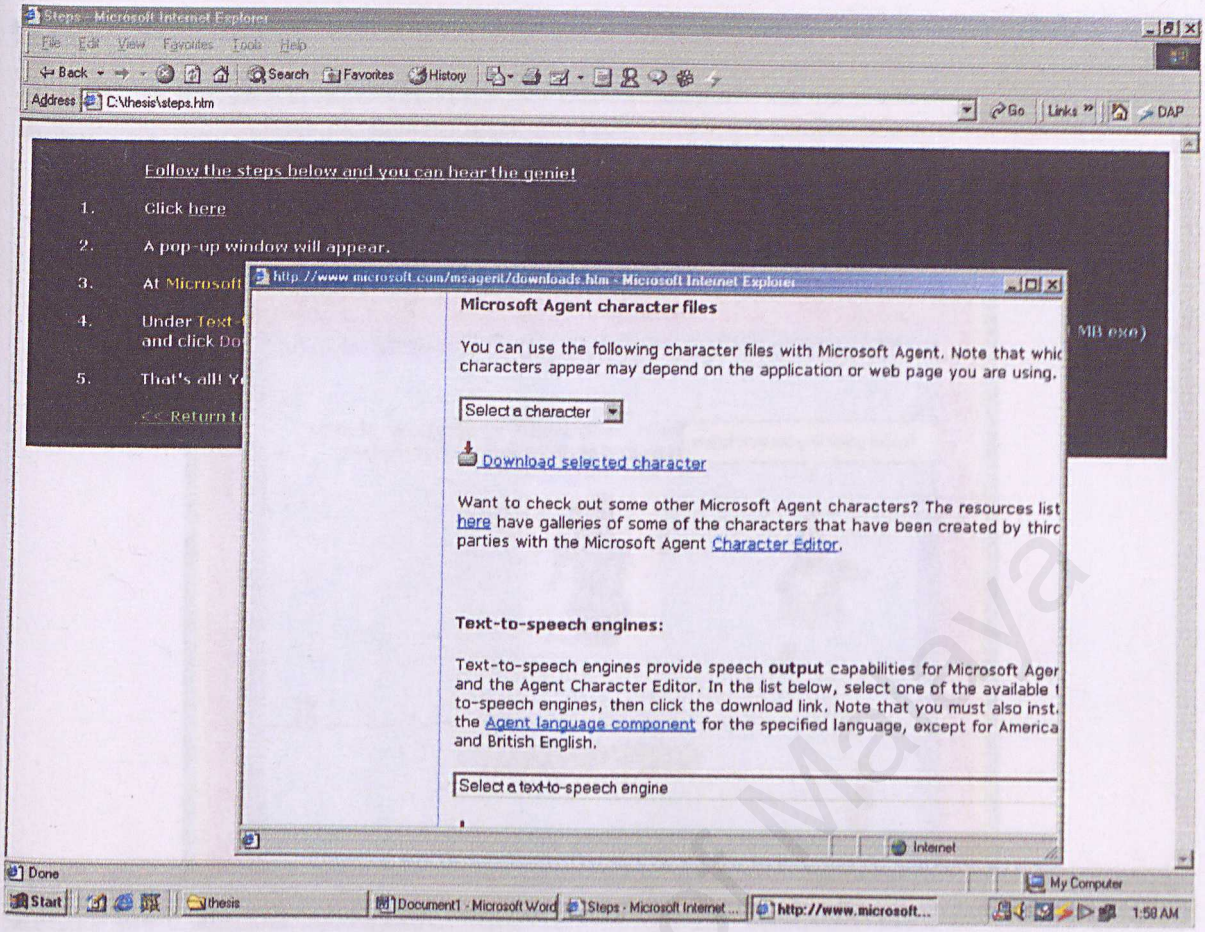


Figure 2.2: Step by step instructions to download the necessary applications

Here, step by step instructions will be given to the user to download the necessary applications so that the wizard can function properly. If a user can see the wizard but couldn't hear it speak, they should download '**Lernout & Hauspie TruVoice Text-To-Speech Engine-American English (1 MB exe)**' under the **Text-to-speech engines** category. If they couldn't see and hear the wizard, they should download both '**Merlin (1.8 MB exe)**' under the **Microsoft Agent character files** and the text-to-speech engine. After installing the necessary applications, they will be able to see and hear the wizard and they can start surfing Smartmath!

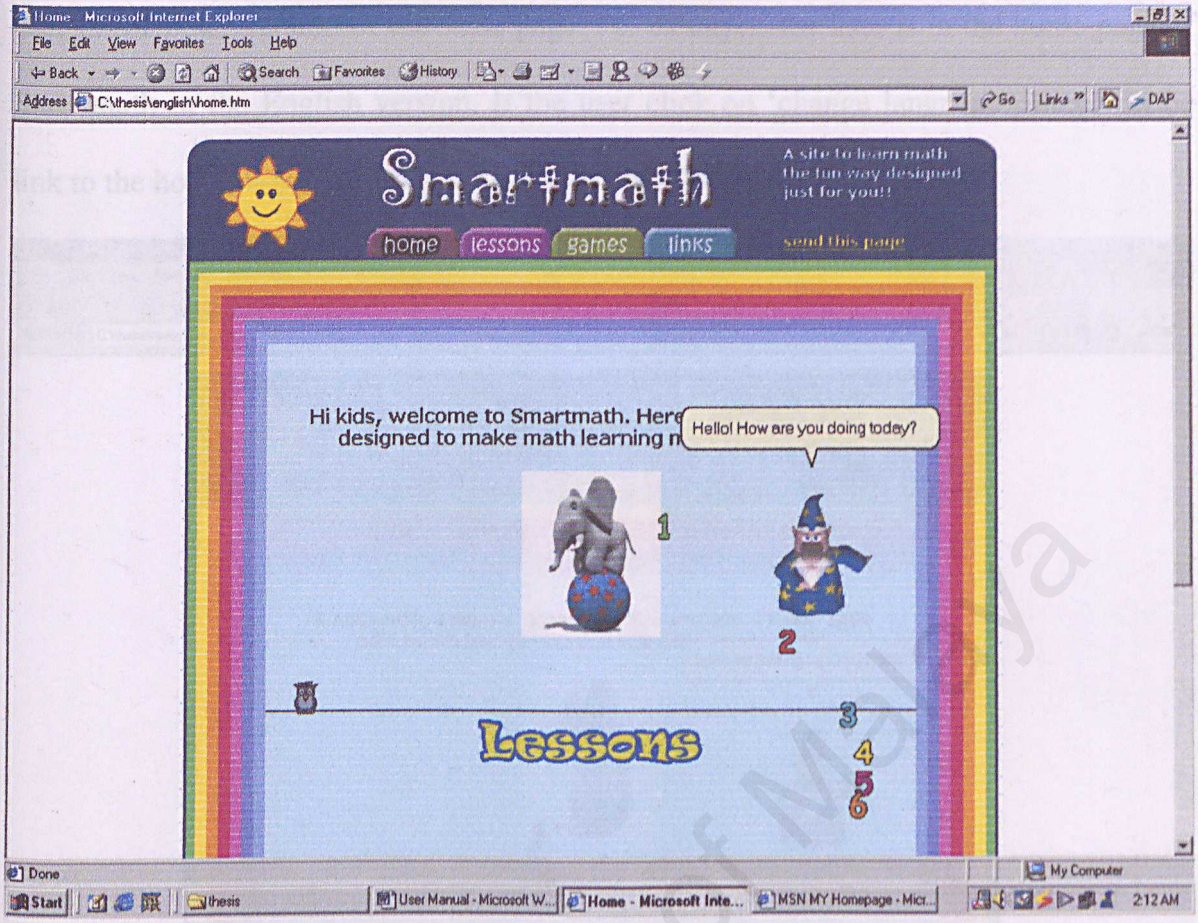


Figure 2.3: Home page of Smartmath (English version)

This is the home page of Smartmath. Users can view the topics of the lessons available by clicking on the 'lessons' button. By clicking on the 'games' button, users can view the games that they can play. The 'links' button will bring users to a page containing links to other educational sites. Users can also click on 'send this page' to the send the URL of this page to their friends.

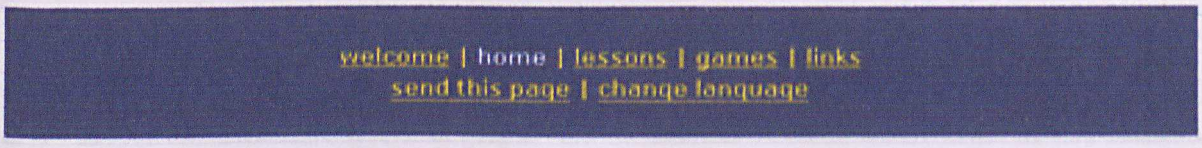


Figure 2.4: Botttom part of a page (English version)

The figure above shows the bottom part of each page. The additional links shown here as compared to the top bar are welcome and change language. By clicking on 'welcome' users will be link to the Welcome page of Smartmath. 'change language' will link the

user to the other version of the same page. For example, the user is in the home page of Smartmath in the English version. If the user click on 'change language', they will be link to the home page of Smartmath in the Bahasa Melayu version.

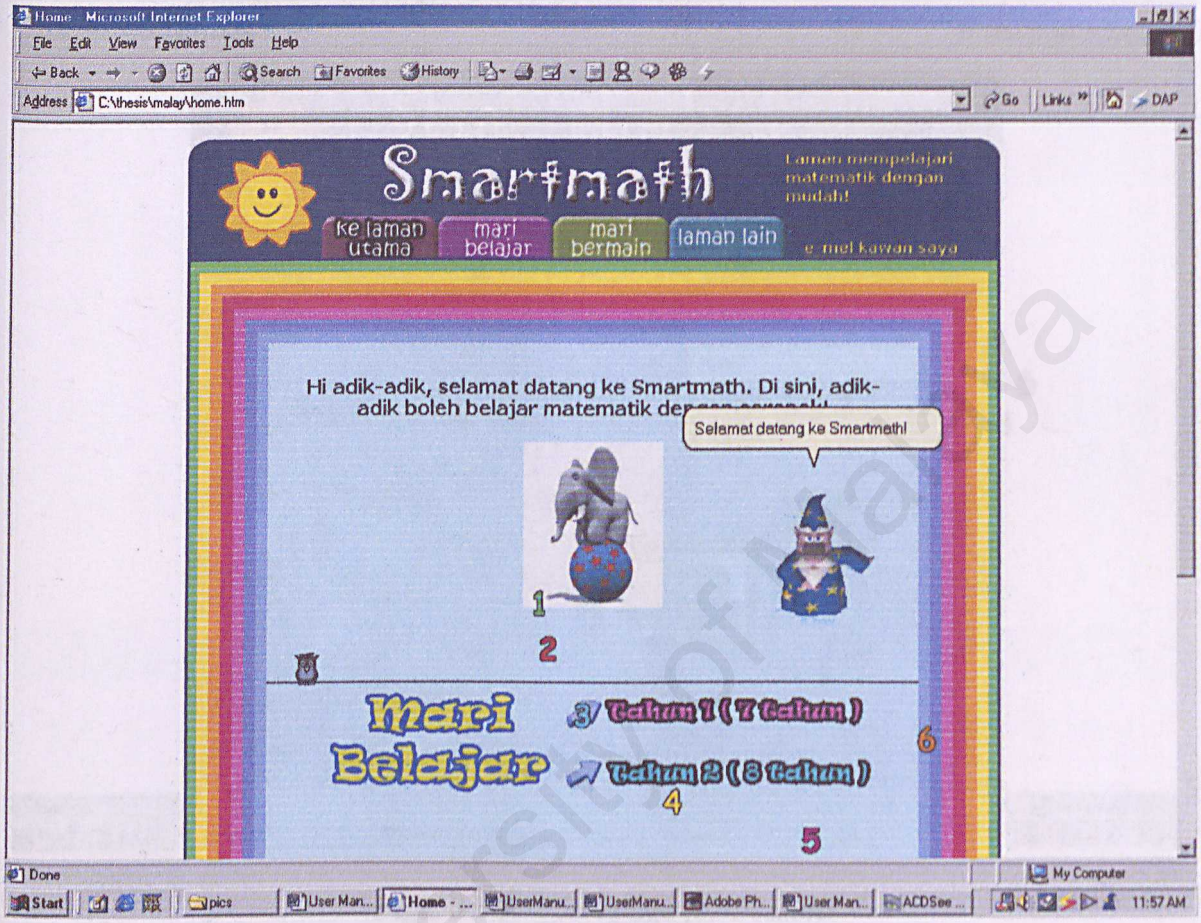


Figure 2.5: Home page of Smartmath (BM version)

To change back to the English version, users can go to the bottom of the page and click 'tukar bahasa'.

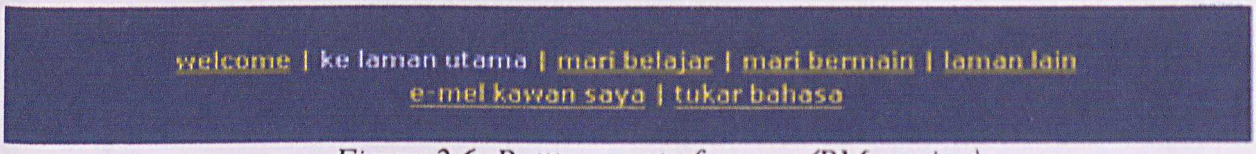


Figure 2.6: Botttom part of a page (BM version)

Chapter 3 - Lessons, Games, Links, and Send this page

3.1 Lessons

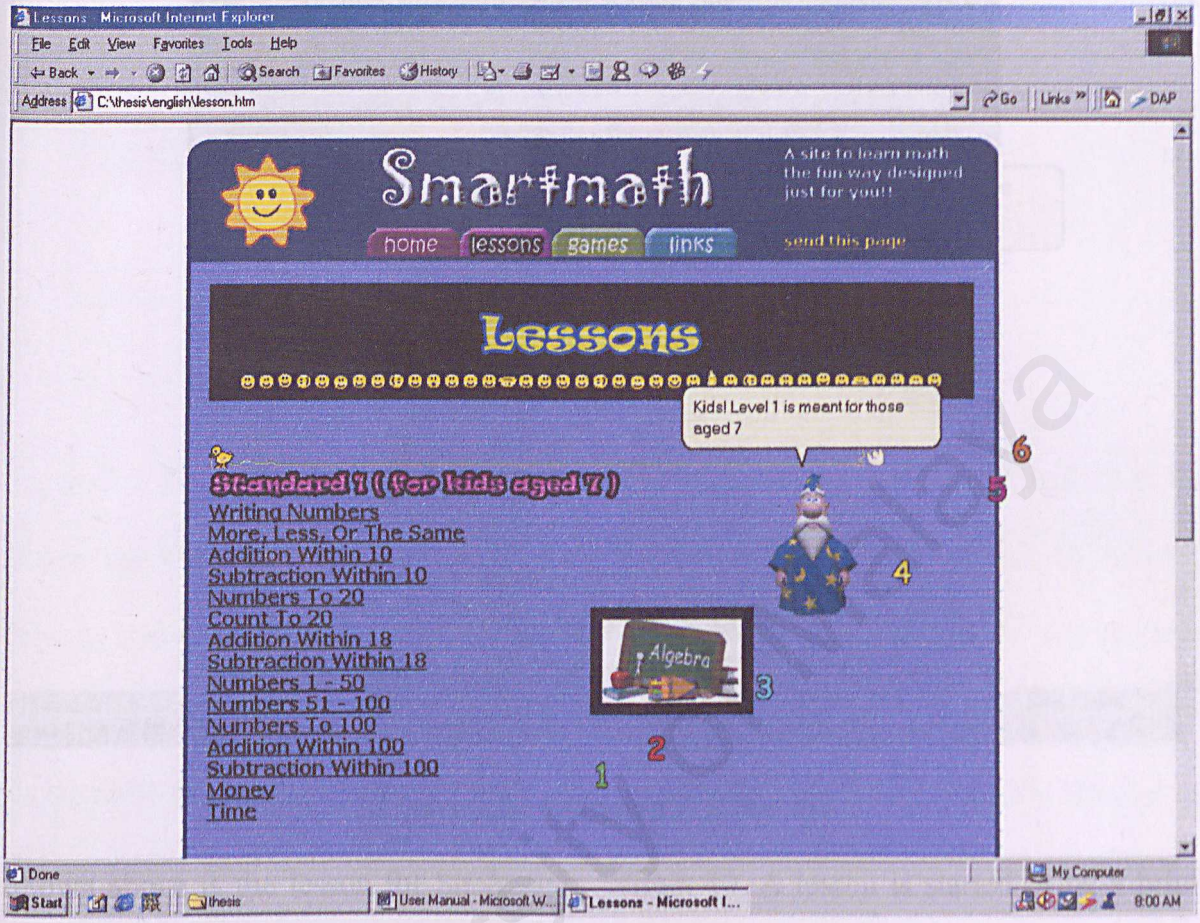


Figure 3.1: Main page for Lessons

Figure 3.1 shows the main page for Lessons. Here, the topics that is available for standard is listed. Users can choose to go to any selected topic by simply clicking on the title. They can also learn the lessons from the beginning and proceed to the next one in the lesson itself. There are 15 topics in each standard.

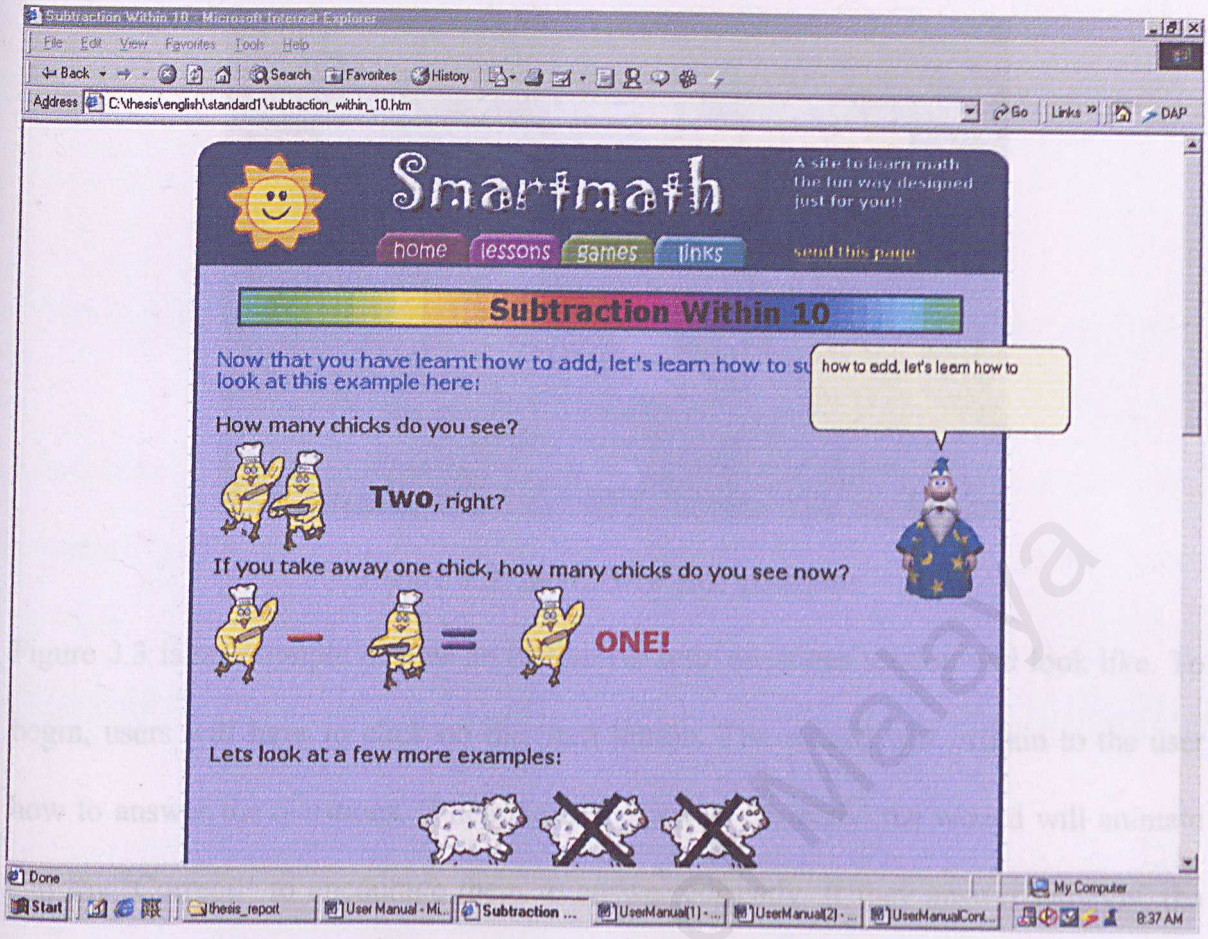


Figure 3.2: An example of lessons

Shown above is the lesson for Subtraction Within 10. A lesson is started of by a brief explanation by the wizard. Then, the user can learn by looking at the explanations in detail and examples given. At the end of a lesson, there is an interactive quiz section which users can practice what they have just learnt.

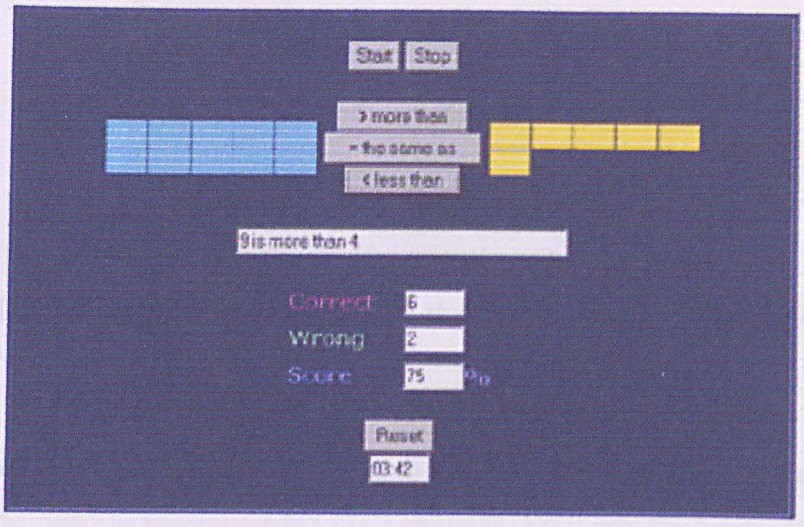


Figure 3.3: Interactive quiz questions

Figure 3.3 is an example of how an interactive quiz questions section will look like. To begin, users will have to click on the Start button. The wizard will explain to the user how to answer the questions. When the user answers correctly, the wizard will animate and say 'Correct!' to encourage them to answer correctly. If they answer wrongly, the wizard will also animate and say 'Oops!' and an alert message box will appear displaying the correct answer.

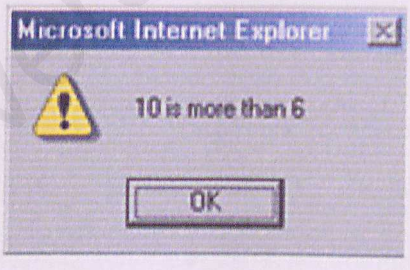


Figure 3.4: Alert message box that displays the correct answer

The number of correct and wrong answers will be displayed. The score will also be displayed. Users can end the interactive quiz section by clicking on the Stop button. An alert message box stating how many correct and wrong answers they manage to answer within how many tries will be displayed.

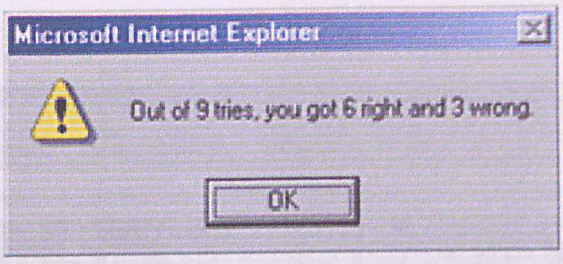


Figure 3.5: An alert message box indicating the number of tries, the number of correct and wrong answers

Users can click on the Start button or Reset to begin answering the interactive quiz again. Users can go the other lessons by clicking on the '<< previous lesson' and 'next lesson >>' link.

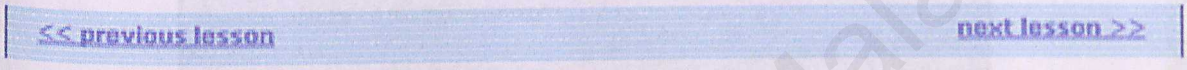


Figure 3.6: Link to previous lesson and next lesson

3.2 Games

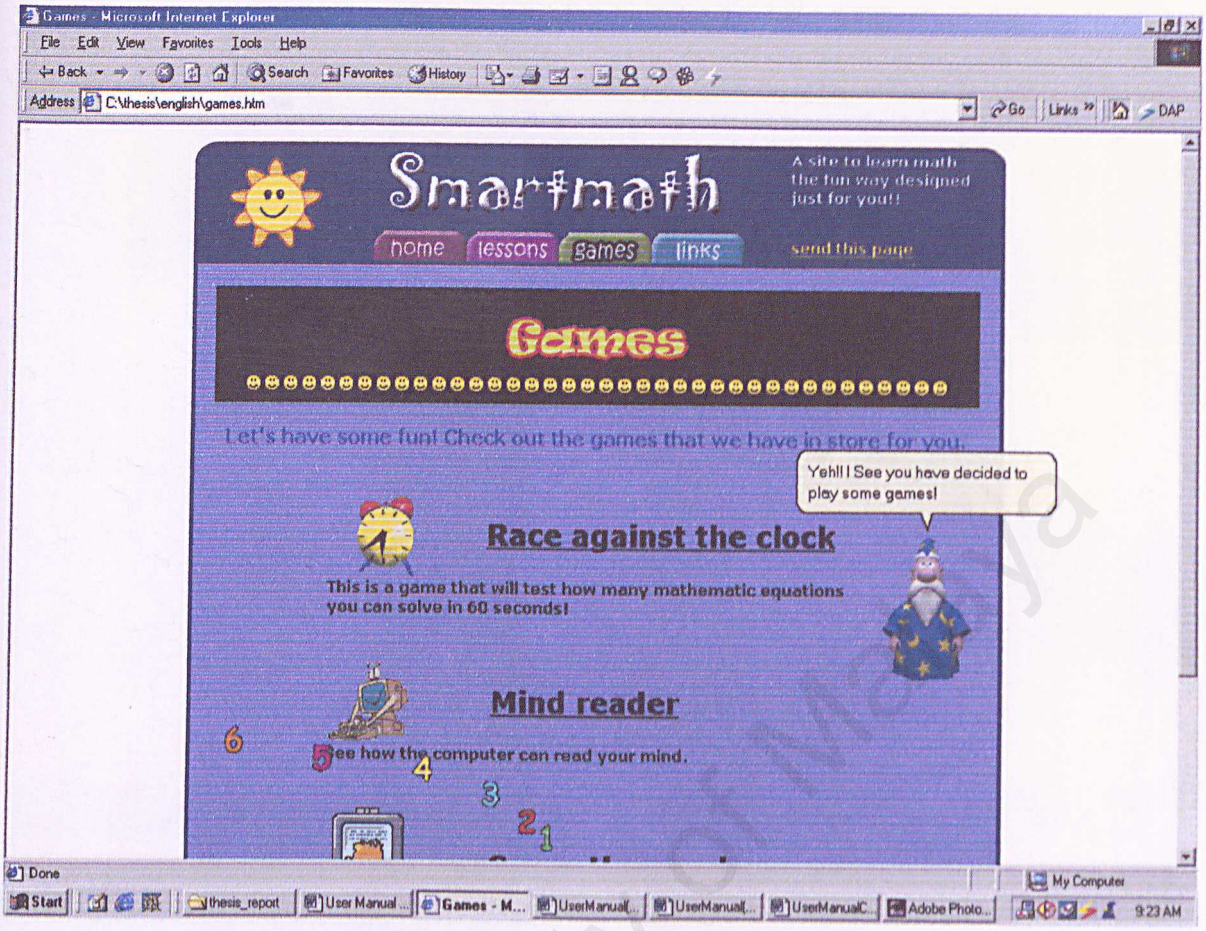


Figure 3.7: Main page for Games

Shown above is the main page for Games. There are 3 types of games that the user can choose from, Race against the clock, Mind reader, and Guess the number.

3.2.1 Race against the clock

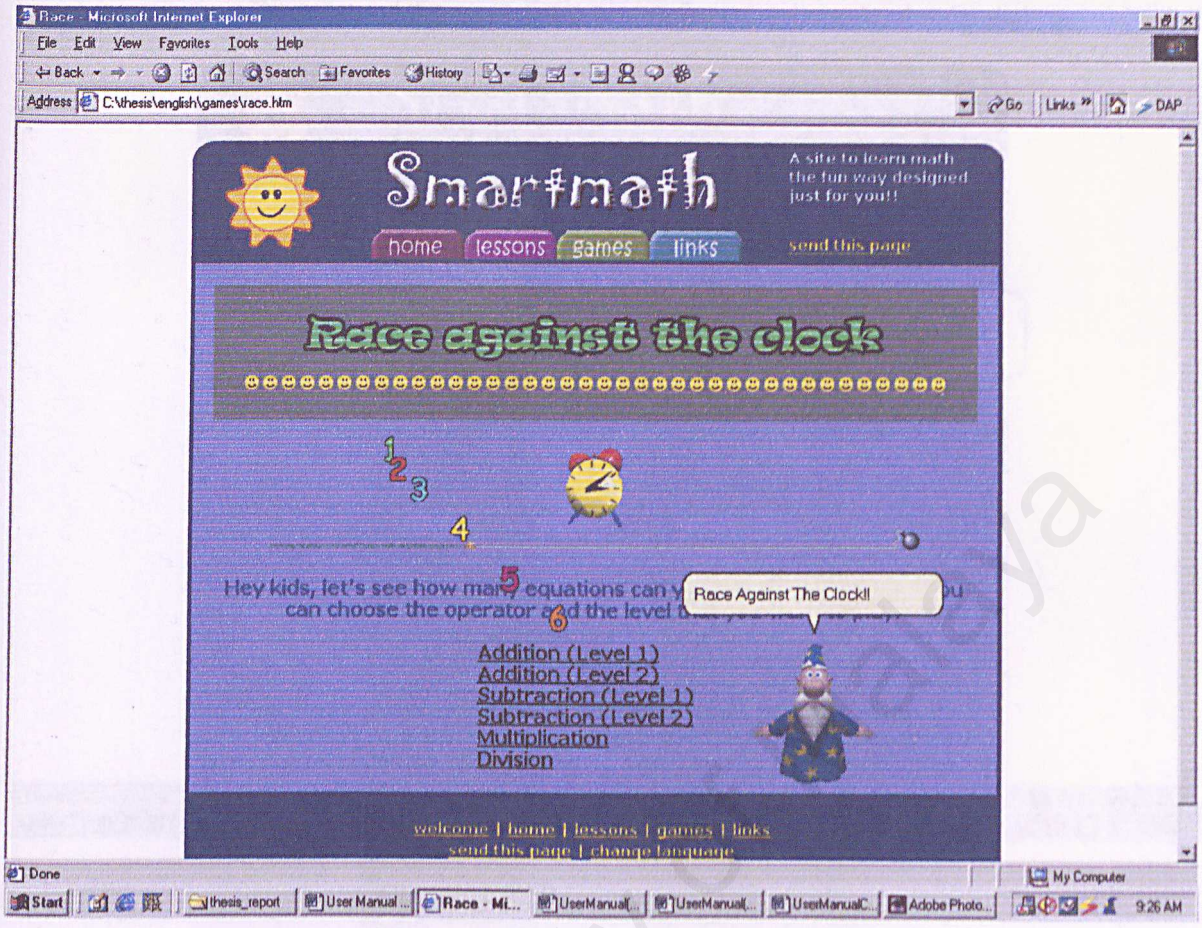


Figure 3.8: Menu page for Race against the clock

Race against the clock is a game to test the user's mathematic skills. In this game, users will have to answer as many questions as they can in 60 seconds. They can choose the level and mathematic operation that they want to play.

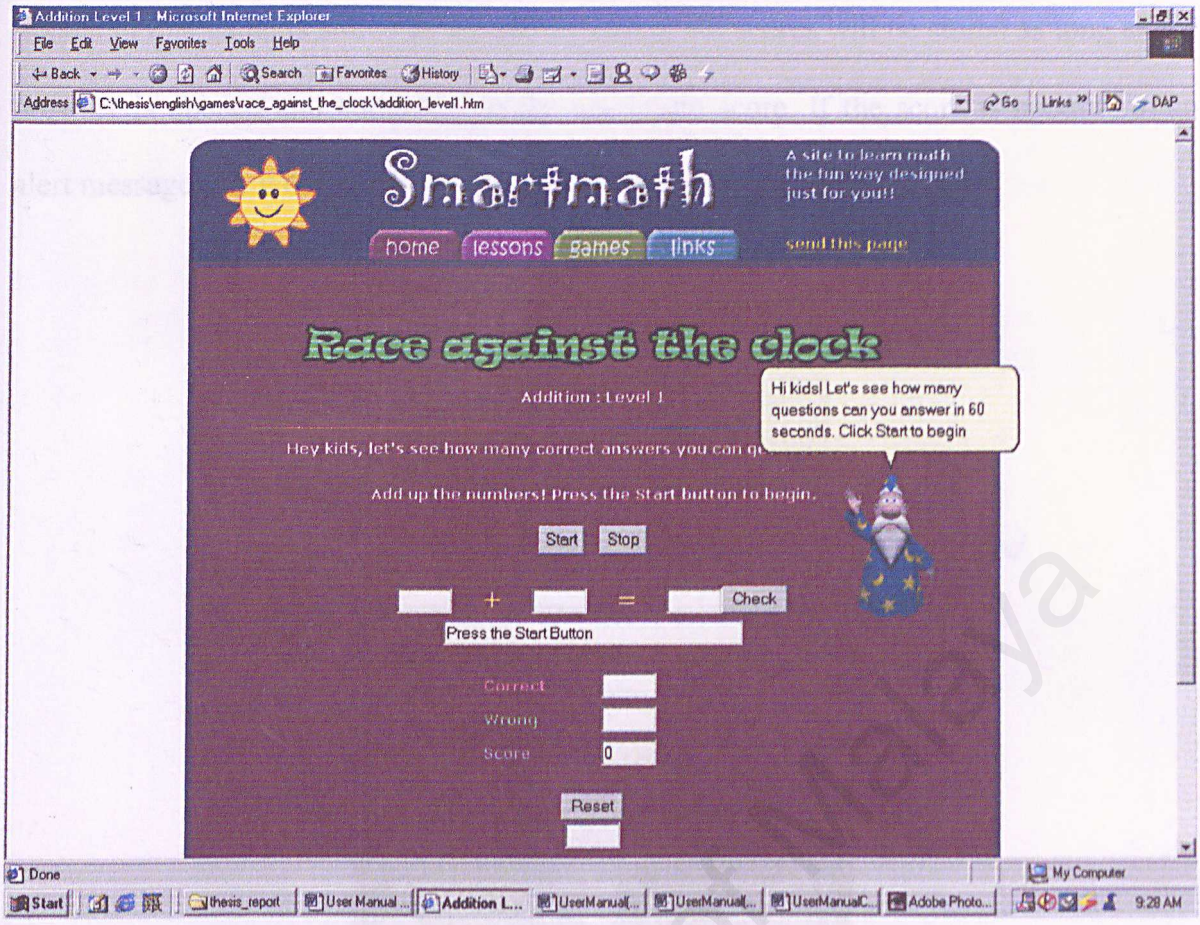


Figure 3.9: Race against the clock

Shown above is the screen for 'Race against the clock'. Here, the wizard will give explanations so the user knows how to play the game. The feedback from the wizard when the user answers correctly are 'Good job!', 'Correct!', 'You are right!', 'Well done!' which will appear randomly. The timer start at 60 seconds and at the end of it, the user's score will be displayed.

Figure 3.10: Alert message box displaying the score

Scores in this game is stored as temporary string. All scores will be stored as long as the browser is opened and compared to the maximum score. If the score is maximum, an alert message box will appear and the wizard will animate.

Figure 3.11: Alert message box stating a new record is created

Figure 3.12: Feedback from the wizard when a new record is created

To start playing the game again, users can click the Start or Reset button.

3.2.2 Mind reader

Figure 3.13: Mind reader

Figure 4.3 above shows the 'Mind reader' game. The wizard will also give instructions on how to play the game. First of all, users should think of a number between 1 and 63 and click Start. Six cards will be displayed consecutively. Users can click 'Yes' if the number is on the card or click 'No' if otherwise. After the sixth card, the number will be displayed.

3.2.3 Guess the number

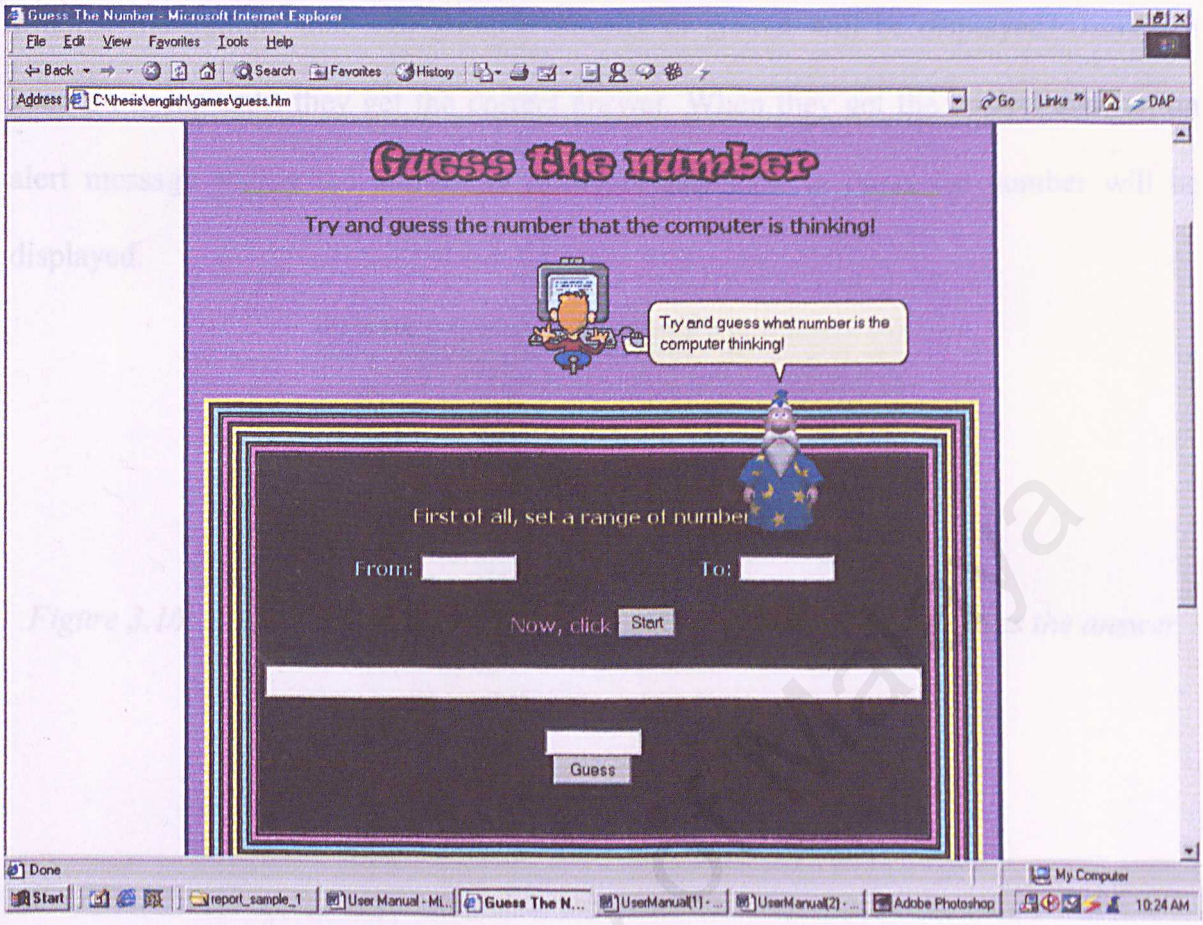


Figure 3.14: Guess the number

Figure 4.4 above shows the 'Guess the number' game. The wizard will guide the users on how to play the game. First of all, the user should set a range of numbers for the computer to generate a random number. If the user set a range that is not valid, an alert message will appear and the user will have to set another range.

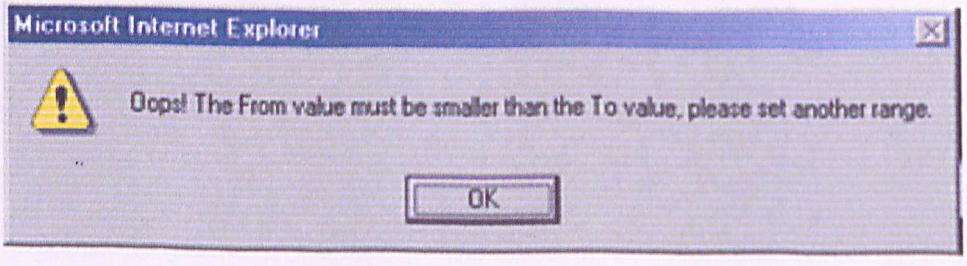


Figure 3.15: Alert message box indicating the range set by the user is not valid.

If the range is valid, users can start guessing the number. Feedback telling the user whether the number that they guess is smaller or greater will be displayed. Users can keep on trying until they get the correct answer. When they get the correct answer, an alert message stating the number of attempts they took to guess the number will be displayed.

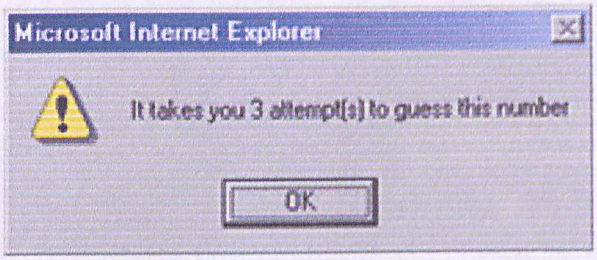


Figure 3.16: Alert message box indicating the number of attempts to guess the answer.

3.3 Links

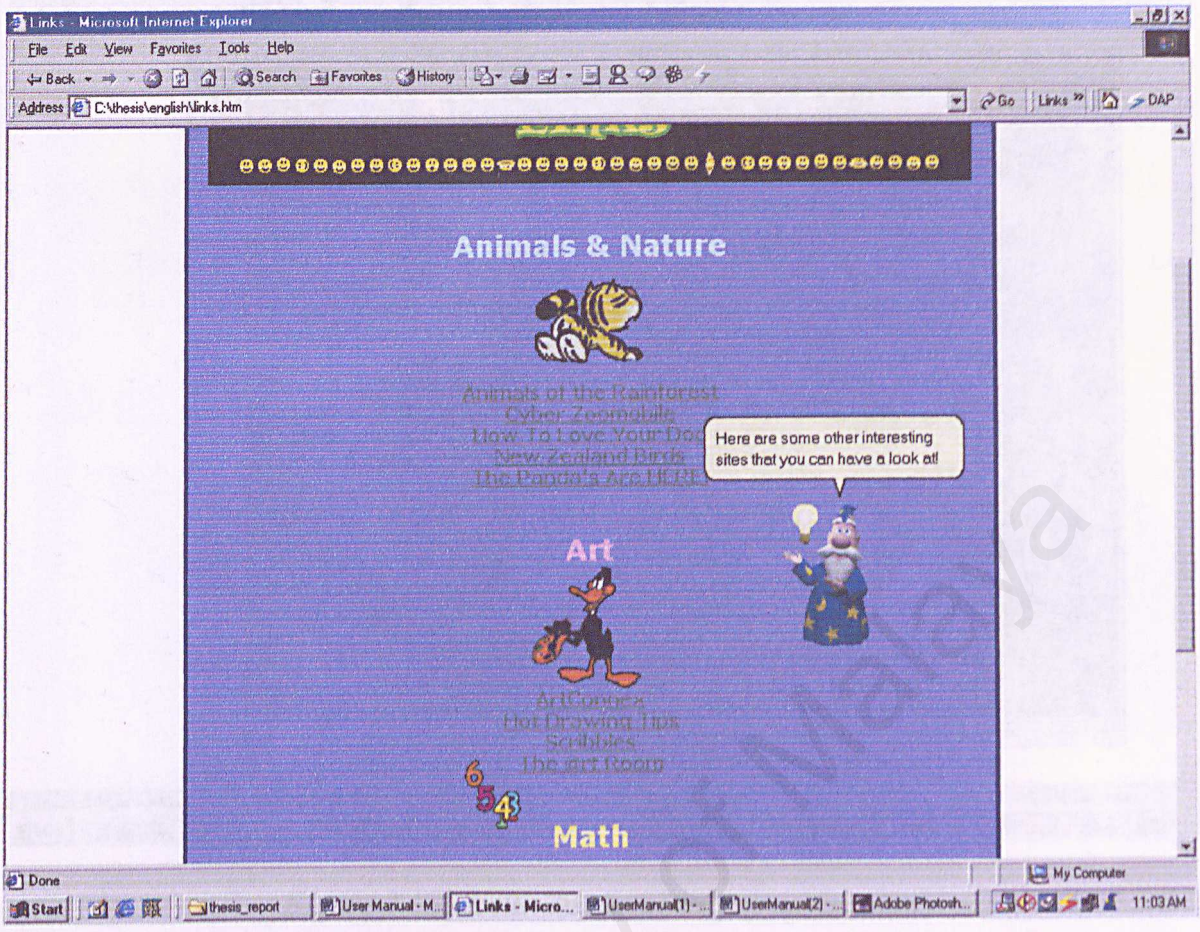


Figure 3.17: The Links page

At this page, the links to other educational will be listed. There are 3 categories that the users can choose from, Animals & Nature, Art, and Math. The site will be displayed as a pop-up window as shown in Figure 3.18.

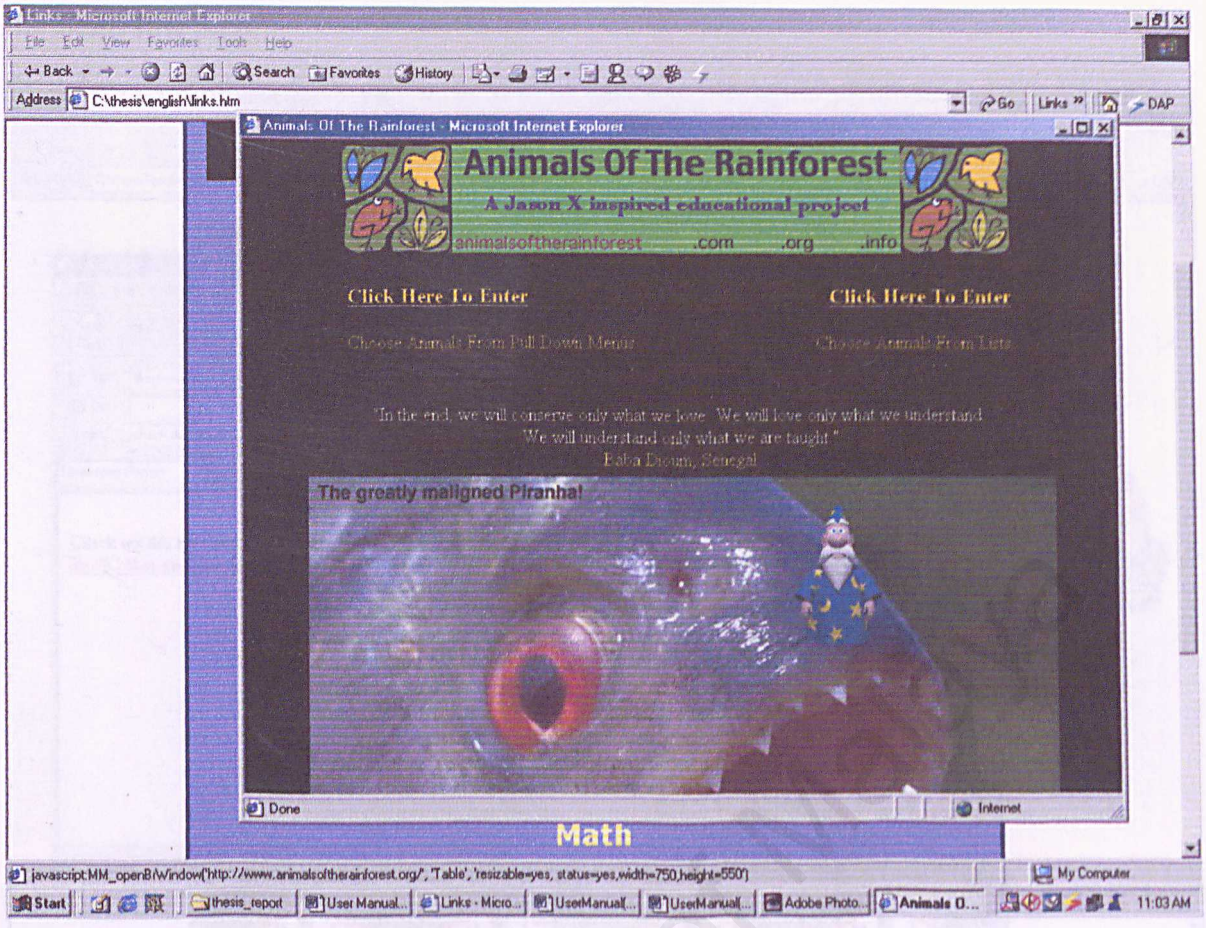


Figure 3.18: A site displayed as pop-up window

3.4 Send this page

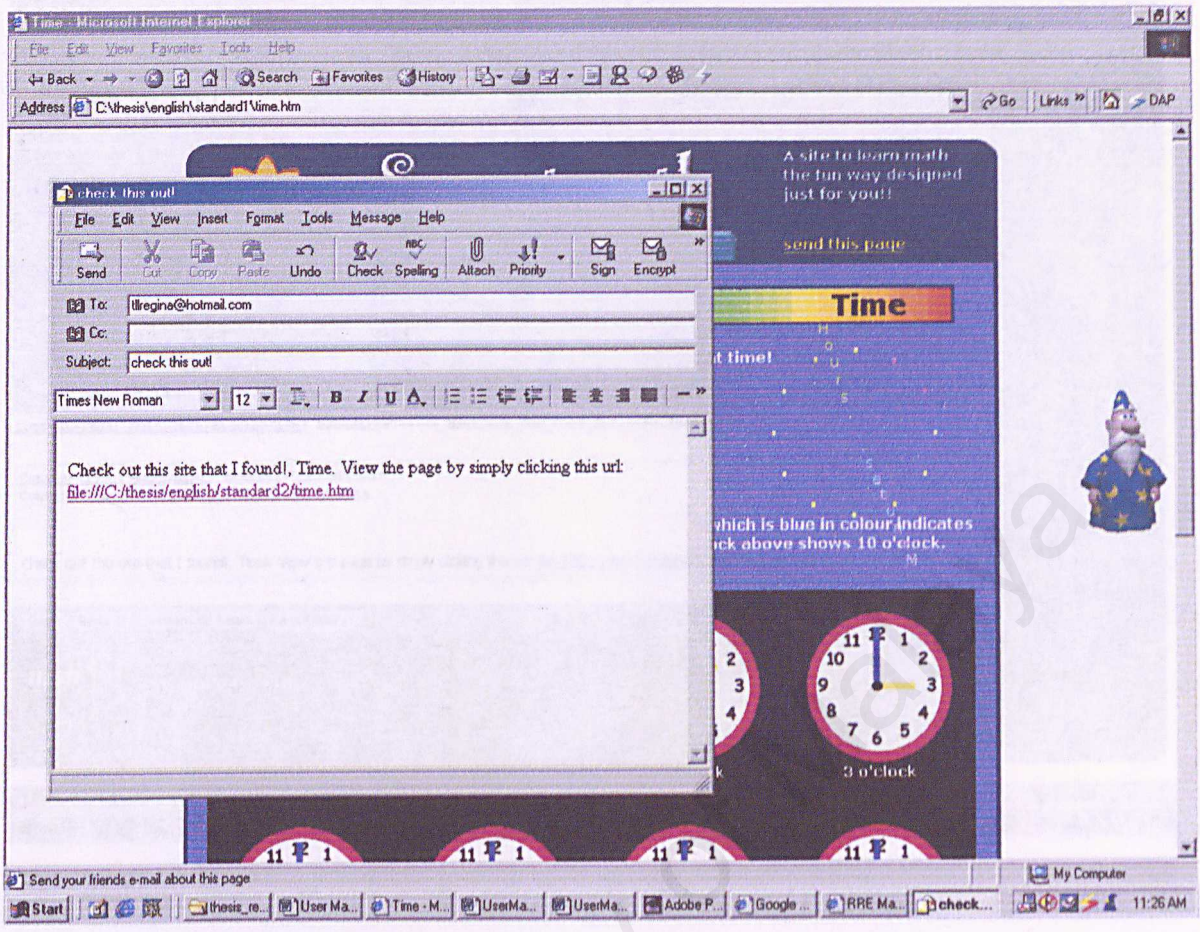


Figure 3.19: Send this page

When the user click on the 'send this page' link, a new Microsoft Outlook message will appear. An example of the default message is:

- Check out this site that I found!, Time. View the page by simply clicking this url: file:///C:/thesis/english/standard2/time.htm (English version)
- Cuba layari laman ini!, Time. dengan klik url ini: file:///C:/thesis/malay/standard1/time.htm (BM version)

Users should input the email address of the recipient and add in extra messages which is optional and the message will be sent to the recipient.

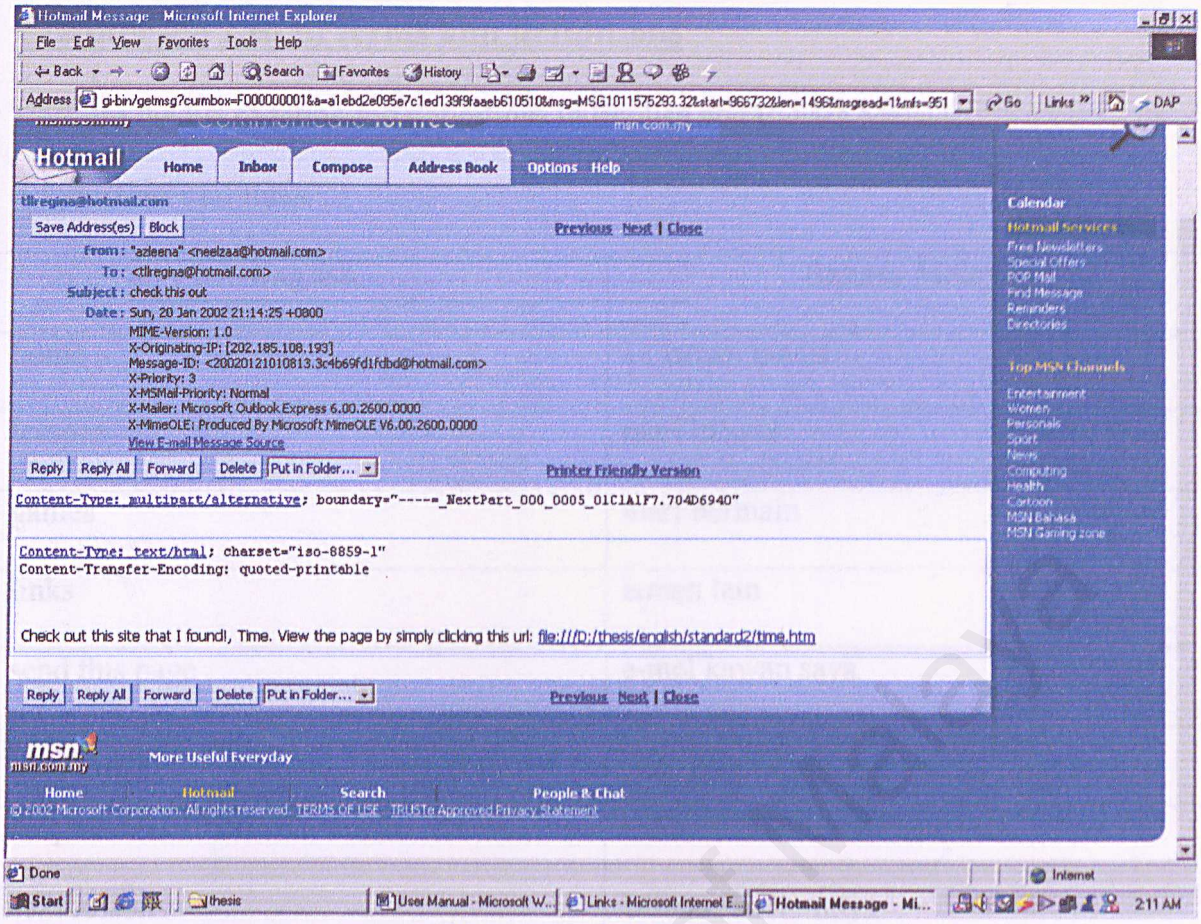


Figure 3.20: An example of the email received

Figure 3.20 shows an example of the email received.

Shown in this user manual is the flow of Smartmath in the English version. The flow and design of Smartmath for the Bahasa Melayu is exactly the same. The difference between these 2 versions are the names of the buttons, hyperlinks, and explanations. The next chapter summarizes the names of the buttons, links, topics in lessons, and types of games for both languages for quick reference.

Chapter 4 – List of terms and definitions

Here are the list of terms and definitions in English and Bahasa Melayu.

4.1 Buttons and links

English	Bahasa Melayu
home	ke laman utama
lessons	mari belajar
games	mari bermain
links	laman lain
send this page	e-mel kawan saya
change language	tukar bahasa
<< previous lesson	<< latihan sebelum ini
next lesson >>	latihan sebelumnya >>
> more than	> lebih daripada
= the same as	= sama dengan
< less than	< kurang daripada
Start	Mula
Stop	Tamat
Check	Semak
Yes	Ya
No	Tidak
Restart	Semula
Guess	Teka

4.2 Lessons

English	Bahasa Melayu
Standard 1 (for kids aged 7)	Tahun 1 (7 tahun)
Writing Numbers	Menulis Nombor
More, Less Or The Same	Lebih Daripada, Kurang Daripada, Atau Sama Dengan
Addition Within 10	Tambah Dalam Lingkungan 10
Subtraction Within 10	Tolak Dalam Lingkungan 10
Numbers To 20	Nombor Sehingga 20
Count To 20	Mengira Sehingga 20
Addition Within 18	Tambah Dalam Lingkungan 18
Subtraction Within 18	Tolak Dalam Lingkungan 18
Numbers 1 - 50	Nombor 1 - 50
Numbers 51 - 100	Nombor 51 – 100
Numbers To 100	Nombor Sehingga 100
Addition Within 100	Tambah Dalam Lingkungan 100
Subtraction Within 100	Tolak Dalam Lingkungan 100
Money	Wang
Time	Masa

English	Bahasa Melayu
Standard 2 (for kids aged 8)	Tahun 2 (8 tahun)
Addition Within 100	Tambah Dalam Lingkungan 100
Subtraction Within 100	Tolak Dalam Lingkungan 100
Multiplication Till 9 X 5	Darab Hingga 9 X 5
Division Till 45 ÷ 5	Bahagi Hingga 45 ÷ 5
Numbers To 300	Nombor Sehingga 300
Addition Within 300	Tambah Dalam Lingkungan 300
Subtraction Within 300	Tolak Dalam Lingkungan 300
Multiplication Till 9 X 9	Darab Hingga 9 X 9
Division Till 81 ÷ 9	Bahagi Hingga 81 ÷ 9
Numbers To 500	Nombor Sehingga 500
Addition Within 500	Tambah Dalam Lingkungan 500
Subtraction Within 500	Tolak Dalam Lingkungan 500
Money (Adding Coins)	Wang (Menambah syiling)
Money (Adding Notes)	Wang (Menambah wang ringgit)
Time	Masa

4.3 Games

English	Bahasa Melayu
Race against the clock	Menentang masa
Mind reader	Peneka minda
Guess the number	Meneka nombor