

**WEB SITE DESIGN WITH ACTIVE SERVER PAGE (ASP) FACILITY**

**FOR**

**ELECTRICAL/COMPUTER ENGINEERING DEPARTMENT,  
FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA**

**BY**

**AGBECHOMA, OGHENEVWOKE GRATEFUL**


**98/6832EE**

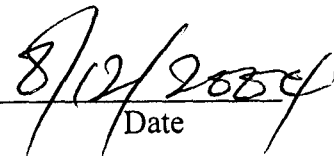
**DEPARTMENT OF ELECTRICAL/COMPUTER ENGINEERING  
SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY  
FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA**

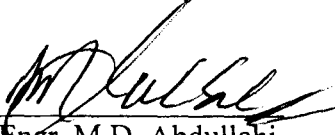
**NOVEMBER, 2004**

ATTESTATION

This is to certify that this project titled, "WEB SITE DESIGN WITH ACTIVE SERVER PAGE (ASP) FACILITY FOR ELECTRICAL/COMPUTER ENGINEERING DEPARTMENT, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA", was carried out by me, Agbechoma, O.G, for the award of a Bachelor of Engineering (B.Eng.) Degree in the Department of Electrical and Computer Engineering of the Federal University of Technology, Minna, Niger State.

  
Engr. M.D. Abdullahi  
(Project Supervisor)

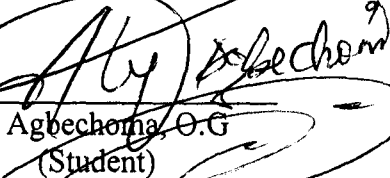
  
Date

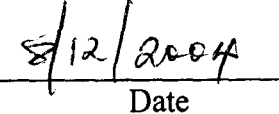
  
Engr. M.D. Abdullahi  
(Head of Department)

  
Date

\_\_\_\_\_  
External Examiner

\_\_\_\_\_  
Date

  
Agbechoma, O.G  
(Student)

  
Date

## DEDICATION

This project work is dedicated to the Almighty God; the only One, True, Living and Wise God, from and in whom the fountain of wisdom flows. For Your unusual supply of strength; Lord, You're the Greatest.

## **ACKNOWLEDGEMENT**

My acknowledgement goes to everyone who contributed in one way or the other to the progress and success of this project. There are too many people to thank in so little time and space.

I'm so very grateful to my supervisor, Engr. M.D. Abdullahi, for his understanding, fatherly and easy-going nature. To the other lecturers, I say thank you, for being there through the years.

I am awfully grateful to my family; my parents, Mr. and Mrs. Agbechoma, for their financial support, and my great siblings, for their love.

I'd like to thank my friend Damola for his great hospitality; I'm totally indebted to you.

To Jide, I say thanks for your creative expertise and insight into learning; truly you have an eye for quality.

To all my friends; the Runs Family - Yinka, Roy, Matthew, Gabriel and Izu; for your moral support and thoughtfulness, as my name goes, I'm grateful.

To Caroline, the one lady I share pure and loving thoughts with, I say thanks for being my prayer tag team.

Yet, the Person most directly responsible for the success of this project is my Sustainer; God, who makes my life complete in Him.

## **ABSTRACT**

This project is based on Web site design for Electrical/Computer Engineering Department of the Federal University of Technology, Minna, which includes a feature of database using Active Server Page (ASP), with which students can check their results easily.

Simple, attractive and easy to load Web pages design is undertaken with a view to attracting the interest of a viewer and meeting the need of the audiences.

Hand-coding, using HTML is employed in making the Web designs and the basic elements include colour contrast, text organization, font selection, page style, page size, graphics used, and consistency.

The uploading of Web pages after they are completed, using one of the Internet services that allows computers to transfer binary files across the Internet, is discussed. The basic idea of a Web page being nothing more than a file, a HTML file to be exact, is indicated.

## TABLE OF CONTENTS

CONTENT	PAGE
Title Page.....	I
Attestation.....	II
Dedication.....	III
Acknowledgement.....	IV
Abstract.....	V
Table of Contents.....	VI

### CHAPTER ONE: BROAD INTRODUCTION

1.1 Introduction.....	1
1.2 In order to design a good Web site.....	2
1.2.1. Decode scope of features.....	2
1.2.2 Decide page layout & navigation scheme.....	2
1.2.3 Determine keywords.....	2
1.2.4 Draft the content.....	3
1.2.5 Decide artistic enhancement.....	3
1.3 The three (3) levels of Web site design sophistication.....	3
1.3.1 Basic.....	3
1.3.2 Database driven.....	3
1.3.3 Multimedia.....	3
1.4 History of the Internet (Basic concept).....	3
1.5 The immensity of the Internet.....	5
1.6 What can be done on the Internet?.....	6

1.7 Who owns the Internet?.....	7
1.8 How the Internet works.....	7

**CHAPTER TWO: INTRODUCTION TO WEBSITES**

2.1 Introduction.....	8
2.2 Web sites? (Literature Review).....	8
2.3 Who is a Web site visitor?.....	9
2.4 Why does one need a Web site?.....	9
2.5 What a Web site can do.....	9
2.6 How does a Web site work?.....	10
2.7 Project idea and Incentive.....	10

**CHAPTER THREE: DESIGNING PAGE LAYOUT**

3.1 Planning your Web site design.....	11
3.1.1 Creating goals for your site.....	11
3.1.2 Organizing the site structure.....	12
3.1.3 Creating your design look.....	13
3.1.4 Designing the navigation scheme.....	14
3.1.5 Planning and gathering your assets.....	14
3.2 Testing World Wide Web sites.....	15
3.3 Usability Guidelines.....	17
3.4 Web site design.....	17
3.4.1 HyperText Markup Language (HTML).....	17
3.4.2 The merits of HTML.....	18

## **CHAPTER FOUR: ACTIVE SERVER PAGE**

4.1 Introduction.....	19
4.2 How ASP Works.....	20
4.3 ASP Objects.....	20
4.3.1 Application.....	21
4.3.2ObjectContext.....	21
4.3.3 Response.....	21
4.3.4 Request.....	22
4.3.5 Server.....	22
4.3.6 Session.....	22
4.4 Expanding ASP Capabilities.....	23
4.5 What you can do with Active Server Pages.....	23
4.6 Putting data on the Internet using Active server page.....	24

## **CHAPTER FIVE: RESULT AND CONCLUSION**

5.1 Introduction.....	27
5.2 Result for the site design.....	27
5.3 Result for the database.....	27
5.4 Recommendation.....	28
5.5 References.....	30
Home Page Sample.....	31
Source Codes.....	32



# CHAPTER ONE

## BROAD INTRODUCTION

### 1.1 INTRODUCTION

The world in our time can be likened to a small settlement where information travels along time itself with great speed. Information, entertainment and education have gone from merely getting acquainted with the mass media and other educational bodies, to readily getting any piece of what you need, in little or no time at all, on the Internet.

The Internet is a worldwide collection of networks and gateways that use the Transmission Control Protocol/Internet Protocol (TCP/IP) suite of protocols (standards) to communicate with one another. It is a set of computer networks that may be dissimilar and are joined together by means of gateways that handle data transfer and conversion of messages. Several definitions will be given in the course of this project report.

Since computers are linked together, they allow for the quick, accurate and hopeful exchange of information such as: electronic mails (i.e. e-mails), video/graphics, sound files and documents and therefore, people no longer have to travel as far or rather it allows individuals to organize more quickly and efficiently than with previous technologies as much as they can, in order to source for any information needed. In other words, the Internet can reach a large audience (mass communication) while simultaneously operating at the level of the individual (interpersonal communication).

In view of the above benefits, the need to develop a Web site for Electrical/Computer Engineering Department of the Federal University of Technology, Minna, Niger State, Nigeria, arises. The Department is a great one for learning, training and making

competent people in various fields of endeavour and as such, its activities should be made available as it were, to the whole world.

The site is to be designed with a combination of formatting and programming languages, as well as hand-coding, as the case maybe.

## **1.2 IN ORDER TO DESIGN A GOOD WEB SITE:-**

### **1.2.1 Decide Scope of Features**

Produce a checklist of Web site features that you want your Web site to have. Visit Web sites of similar purpose to generate ideas for your own. Determine the degree of Web site sophistication needed and the scope of user interaction that will be provided, e.g., online credit card processing, shopping, bulletin boards, chat rooms, site search, etc.

### **1.2.2 Decide Page Layout & Navigation Scheme**

Estimate the total number of Web pages and, name and define the purpose for each one. Plan the positioning and contents of the pages common areas - headers, footers and sidelinks. Also, decide upon a navigation scheme that will be consistent throughout the Web site. This may be as simple as a static set of side links on every page or as sophisticated as a pulldown cascading menu system which becomes more detailed as you go deeper into the site. Visit Web sites of similar purpose or surf the Internet freely to generate ideas of your own and be sure to consider the use of free Web design templates.

### **1.2.3 Determine Keywords**

Do your keyword analysis to determine your most effective primary and secondary keyword phrases with respect to search engine positioning.

#### **1.2.4 Draft the Content**

Produce the text content that will appear on each page. Review the rules for incorporating keyword phrases into page content and the specific concerns relative to writing for the Internet before beginning.

#### **1.2.5 Decide Artistic Enhancement**

Plan the artistic aspects; graphics, animations and text appearances.

### **1.3 THE THREE (3) LEVELS OF WEB SITE DESIGN SOPHISTICATION**

#### **1.3.1 Basic**

All Web sites, whether Personal, Hobbyist, Organization or Small Business, have some or all of the basic ingredients e.g. colour contrast, text organization, font selection, e.t.c.

#### **1.3.2 Database Driven**

Web sites can also have a backend database for storing and dynamically displaying various types of data.

#### **1.3.3 Multimedia**

Web sites can also have a multimedia enhanced appeal. This is entertaining but is not always effective, depending on your Web sites purpose. The user may not want to be bothered by a lot of intrusive and time consuming flashiness.

### **1.4 HISTORY OF THE INTERNET (BASIC CONCEPT)**

It is clear that the Internet is one of the most fashionable areas of computing. It is effecting all subjects in higher education, not only altering teaching opportunities because of its new perspectives on communication and dissemination of information, but

also opening up exciting new resources for students and lecturers alike.

However, a synopsis of the history of the Internet and its rapid development is given below:-

After the launch of the Soviet satellite Sputnik in 1957, the US established the Advanced Research Projects Agency to investigate ways of increasing the military's use of digital communications . Although this was a Defence Department agency it soon saw the need to bring in the major US research Universities. In 1969, a four node network (UCLA, Stanford Research Institute, UC Santa Barbara, and the University of Utah) was established running under a 50kbps circuit. There are details of one of the first Internet communications. Maintaining communication via the telephone, the message LOGWIN was typed in on one terminal in the hope that it would appear on the screen at the remote node. The following conversation is recorded:

'We typed an L into our computer and we asked on the phone 'Do you see the L?'

'Yes we see the L' came the response'

'We typed the O' and we asked 'Do you see the O?'

'Yes we see the O'

'Then we typed a G, and the system crashed.'

The network soon expanded to 15 nodes and in 1973 there was the first cross-Atlantic links to England and Norway. 1981 saw the advent of BITNET ('Because It's Time Network') organized by the City University of New York with an initial connection to Yale. This network provided accessible electronic mail and Listserv servers. In 1984, JANET (the UK's Joint Academic Network) was established, linking higher education Institutions throughout the country, and allowing access to the global Internet.

1992 saw the beginning of the World-Wide Web (although in truth it had been developed at CERN some years previous, after an initial proposal by Tim Berners-Lee ). The success of the Web was assured with NCSA's release of its free browser Mosaic in 1993, allowing a user-friendly way of graphically accessing the many Internet resources that were available. To give some idea of the growth of the Internet the 4-node network of 1967 grew to 28,174 nodes in 1987 and reached 3,212,000 in 1994. It is now measured at approaching 40 million.

What exactly is the Internet? The Internet stands for 'inter-networking', i.e. when two or more computers are inter-networked allowing for communication between them. It is now taken to relate to the global network that allows you to send a message to a machine elsewhere on the network, in this case a global one, or to look at information on a remote machine. There is no Internet as such. It is made up a series of Wider Area Networks (e.g. JANET) talking to each other, which in turn are made up of Local Area Networks (e.g. the network for an individual University), i.e. networks within networks.

## **1.5 THE IMMENSITY OF THE INTERNET**

The Internet is a vast network that connects many independent networks spanning over 170 countries in the world. It links computers of many different types, sizes, and operating systems, and, of course, the many people of those countries that use the Internet to communicate.

The one thing all these different computers have in common is the use of the Internet Protocol, abbreviated as IP, which allows computers of different types to communicate with each other. You will often see reference to the longer abbreviation, TCP/IP, which

stands for Transmission Control Protocol/Internet Protocol. Your own computer uses TCP/IP software to enable it to link to this service.

## **1.6 WHAT CAN BE DONE ON THE INTERNET?**

The Internet Protocol makes it possible for you to communicate in various ways, find things that interest you, and exchange information and files. The most common things you can do are:

1. Send and receive email with people all over the world. Almost as fast as the telephone, there is never a busy signal, and you never play phone tag.
2. Join discussion groups about a common subject with Usenet Newsgroups and email discussion lists.
3. Get or exchange software and files with the File Transfer Protocol (FTP).
4. Log into and use many computers around the world using telnet. Telnet lets you use the resources on a remote computer such as games, databases, library catalogs, and many more interesting things.
5. Connect to thousands of different computers using gopher menu systems, which make navigation from one site to another easy.
6. Explore the World Wide Web, which can use all of the above, and adds easy links to other resources and adds multimedia-graphics, sound, and video capabilities.
7. Talk by keyboard with Internet Relay Chat (IRC), which lets small groups of users meet in conference to talk with each other by typing on their keyboards.

## 1.7 WHO OWNS THE INTERNET?

No organization, corporation or government owns or runs the Internet. Instead, many people and organizations voluntarily participate in task force groups who meet to develop standards for the many various technical needs of running the Internet. Decisions are made by consensus among all who choose to participate, and every point of view is heard in the long process of hashing out decisions and setting new standards.

The equipments; computers, cables, routers, and so on are owned by government and private organizations and are paid for by taxes and user fees. In the early history of the Internet, the US government paid for many of the development and operating costs through government grants. In recent years, the US government has stepped aside except for the portions that link government organizations and let private enterprises develop the nets.

## 1.8 HOW THE INTERNET WORKS

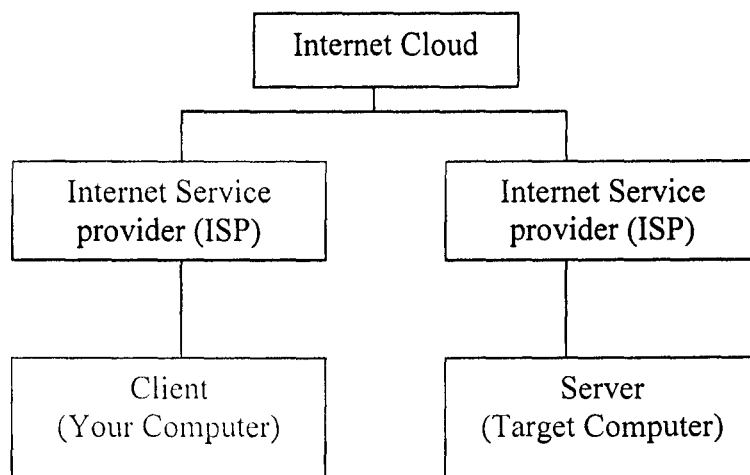


Fig. 1.1 A Basic Diagram of How the Internet Works

## **CHAPTER TWO**

### **2.1 INTRODUCTION**

In the past decade, the World Wide Web (WWW) has become more accessible to the public. Many Companies have created their own Web sites to assist their customers and to bring out any general information about the Company to the public. People can also perform research and other activities, such as chatting, playing games, listening to music e.t.c. Due to the growth of the WWW, Web page designs and page layouts have become very important topics these days. A better-organized Web page can be more attractive to people and can also get the information to the public more effectively.

To have a Web site that stands out from the crowd, an attractive and easy to use page layout is very important. Using technology, we can design a page with different combinations of layouts. It is crucial to detect which combination yields the best result to the viewers. In this project, I will study the different combinations of page layouts and report on their effectiveness.

### **2.2 LITERATURE REVIEW**

In simplest form, a Web site is merely a medium for communicating. A Web site is one's public face; it is one's Web site that will establish their corporate identity.

It is your Web site that will either consolidate or else redefine your brand. It is your Web site that will help build customer loyalty or else dissipate it. A Web site will help increase sales and profits in a business outfit.



If you're a consumer, then a Web site is a product that allows you access information, initiate communication and carry out transactions. If you're a producer, then a Web site is a product that allows you to place your information so that others can access it. Further, it allows you to communicate with people and to transact with them.

### **2.3 WHO IS A WEB SITE VISITOR?**

A web site visitor is like a ghost that visits your Company, looking over your shoulder, peering at your products and information, poking into your support function, seeing how you deal with personnel, peeping at your corporate history, passing through your reception (homepage) to find out more about you, ask you questions, give you feedback, and even buy your products or service.

### **2.4 WHY DOES ONE NEED A WEB SITE?**

A Few years ago having a Web site was a vanity but today having a Web site and your own Domain name is a necessity and very common. People are using the Internet for everything from booking vacations to buying flowers to their loved ones. Students also are not left out in this development as via the Internet, admissions and other academic necessities are gotten. It's quick, convenient, and you never need to leave your desk.

### **2.5 WHAT A WEBSITE CAN DO**

A Web site can be used for a large number of tasks such as:

**Marketing product / service;**

**Providing support to customers;**

**Receiving feedback from visitors;**

**Sharing/distributing files and images;**

And, of course, show the world, services you can render and how technologically balanced you are.

## **2.6 HOW DOES A WEB SITE WORK?**

A Web site is simply a collection of files that reside on a computer connected to the Internet. When people visit your site, they are actually contacting that Web server / computer, and that computer gives the visitor the files they want to see.

## **2.7 PROJECT IDEA AND INCENTIVE**

The purpose of this project is to create a Web site for Electrical/Computer Engineering Department of the Federal University of Technology, Minna, with a result database management facility. Over the years, the Institution has grown in various ways vis-à-vis, number of students, academic excellence and other areas. In this vein, the acquisition of a Web site as well as a database management facility for students to check their results is necessary.

It is known that communication is the connection between two persons and hence, the development of a website for Electrical/Computer Engineering Department of the Federal University of Technology, Minna, creates a medium for the public to acquire information.

Also, going to the Internet to check result can significantly reduce the stress of queuing up to check results. The Department's activities, as well, can be made known to the public in case of future needs, probably admissions, projects and as the case may be.

## CHAPTER THREE

### DESIGNING PAGE LAYOUT

#### 3.1 PLANNING YOUR WEB SITE DESIGN

When you begin thinking about creating a Web site, you should follow a series of planning steps to make sure your site is successful. Even if you are just creating a personal home page that only friends and family will see, it can still be to your advantage to plan the site carefully in order to make sure everyone will be able to use it easily. The following basic steps would help a great deal to realizing the establishment of a good Web site.

1. Creating goals for your site
2. Organizing the site structure
3. Creating your design look
4. Designing the navigation scheme
5. Planning and gathering your assets

##### 3.1.1 Creating goals for your site:

Deciding what your site goals are, should be the first step you take when creating a Web site. Ask yourself, or your client, about what you hope to accomplish by having a Web site. Write down your goals so that you remember them as you go through the design process. Goals help you focus and target your Web site to your particular needs.

A Web site that provides news about a specific subject should have a different look and navigation than a Web site that sells products. The complexity of your goals will affect the navigation, the media that you use (Macromedia Flash, Macromedia Director, and so on), and even the look and feel of your site.

### **3.1.2 Organizing the site structure:**

Organizing your site carefully from the start can save you frustration and time later on. If you begin creating documents without thinking about where in your folder hierarchy they should go, you may end up with a huge, unwieldy folder full of files, or with related files scattered through a half-dozen similarly named folders.

The usual way to set up a site is to create a folder on your local hard disk that contains all the files for your site (referred to as the local site), and to create and edit documents within that folder. You then copy those files to a Web server when you are ready to publish your site and allow the public to view it. This approach is better than creating and editing files on the live public Web site itself, because it allows you to test changes in the local site before making them publicly viewable. When you're finished, you can upload the local site files and update the entire public site at once.

**Break down your site into categories and put related pages in the same folder. For example, your company press releases, contact information, and job postings might all go in one folder, but your online catalog pages might go in a different folder. Use subfolders where necessary. This type of organization will make your site easier to maintain and navigate.**

Decide where to put items such as images and sound files. For example, it's convenient to place all your images in one location, so that when you want to insert an image into a page, you know where to find it. Designers sometimes place all of the non-HTML items to be used on a site in a folder called Assets. This folder may contain other folders—for example, an Images folder, a Macromedia Shockwave folder, and a Sound folder. Or you

might have a separate Assets folder for each group of related pages on your site, if there aren't many assets shared among such groups.

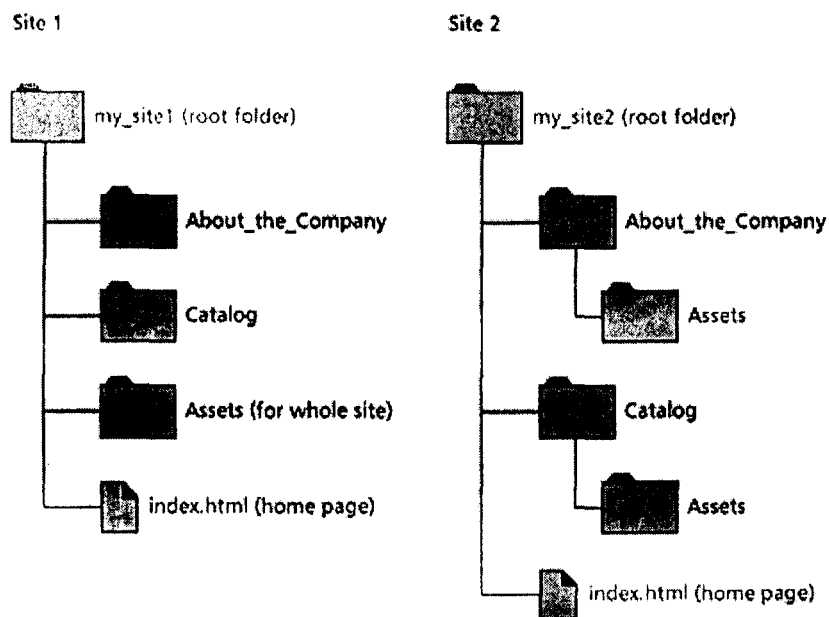


Fig 3.1. Site Organization

Use the same structure for local and remote sites. Your local site and your remote Web site should have exactly the same structure. If you create a local site using Macromedia Dreamweaver and then upload everything to the remote site, Dreamweaver ensures that the local structure is precisely duplicated in the remote site.

### 3.1.3 Creating your design look:

You can save a lot of time later if you plan your design and layout before you begin working in Dreamweaver. You can simply create a mock-up drawing on paper of how you want the site layout to look and then follow it as you build your site; or you can create a composite drawing of your site using software such as Macromedia FreeHand or Macromedia Fireworks.

Maintaining consistency in your page layout and design helps to ensure a good user experience. The user should be able to click through the pages in your site without getting confused. If all the pages have a different look, or the navigation is in a different place on each page, it might frustrate the user. Make sure your site provides a consistent look for your user.

### **3.1.4 Designing the navigation scheme:**

As you design your site, think about the experience you want your visitors to have. Think about how a visitor to your site will be able to move from one area to another. Navigation should be consistent throughout your site. If you place a navigation bar across the top of your home page, try to keep it there for all the linked pages. Consider the following points:

1. Visitors should know where they are in your site and how to return to your top-level page.
2. Search features and indexes make it easier for visitors to find information they are looking for.
3. Feedback features provide a way for visitors to contact the Web master (if appropriate) if something is wrong with the site, and to contact other relevant people associated with the company or the site.

### **3.1.5 Planning and gathering your assets:**

Once you know what your design and layout will look like, you can create and gather the assets that you will need. Assets can be items such as images, text, or media (Flash, Shockwave, and so on). Make sure you have all of these items gathered and ready to go

before you begin developing your site. Otherwise, you'll have to continually stop development to find an image or to create a button.

If you are using images and graphics from a clip-art site or someone else is creating them, make sure you collect them and put them in a folder on your site. If you are creating the assets yourself, make sure you create them all before you start development, including any images you need if you are using roll-overs. Then organize your assets so you can access them easily while creating your site in a web editor e.g. Dreamweaver.

You can reuse page layouts and page elements in various documents by using templates and libraries in your web editor. However, it is easier to create new pages with templates and libraries than it is to apply them to existing documents.

Use templates if many of your pages will use the same layout. Plan and design a template for that layout, and then you can create new pages based on that template. If you decide to change the layout for all the pages, you can simply change the template.

### **3.2 TESTING WORLD WIDE WEB SITES**

Web sites are essentially client/server applications - with Web servers and browser clients. Consideration should be given to the interactions between html pages, TCP/IP communications, Internet connections, firewalls, applications that run in Web pages (such as applets, javascript, plug-in applications), and applications that run on the server side (such as cgi scripts, database interfaces, logging applications, dynamic page generators, asp, etc.). Additionally, there are a wide variety of servers and browsers, various versions of each, small but sometimes significant differences between them, variations in connection speeds, rapidly changing technologies, and multiple standards

and protocols. The end result is that testing for web sites can become a major ongoing effort. Other considerations might include:

1. What are the expected loads on the server (e.g., number of hits per unit time?), and what kind of performance is required under such loads (such as Web server response time, database query response times). What kinds of tools will be needed for performance testing (such as Web load testing tools, other tools already in house that can be adapted, Web robot downloading tools, etc.)?
2. Who is the target audience? What kind of browsers will they be using? What kind of connection speed will they be using? Are they intra-organization (thus with likely high connection speeds and similar browsers) or Internet-wide (thus with a wide variety of connection speeds and browser types)?
3. What kind of performance is expected on the client side (e.g., how fast should pages appear, how fast should animations, applets, etc. load and run)?
4. What processes will be required to manage updates to the Web site's content, and what are the requirements for maintaining, tracking, and controlling page content, graphics, links, etc.
5. Which HTML specification will be adhered to? How strictly? What variations will be allowed for targeted browsers?
6. How are cgi programs, applets, javascripts, ActiveX components, etc. to be maintained, tracked, controlled, and tested?
7. What kinds of security (firewalls, encryptions, passwords, etc.) will be required and what is it expected to do? How can it be tested?



Some sources of site security information include the Usenet newsgroup 'comp.security.announce' and links concerning web site security in the 'Other Resources' section.

### **3.3 USABILITY GUIDELINES**

In Web designs, some guidelines are to be considered but these are subjective and may or may not apply to a given situation; but one should know that:

1. Pages should be 3-5 screens max unless content is tightly focused on a single topic. If larger, provide internal links within the page.
2. The page layouts and design elements should be consistent throughout a site, so that it's clear to the user that they're still within a site.
3. Pages should be as browser-independent as possible, or pages should be provided or generated based on the browser-type.
4. All pages should have links external to the page; there should be no dead-end pages.
5. The page owner, revision date, and a link to a contact person or organization should be included on each page.

### **3.4 WEB SITE DESIGN**

#### **3.4.1 HYPERTEXT MARKUP LANGUAGE (HTML)**

HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page's words and images for the user. Each individual

markup code is referred to as an element (also referred to as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

HTML is a formal Recommendation by the World Wide Web Consortium (W3C) and is generally adhered to by the major browsers, Microsoft's Internet Explorer and Netscape's Navigator. The current version of HTML is HTML 4.0. However, both Internet Explorer and Netscape implement some features differently and provide non-standard extensions. Web developers using the more advanced features of HTML 4 may have to design pages for both browsers and send out the appropriate version to a user. Significant features in HTML 4 are sometimes described in general as dynamic HTML. What is sometimes referred to as HTML 5 is an extensible form of HTML called Extensible Hypertext Markup Language (XHTML).

### **3.4.2 THE MERITS OF HTML**

1. The cascading style sheet, the ability to control Web page content at multiple levels.
2. The ability to create richer forms.
3. Support for frames (which is already supported by the major browsers).
4. Enhancements for tables that make it possible to use captions to provide table content for Braille or speech users.
5. The capability to manage pages so that they can be distributed in different languages.

## CHAPTER 4

### 4.1 ACTIVE SERVER PAGE (ASP)

If an individual is new to the whole scripting scene and doesn't have any kind of programming experience behind him/her, then Active Server Pages (ASP) is probably one of the easiest ways to start in the world of Web application development.

Active Server Pages was a technology introduced by Microsoft in Internet Information Server 2.0. Code named Denali, the technology uses a small Internet Server Programming Application Interface filter (ISAPI Filter) that loads any required scripting language interpreters when a user first requests an ASP page. Because the filter is in memory for as long as the server is running, there is no additional overhead of starting new processes and so it is thought to be much kinder on server resources than the traditional Common Gateway Interface (CGI) applications.

An Active Server Page (ASP) is an HTML page that includes one or more scripts (small embedded programs) that are processed on a Microsoft Web server before the page is sent to the user. An ASP is somewhat similar to a server-side include or a Common Gateway Interface (CGI) application in that all involve programs that run on the server, usually tailoring a page for the user. Typically, the script in the Web page at the server uses input received as the result of the user's request for the page to access data from a database and then builds or customizes the page on the fly before sending it to the requestor. To distinguish them from normal HTML pages, Active Server Pages are given the ".asp" extension.

## **4.2 HOW ASP WORKS**

When an ASP page is requested, the ASP ISAPI filter executes any server side code it finds in the page before sending back a more browser friendly response. Normally this is HTML, but can be HTML with embedded scripts, images or any other file type so long as the correct headers are sent along with the response.

The response an ASP page gives to a request can be manipulated by the use of ASP objects, each have their own properties and methods that can be manipulated by any scripting language that has been configured to run with the server (typically VBScript).

## **4.3 ASP OBJECTS**

1. Application
- 2.ObjectContext
3. Response
4. Request
5. Server
6. Session

The ASP objects listed above are of little use on their own, you will need to be familiar with at least one scripting language before you can start to build your own pages that use these objects. VBScript is the common choice, not only is it the default scripting language that is assumed for ASP pages, but it is the scripting language that is most like English.

### **4.3.1 Application**

The Application object is one that represents the whole domain space, without it ASP will not work. Before ASP can work, you have to create the application; this tells the server to load the ASP ISAPI filter for the files in that particular space. The application object will then run in the background on the server until it is stopped or restarted.

Any variables that you store in the application will be available to all your ASP pages, and so this is an easy way to pass data from one script to another. However it does use system resources (such as memory) and so the use of application level variables is not very efficient and can in some cases slow down your site.

### **4.3.2ObjectContext**

The objectcontext object allows you to create and run transactions. The best way to describe a transaction is a piece of code that can be undone should there be problems.

### **4.3.3 Response**

The response object will send the results of any server side scripts back to the user's browser. It has methods to add headers telling the browser when to expire the page, what character set to use and can even detect if the connection originally used to request the page has been closed. This can be a handy feature as it can stop scripts when there is no one there to see the page, thus saving data transfer and server resources.

Not only does the response object control what gets sent back to the browser, but also

how it gets sent back. It is possible to buffer the response and send it back when the whole script is finished, or you could send the data back in chunks to let your visitors know that it is still working.

#### **4.3.4 Request**

The request object holds all the information that has been sent from the user that can be used in your scripts. This includes any page headers, cookies, form or URL data. Fortunately, all this data has been parsed to some degree and so you don't need to worry about breaking it all down and putting it back together again; just remember to make sure that it is safe to use.

#### **4.3.5 Server**

The server object is mainly used as a utility object; helping create new instances of add-in ASP objects, find the real path of a file and converting output to make it more browser friendly. It can also be used to set the server time out response, which is handy for scripts that need a longer time to execute without sending any output to the browser.

#### **4.3.6 Session**

The session object works a little like the application object, but is specific to the user and the browser they are using. It can make a viable alternative to cookies used to store and move data between scripts, but as sessions tend to expire very quickly (usually 20 minutes after last activity) they can only store data temporarily.

#### **4.4 EXPANDING ASP CAPABILITIES**

There are other objects that are additional to the built in objects listed above that can be used to provide additional functionality.

1. **FileSystemObject** - allows ASP pages to interact directly with files stored anywhere on the host server (which is why a few hosts have this disabled). This is a useful object; Web site owners rely on it to create the pages.
2. **CDONTS (Collaboration Data Objects for Windows NT Server)** - used to send and receive messages, this object is Microsoft's offering of adding mail capabilities to a site. As the name suggests, it is a little dated now and most of the hosts I've asked tend to prefer other 3rd party objects to do the same job; JMail and ASPEmail being two preferred replacements.
3. **ADO (ActiveX Data Objects)** - used to interact with databases (either physical or those created in working memory).

The fact is that ASP may not be as fast as some of the scripting languages available today, but it is the easiest for a beginner to learn and with a little imagination and a logical mind it can do a wide variety of things even in designing a School's Web site.

#### **4.5 WHAT YOU CAN DO WITH ACTIVE SERVER PAGES**

There are many things you can do with Active Server Pages.

1. You can display date, time, and other information in different ways.
2. You can make a survey form and ask people who visit your site to fill it out, send emails, save the information to a file, etc.

Active Server Pages (ASP) solves all the problems associated with Common Gateway Interface (CGI) and server Application Programming Interface (API)s. In addition to

being just as efficient as Internet Server Application Programming Interface (ISAPI) applications, ASP is a lot more simple to learn and much easier to use.

With ASP, you simply write your code in the HTML page itself. The HTML tags and the code are side by side. You write the code in a simple scripting language that is easy to learn and easy to use. Then you save the page to your Web site and it's ready to go. No compiling and no complex interfacing.

ASP makes it much quicker and easier to create highly interactive Web sites. It also makes your pages easier to maintain and update in the future. It is a technology in itself and a feature built into a different piece of software. For ASP you need either Microsoft Internet Information Server or a Personal Web Server.

#### **4.6 PUTTING DATA ON THE INTERNET USING ACTIVE SERVER PAGE**

Opening a connection to data source on every page just before it is required and closing immediately after that in the ASP page, is a good performance benefit. Releasing the database connection quickly allows database connection pooling. This way, the connection is optimally used, as it is active for a short duration when it is needed. For maintenance sake, if you do not want to have connection information on every page, you can create a database access ASP page with connection information and then include that file in every ASP page where you intend to use that connection.

Another technique used for opening a connection is, assigning the Connection object to an application -level object. Though this allows for easy maintenance as any changes to the connection string are to be applied at only one place, it has serious performance



issues. This technique maintains an open connection to the database throughout the life of the application. That means, system has to maintain an open connection to the database even if it is not being used.

A special way of the implementation of ASP is encapsulating it in an ActiveX component, usually a .dll file. The ActiveX components used from ASP pages are often called ASP components. When the ADO objects are wrapped in an ASP component, this ASP data access component becomes an extra layer between ASP pages and the database. So, instead of having several ASP pages making the same direct database access, an ASP data access component can be created and used to perform this recurring operation and then instantiated in every ASP page needing database access. With this approach, the changes to underlying data structure, results in changes to just one component rather than all pages that use that particular changed data item, thus increasing maintainability. As components are reusable entities, the same component can be used many times without having to recode the data access each and every time it is needed.

However, the choice of implementation of data access really varies based on the requirements of the application being developed. If the application does not need frequent modification to the database structure or the database and the connection information is not going to change frequently, then encapsulating that info in an ASP component may be a good idea. Also, if application includes, complex database operations, business rules, school rules, which are not going to change frequently then ASP components, can provide significant performance benefits. Otherwise, instead of using ASP components for data access, the connection information can be used from a DSN on web server in an

include ASP file and this file can be included in all ASP files needing data access. The stored procedures can be used for recurring or complex queries, or even for encapsulating business rules.

## CHAPTER FIVE

### RESULT AND CONCLUSION

#### 5.1 INTRODUCTION

This is basically the result of the Project “Web site Design with Active server Page Facility” carried out for Electrical/Computer Engineering Department of the Federal University of Technology, Minna.

The results of the design as well as the database management were tested to check for user-friendly competence.

#### 5.2 RESULT FOR THE SITE DESIGN

The created Web site was given the name “ecefutmx”. The site functions after its files have been uploaded to a Web hoster and hence it can be accessed by typing its Uniform Resource Locator (URL), i.e. <http://www.ecefutmx.edu> in the browser.

#### 5.3 RESULT FOR THE DATABASE

The database facility has been designed in such a way that students can check their results wherever they are. The lecturers as well can enter scores of individual students, and their SGPAs and CGPAs are computed by the facility. Students can check their results on-line by typing <http://www.ecefutmx.edu> on the browser and clicking “RESULT”. Alumni can view their past results as well.

## **CONCLUSION**

The purpose and objective of this project was maximally achieved, i.e. the world at large and the students of the Department are given every piece of major information they need. Rooms have been made for easy accessibility, maintainability and future expansion by recreation.

Hand-coding using HTML is a fundamental and basic tool for designing Web sites. It is easy to comprehend and creates an environment for interaction with such great Web editors like the Dreamweaver (Dreamweaver has various editions), Flash e.t.c.

Active Server Pages (ASP) was used for the database management because it is probably one of the easiest ways to start in the world of Web application development. It provides a quick overview to help you get an idea of how the technology works.

This site is ready for publishing and therefore, can be uploaded to the Internet using the services of CuteFTP or any desired File transfer Protocol.

## **RECOMMENDATION**

After analyzing all the researched information and feedback from different users, I recommend that a general-purpose Web site should be planned in such a way to ease future upgrading, modernization and development. Concerned persons wishing to advance on this project may reflect on the following recommendations as tasks that can be integrated into this work.

1. First of all, an Internet connection is needed for the Institution.
2. The use of current web editors to maximize and implement highly interactive Web pages.

3. A database that can be accessed by aspiring students of the University to check if they are admitted.

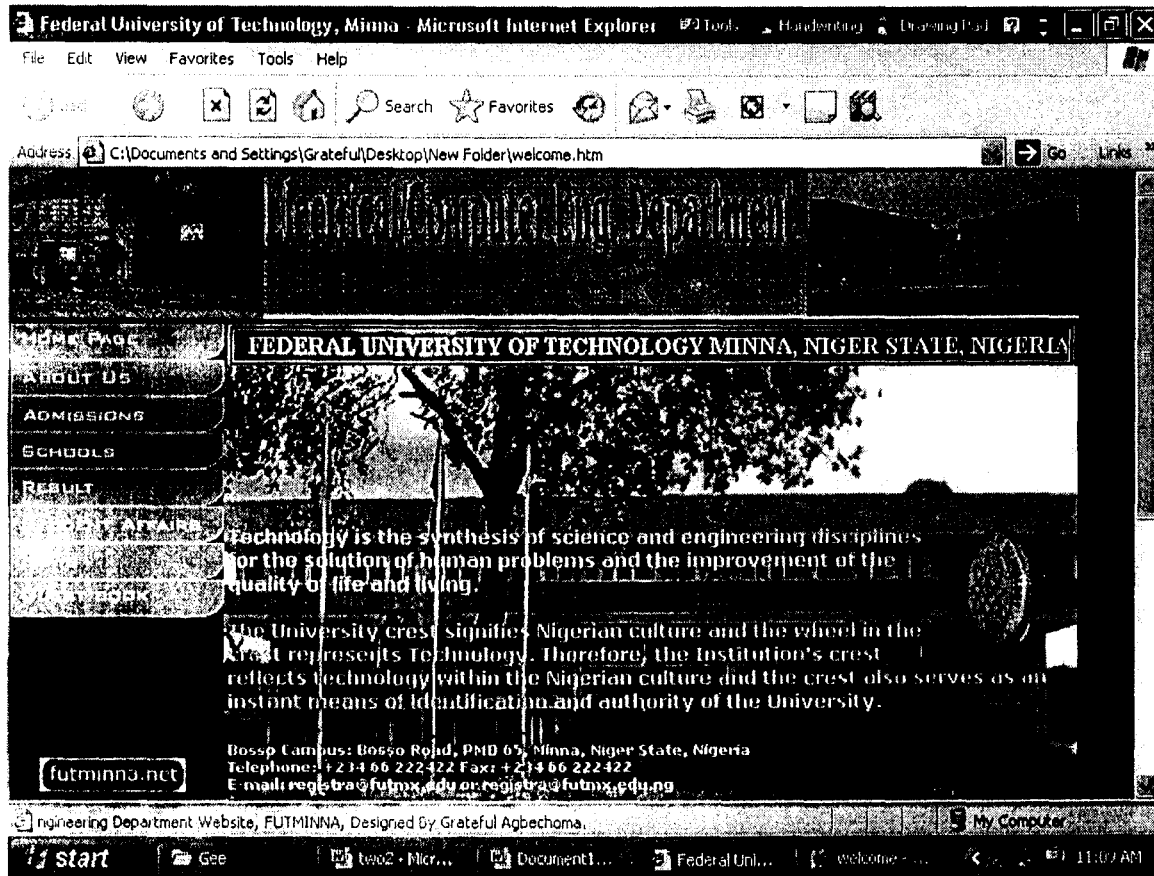
To investigate the effectiveness of different types of page design, the following lines of inquiry should be pursued:

1. What are the different types of page layout?
2. How to measure a good page layout?
3. What are the advantages and disadvantages of different types of page layouts?
4. What are the differences between frame and non-frame page layouts?
5. How does a page with navigation bar compare to a page without a navigating bar?
6. How does the cost of creating different Web pages affect each Web page design?
7. What is the most efficient way (both time and cost) of creating a Web page?
8. How does the page layout affect the cost of making the page?
9. How does the different page layout ease the viewer's ability to browse the page?
10. How does the page differ in displaying the same layout in different Web browsers?

## REFERENCES

1. ANDREW TRUBAC, "Hypertext Markup Language (HTML)"  
<http://www.whatis.techtarget.com>
2. ATUL TOTRE, "The Data Access Dilemma"  
[http://www.asp-help.com/database-db\\_dataaccess](http://www.asp-help.com/database-db_dataaccess)
3. EMILY RICKETTS, "Planning your Web site design"  
<http://www.macromedia.com>
4. GERRY MCGOVERN, "What is a website?"  
<http://www.gerrymcgovern.com>
5. JOE BARTA, "Professional Website Designs", 2<sup>nd</sup> Edition, Maximum Publishers, 1998.
6. RICK HOWER, "Software QA and Test Resource"  
<http://www.softwareqatest.com/testing site/>
7. ROSEMARIE WISE, "Active Server Pages"  
<http://www.akamarketing.com/active-server-pages.html>
8. WALT HOWE, "History of the Internet"  
<http://www.walthowe.com>

# Appendix A:



## Appendix B:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<html>
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_jumpMenu(targ,selObj,restore){ //v3.0
    eval(targ+".location='"+selObj.options[selObj.selectedIndex].value+"'");
    if (restore) selObj.selectedIndex=0;
}

function MM_findObj(n, d) { //v4.01
    var p,i,x; if(!d) d=document; if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++) x=d.forms[i][n];
    for(i=0;!x&&d.layers&&i<d.layers.length;i++)
x=MM_findObj(n,d.layers[i].document);
    if(!x && d.getElementById) x=d.getElementById(n); return x;
}

function MM_jumpMenuGo(selName,targ,restore){ //v3.0
    var selObj = MM_findObj(selName); if (selObj) MM_jumpMenu(targ,selObj,restore);
}
//-->
</script>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Federal University of Technology, Minna</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_displayStatusMsg(msgStr) { //v1.0
    status=msgStr;
    document.MM_returnValue = true;
}
//-->
</script>
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_reloadPage(init) { //reloads the window if Nav4 resized
    if (init==true) with (navigator) {if
((appName=="Netscape")&&(parseInt(appVersion)==4)) {
```



```

    document.MM_pgW=innerWidth; document.MM_pgH=innerHeight;
onresize=MM_reloadPage; } }
    else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH)
location.reload();
}
MM_reloadPage(true);
//-->
</script>
</head>

```

```

<body bgcolor="#000066" text="black" link="blue" vlink="purple" alink="red"
onload="namosw_scrollstatus('Welcome To Electrical/Computer Engineering
Department Website,', 'FUTMINNA,', 'Designed By Grateful Agbechoma.')">
<div id="Layer2" style="position:absolute; left:21px; top:401px; width:104px;
height:26px; z-index:2">
  <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="103" height="24">
  <param name="movie" value="button5.swf">
  <param name="quality" value="high">
  <param name="base" value=".">
  <param name="bgcolor" value="#FFFFFF00">
  <embed src="button5.swf" base="." quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" width="103" height="24"
bgcolor="#FFFFFF00"></embed>
  </object>
</div>
<meta name="generator" content="Namo WebEditor v3.0">
<script language="JavaScript">
<!--
function namosw_scrollstatus()
{
  var str = "";
  if (namosw_scrollstatus.arguments.length > 0) {
    var i;
    for (str = "", i = 0; i < 100; i++)
      str += ' ';
    for (i = 0; i < namosw_scrollstatus.arguments.length; i++)
      str += namosw_scrollstatus.arguments[i] + ' ';
    document.namosw_scrollstatus = str;
  } else {
    var src_str = document.namosw_scrollstatus;
    str = src_str.substring(1, src_str.length);
    str += src_str.substring(0, 1);
    document.namosw_scrollstatus = str;
  }
}

```

```

}
window.status = str;
setTimeout('namosw_scrollstatus()', 100);
}

</script>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Federal University of Technology</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#000066" topmargin="0" bottommargin="15" leftmargin="0"
rightmargin="0">

<table cellspacing="20" cellpadding="0" width="150" align="left"><tr>
<tr><a></a></tr>
<tr><a href="aboutus.htm"></a></tr>
<tr><a href="admissions.htm"></a></tr>
<tr><a href="schools.htm"></a></tr>
<tr><a href="result.htm"></a></tr>
<tr><a href="studentaffairs.htm"></a></tr>
<tr><a href="news.htm"></a></tr>
<tr><a href="guest.htm"></a></tr>

</td></table>

<table width="75%" border="1">
<tr>
<td><marquee>
<font color="#FFFFFF"><strong>FEDERAL UNIVERSITY OF TECHNOLOGY
MINNA, NIGER
STATE, NIGERIA</strong></font>
</marquee></td>
</tr>

```

