Web-Based Information System For Rural Community In Kalutara District

By

H.A.C.S.Hettiarachchi

00/AS/002

Dissertation submitted in partial fulfillment of the requirements for the Degree of Bachelor of Science In Physical Sciences

Faculty of Applied Sciences

Sabaragamuwa University of Sri Lanka

Buttala

February 2004

DECLARATION

I hereby declare that the work reported in the project report was exclusively carried out by me, under the supervision of Prof. Lal Bopearachy and Mr.Jayalath Ekanayake. Any part of this project report has not been submitted earlier or concurrently for same or any other degree.

10/5/2004
Date

H.A.C.S.Hettiarachchi

To the best of my knowledge the above particulars are correct.

External Supervisor

Prof. Lal Bopearachy
Team Leader,
Community Information Network Project
2nd Floor, "Isurupaya"
Battaramulla.

Signature
Date: 11/05/2004

Internal Supervisor

Mr.Jayalath Ekanayake.

Lecturer,
Department of Physical Sciences,
Faculty of Applied Science,
Sabaragamuwa University of Sri Lanka.

Signature
Date: 12/05/2004

Head of the Department

Dr,(Mrs). N.Wickramaratne.
Head/Dept Physical Sciences,
Faculty of Applied Science,
Sabaragamuwa University of Sri Lanka.

Signature
Date: 12/05/2004

ACKNOWLEDGEMENT

First and foremost I wish to express my deepest gratitude to my internal supervisor Mr.Jayalath Ekanayake, Lecturer, Department of Physical Sciences, Faculty of Applied Sciences, Sabaragamuwa University of Sri Lanka for his assistance, encouragement, guidance and his valuable time to make this study a success.

I express my sincere gratitude to my external supervisor Prof.Lal Bopearachy, Team Leader, Community Information network project, for assistance and guiding me towards a successful completion.

My special award of thank to Dr.D.B.M.Wickramarathne, Dean, Faculty of Applied Sciences, Dr.(Mrs).N.Wickramarathne, Department Head,Department of Physical Sciences, Faculty of Applied Sciences, Sabaragamuwa University of Sri Lanka for all kind of guidance and helpfulness.

Finally my deepest gratitude is express to Mr. Percy Perera, IT Manager, TEAMS Consultants (Pvt) Ltd for providing the opportunity to carryout this study under Community Information Network project.

ABSTRACT

With the latest developments in Sri Lanka there was an increasing demand for the use of Information technology in the day-to-day life. Though the demand was very high the knowledge and the information technology facilities had not reached for almost entire community of Sri Lanka, especially the rural and the poor community living in Kalutara District. The people in Kalutara District did not use the technologies such as Internet, E-Mail, Net-meeting, due to the high cost.

The main objective of this project was to develop a web-based information system for the Kalutara District. This system created opportunities for the general public especially lower income groups and disabilities, to participate in the process of economic development by providing vital information accurately and quickly using advance communication technology. Moreover the information and other services provided free of charges.

To meet this requirement a web site was developed using Dreamweaver MX software,. Forms were designed using Active Server Page technology that enabled to enter details of the questionnaires to a database. The graphics were designed by using Fireworks MX, Photoshop 7 and CorelDraw 11. The web site was uploaded to Internet by using ws_ftp pro.

The result of this project was a web-based information system for the rural community in Kalutara District. This information system provided an easy access for vital information about Kalutara District, services such an Internet, E-Mail, Net-meeting and Information about all over the world. There were ten Village Information Centers (VIC) and one Community Information Centre (CIC) at Horana. The community in this area could use either one of VIC or CIC to log on to www.kalutara.org web site. The information on the web site was published in three main languages that were commonly used in Sri Lanka.

A search engine could be added to provide search facilities and an online registration facility could be provided. A web server could be maintained at the Community Information Centre and the web site also could be hosted from there.

CONTENTS

LIST OF FIGURES

ABBREVIATIONS

ADB	Asian Development Bank
ASP	Active Sever Page
CD	Compact Disc
CIC	Community Information Center
CSS	Cascading Style Sheets
DHTML	Dynamic Hyper Text Markup Language
DMC	Developing Member Country
E-Commerce	Electronic Commerce
E-Governance	Electronic Governance
HTML	Hyper Text Markup Language
ICT	Information and Communication Technology
IIS	Internet Information Services
IT	Information Technology
JSP	Java Server Page
SQL	Structured Query Language
VIC	Village Information Center

1. Introduction

1.1 Background

With the latest developments in Sri Lanka there is an increasing demand for the use of Information technology in the day-to-day life. Therefore the community is seeking for new information to meet with their day-to-day requirements in almost in every field. Though the demand is very high, the knowledge and the information technology facilities have not yet reached the entire community of Sri Lanka, specially the rural and the poor community. The cutting edge technologies such as Internet, E-Mail, Net-meeting, chatting have not reached the under privileged community so far as due to the cost of the use in these technologies are very high. Therefore as apart of the development process it is important to disseminate the knowledge in information technology (IT) to the under privileged communities. The Asian Development Bank has taken up the mission of dissemination of IT knowledge to the under privileged community and has selected the Kalutara, Ratnapura and Gampaha District as for their initial mission launching areas.

1.2 Objective

The main objective of this project was to develop a web-base information system and a community information network for the Kalutara District to reduce poverty in Kalutara District.

The web site was designed according to the requirements of the people in Kalutara District. Further 10 Village Information Centers (VIC) was proposed within the villagers in Kalutara District so that the villages can visit to either one of VIC or the Community Information Center (CIC) and log on to the web site that would enable them to obtain the information required.

1.3.1 Asian Development Bank at a glance

Asian Development Bank (ADB) is a multilateral development finance institution dedicated to reducing poverty in Asia and the Pacific. ADB was established in 1966 and now they have 63 members, mostly from the region. The headquarters of ADB is in Manila.

ADB is a non-profit, multilateral development finance institution that engages in mostly public sector lending for development purposes in its developing member countries. ADB's clients are its member governments, who are also its shareholders.

ADB's overarching goal is to reduce poverty in Asia and the Pacific. It helps to improve the quality of people's lives by providing loans and technical assistance for a broad range of development activities.

The Asian Development Bank extends loans and equity investments to its developing member countries (DMCs) for their economic and social development and provides technical assistance for the planning and execution of development projects and programs and for advisory services.

ADB promotes and facilitates investment of public and private capital for development, and responds to requests for assistance in coordinating development policies and plans of its developing member countries.

1.3.2 IT development in Sri Lanka

The government of Sri Lanka has developed in 2003 a vision statement titled "e-Sri Lanka", an information and communication technology (ICT) development road map. The goal is to take information and communication technology (ICT) to the village, and to focus on the government development activities and various facilities to the general public. Out of the total population of Sri Lanka, 40% is classified as poor or vulnerable to poverty. Poverty is predominantly a rural phenomenon, with nearly 90% of the poor residing in rural areas where vital information for survival and additional income is lacking. Therefore, it is very difficult for the poor to share the benefits of economic growth, as they do not know what opportunities are available for them.

1.3.3 Issues

About 40% of Sri Lanka's population can be classified as poor or vulnerable to poverty. This level of poverty has remained constant during the 1990s, despite a real gross domestic product growth rate of 5% per annum. Moreover, the benefits of growth are not being evenly distributed throughout the country, as the most growth has been concentrated in the Western Province. Poverty is predominantly a rural phenomenon, with nearly 90% of the poor residing in rural areas where vital information for survival and additional income is lacking. Therefore, it is very difficult for the poor to share the benefits of economic growth, as they do not know what opportunities are available for them.

ICT has enormous potential to improve the livelihoods of low income people by reducing the cost of providing services to traditionally marginalized communities and facilitating the build up of constructive social capital. ICT contributes to poverty reduction by enhancing the efficiency of the economy, enabling better delivery of public services, and creating new employment opportunities for the poor and disabled. Access to information plays a crucial role in improving living conditions of the poor. ICT promotes integration of isolated communities into the global economy. ICT improves the delivery of public services, which is especially important for rural and isolated communities. However, many poor and disabled people in Sri Lanka are not able to benefit from ICT because of cultural barriers, low income, and lack of suitable facilities. Due to high Internet access fees and low computer ownership, Internet use per 100 inhabitants is 0.78 in 2001, which needs to be increased.

About 900,000 – 1.4 million people are disabled in Sri Lanka. In addition, about 10,000 soldiers were disabled as a result of 20 years of internal conflict. About 47% of the disabled population is under 14 years of age and their disabilities are related to prevailing conditions of poverty and malnutrition. In Sri Lanka, people with disabilities who are also poor have had little access to educational and training programs suited to their

special needs. In a country with a high unemployment rate, disabled people have few opportunities to compete for jobs and training programs. As computer – related jobs are new and relatively few people have been trained in the country, some disabled people could have a good chance of being employed if trained appropriately in information technology. ICT can help overcome geographical barriers to communication, simulate expensive and risky laboratory experiments at low cost and no personal risk, provide tutoring, and offer a window to unforeseen markets. Experience in other countries show that disabled people have benefited from the freedom, support, and opportunities offered through the use of computers. However, Sri Lanka has few training centers that serve the poor and disabled.

ADB's assistance has been focused on reducing poverty in many sectors, but the poor have not been provided vital information so they can plan effectively to escape the poverty trap. ADB and the World Bank promote computer literacy in the schools, but access is limited (only 15 computers each to 1,200 of Sri Lanka's 10,000 schools) and computer teaching is basic. No attempt has been made to provide vital information through the Internet to the rural poor, because of relatively high costs and lack of experience. A pilot study is needed to test the effectiveness of information service to the poor and disabled people.

The most important determinants for the high incidence of poverty in Sri Lanka have been the 20-year war and a legacy of poor economic management. The Government is fully committed to overcoming both of these challenges. The Government's long-term goals are to achieve prosperity through sustainable development, including a 50% decline in the level of absolute poverty. The Government and ADB agreed to the following strategic focus to reduce poverty: (I) creating opportunities for pro-poor growth, (II) strengthening the social protection system, and (III) empowering the poor and strengthening the governance. ADB agreed to consolidate its assistance to seven sectors, including education, which contributes greatly in reducing poverty in Sri Lanka.

The Government drafted a plan to use ICT to develop Sri Lanka's economy, reduce poverty, and improve the quality of life and the opportunities for all people. ADB's country operational strategy emphasizes the importance of human development to promote continued economic growth and closer integration into the global economy. ADB's Policy on Education places high priority on improving the quality of education; increasing equity, access, and retention. Especially for the poor, women and other marginalized groups; and strengthening management, governance and efficiency. The proposed TA is fully in line with ADB's education policy and country strategy in that it aims to improve the quality of education, reduce digital divide, and develop human resources for the information age.

2. Literature Review

2.1.1 Electronic Commerce (e-commerce)

Electronic-commerce: way of doing real-time business transactions via telecommunications networks, when the customer and the merchant are in different geographical locations. Electronic commerce is a broad concept that includes virtual browsing of goods on sale, selection of goods for purchase, and payment methods. Electronic commerce operates on a bona fide basis, without prior arrangements between customers and merchants. E-commerce operates via the Internet using all or any combination of technologies designed to exchange data (such as e-mail), to access data (such as shared databases or electronic bulletin boards), and to capture data (through the use of bar coding and magnetic or optical character readers).

2.1.2 How e-Commerce works:

The consumer moves through the Internet to the merchant's web site from where, he decides what he wants to purchase something, Thereafter he is moved to the online transaction server, where all of the information he gives is encrypted. Once order is placed, the information moves through a private gateway to a Processing Network, where the issuing and acquiring banks complete or deny the transaction. This generally takes place in no more than 5-7 seconds. There are many different payment systems available to accommodate the varied processing needs of merchants, from those who have a few orders a day to those who process thousands of transactions daily. With the addition of Secure Socket Layer technology, e-Commerce is also a very safe way to complete transactions.

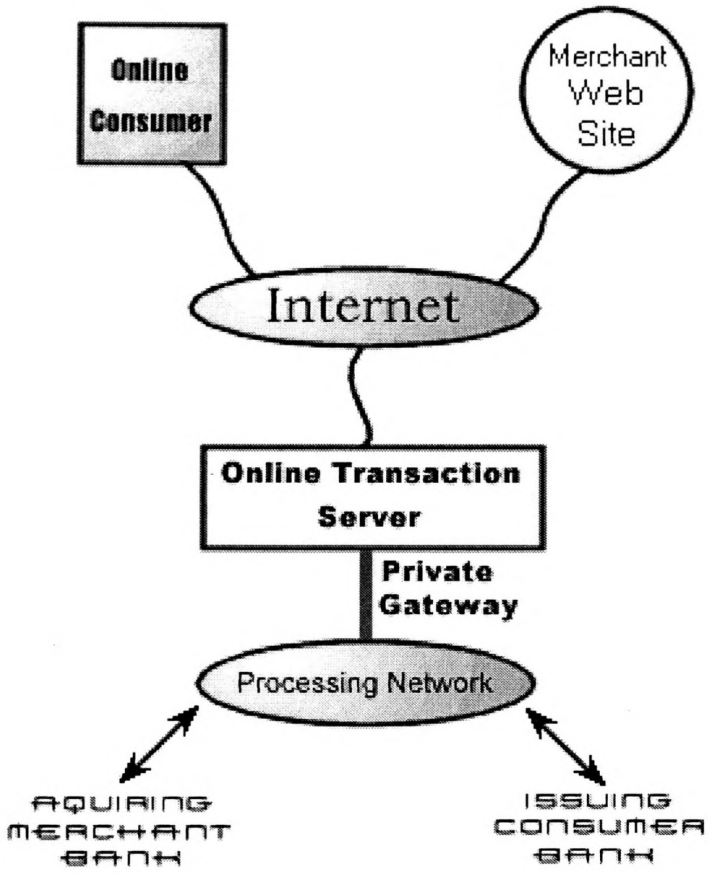


Figure 2.1 How e-commerce work

2.2 Electronic Governance (e-Governance)

The organization of the basic conditions of the information society is called adjusting e-Government briefly or as e-Governance, this is independently of whether it is made by national or by private organizations. The influence of basic conditions, made by the state, reflects directly its political direction of attack within the range of the information and communication technologies again. Accordingly the national part of the e-Governance is called e-Policy. The imbedding of the e-Government into the phases of e-Governance strictly speaking is to clarify the following diagram (Heriot Watt University – Department of Computing and Electrical Engineering, 2001)

Figure 2.2 Phases of e-governance

2.3 Software Engineering Methodology

2.3.1 Engineering software

Developing a software system is usually not done over night, but is a complex and time-consuming process. In order to control this process, reducing the complexity and uncertainties surrounding the to-be software system, we try to adhere to some kind of framework that introduces certain degrees of structure to the overall development process.

Software engineering methodologies are the framework that tells us how we should go about developing our software systems. These frameworks define different phases of the development process, such as planning, requirements analysis, design, testing and maintenance.

The most popular methodologies for software engineering are sometimes referred to as software engineering paradigms. The choice of which methodology to use in a development project is closely related to the size of the software system and the environment it is supposed to function in. The environment in itself constitutes a larger system – though most often not a computer software system, but rather some kind of organization. The different paradigms presented in this chapter all spring from this view of the world as a system of systems (Press mann, R.S. 2001).

2.3.2 Definition of Software Engineering

The definition offered here is a development of Blum's definition. It includes the notion that what is being produced is a process, and explicitly refers to the use of software in providing a solution. Thus, software Engineering is the application of tools, methods and disciplines in a cost effective way to produce and maintain a solution to a real-world processing problem, automated partly or wholly by software.

2.3.3 The life-cycle paradigm

The life-cycle paradigm of software engineering is sometimes called the "waterfall model", as it demands a sequential approach to the development process. The work is started at the system level and passes through phases of analysis, design, coding, testing and maintenance. Six activities constitute the overall development process.

- **System engineering and analysis**

This activity is characterized by system-level requirements gathering for all system elements, which then is reduced to a subset of requirements that are relevant to the software system being developed. Overall design and analysis tasks are also executed in order to understand the full system of which the software system being developed is only a part.

- **Software requirements analysis**

This activity is usually executed together with the customer, as the goal is to document all function, performance and interfacing requirements for the software.

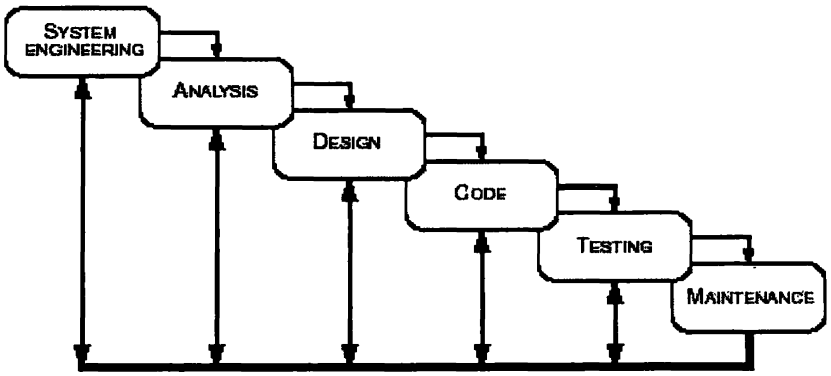


Figure 2.3 The life-cycle paradigm of software development

• **Design**

When creating the design of the software system the requirements are transformed into a representation of software that can be assessed for quality before the actual coding begins. Data structures, architecture, procedural detail and interface characterization are outlined and documented in a design specification.

• **Coding**

This activity is the transition of the design specification into a software program.

• **Testing**

This activity is to be executed using documented test methods in order to ensure that as many errors as possible is unveiled, and that the software is in accordance with the requirements.

• **Maintenance**

Rather than being an atomic activity, maintenance reapplies all other activities, as new requirements are defined in order to adapt the software to the changes occurring in its environment and to correct errors encountered after the software has been deployed.

2.3.4 The prototyping paradigm

Contrary to the static, procedural approach offered by the life-cycle paradigm, the prototyping paradigm can be used. Similarly to the life-cycle paradigm, this process begins by gathering requirements of the system. The developer meet with customers determines the overall objectives of the software and identifies any known requirements. A quick design then occurs, focusing on areas visible to the users, such as user interface and basic functionality. The design model is then used to implement a first prototype, which may take one of three forms:

- **Interactions prototype**

This is a paper or computer software prototype that makes it possible for users to understand how to interact with the software system.

- **Subset function prototype**

This is a working software program that implements a subset of the required functionality.

- **Existing program**

This is an existing program that implements most or all of the required functionality, but has features that should be improved in a later development effort.

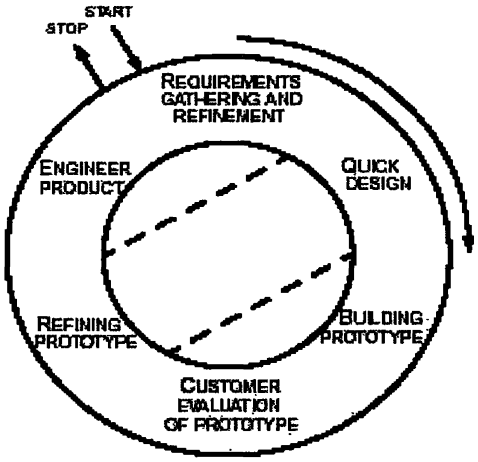


Figure 2.4 The prototyping paradigm of software development

When the prototype is created, the customer reviews it. Typically this review gives feedback to the developers that help remove uncertainties in the requirements of the software system, and starts an iteration of refinement in order to further clarify requirements by improving the prototype, or by building new prototypes. This process will result in one of two kinds of complete prototypes.

- **A throw-away prototype**

This is may be a usable software program, but is not suitable as the final software product, for various reasons such as poor performance, maintainability or overall quality.

- **A prototype to refine and deliver**

This kind of prototype is enhanced and possibly reworked in various areas so that it is suitable to deliver as the final software product. It should be noted that there are generally more reasons to throw away a prototype than there are not to -- Brooks expresses this view like this:

The basis of this argument is that the first system built usually is too slow, too big or too awkward in use. The prototype serves a better purpose of refining requirements than refining code (Sommerville, I. 1996).

2.3.5 The spiral model paradigm

While the waterfall methodology offers an orderly structure for software development, demands for reduced time-to-market make its series steps inappropriate. The spiral methodology reflects the relationship of tasks with rapid prototyping, increased parallelism, and concurrency in design and build activities. The spiral method can still be planned methodically, with tasks and deliverables identified for each step in the spiral. Some advantages of the spiral model are defers elaboration of low risk software elements, incorporates prototyping as a risk reduction strategy and gives an early focus to reusable software (Sommerville, I. 1996).

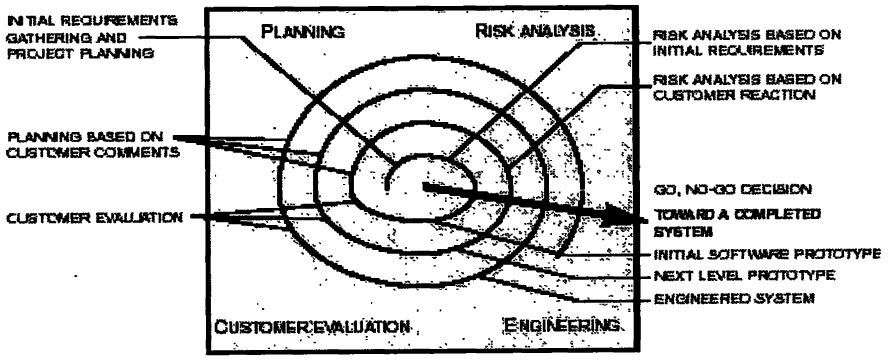


Figure 2.5 The spiral model of software development

2.4 Web Site Designing

A web site is a collection of many files linked together. A web page is a document written in HTML (hypertext mark-up language), which is viewable, by anyone -- at any time -- who has a web browser. Interactive links guide the viewer through the "web" of connections.

2.4.1 Common website building tools:

Notepad – comes as a packaged with any Windows version. Most likely to be found by taking start--> programs - -> Accessories. One can build their own website using Notepad, or any text editor. The only down side is, that in order to use notepad for web site building a knowledge in "Markup Language" is necessary. There are many different markup languages to use such as HTML, XHTML, DHTML or XML and Java. Learning these languages can become very time consuming – beneficial, but perhaps not worth, if only a simple web site is needed.It is possible to see a lot of what are called tags, such as <head>, <body> <title> and their endings, </head> , </body> , </title> which are the typical elements of Markup languages that make a web page look the way it does.

Microsoft FrontPage 2000/XP – FrontPage is one of the more popular HTML editors, developed by Microsoft, and is recommend for a beginner. (If one is familiar with Microsoft Word could be pretty comfortable using this program.) Font Page typically comes on the Microsoft Office CD.

Dreamweaver MX – Dreamweaver MX 2004 is the professional choice for building web sites and applications. It provides a powerful combination of visual layout tools, application development features, and code editing support. With robust features for CSS-based design and integration, Dreamweaver enables web designers and developers to easily create and manage any website (W3schools, 2004).

2.4.2 About HTML File

HTML stands for **H**yper **T**ext **M**arkup **L**anguage and a HTML file is a text file containing small markup tags. The markup tags tell the Web browser how to display the page. An HTML file must have an **htm** or **html** file extension. A HTML file can be created using a simple text editor.

Example for simple HTML Code:

```
<html>
<head>
<title>Title of page</title>
</head>
<body>
This is my first homepage.
</body>
</html>
```

Example Explained

The first tag in the HTML document is <html>. This tag tells the browser that this is the start of an HTML document. The last tag in the document is </html>. This tag tells the browser that this is the end of the HTML document. The text between the <head> tag and the </head> tag is header information. Header information is not displayed in the browser window. The text between the <title> tags is the title of the document. The title is displayed in the browser's caption. The text between the <body> tags is the text that will be displayed in the browser. The text between the and tags will be displayed in a bold font.

2.5 Web Application development with Macromedia Dreamweaver MX

2.5.1 Basics of Web Applications

A web application is a website that contains pages stored on a web server with partly or entirely undetermined content. The final content of a page is determined only when the user requests a page from the web server.

Because the final content of the page varies from request to request based on the user's actions, this kind of page is called a dynamic page.

2.5.2 Common uses for web applications

Web applications have many uses for both users and developers. It lets the user find information quickly and easily on a content-rich website. This kind of web application gives users the ability to search, organize, and navigate content as they see fit. Examples include company intranets, Microsoft MSDN, and Amazon.com.

Web applications can Collect, save, and analyze data provided by users. In the past, data entered in HTML forms was sent as e-mail messages to employees or CGI applications for processing. A web application can save form data directly into a database and also extract the data and create web-based reports for analysis. Examples include online banking pages, store check-out pages, surveys, and user-feedback forms.

Update websites that have constantly changing content.A web application frees the web designer from continually updating the site's HTML. Content providers such as news editors provide the web application with content and the web application updates the site automatically. Examples include the Economist and CNN.

2.5.3 Processing regular web pages

A regular website comprises a set of related HTML pages and files hosted on a computer running a web server.

A web server is software that serves web pages in response to requests from web browsers. A page request is generated when a user clicks a link on a web page, chooses a bookmark in a browser, or enters a URL in a browser's Address text box and clicks Go. The final content of a regular web page is

determined by the page designer and doesn't change when the page is requested.

When the web server receives a request for a static page, the server reads the request, finds the page, and sends it to the requesting browser, as shown in the following figure:

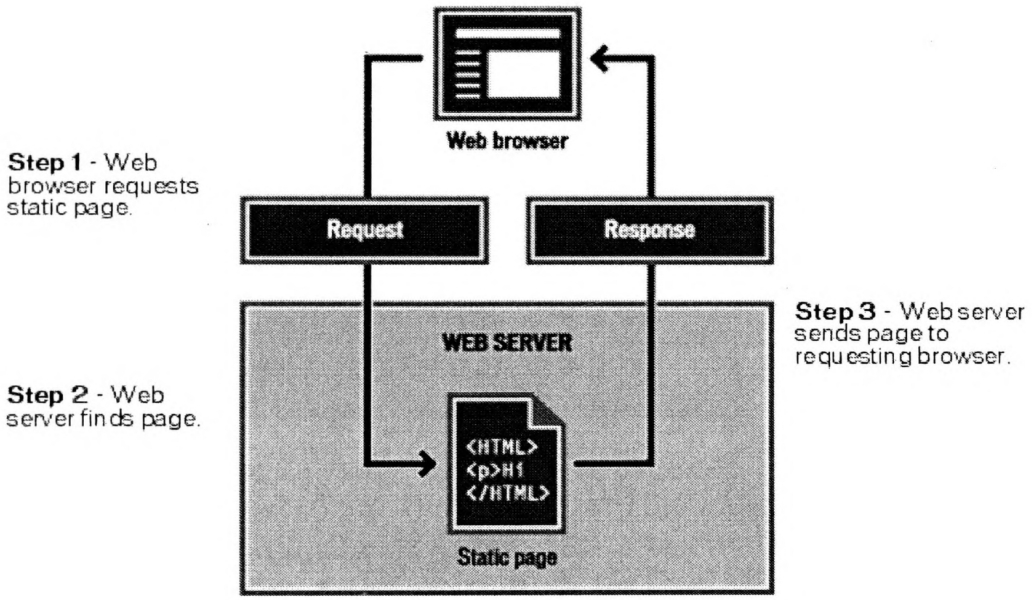


Figure 2.6 Processing regular web pages

2.5.4 Processing dynamic pages

When a web server receives a request for a regular web page, the server sends the page to the requesting browser without further ado. The web server reacts differently when it receives a request for a dynamic page: it passes the page to a special software extension responsible for finishing the page. This special software is called an application server.

The application server reads the code on the page, finishes the page according to the instructions in the code, then removes the code from the page. The result is a static page that the application server passes back to the web server, which then sends the page to the requesting browser. All the

browser gets when the page arrives is pure HTML. Here's a view of the process:

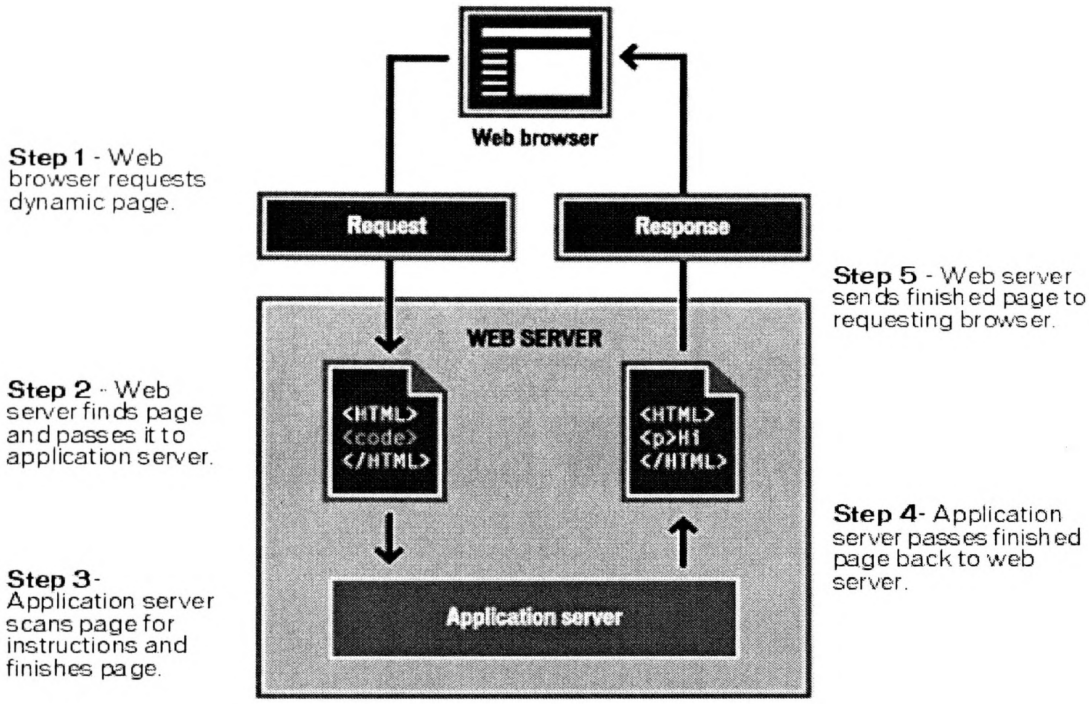


Figure 2.7 Processing dynamic pages

2.5.5 Accessing a database

An application server lets user to work with server-side resources such as databases. For example, a dynamic page may instruct the application server to extract data from a database and insert it into the page's HTML.

The instruction to extract data from a database is called a database query. A query consists of search criteria expressed in a database language called SQL (Structured Query Language). The SQL query is written into the page's server-side scripts or tags.

An application server cannot communicate directly with a database because the database's proprietary format renders the data undecipherable in much the same way that a Word document opened in Notepad is undecipherable. The application server can communicate only through the intermediary of a

once enjoy the control of hand-coding HTML or prefer to work in a visual editing environment, Dreamweaver provides the user with helpful tools to enhance web creation experience.

The visual editing features in Dreamweaver let the user quickly create pages without writing a line of code. One can view all his site elements or assets and drag them from an easy-to-use panel directly into a document. Yhe user can streamline his development workflow by creating and editing images in Macromedia Fireworks, then importing them directly into Dreamweaver, or by adding Macromedia Flash objects you create directly in Dreamweaver.

Dreamweaver also includes many coding-related tools and features, including code editing tools in the Code view (such as code coloring and tag completion); reference material on HTML, CSS, JavaScript, CFML, ASP, and JSP; and a JavaScript Debugger. Macromedia Roundtrip HTML technology imports your hand-coded HTML documents without reformatting the code.

And Dreamweaver now incorporates and expands on all of the capabilities from Macromedia UltraDev, helping to build dynamic database-backed web applications using server languages such as ASP, ASP.NET, ColdFusion Markup Language (CFML), JSP, and PHP.

2.6 About Cascading Style Sheets

Cascading Style Sheets (CSS) are a collection of formatting rules which control the appearance of content in a web page. With CSS styles users have great flexibility and control of the exact page appearance, from precise positioning of layout to specific fonts and styles.

CSS styles let the users control many properties that cannot be controlled using HTML alone. For example, one can assign custom list bullets and specify different font sizes and units (pixels, points, and so on). By using CSS styles and setting font sizes in pixels, one can ensure a more consistent treatment of the page layout and appearance in multiple browsers. In addition to text formatting, one can control the format and positioning of a block-level

elements in a web page. For example, one can set margins, borders, float text around other text, and so on.

A CSS style rule consists of two parts—the selector and the declaration. The selector is the name of the style (such as TR, or P) and the declaration defines what the style elements are. The declaration consists of two parts, the property (such as font-family), and value (such as Helvetica). The term cascading refers to onces ability to apply multiple style sheets to the same web page. For example, one can create one style sheet to apply color and another to apply margins, and apply them both to the same page to create the design.

A major advantage of CSS styles is easy update capability; when one update a CSS style, the formatting of all the documents that use that style are automatically updated to the new style.
One can define the following types of CSS style sheets in Dreamweaver:

- Custom CSS styles, also called class styles, let the users to set style attributes to any range or block of text.
- HTML tag styles redefine the formatting for a particular tag, such as h1. When user create or change a CSS style for the h1 tag, all text formatted with the h1 tag is immediately updated.
- CSS selector styles redefine the formatting for a particular combination of tags (for example, td h2 applies whenever an h2 header appears inside a table cell) or for all tags that contain a specific id attribute (for example, #myStyle applies to all tags that contain the attribute-value pair ID="myStyle").
- Easily change the look of web pages even after they're created. Since the styles are defined in one place the look can be changed of the entire site at once.
- Define font sizes and similar attributes with the same accuracy as the user has with a word processor - not being limited to just the seven different font sizes defined in HTML.
- Position the content of web pages with pixel precision.

- Redefine entire HTML tags. Say for example, if one wanted the bold tag to be red using a special font - this can be done easily with CSS.
- Define customized styles for links - such as getting rid of the underline.
- Define layers that can be positioned on top of each other (often used for menus that pop up).

CSS style sheets reside in the head area of a document. CSS styles can define the formatting attributes for HTML tags, ranges of text identified by a class attribute. Dreamweaver MX recognizes styles defined in existing documents as long as they conform to CSS style guidelines.

2.6.1 How it works:

A style is a definition of fonts, colors, etc. Each style has a unique name: a **selector**. The selectors and their styles are defined in one place. In the HTML contents simply refer to the selectors whenever user want to activate a certain style.

For example:

Instead of defining fonts and colors each time user can start a new table cell, A style can be define and then, simply refer to that style in the table cells.

Classic HTML

```
<table>
<tr><td bgcolor="#FFCC00" align="left"><font face="arial"
size="2" color="red"><b>this is line 1</b></font></td></tr>

<tr><td bgcolor="#FFCC00" align="left"><font face="arial"
size="2" color="red"><b>this is line 2</b></font></td></tr>

<tr><td bgcolor="#FFCC00" align="left"><font face="arial"
size="2" color="red"><b>this is line 3</b></font></td></tr>
</table>
```

With CSS (assuming that a selector called subtext is defined)

```
<table>
<tr><td class="subtext">this is line 1</td></tr>
<tr><td class="subtext">this is line 2</td></tr>
<tr><td class="subtext">this is line 3</td></tr>
</table>
```

2.6.2 Usage of CSS in this project

```
<HTML><HEAD><TITLE>CCS</TITLE>

<STYLE type=text/css>A:link {
        COLOR: #238cbe; TEXT-DECORATION: none
}
A:active {
        COLOR: #ffeeee; TEXT-DECORATION: none
}
A:visited {
        COLOR: #238cbe; TEXT-DECORATION: none
}
A:hover {
        COLOR: #94d652; TEXT-DECORATION: none
}
</STYLE>
</HEAD>
```

2.7 About Java Script

JavaScript is an easy-to-use programming language that can be embedded in the header of the web pages. It can enhance the dynamics and interactive features of the page by allowing one to perform calculations, check forms, write interactive games, add special effects, customize graphics selections,

and create security passwords and more. JavaScript is a scripting language. A scripting language is a lightweight programming language. A JavaScript is lines of executable computer code and it is usually embedded directly in HTML pages. JavaScript is an interpreted language (means that scripts execute without preliminary compilation). Everyone can use JavaScript without purchasing a license. It is supported by all major browsers, like Netscape and Internet Explorer

2.7.1 Usage of Java Script

JavaScript gives HTML designers a programming tool - HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small "snippets" of code into their HTML pages.

JavaScript can put dynamic text into an HTML page - A JavaScript statement like this: document.write("<h1>" + name + "</h1>") can write a variable text into an HTML page.

JavaScript can react to events - A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element.

JavaScript can read and write HTML elements - A JavaScript can read and change the content of an HTML element.

JavaScript can be used to validate data - A JavaScript can be used to validate form data before it is submitted to a server, this will save the server from extra processing (W3schools, 2004).

2.7.2 Usage of Java Script in this project

```
<html>
<head>
<title>Calendar </title>
</head>
<body>
<table width="353" height="44" border=0 cellpadding=0 cellspacing=0>
 <tbody>
    <tr>
    <td height="43"> <div align=left>
      <form name=doublecombo>
       <p>
        <select class=backcolor
      onChange=redirect(this.options.selectedIndex) size=1 name=example>
          <option>2003</option>
          <option>2004</option>
          <option>2005</option>
        </select>
        <select
      class=backcolor size=1 name=stage2>
          <option value="Untitled-5.htm">January</option>
          <option value="Untitled-6.htm">February</option>
          <option value="http://k">March</option>
          <option value="http://>April</option>
          <option value="http://>May</option>
          <option value="http://net">June</option>
          <option value=" http://www.kalutara.org/Index.htm ">July</option>
          <option value="http://www.shinka.net">August</option>
          <option value="http://www.shinka.net">September</option>
          <option value="http://www.shinka.net">October</option>
          <option value="http://www.shinka.net">November</option>
          <option value="http://www.shinka.net">December</option>
        </select>
        <input onClick=go() type=button value=Go! name=test>
       </p>
      </form>
     </div></td>
   </tr>
 </tbody>
</table>
<p>
 <script>
var groups=document.doublecombo.example.options.length
var group=new Array(groups)
for (i=0; i<groups; i++)
group[i]=new Array()

group[0][0]=new Option("January","http://www.kalutara.org/Index.htm")
group[0][1]=new Option("February","http://www.kalutara.org/Index.htm")
```

```
group[0][2]=new Option("March," http://www.kalutara.org/Index.htm ")
group[0][3]=new Option("April," http://www.kalutara.org/Index.htm ")
group[0][4]=new Option("May," http://www.kalutara.org/Index.htm ")
group[0][5]=new Option("June," http://www.kalutara.org/Index.htm ")
group[0][6]=new Option("July," http://www.kalutara.org/Index.htm ")
group[0][7]=new Option("August," http://www.kalutara.org/Index.htm ")
group[0][8]=new Option("September," http://www.kalutara.org/Index.htm ")
group[0][9]=new Option("October," http://www.kalutara.org/Index.htm ")
group[0][10]=new Option("November," http://www.kalutara.org/Index.htm ")
group[0][11]=new Option("December," http://www.kalutara.org/Index.htm ")

group[1][0]=new Option("January," http://www.kalutara.org/Index.htm ")
group[1][1]=new Option("February," http://www.kalutara.org/Index.htm ")
group[1][2]=new Option("March," http://www.kalutara.org/Index.htm ")
group[1][3]=new Option("April," http://www.kalutara.org/Index.htm ")
group[1][4]=new Option("May," http://www.kalutara.org/Index.htm ")
group[1][5]=new Option("June," http://www.kalutara.org/Index.htm ")
group[1][6]=new Option("July," http://www.kalutara.org/Index.htm ")
group[1][7]=new Option("August," http://www.kalutara.org/Index.htm ")
group[1][8]=new Option("September," http://www.kalutara.org/Index.htm ")
group[1][9]=new Option("October","http://www.slt.lk/athome.htm")
group[1][10]=new Option("November," http://www.kalutara.org/Index.htm ")
group[1][11]=new Option("December," http://www.kalutara.org/Index.htm ")

group[2][0]=new Option("January," http://www.kalutara.org/Index.htm ")
group[2][1]=new Option("February," http://www.kalutara.org/Index.htm ")
group[2][2]=new Option("March," http://www.kalutara.org/Index.htm ")
group[2][3]=new Option("April," http://www.kalutara.org/Index.htm ")
group[2][4]=new Option("May," http://www.kalutara.org/Index.htm ")
group[2][5]=new Option("June," http://www.kalutara.org/Index.htm ")
group[2][6]=new Option("July," http://www.kalutara.org/Index.htm ")
group[2][7]=new Option("August," http://www.kalutara.org/Index.htm ")
group[2][8]=new Option("September," http://www.kalutara.org/Index.htm ")
group[2][9]=new Option("October," http://www.kalutara.org/Index.htm ")
group[2][10]=new Option("November," http://www.kalutara.org/Index.htm ")
group[2][11]=new Option("December," http://www.kalutara.org/Index.htm ")

var temp=document.doublecombo.stage2

function redirect(x){
for (m=temp.options.length-1;m>0;m--)
temp.options[m]=null
for (i=0;i<group[x].length;i++){
temp.options[i]=new Option(group[x][i].text,group[x][i].value)
}
temp.options[0].selected=true
}

function go(){
var newWindow;
```

```
var location=temp.options[temp.selectedIndex].value
newWindow =
window.open(location,'','height=280,width=300,toolbar=no,minimize=no,status
=no,memubar=no,location=no,scrollbars=no')
}
//-->
</script>
</p>
</body>
</html>
```

2.8 About Active Server Pages

Active Server Pages (ASPs) are Web pages that contain server-side scripts in addition to the usual mixture of text and HTML (Hypertext Markup Language) tags. Server-side scripts are special commands put in Web pages that are processed before the pages are sent from once Personal Web Server to the Web browser of someone who's visiting the Web site. . When the URL is typed in the Address box or click a link on a Web page, asking a Web server on a computer somewhere to send a file to the Web browser (sometimes called a "client") on client's computer. If that file is a normal HTML file, it looks exactly the same when client's Web browser receives it as it did before the Web server sent it. After receiving the file, Client's Web browser displays its contents as a combination of text, images, and sounds.

In the case of an Active Server Page, the process is similar, except there's an extra processing step that takes place just before the Web server sends the file. Before the Web server sends the Active Server Page to the Web browser, it runs all server-side scripts contained in the page. Some of these scripts display the current date, time, and other information. Others process information the user has just typed into a form, such as a page in the Web site's guest book.

To distinguish them from normal HTML pages, Active Server Pages are given the ".asp" extension. ASP is a program that runs inside **IIS.** IIS stands for Internet Information Services and it comes as a free component with Windows 2000 (Mitchell, S. 2000).

2.8.1 ASP Compatibility

ASP is Microsoft Technology .To run IIS the systems must have Windows NT 4.0 or later version. To run PWS the systems must have Windows 95 or later version. ChiliASP is a technology that runs ASP without Windows OS. InstantASP is another technology that runs ASP without Windows.

2.8.2 What is an ASP File?

An ASP file is just the same as an HTML file. It can contain text, HTML, XML, and scripts. Scripts in an ASP file are executed on the server. An ASP file has the file extension ".asp".

2.8.3 What Can User Do with Active Server Pages?

There are many things can be done with Active Server Pages. One can display date, time, and other information in different ways. A survey form and ask people who visit once web site to fill it out, send emails, save the information to a file, etc. It can dynamically edit, change or add any content of a Web page. ASP responds to user queries or data submitted from HTML forms and access any data or databases and return the results to a browser. It customizes a Web page to make it more useful for individual users. The advantages of using ASP instead of CGI and Perl, are those of simplicity and speed. A security cab be provided since ASP code can not be viewed from the browser. Since ASP files are returned as plain HTML, they can be viewed in any browser. Using clever ASP programming, the network traffic can be minimized.

2.8.4 How Does ASP Differ from HTML?

When a browser requests an HTML file, the server returns the file. But when a browser requests an ASP file, IIS passes the request to the ASP engine. The ASP engine reads the ASP file, line by line, and executes the scripts in the file. Finally, the ASP file is returned to the browser as plain HTML.

2.8.5 What Do Active Server Pages Look Like?

The appearance of an Active Server Page depends on who or what is viewing it. To the Web browser that receives it, an Active Server Page looks just like a normal HTML page. If a visitor to your Web site views the source code of an Active Server Page, that's what they see: a normal HTML page. However, the file located in the server looks very different. In addition to text and HTML tags, you also see server-side scripts. This is what the Active Server Page looks like to the Web server before it is processed and sent in response to a request.

2.8.6 The Basic Syntax Rule

An ASP file normally contains HTML tags, just like an HTML file. However, an ASP file can also contain **server scripts**, surrounded by the delimiters **<%** and **%>**. Server scripts are **executed on the server,** and can contain any expressions, statements, procedures, or operators valid for the scripting language you prefer to use (Weissinger, A.K. 2000).

2.8.6.1 The Response Object

The **Write** method of the ASP **Response Object** is used to send content to the browser. For example, the following statement sends the text "Hello World" to the browser:

```
<%
response.write("Hello World!")
%>
```

2.8.6.2 VBScript

You may use different scripting languages in ASP files. However, the default scripting language is VBScript:

```
<html>
<body>
<%
response.write("Hello World!")
%>
</body>
</html>
```

3. Methodology

The system of the project was developed under four phases during the project period.

1. Designing the website.
2. Designing a form with necessary ASP files to enter the details of survey questionnaires and connect the form with the SQL database through ODBC data source.
3. Uploading the materials to Internet.
4. Maintaining the web site.

3.1.1 Material used

The website was designed by means of Dreamweaver MX software. This software provides facility to build websites more efficiently and more quickly. According to software development process coding is very important phase. To design the web pages of the site, HTML, java script and ASP codes were written by using Dreamweaver MX software.

3.1.2 System Requirements for the development

Windows

- 600 MHz Intel Pentium III processor or equivalent
- Windows 98 SE (4.10.2222 A), Windows 2000, Windows XP, or Windows Server 2003
- 128 MB RAM (256 MB recommended)
- 275 MB available disk space

Macintosh

- 500 MHz Power PC G3 processor
- Mac OS X 10.2.6 and later, 10.3

- 128 MB computer RAM (256 MB recommended)
- 275 MB available disk space

3.2 Web-site design

The website was designed according to following principles.

- Pages share the same color scheme, navigation icons, and identifying graphics, which create a smooth transition from the main page to the secondary page, and develop a unified look and feel for the web site.
- Consistent background graphics
- Consistent placement of navigation information
- Consistent font usage
- Logo that brands the site

The continuity between these pages enforces the feeling of a whole piece of work. The understated fonts and colors and the quiet background graphics provide an appropriate museum-like impression.

At the beginning following codes were generated by using Dreamweaver MX.

```
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
</body>
</html>
```

Then a layer was added to the page by using following codes. These codes were included inside the body tag.

Then after a table was included to design the rest of the page setup.

<table width="1000" border="0" cellpadding="0" cellspacing="0" >
<tr><td width="147" rowspan="2" valign="top" bgcolor="#000000">
<table width=140 border=0 align="left">
<tbody><tr align=middle>
<td align=center width=118 height=26>
<font color=#ffffff size="3"> </td></tr>
</tbody>
</table>

Images were included using the following codes.

<img src="../../Images/I_Home_Page/BLUERED.GIF" width="12" height="12" border="0">

To add JavaScript, <SCRIPT LANGUAGE="JavaScript"> tag was used. JavaScript functions were included under this tag.

```
function scrollit(seed) {
var m1  = "Welcome to www.kalutara.org  ";
var m2  = "..... You can find a lot of information you need here!    ";
var m3  = ".... ";
var m4  = "Help The Poor And Disable ";
var msg=m1+m2+m3+m4;
var out = " ";
var c   = 1;
if (seed > 100) {
seed--;
cmd="scrollit("+seed+")";
timerTwo=window.setTimeout(cmd,100);
}
```

The other pages of the website were also designed according to the above procedure.

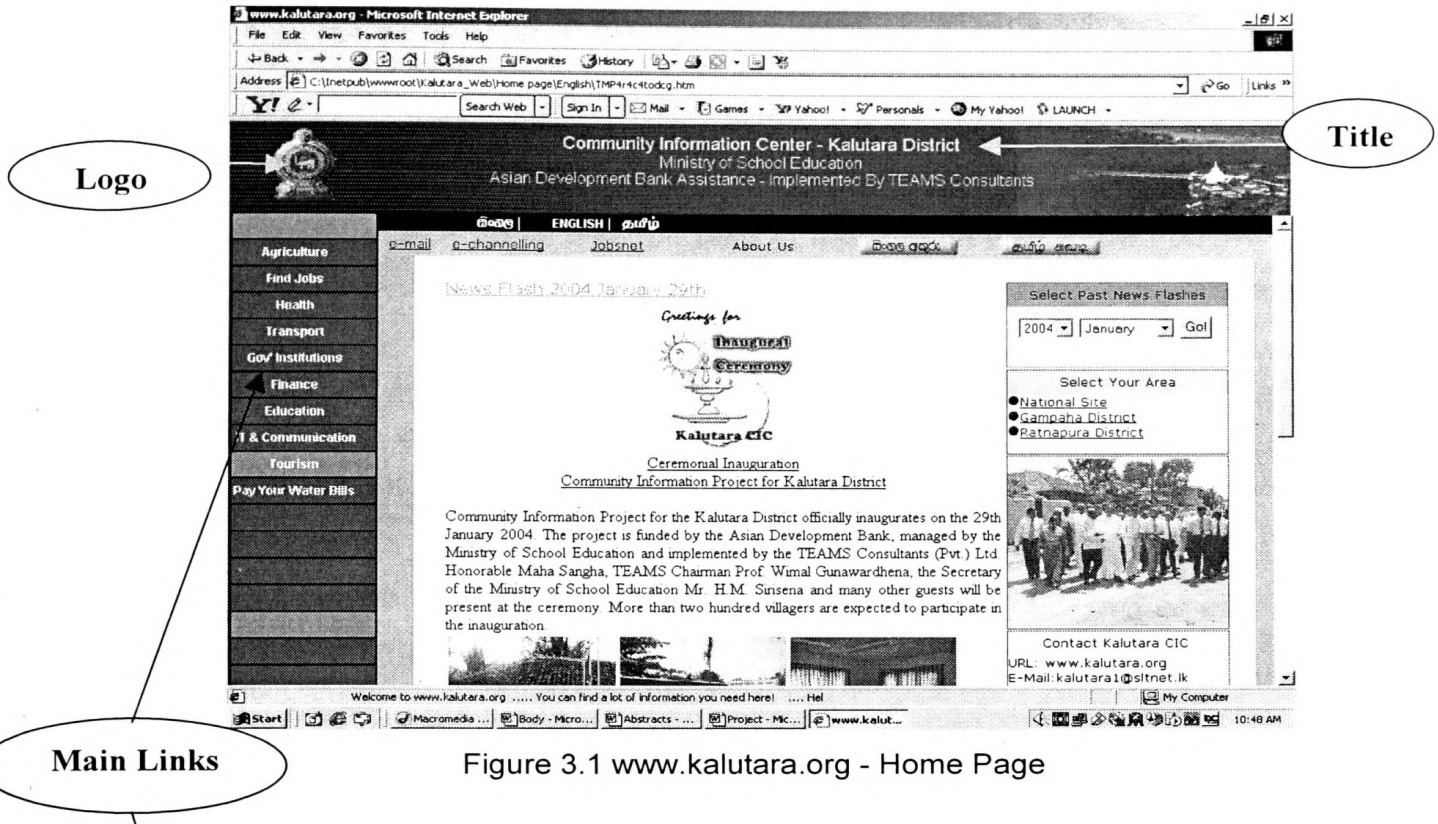


Figure 3.1 www.kalutara.org - Home Page

Figure 3.2 www.kalutara.org - Secondary Page

3.3 Designing the survey form

The interface for the survey form (HTML coding) and the ASP file were designed by means of Dreamweaver MX Software. The information of the questionnaires was entered with the help of this interface as shown below. To design the survey form following codes was generated using Dreamweaver MX.

Following tags were included in the body tag of the HTML page.

```
<form name="form1" method="post" action="insert.asp">
<table width="80%" height="736" border="0" align="center" cellpadding="0"
cellspacing="0">
<td colspan="2"><font size="2" face="Verdana, Arial, Helvetica, sans-serif">
<select name="sltInvesOf">
<option selected>Select an Officer</option>
<option value="Prasad">Prasad</option>
</select>
```

Text fields were added using following codes.

```
<tr><td height="24"><font size="2" face="Verdana, Arial, Helvetica, sans-
serif">Date :</font></td>
 <td colspan="2"><font size="2" face="Verdana, Arial, Helvetica, sans-serif">
 <input name="txtDate" type="text" id="txtDate">
 </font>
</td></tr>
```

Radio buttons, check boxes and text arias were included with the help of following codes.

```
<input name="rbcivil" type="radio" value="Single">
<input name="chkAvaCom1" type="checkbox" value="Telephone">
<textarea name="txtinfosupp1" cols="50"></textarea>
```

The rest of the form was designed according to the above procedure.

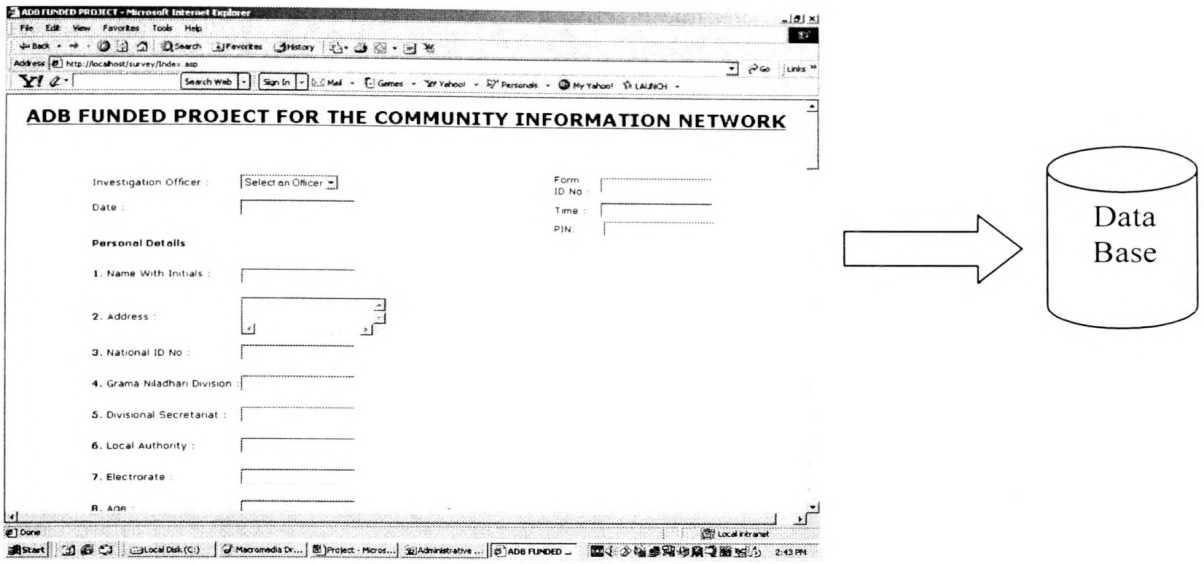


Figure 3.3 The interface of the survey form

3.4 Uploading the materials to Internet

The web site was uploaded to Internet using ws_ftp pro. This software was downloaded to the computer that has the real IP address in the network and the materials of the web site were up loaded to the data folder that is available in the Sri Lanka Telecom Server.

3.5 Maintaining the website

The web site for community of kalutara district was uploaded to Internet and it is to be updated daily. Information of the kalutara district is included to the site and it is kept as a very fresh site.

4. Results And Discussion

A Web-based information system for the Kaluthara District was developed. The name of the website is called as www.kalutara.org. The web site has been uploaded to the Internet and those who have Internet facility can visit the site from anywhere in the world. The web site can be updated remotely. Therefore a person who wants Information about following key fields can be obtained quickly from this web site.

- Agriculture
- Health
- Education
- Government Institutions
- Jobs
- Private Institutions
- Transport
- Trade
- IT and Communication
- Tourism
- Banks details and facilities provided by them (Loans, Leasing, Savings etc.)

In addition to these topics following extra facilities are available in this site.

- e-channeling
- e-mail
- Fax
- News Flash (Daily)
- News Letter (Monthly)

The past news flashes as well as past news letters can be selected according to the date when they were issued. This can be achieved by using the pop up

calendar. After selecting the year and the month from double pull down menu the appropriate date can be select from the pop up calendar.

Select Year → Select Month → Press Go Button → Click On date of the News Flash

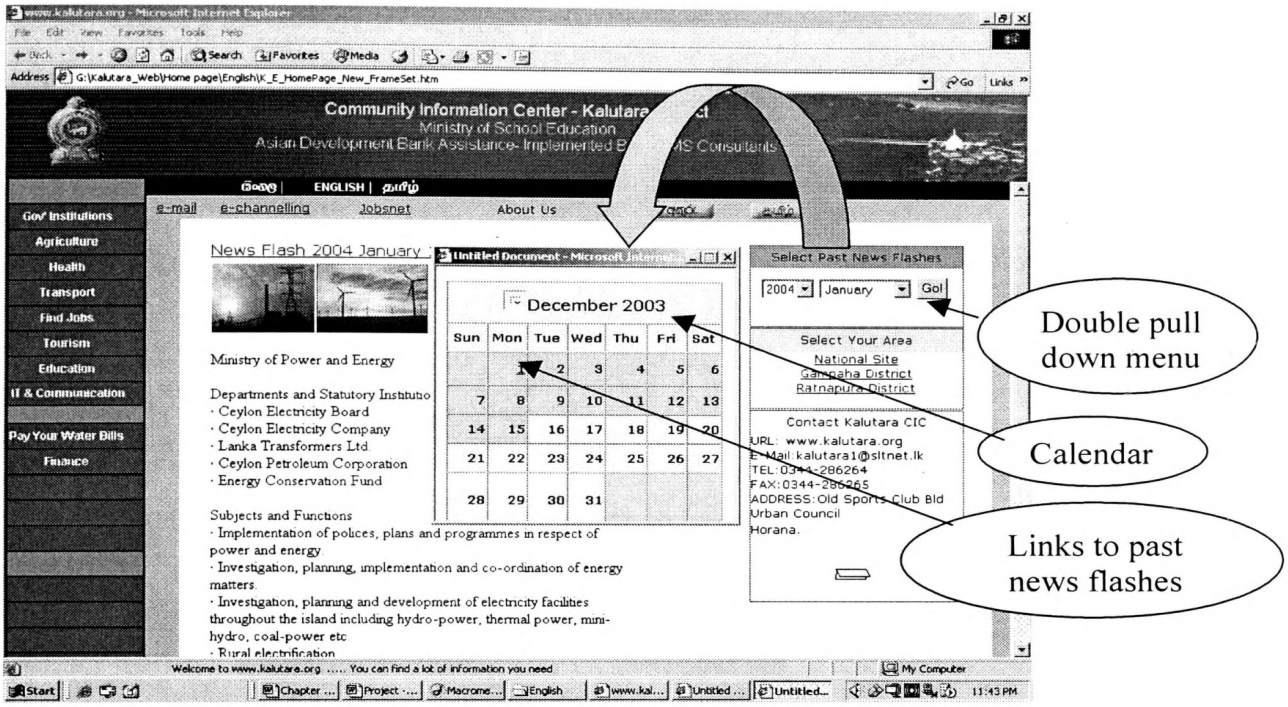


Figure 4.1 How pop up calendar work

The details of the survey questionnaires can be entered to the SQL database by means of the survey form effortlessly, that has been developed under this project.

The result out of this project provides an easy access for vital information in Kalutara District as well as other Information over the world through modern Information Technology. The specific areas of information have been entered to the site so far and vast amount of details of so many specific areas are to be added. Through this information the life style of poor and disable could be changed. The site was designed in three main languages. Sinhala, Tamil and English that are commonly used in Sri Lanka. Therefore the multi-cultural community can easily go through into this site with their mother language. The facility to down load Sinhala and Tamil fonts has been provided on the home

page of the site. To choose the language, every page has a link for the language as shown below.

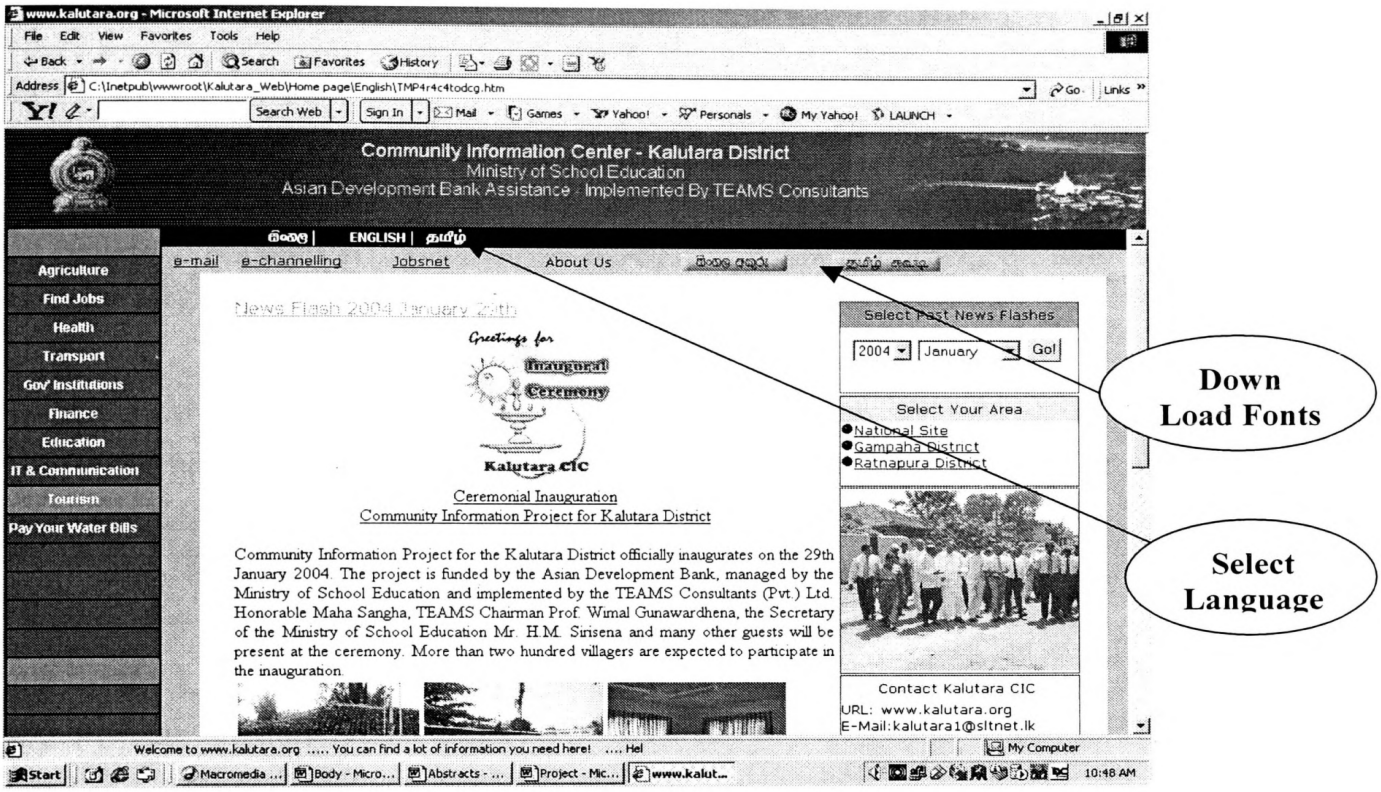


Figure 4.2 How to select the language

The daily news flash page was included to publish daily important news regarding the Kalutara District and for important and necessary cases the video clips were added to the relevant pages.

5. Conclusions

5.1 Conclusions

When providing information we have to think about printing pages to get hard copies. To meet this requirement pages of this web site were designed to get print out of the page without loosing the containing. Printed croppies can be obtained by clicking icon of printing on standard toolbar.

The www.kalutara.org web site can be accessed both online and offline in the Community Information Center (CIC) At Horana. The site was locally hosted at Horana Community Information Center. Therefore the access to the site for offline is available at Horana CIC premises. The site can be access from anywhere of the world through the Internet successfully.

5.2 Future studies

To provide searching method, a search engine could be added to the site. An online registration facility could be provided to the community for their convenience. In the Community Information Center, a web server could be maintained and all the contain of the web site could be kept in that sever. After wards, uploading the web site materials to the Sri Lanka Telecom Server need not to be done.

References

Heriot Watt University – Department of Computing and Electrical Engineering. (2001), Heriot-Watt University, http://www.hw.ac.uk/sbc/sbcampus

Mitchell, S. (2000), Designing Active Server Pages. 1st Ed. O'Reilly & Associates. pp.348.

Press mann, R.S. (2001), Software Engineering- A Practitioner's Approach. 5th Ed. Mc Craw-hill Companies, New York, pp.860.

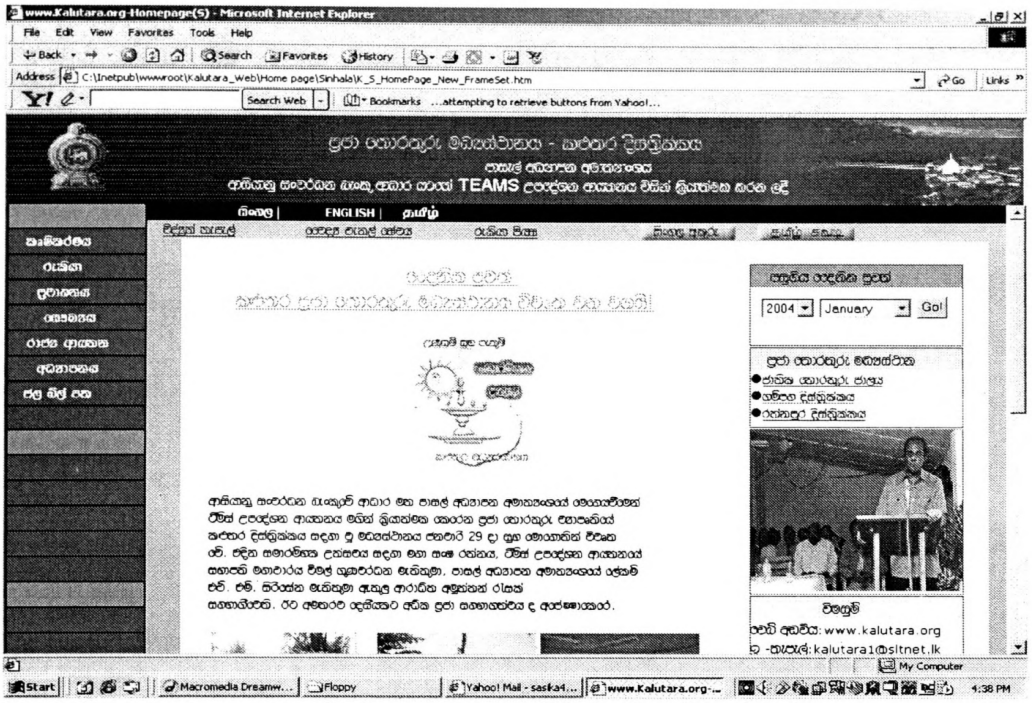
Sommerville, I. (1996), Software Engineering. 5th Ed. Addison-Wesley, Inc. pp 8- 45.

Weissinger, A.K. (2000), ASP in a Nutshell, O'Reilly & Associates. pp.322.

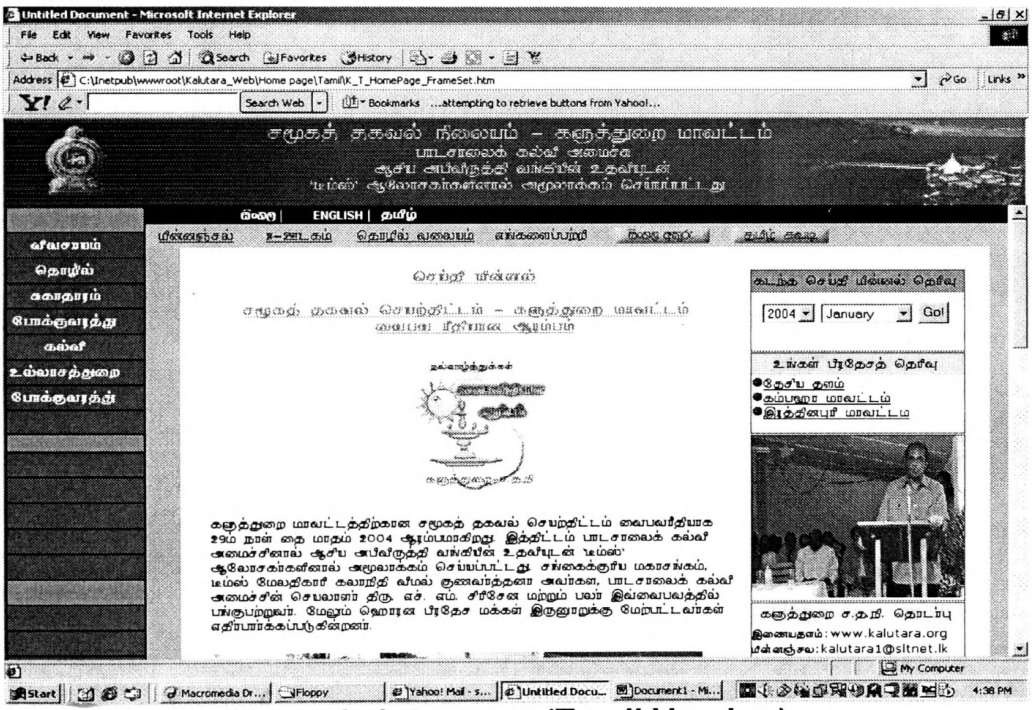
W3schools (1999-2004), JavaScript Tutorials, W3schools, http://www.w3schools.com/js/default.asp

W3schools (1999-2004), Web Building Tutorials, W3schools, http://www.w3schools.com/site/default.asp

APPENDIX



www.kalutara.org (Sinhala Version)



www.kalutara.org (Tamil Version)

National Digitization Project

National Science Foundation

Institute : Sabaragamuwa University of Sri Lanka

1. Place of Scanning : Sabaragamuwa University of Sri Lanka, Belihuloya

2. Date Scanned : 2017-09-25

3. Name of Digitizing Company : Sanje (Private) Ltd, No 435/16, Kottawa Rd, Hokandara North, Arangala, Hokandara

4. Scanning Officer

Name : B.A.C. Sadaruwan

Signature : .....

Certification of Scanning

I hereby certify that the scanning of this document was carried out under my supervision, according to the norms and standards of digital scanning accurately, also keeping with the originality of the original document to be accepted in a court of law.

Certifying Officer

Designation : Librarian

Name : T. N. Neighsoorei

Signature : .....

Mrs. T.N. NEIGHSOOREI
(MSSc.PGD.ASLA,BA)
Librarian
Sabaragamuwa University of Sri Lanka
P.O.Box 02 Belihuloya,Sri Lanka
Tele:0094 45 2280045
Fax:0094 45 2280045

Date : 2017-09-25

"This document/publication was digitized under National Digitization Project of the National Science Foundation, Sri Lanka"